aws

Lustre 사용 설명서

FSx for Lustre

FSx for Lustre: Lustre 사용 설명서

Table of Contents

Amazon FSx for Lustre란? 1
다양한 배포 옵션 2
다양한 스토리지 옵션 2
FSx for Lustre 및 데이터 리포지토리 2
FSx for Lustre S3 데이터 리포지토리 통합 2
FSx for Lustre 및 온프레미스 데이터 리포지토리 3
파일 시스템 액세스 3
서비스와의 통합 AWS 4
보안 및 규정 준수 4
가정 5
Amazon FSx for Lustre 요금 5
Amazon FSx for Lustre 포럼 5
Amazon FSx for Lustre를 처음 사용하시나요? 5
설정 7
Amazon Web Services 가입 7
등록하십시오. AWS 계정 7
관리자 액세스 권한이 있는 사용자 생성 8
Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가 9
FSx for Lustre가 S3 버킷에 대한 액세스를 확인하는 방법 10
다음 단계 11
시작하기 12
필수 조건 12
FSx for Lustre 파일 시스템 만들기 13
Lustre 클라이언트 설치 18
파일 시스템을 탑재 19
워크로드 실행 20
리소스 정리 21
파일 시스템 배포 옵션 22
배포 옵션 22
스크래치 파일 시스템 23
영구 파일 시스템 23
퍼시스턴트_2 배포 유형 24
퍼시스턴트_1 배포 유형 24
사용 가능한 리전 25

데이터 리포지토리 사용 27
데이터 리포지토리 개요 27
POSIX 메타데이터 지원 29
하드 링크 및 S3로 내보내기 31
S3 버킷에 POSIX 권한 연결 32
S3 버킷에 파일 시스템 연결 34
연결된 S3 버킷에 대한 리전 및 계정 지원 36
S3 버킷 링크 생성 36
서버 측 암호화된 Amazon S3 버킷 사용 45
데이터 리포지토리에서 변경 내용 가져오기 48
S3 버킷에서 업데이트 자동 가져오기 49
데이터 리포지토리 작업을 사용하여 변경 내용 가져오기 54
파일 시스템에 파일 미리 로드 56
데이터 리포지토리로 변경 내용 내보내기 57
업데이트를 S3 버킷으로 자동 내보냅니다. 58
데이터 리포지토리 작업을 사용하여 변경 내용 내보내기 61
HSM 명령을 사용하여 파일 내보내기 64
데이터 리포지토리 작업 65
데이터 리포지토리 작업 유형 65
작업 상태 및 세부 정보 66
데이터 리포지토리 작업 사용 67
작업 완료 보고서 사용 74
작업 실패 문제 해결 75
파일 릴리스 79
데이터 리포지토리 작업을 사용하여 파일을 릴리스합니다. 80
온프레미스 데이터에 Amazon FSx 사용 84
데이터 리포지토리 이벤트 로그 85
이전 배포 유형으로 작업하기 98
Amazon S3 버킷에 파일 시스템 연결 98
S3 버킷에서 업데이트 자동 가져오기 106
성능 111
FSx for Lustre 파일 시스템의 작동 방식 111
파일 시스템 성능 총계 112
예제: 기준 및 버스트 처리량 총계 117
파일 시스템 메타데이터 성능 117
파일 시스템 스토리지 레이아웃 118

파일 시스템의 스트라이핑 데이터 119
스트라이핑 구성 수정 120
프로그레시브 파일 레이아웃 121
성능 및 사용량 모니터링 123
성능 팁 123
파일 시스템 액세스 126
Lustre 파일 시스템 및 클라이언트 커널 호환성 126
Lustre 클라이언트 설치 중 130
Amazon Linux 130
CentOS, Rocky Linux, Red Hat 132
Ubuntu 142
SUSE Linux 148
Amazon EC2에서 마운트 150
Amazon ECS에 마운트 152
Amazon ECS 작업을 호스팅하는 Amazon EC2 인스턴스에서 마운트 153
도커 컨테이너에서 마운트 154
온프레미스 또는 다른 VPC에서 마운트 155
Amazon FSx 자동 마운트 157
/etc/fstab 사용하여 자동 마운트 157
특정 파일 세트 마운트 160
파일 시스템 마운트 해제 161
EC2 스팟 인스턴스 사용 161
Amazon EC2 스팟 인스턴스 중단 처리 162
파일 시스템 관리 165
백업 165
FSx for Lustre의 백업 지원 167
자동 일일 백업 작업 167
사용자 시작 백업 작업 167
아마존 AWS Backup FSx와 함께 사용 168
백업 복사 169
동일한 내에서 백업 복사 AWS 계정 171
백업 복원 172
백업 삭제 173
스토리지 할당량 173
할당량 적용 174
할당량 유형 174

할당량 제한 및 유예 기간 175
할당량 설정 및 보기 176
할당량 및 Amazon S3 연결 버킷 179
할당량 및 백업 복원 180
스토리지 용량 180
스토리지 용량 증가 시 고려 사항 181
스토리지 용량을 늘려야 하는 경우 181
동시 스토리지 크기 조정 및 백업 요청을 처리하는 방법 182
스토리지 용량을 늘리는 방법 182
스토리지 용량 증가 모니터링 184
메타데이터 성능 관리 187
Lustre 메타데이터 성능 구성 188
메타데이터 성능 향상 시 고려 사항 189
메타데이터 성능을 향상시켜야 하는 경우 190
메타데이터 성능을 높이는 방법 190
메타데이터 구성 모드 변경 191
메타데이터 구성 업데이트 모니터링 193
처리량 용량 195
처리량 용량 업데이트 시 고려 사항 196
처리량 용량을 수정해야 하는 경우 196
처리량 용량을 수정하는 방법 196
처리량 용량 변화 모니터링 198
데이터 압축 200
데이터 압축 관리 200
이전에 작성한 파일 압축 203
파일 크기 보기 203
지표 사용 CloudWatch 204
Root루트 스쿼시 204
루트 스쿼시 작동 방식 205
루트 스쿼시 관리 206
파일 시스템 상태 211
리소스 태그 지정 212
태그 기본 사항 212
리소스 태그 지정 213
태그 제한 213
권한 및 태그 214

유지 관리 214
파일 시스템 삭제 215
를 사용하여 FSx for Lustre로 마이그레이션하기 DataSync 216
를 사용하여 파일을 마이그레이션합니다. AWS DataSync 216
필수 조건 216
DataSync 마이그레이션 기본 단계 217
파일 시스템 모니터링 218
를 통한 모니터링 CloudWatch 218
파일 시스템 지표 218
파일 시스템 메타데이터 메트릭 222
AutoImport AutoExport 및 지표 225
Amazon FSx for Lustre 측정기준 226
Amazon FSx for Lustre 지표 사용 방법 227
CloudWatch 지표 액세스 228
경보 생성 229
CloudWatch 로그를 사용한 로깅 230
로깅 개요 231
로그 대상 231
로깅 관리 232
로그 보기 234
를 사용하여 로그인하기 AWS CloudTrail 234
Amazon FSx for Lustre 정보는 다음에서 확인할 수 있습니다. CloudTrail 235
Amazon FSx for Lustre 로그 파일 항목 이해 236
보안 239
데이터 보호 240
데이터 암호화 241
인터네트워크 트래픽 개인 정보 245
자격 증명 및 액세스 관리 246
고객 246
보안 인증 정보를 통한 인증 247
정책을 사용한 액세스 관리 250
FSx for Lustre와 IAM 252
자격 증명 기반 정책 예시 258
AWS 관리형 정책 261
문제 해결 273
Amazon FSx에서 태그 사용 275

서비스 링크 역할 사용 ... 281
Amazon VPC를 사용한 파일 시스템 액세스 제어 ... 287
Amazon VPC 보안 그룹 ... 288
Lustre 클라이언트 VPC 보안 그룹 규칙 ... 291
Amazon VPC 네트워크 ACL ... 293
규정 준수 검증 ... 293
인터페이스 VPC 엔드포인트 ... 295
Amazon FSx 인터페이스 VPC 엔드포인트에 대한 고려 사항 ... 295
Amazon FSx API에 대한 인터페이스 VPC 엔드포인트 생성 ... 295
Amazon FSx에 대한 VPC 엔드포인트 정책 생성 ... 296
할당량 ... 297
늘릴 수 있는 할당량 ... 297
각 파일 시스템의 리소스 할당량 ... 298
추가 고려 사항 ... 299
문제 해결 ... 300
파일 시스템 생성 실패 ... 300
잘못 구성된 보안 그룹 때문에 파일 시스템을 생성할 수 없습니다. ... 300
S3 버킷에 연결된 파일 시스템을 생성할 수 없습니다. ... 301
파일 시스템 마운트 실패 ... 301
파일 시스템 마운트 즉시 실패 ... 301
파일 시스템 마운트가 중단된 후 실패하고 제한 시간 초과 오류가 표시됨 ... 302
자동 마운트 실패 및 인스턴스 무응답 ... 302
시스템 부팅 중에 파일 시스템 마운트 실패 ... 302
DNS 이름을 사용한 파일 시스템 마운트에 실패 ... 303
파일 시스템 액세스 불가 ... 304
파일 시스템 탄력적 네트워크 인터페이스에 연결된 탄력적 IP 주소 삭제됨 ... 304
수정 또는 삭제된 파일 시스템 탄력적 네트워크 인터페이스 ... 304
DRA 생성 실패 ... 304
디렉터리 이름을 바꾸는 데 시간이 오래 걸립니다. ... 306
잘못 구성된 연결된 S3 버킷 ... 306
스토리지 문제 ... 308
스토리지 대상에 공간이 없어서 쓰기 오류가 발생했습니다. ... 308
OST의 스토리지 불균형 ... 308
CSI 드라이버 ... 312
추가 정보 ... 313
사용자 지정 백업 일정 설정 ... 313

아키텍처 개요 313
AWS CloudFormation 템플릿 314
배포 자동화 314
추가 옵션 316
사용 설명서 기록 318
..... cccxxxiv

Amazon FSx for Lustre란?

FSx for Lustre를 사용하면 널리 사용되는 고성능 Lustre 파일 시스템을 쉽고 비용 효율적으로 시작하고 실행할 수 있습니다. Lustre는 기계 학습, 고성능 컴퓨팅 (HPC), 비디오 처리, 금융 모델링 등 속도가 중요한 워크로드에 사용합니다.

오픈 소스 Lustre 파일 시스템은 빠른 스토리지가 필요한 애플리케이션, 즉 스토리지가 컴퓨팅 속도를 따라잡아야 하는 애플리케이션을 위해 설계되었습니다. Lustre는 전 세계적으로 계속 증가하는 데이터 세트를 빠르고 저렴하게 처리해야 하는 문제를 해결하기 위해 개발되었습니다. 세계에서 가장 빠른 컴퓨터를 위해 설계된 널리 사용되는 파일 시스템입니다. 1밀리초 미만의 지연 시간, 최대 수백 GBps의 처리량, 최대 수백만 IOPS를 제공합니다. Lustre에 대한 자세한 내용은 Lustre 웹 사이트를 참조하세요.

완전관리형 서비스인 Amazon FSx를 사용하면 스토리지 속도가 중요한 워크로드에 Lustre를 더 쉽게 사용할 수 있습니다. FSx for Lustre를 사용하면 Lustre 파일 시스템의 설정 및 관리에 따르는 기존의 복잡성을 없애고, 몇 분 안에 엄격한 테스트를 거친 고성능 파일 시스템을 가동하여 실행할 수 있습니다. 또한 다양한 배포 옵션을 제공하므로 필요에 맞게 비용을 최적화할 수 있습니다.

FSx for Lustre는 POSIX와 호환되므로 변경하지 않고도 현재 Linux 기반 애플리케이션을 사용할 수 있습니다. FSx for Lustre는 네이티브 파일 시스템 인터페이스를 제공하며 일반 파일 시스템이 Linux 운영 체제에서 작동하는 것처럼 작동합니다. 또한 read-after-write 일관성을 제공하고 파일 잠금을 지원합니다.

주제

- 다양한 배포 옵션
- 다양한 스토리지 옵션
- FSx for Lustre 및 데이터 리포지토리
- FSx for Lustre 파일 시스템 액세스
- 서비스와의 통합 AWS
- 보안 및 규정 준수
- 가정
- Amazon FSx for Lustre 요금
- Amazon FSx for Lustre 포럼
- Amazon FSx for Lustre를 처음 사용하시나요?

다양한 배포 옵션

Amazon FSx for Lustre는 다양한 데이터 처리 요구 사항을 수용할 수 있는 다양한 스크래치 및 영구 파일 시스템을 제공합니다. 스크래치 파일 시스템은 임시 스토리지 및 단기 데이터 처리에 적합합니다. 데이터는 복제되지 않으며 파일 서버에 장애가 발생하는 경우 지속되지 않습니다. 영구 파일 시스템은 장기 스토리지 및 처리량 중심의 워크로드에 적합합니다. 영구 파일 시스템에서는 데이터가 복제되어, 장애가 발생할 경우 파일 서버가 교체됩니다. 자세한 정보는 FSx for Lustre 파일 시스템 배포 옵션을 참조하세요.

다양한 스토리지 옵션

Amazon FSx for Lustre는 다양한 데이터 처리 요구 사항에 최적화된 솔리드 스테이트 드라이브(SSD) 및 하드 디스크 드라이브(HDD) 스토리지 유형 중에서 선택할 수 있는 옵션을 제공합니다.

- SSD 스토리지 옵션 - 일반적으로 작고 무작위 파일 작업이 많으며 지연 시간이 짧고 IOPS 집약적인 워크로드의 경우 SSD 스토리지 옵션 중 하나를 선택합니다.
- HDD 스토리지 옵션 - 일반적으로 대용량의 순차적 파일 작업이 필요한 처리량 집약적인 워크로드의 경우 HDD 스토리지 옵션 중 하나를 선택합니다.

HDD 스토리지 옵션으로 파일 시스템을 프로비저닝하는 경우 HDD 스토리지 용량의 20%에 해당하는 읽기 전용 SSD 캐시를 선택적으로 프로비저닝할 수 있습니다. 자주 액세스하는 파일에 대해 1밀리초 미만의 지연 시간 및 더 높은 IOPS가 제공됩니다. SSD 기반 파일 시스템과 HDD 기반 파일 시스템은 SSD 기반 메타데이터 서버로 프로비저닝됩니다. 따라서 파일 시스템 작업의 대부분을 차지하는 모든 메타데이터 작업이 1밀리초 미만의 지연 시간으로 제공됩니다.

이러한 스토리지 옵션 성능에 대한 자세한 내용은 Amazon FSx for Lustre 성능 섹션을 참조하세요.

FSx for Lustre 및 데이터 리포지토리

FSx for Lustre 파일 시스템을 Amazon S3의 데이터 리포지토리 또는 온프레미스 데이터 스토어에 연결할 수 있습니다.

FSx for Lustre S3 데이터 리포지토리 통합

FSx for Lustre는 Amazon S3와 통합되므로 Lustre 고성능 파일 시스템을 사용하여 클라우드 데이터 세트를 더 쉽게 처리할 수 있습니다. FSx for Lustre 파일 시스템은 Amazon S3 버킷에 연결된 경우 S3 객체를 파일로 투명하게 표시합니다. Amazon FSx는 파일 시스템 생성 시 S3 버킷에 있는 모든 기존 파

일 목록을 가져옵니다. Amazon FSx는 파일 시스템이 생성된 후 데이터 리포지토리에 추가된 파일 목록을 가져올 수도 있습니다. 워크플로 요구 사항에 맞게 가져오기 기본 설정을 지정할 수 있습니다. 이 파일 시스템에서는 파일 시스템 데이터를 S3에 다시 작성하는 것도 가능합니다. 데이터 리포지토리 작업은 FSx for Lustre 파일 시스템과 Amazon S3의 내구성 있는 데이터 리포지토리 간의 데이터 및 메타데이터 전송을 간소화합니다. 자세한 내용은 Amazon FSx for Lustre에서 데이터 리포지토리 사용 및 데이터 리포지토리 작업 섹션을 참조하세요.

FSx for Lustre 및 온프레미스 데이터 리포지토리

Amazon FSx for Lustre를 사용하면 또는 를 사용하여 데이터를 가져와서 온프레미스에서 로 데이터 처리 워크로드를 버스트할 수 있습니다. AWS 클라우드 AWS Direct Connect AWS VPN자세한 정보는 온프레미스 데이터에 Amazon FSx 사용을 참조하세요.

FSx for Lustre 파일 시스템 액세스

단일 FSx for Lustre 파일 시스템에 연결된 컴퓨팅 인스턴스 유형 및 Linux Amazon Machine Image(AMI)를 필요에 맞게 사용할 수 있습니다.

Amazon FSx for Lustre 파일 시스템은 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스에서 실행되는 컴퓨팅 워크로드, Amazon Elastic Container Service(Amazon ECS) 도커 컨테이너 및 Amazon Elastic Kubernetes Service(Amazon EKS)에서 실행되는 컨테이너에서 액세스할 수 있습니다.

- Amazon EC2 - 오픈 소스 Lustre 클라이언트를 사용하여 Amazon EC2 컴퓨팅 인스턴스에서 파일 시스템에 액세스합니다. Amazon EC2 인스턴스는 네트워킹 구성이 VPC 내의 서브넷 전체에 대한 액세스를 허용하는 경우 동일한 Amazon Virtual Private Cloud(Amazon VPC) 내의 다른 가용 영역에서 파일 시스템에 액세스할 수 있습니다. Amazon FSx for Lustre 파일 시스템을 마운트한 후에는 로컬 파일 시스템에서와 마찬가지로 파일 및 디렉터리를 사용할 수 있습니다.
- Amazon EKS - Amazon EKS 사용 설명서에 설명된 대로, 오픈 소스 FSx for Lustre CSI 드라이버를 사용하여 Amazon EKS에서 실행되는 컨테이너에서 Amazon FSx for Lustre에 액세스합니다. Amazon EKS에서 실행되는 컨테이너는 Amazon FSx for Lustre에서 지원하는 고성능 영구 볼륨(PV)을 사용할 수 있습니다.
- Amazon ECS - Amazon EC2 인스턴스의 Amazon ECS Docker 컨테이너에서 Amazon FSx for Lustre에 액세스합니다. 자세한 정보는 Amazon Elastic Container Service에 마운트을 참조하세요.

Amazon FSx for Lustre는 Amazon Linux 2 및 Amazon Linux, Red Hat Enterprise Linux(RHEL), CentOS, Ubuntu, SUSE Linux 등 가장 널리 사용되는 리눅스 기반 AMI와 호환됩니다. Lustre 클라이

언트는 Amazon Linux 2 및 Amazon Linux에 포함되어 있습니다. RHEL, CentOS 및 우분투의 경우 Lustre 클라이언트 리포지토리는 이러한 운영 체제와 호환되는 클라이언트를 제공합니다. AWS

FSx for Lustre를 사용하면 또는 를 통해 데이터를 가져와서 컴퓨팅 집약적인 워크로드를 온프레미스에서 로 버스트할 수 있습니다. AWS 클라우드 AWS Direct Connect AWS Virtual Private Network온프레미스에서 Amazon FSx 파일 시스템에 액세스하고, 필요에 따라 데이터를 파일 시스템으로 복사하고, 클라우드 내 인스턴스에서 컴퓨팅 집약적인 워크로드를 실행할 수 있습니다.

FSx for Lustre 파일 시스템에 액세스할 수 있는 클라이언트, 컴퓨팅 인스턴스 및 환경에 대한 자세한 내용은 파일 시스템 액세스 섹션을 참조하세요.

서비스와의 통합 AWS

Amazon FSx for Lustre는 아마존과 입력 데이터 소스로 통합됩니다. SageMaker FSx for SageMaker Lustre와 함께 사용하면 Amazon S3에서 초기 다운로드 단계를 생략하여 기계 학습 교육 작업을 가속화할 수 있습니다. 또한 동일한 데이터 세트에서 반복적인 작업을 위해 공통 객체를 반복적으로 다운로드하지 않아 S3 요청 비용이 절약되므로 총 소유 비용(TCO)이 절감됩니다. 자세한 내용은 무엇입니까를 참조하십시오. SageMaker Amazon SageMaker 개발자 가이드에서 Amazon FSx for Lustre를 데이터 소스로 사용하는 방법에 대한 자세한 내용은 Machine Learning 블로그의 Amazon FSx SageMaker for Lustre와 Amazon EFS SageMaker 파일 시스템을 사용하여 Amazon에서의 교육 가속화를 참조하십시오.AWS

FSx for AWS Batch Lustre는 EC2 시작 템플릿 사용과 통합됩니다. AWS Batch 에서 고성능 컴퓨팅 (HPC) AWS 클라우드, 기계 학습 (ML) 및 기타 비동기 워크로드를 비롯한 배치 컴퓨팅 워크로드를 실행할 수 있습니다. AWS Batch 작업 리소스 요구 사항에 따라 인스턴스 크기를 자동 및 동적으로 조정합니다. 자세한 내용은 What Is를 참조하십시오. AWS BatchAWS Batch 사용 설명서에서.

FSx for Lustre는 와 통합됩니다. AWS ParallelCluster AWS ParallelCluster AWS HPC 클러스터를 배포하고 관리하는 데 사용되는 지원되는 오픈 소스 클러스터 관리 도구입니다. 클러스터 생성 프로세스 중에 FSx for Lustre 파일 시스템을 자동으로 생성하거나 기존 파일 시스템을 사용할 수 있습니다.

보안 및 규정 준수

FSx for Lustre 파일 시스템은 저장 및 전송 중 암호화를 지원합니다. Amazon FSx는 () 에서 관리되는 키를 사용하여 저장된 파일 시스템 데이터를 자동으로 암호화합니다. AWS Key Management Service AWS KMS지원되는 Amazon EC2 인스턴스에서 액세스하는 AWS 리전 경우 특정 파일 시스템에서 전송 중인 데이터도 자동으로 암호화됩니다. 전송 데이터 암호화가 지원되는 경우를 포함하여 FSx for AWS 리전 Lustre의 데이터 암호화에 대한 자세한 내용은 을 참조하십시오. Amazon FSx for Lustre 의

데이터 암호화 Amazon FSx는 ISO, PCI-DSS 및 SOC 인증을 준수하는 것으로 평가되었으며 HIPAA 인증을 받았습니다. 자세한 정보는 FSx for Lustre 보안을 참조하세요.

가정

이 가이드에서는 다음과 같은 가정을 합니다.

- Amazon Elastic Compute Cloud(Amazon EC2)를 사용할 경우 해당 서비스에 익숙하다고 가정합니다. Amazon EC2 사용 방법에 대한 자세한 내용은 Amazon EC2 설명서를 참조하세요.
- Amazon Virtual Private Cloud(Amazon VPC) 사용에 익숙하다고 가정합니다. Amazon VPC 사용 방법에 대한 자세한 내용은 Amazon VPC 사용 설명서를 참조하세요.
- Amazon VPC 서비스를 기반으로 하는 VPC의 기본 보안 그룹에 대한 규칙을 변경하지 않은 것으로 가정합니다. 변경한 경우 Amazon EC2 인스턴스에서 Amazon FSx for Lustre 파일 시스템으로의 네트워크 트래픽을 허용하는 데 필요한 규칙을 추가했는지 확인합니다. 자세한 내용은 Amazon VPC를 사용한 파일 시스템 액세스 제어를 참조하세요.

Amazon FSx for Lustre 요금

Amazon FSx for Lustre를 사용하면 하드웨어 또는 소프트웨어 선결제 비용이 없습니다. 최소 약정, 설치 비용 또는 추가 비용 없이 사용한 리소스에 대해서만 비용을 지불하면 됩니다. 서비스와 관련된 요금 및 비용에 대한 내용은 Amazon FSx for Lustre 요금을 참조하세요.

Amazon FSx for Lustre 포럼

Amazon FSx for Lustre를 사용하는 동안 문제가 발생하는 경우 포럼을 확인합니다.

Amazon FSx for Lustre를 처음 사용하시나요?

Amazon FSx for Lustre를 처음 사용한다면, 다음 섹션을 순서대로 읽어보기를 권장합니다.

1. 첫 번째 Amazon FSx for Lustre 파일 시스템을 만들 준비가 되었으면 Amazon FSx for Lustre 시작하기 섹션을 참조하세요.
2. 성능에 대한 자세한 내용은 Amazon FSx for Lustre 성능 섹션을 참조하세요.
3. 파일 시스템을 Amazon S3 버킷 데이터 리포지토리에 연결하는 방법에 대한 자세한 내용은 Amazon FSx for Lustre에서 데이터 리포지토리 사용 섹션을 참조하세요.
4. Amazon FSx for Lustre 보안 세부 정보는 FSx for Lustre 보안 섹션을 참조하세요.

5. 처리량 및 파일 시스템 크기를 포함한 Amazon FSx for Lustre의 확장성 제한에 대한 자세한 내용은 할당량 섹션을 참조하세요.
6. Amazon FSx for Lustre API에 대한 자세한 내용은 Amazon FSx for Lustre API 참조를 참조하세요.

Amazon FSx for Lustre 설정

Amazon FSx for Lustre를 처음 사용하는 경우 먼저 Amazon Web Services 가입 섹션에 있는 작업을 완료해야 합니다. 시작하기 자습서를 완료하려면 파일 시스템에 연결할 Amazon S3 버킷에 Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가에 나열된 권한이 있는지 확인합니다.

주제

- Amazon Web Services 가입
- Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가
- FSx for Lustre가 연결된 S3 버킷에 대한 액세스를 확인하는 방법
- 다음 단계

Amazon Web Services 가입

설정하려면 다음 작업을 완료하십시오. AWS

1. 등록하십시오. AWS 계정
2. 관리자 액세스 권한이 있는 사용자 생성

등록하십시오. AWS 계정

계정이 없는 경우 다음 단계를 완료하여 계정을 만드세요. AWS 계정

가입하려면 AWS 계정

1. https://portal.aws.amazon.com/billing/signup을 여세요.
2. 온라인 지시 사항을 따르세요.

   등록 절차 중에는 전화를 받고 키패드로 인증 코드를 입력하는 과정이 있습니다.

   에 AWS 계정가입하면 AWS 계정 루트 사용자a가 생성됩니다. 루트 사용자에게는 계정의 모든 AWS 서비스 및 리소스 액세스 권한이 있습니다. 보안 모범 사례는 사용자에게 관리 액세스 권한을 할당하고, 루트 사용자만 사용하여 루트 사용자 액세스 권한이 필요한 작업을 수행하는 것입니다.

AWS 가입 절차가 완료된 후 확인 이메일을 보냅니다. 언제든지 https://aws.amazon.com/으로 가서 내 계정(My Account)을 선택하여 현재 계정 활동을 보고 계정을 관리할 수 있습니다.

관리자 액세스 권한이 있는 사용자 생성

등록한 AWS 계정후에는 일상적인 작업에 루트 사용자를 사용하지 않도록 관리 사용자를 보호하고 AWS IAM Identity Center활성화하고 생성하십시오 AWS 계정 루트 사용자.

보안을 유지하세요. AWS 계정 루트 사용자

1. 루트 사용자를 선택하고 AWS 계정 이메일 주소를 입력하여 계정 AWS Management Console소유자로 로그인합니다. 다음 페이지에서 비밀번호를 입력합니다.

   루트 사용자를 사용하여 로그인하는 데 도움이 필요하면AWS 로그인 사용 설명서의 루트 사용자로 로그인을 참조하세요.
2. 루트 사용자의 다중 인증(MFA)을 활성화합니다.

   지침은 IAM 사용 설명서의 AWS 계정 루트 사용자 (콘솔) 에 대한 가상 MFA 디바이스 활성화를 참조하십시오.

관리자 액세스 권한이 있는 사용자 생성

1. IAM Identity Center를 활성화합니다.

   지침은 AWS IAM Identity Center 사용 설명서의 AWS IAM Identity Center설정을 참조하세요.
2. IAM Identity Center에서 사용자에게 관리 액세스 권한을 부여합니다.

   를 ID 소스로 사용하는 방법에 대한 자습서는 사용 설명서의 기본값으로 IAM Identity Center 디렉터리사용자 액세스 구성을 참조하십시오. IAM Identity Center 디렉터리 AWS IAM Identity Center

관리 액세스 권한이 있는 사용자로 로그인

- IAM IDentity Center 사용자로 로그인하려면 IAM IDentity Center 사용자를 생성할 때 이메일 주소로 전송된 로그인 URL을 사용합니다.

  IAM Identity Center 사용자를 사용하여 로그인하는 데 도움이 필요하면 사용 설명서의 AWS 액세스 포털에 로그인을 참조하십시오.AWS 로그인

추가 사용자에게 액세스 권한 할당

1. IAM Identity Center에서 최소 권한 적용 모범 사례를 따르는 권한 세트를 생성합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Create a permission set를 참조하세요.
2. 사용자를 그룹에 할당하고, 그룹에 Single Sign-On 액세스 권한을 할당합니다.

   지침은AWS IAM Identity Center 사용 설명서의 Add groups를 참조하세요.

Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가

Amazon FSx for Lustre는 Amazon S3와 긴밀하게 통합되어 있습니다. 이러한 통합으로 FSx for Lustre 파일 시스템에 액세스하는 애플리케이션이 연결된 Amazon S3 버킷에 저장된 객체에도 원활하게 액세스할 수 있습니다. 자세한 정보는 Amazon FSx for Lustre에서 데이터 리포지토리 사용을 참조하세요.

데이터 리포지토리를 사용하려면 먼저 관리자 사용자의 계정과 연결된 역할에서 Amazon FSx for Lustre에 특정 IAM 권한을 허용해야 합니다.

콘솔을 사용해 역할의 인라인 정책 포함

1. 에 AWS Management Console 로그인하고 https://console.aws.amazon.com//iam/ 에서 IAM 콘솔을 엽니다.
2. 탐색 창에서 역할을 선택합니다.
3. 목록에서 정책을 삽입할 그룹, 역할 이름을 선택합니다.
4. 권한 탭을 선택합니다.
5. 페이지의 하단으로 스크롤하고 인라인 정책 추가를 선택합니다.

   > **Note**
   >
   > IAM의 서비스 연결 역할에는 인라인 정책을 포함할 수 없습니다. 링크된 서비스가 역할 권한을 수정할 수 있는지 여부를 결정하기 때문에 서비스 콘솔이나 API 또는 AWS CLI에서 정책을 추가할 수 있습니다. 서비스에 대한 서비스 연결 역할 설명서를 보려면 IAM과 함께 작동하는AWS 서비스를 참조하고 해당 서비스의 서비스 연결 역할 열에서 예를 선택합니다.
6. 시각적 편집기를 사용하여 정책 만들기를 선택합니다.
7. 다음 정책 설명을 추가합니다.

```
{
    "Version": "2012-10-17",
    "Statement": {
        "Effect": "Allow",
        "Action": [
            "iam:CreateServiceLinkedRole",
            "iam:AttachRolePolicy",
            "iam:PutRolePolicy"
        ],
        "Resource": "arn:aws:iam::*:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/*"
    }
}
```

인라인 정책을 생성하면 이 정책이 역할에 자동으로 포함됩니다. 서비스 연결 역할에 대한 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요.

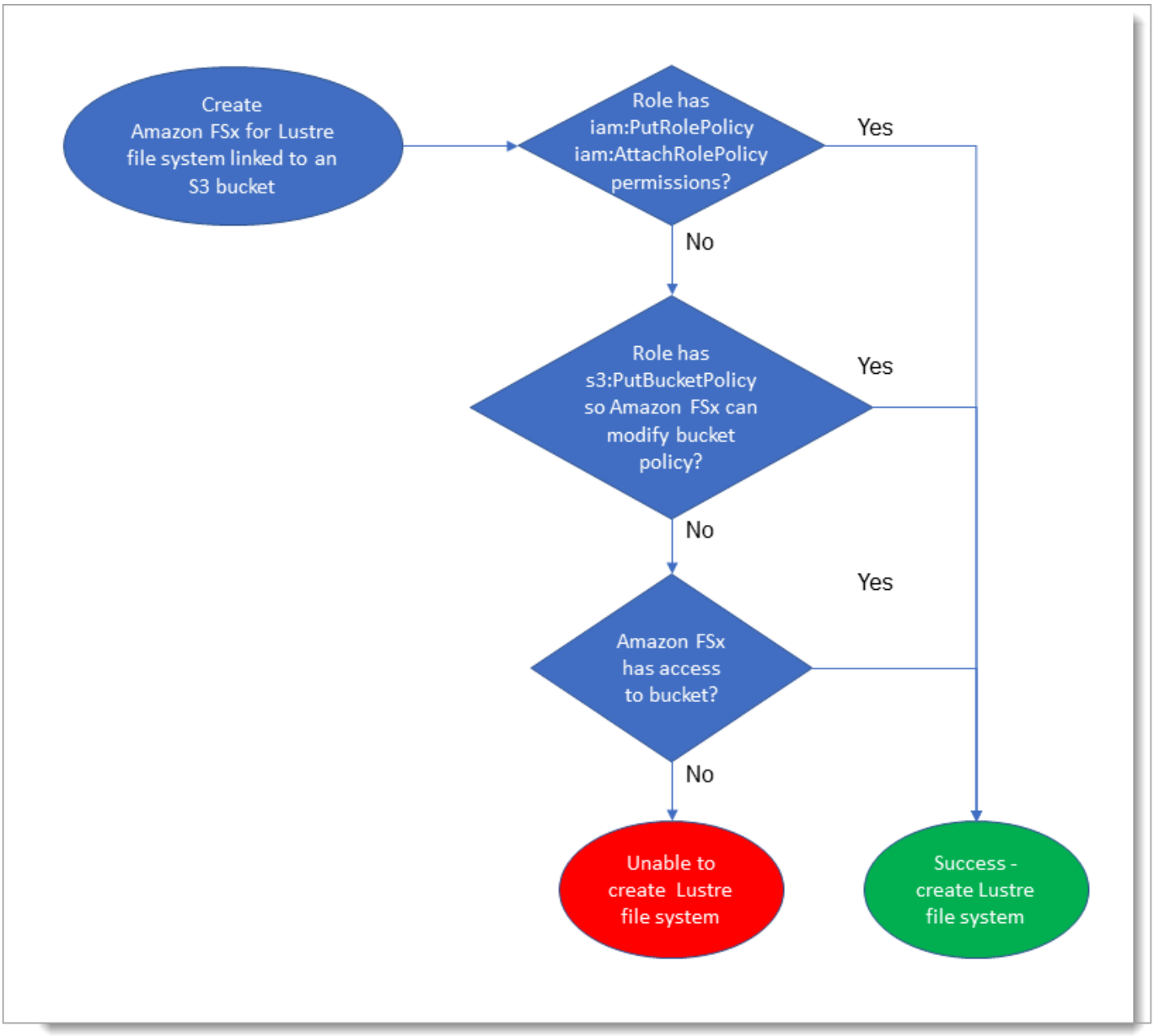
FSx for Lustre가 연결된 S3 버킷에 대한 액세스를 확인하는 방법

FSx for Lustre 파일 시스템을 생성하는 데 사용하는 IAM 역할에 `iam:AttachRolePolicy` 및 `iam:PutRolePolicy` 권한이 없는 경우 Amazon FSx는 S3 버킷 정책을 업데이트할 수 있는지 여부를 확인합니다. Amazon FSx는 IAM 역할에 `s3:PutBucketPolicy` 권한이 포함된 경우 Amazon FSx 파일 시스템이 S3 버킷으로 데이터를 가져오거나 내보낼 수 있도록 버킷 정책을 업데이트할 수 있습니다. 버킷 정책을 수정할 수 있는 경우 Amazon FSx는 버킷 정책에 다음 권한을 추가합니다.

- `s3:AbortMultipartUpload`
- `s3:DeleteObject`
- `s3:PutObject`
- `s3:Get*`
- `s3:List*`
- `s3:PutBucketNotification`
- `s3:PutBucketPolicy`
- `s3:DeleteBucketPolicy`

Amazon FSx가 버킷 정책을 수정할 수 없는 경우 기존 버킷 정책이 Amazon FSx에 버킷에 대한 액세스 권한을 부여하는지 확인합니다.

이러한 옵션이 모두 실패하면 파일 시스템 생성 요청이 실패합니다. 다음 다이어그램은 파일 시스템이 연결될 S3 버킷에 액세스할 수 있는지 여부를 결정할 때 Amazon FSx가 따르는 검사를 보여줍니다.



다음 단계

FSx for Lustre를 사용하기 시작하려면 Amazon FSx for Lustre 리소스 생성 지침에 대한 Amazon FSx for Lustre 시작하기 섹션을 참조하세요.

Amazon FSx for Lustre 시작하기

다음에서는 Amazon FSx for Lustre 사용하는 방법을 알아봅니다. 다음 단계에서 Amazon FSx for Lustre 파일 시스템을 생성하고 컴퓨팅 인스턴스에서 액세스하는 과정을 안내합니다. 선택적으로 Amazon FSx for Lustre 파일 시스템을 사용하여 파일 기반 애플리케이션으로 Amazon S3 버킷의 데이터를 처리하는 방법을 보여줍니다.

이 시작하기 연습에는 다음 단계가 포함됩니다.

주제

- 필수 조건
- FSx for Lustre 파일 시스템 만들기
- Lustre 클라이언트 설치 및 구성
- 파일 시스템을 탑재
- 워크로드 실행
- 리소스 정리

필수 조건

이 시작하기 실습에서는 다음 작업을 수행해야 합니다.

- Amazon FSx for Lustre 파일 시스템과 Amazon EC2 인스턴스를 생성하는 데 필요한 권한을 가진 AWS 계정입니다. 자세한 정보는 Amazon FSx for Lustre 설정을 참조하세요.
- Amazon VPC 보안 그룹을 생성하여 FSx for Lustre 파일 시스템과 연결하고, 파일 시스템을 생성한 후에는 변경하지 마십시오. 자세한 내용은 Amazon FSx 파일 시스템을 위한 보안 그룹 생성하기를 참조하십시오.
- Amazon VPC 서비스를 기반으로 하는 Virtual Private Cloud(VPC)에서 지원되는 Linux 릴리스를 실행하는 Amazon EC2 인스턴스입니다. 이 시작하기 연습에서는 Amazon Linux 2023을 사용하는 것이 좋습니다. 이 EC2 인스턴스에 Lustre 클라이언트를 설치한 다음, EC2 인스턴스에 FSx for Lustre 파일 시스템을 마운트합니다. EC2 인스턴스 생성에 대한 자세한 내용은 Amazon EC2 사용 설명서의 시작하기: 인스턴스 시작 또는 인스턴스 시작을 참조하십시오.

  러스터 클라이언트는 아마존 리눅스, 아마존 리눅스 2, 아마존 리눅스 2023, 센토스 및 레드햇 엔터프라이즈 리눅스 7.7~7.9, 8.2~8.9, 9.0, 9.3, 9.4, 록키 리눅스 8.4~8.9, 9.0, 9.3, 9.4, SUSE 리눅스 엔

터프라이즈 서버 12 SP3, SP4, 우분투 18.04, 20.04를 지원합니다., 그리고 22.04입니다. 자세한 정보는 Lustre 파일 시스템 및 클라이언트 커널 호환성을 참조하세요.

이번 시작하기 연습을 위해 Amazon EC2 인스턴스를 생성할 경우 다음 사항에 유의하세요.

- 기본 VPC에서 인스턴스를 생성하는 것이 좋습니다.
- EC2 인스턴스를 생성할 경우 기본 보안 그룹을 사용하는 것이 좋습니다.

- 각 FSx for Lustre 파일 시스템에는 각 메타데이터 서버 (MDS) 에 대해 하나의 IP 주소가 필요하고 각 스토리지 서버 (OSS) 에 대해 하나의 IP 주소가 필요합니다.
  - 메타데이터 구성이 있는 Persistent_2 파일 시스템의 경우 각 12000 메타데이터 IOPS 값에는 파일 시스템이 있는 서브넷 내에 하나의 IP 주소도 필요합니다.
  - 영구 SSD 파일 시스템은 OSS당 2.4TiB의 스토리지가 프로비저닝됩니다.
  - 처리량 용량이 12MB/s/TiB인 영구 HDD 파일 시스템에는 OSS당 6TiB의 스토리지가 프로비저닝됩니다.
  - 처리량 용량이 40MB/s/TiB인 영구 HDD 파일 시스템에는 OSS당 1.8TiB의 스토리지가 프로비저닝됩니다.
  - 스크래치_2 파일 시스템은 OSS당 2.4TiB의 스토리지가 프로비저닝됩니다.
  - 스크래치_1 파일 시스템은 OSS당 3.6TiB의 스토리지가 프로비저닝됩니다.
- 워크로드가 처리할 데이터를 저장하는 Amazon S3 버킷입니다. S3 버킷은 FSx for Lustre 파일 시스템을 위한 연결된 내구성 있는 데이터 리포지토리가 됩니다.
- 생성하려는 Amazon FSx for Lustre 파일 시스템 유형(스크래치 파일 시스템 또는 영구)을 확인합니다. 자세한 정보는 FSx for Lustre의 파일 시스템 배포 옵션 을 참조하세요.

FSx for Lustre 파일 시스템 만들기

그 다음에는 콘솔에서 파일 시스템을 생성합니다.

파일 시스템 생성

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템 생성을 선택하여 파일 시스템 생성 마법사를 시작합니다.
3. FSx for Lustre를 선택한 후 다음을 선택하여 파일 시스템 생성 페이지를 표시합니다.
4. 파일 시스템 세부 정보 섹션에 정보를 입력합니다.

   - 파일 시스템 이름 선택 사항에는 파일 시스템의 이름을 입력합니다. 최대 256개의 유니코드 문자, 공백 및 숫자와 특수 문자 + - =. _: /를 사용할 수 있습니다.

- 배포 및 스토리지 유형에서 다음 옵션 중 하나를 선택합니다.

  SSD 스토리지는 일반적으로 작고 무작위 파일 작업이 필요한 지연 시간이 짧고 IOPS 집약적인 워크로드를 제공합니다. HDD 스토리지는 일반적으로 대용량의 순차적 파일 작업이 필요한 처리량 집약적인 워크로드를 제공합니다.

  스토리지에 대한 자세한 내용은 다양한 스토리지 옵션 섹션을 참조하세요.

  배포 유형에 대한 자세한 내용은 FSx for Lustre 파일 시스템 배포 옵션 섹션을 참조하세요.

  전송 중 데이터 암호화를 사용할 수 AWS 리전 있는 위치에 대한 자세한 내용은 을 참조하십시오전송 중 데이터 암호화.

  - 장기 스토리지와 최고 수준의 IOPS/처리량이 필요한 지연 시간에 민감한 워크로드에는 영구, SSD 배포 유형을 선택합니다. 파일 서버는 가용성이 높고, 데이터는 파일 시스템의 가용 영역 내에 자동으로 복제되며, 전송 데이터 암호화를 지원합니다. 영구, SSD는 최신 영구 파일 시스템인 Persistent 2를 사용합니다.
  - 장기 스토리지와 지연 시간에 민감하지 않은 처리량 중심의 워크로드에는 영구, HDD 배포 유형을 선택합니다. 파일 서버는 가용성이 높고, 데이터는 파일 시스템의 가용 영역(AZ) 내에 자동으로 복제되며, 지원되는 인스턴스 유형에 대한 전송 중 데이터 암호화를 지원합니다. 퍼시스턴트, HDD는 퍼시스턴트 1 배포 유형을 사용합니다.

    SSD 캐시를 선택하면 HDD 스토리지 용량의 20%에 해당하는 크기의 SSD 캐시를 생성하여 밀리초 미만의 지연 시간과 자주 액세스하는 파일에 더 높은 IOPS를 제공할 수 있습니다.
  - 데이터의 임시 저장 및 단기 처리를 위해 스크래치, SSD 배포 유형을 선택합니다. 스크래치, SSD는 Scratch 2 파일 시스템을 사용하며 전송 중 데이터 암호화를 제공합니다.
- 파일 시스템에 사용할 스토리지 단위당 처리량을 선택합니다. 이 옵션은 영구 배포 유형에만 유효합니다.

  스토리지 유닛당 처리량은 프로비저닝된 각 1테비바이트(TiB)의 각 1테비바이트(TiB)에 대한 읽기 및 쓰기 처리량을 MB/s/TiB 단위로 나타냅니다. 프로비저닝한 처리량에 대해 비용을 지불합니다.

  - 영구 SSD 스토리지의 경우 125, 250, 500 또는 1,000MB/s/TiB 중 하나를 선택합니다.
  - 영구 SSD 스토리지의 경우 12 또는 40MB/s/TiB 중 하나를 선택합니다.

  파일 시스템을 생성한 후 필요에 따라 스토리지 단위당 처리량을 늘리거나 줄일 수 있습니다. 자세한 내용은 처리량 용량 관리 섹션을 참조하세요.
- 스토리지 용량에서 파일 시스템의 스토리지 용량을 TiB 단위로 설정합니다.

- 영구, SSD 배포 유형의 경우 이 값을 1.2TiB, 2.4TiB 또는 2.4 TiB 만큼의 증분 단위로 설정합니다.
- 영구, HDD 배포 유형의 경우 이 값은 12MB/s/TiB 파일 시스템의 경우 6.0TiB씩 증가하고 40MB/s/TiB 파일 시스템의 경우 1.8TiB씩 증가할 수 있습니다.

파일 시스템을 생성한 후 필요에 따라 스토리지 용량을 늘릴 수 있습니다. 자세한 정보는 스토리지 용량 관리을 참조하세요.

- 메타데이터 구성의 경우 파일 시스템의 메타데이터 IOPS 수를 프로비저닝하는 두 가지 옵션이 있습니다.
  - Amazon FSx가 파일 시스템의 스토리지 용량을 기반으로 파일 시스템의 메타데이터 IOPS를 자동으로 프로비저닝하고 확장하도록 하려면 자동 (기본값) 을 선택합니다.
  - 파일 시스템에 프로비저닝할 메타데이터 IOPS 수를 지정하려면 사용자 프로비저닝을 선택합니다. 유효한 값은1500,,3000, 6000 1200012000, 의 배수 (최대) 입니다. 192000

  메타데이터 IOPS에 대한 자세한 내용은 을 참조하십시오. Lustre 메타데이터 성능 구성
- 데이터 압축 유형의 경우 NONE을 선택하여 데이터 압축을 끄거나 LZ4를 선택하여 LZ4 알고리즘을 사용한 데이터 압축을 활성화합니다. 자세한 내용은 Lustre 데이터 압축 섹션을 참조하세요.

모든 FSx for Lustre 파일 시스템은 Amazon FSx 콘솔을 사용하여 생성한 경우 Lustre 버전 2.15를 기반으로 구축됩니다.

5. 네트워크 및 보안 섹션에서 다음 네트워킹 및 보안 그룹 정보를 입력합니다.

- Virtual Private Cloud(VPC)의 경우 파일 시스템에 연결할 VPC를 선택합니다. 이번 시작하기 연습에서는 Amazon EC2 인스턴스용으로 선택한 VPC와 동일한 VPC를 선택합니다.
- VPC 보안 그룹의 경우 VPC의 기본 보안 그룹 ID가 이미 추가되어 있어야 합니다. 기본 보안 그룹을 사용하지 않는 경우, 이 시작 연습에 사용하는 보안 그룹에 다음 인바운드 규칙을 추가했는지 확인합니다.

유형	프로토콜	포트 범위	소스	설명
모든 TCP	TCP	0-65535	사용자 지정 *the_ID_of_this_sec*	인바운드 Lustre 트래픽 규칙

유형	프로토콜	포트 범위	소스	설명
			urity_gro up	

다음 화면 캡처는 인바운드 규칙 편집의 예를 보여줍니다.

> **Important**
>
> 사용 중인 보안 그룹이 에 제공된 구성 지침을 따르는지 확인하십시오. Amazon VPC를 사용한 파일 시스템 액세스 제어 보안 그룹 자체 또는 전체 서브넷 CIDR에서 포트 988 및 1018~1023의 인바운드 트래픽을 허용하도록 보안 그룹을 설정해야 합니다. 이 트래픽은 파일 시스템 호스트가 서로 통신할 수 있도록 하는 데 필요합니다.

- 서브넷의 경우 사용 가능한 서브넷 목록에서 원하는 값을 선택합니다.

6. 암호화 섹션의 경우 생성 중인 파일 시스템 유형에 따라 사용할 수 있는 옵션이 달라집니다.

   - 영구 파일 시스템의 경우 AWS Key Management Service (AWS KMS) 암호화 키를 선택하여 파일 시스템에 저장된 데이터를 암호화할 수 있습니다.
   - 스크래치 파일 시스템의 경우 저장된 데이터는 에서 관리하는 키를 사용하여 암호화됩니다. AWS
   - 스크래치 2 및 영구 파일 시스템의 경우 지원되는 Amazon EC2 인스턴스 유형에서 파일 시스템에 액세스하면 전송 중 데이터가 자동으로 암호화됩니다. 자세한 정보는 전송 중 데이터 암호화을 참조하세요.

7. 데이터 리포지토리 가져오기/내보내기 - 선택 사항 섹션의 경우 파일 시스템을 Amazon S3 데이터 리포지토리에 연결하는 기능은 기본적으로 비활성화되어 있습니다. 이 옵션을 활성화하고 기존 S3 버킷에 데이터 리포지토리를 연결하는 자세한 내용은 파일 시스템을 생성하는 동안 S3 버킷 연결(콘솔) 섹션을 참조하세요.

> **Important**
>
> - 이 옵션을 선택하면 백업도 비활성화되며 파일 시스템을 생성하는 동안에는 백업을 활성화할 수 없습니다.
> - 하나 이상의 Amazon FSx for Lustre 파일 시스템을 Amazon S3 버킷에 연결하는 경우, 연결된 파일 시스템이 모두 삭제될 때까지 Amazon S3 버킷을 삭제하지 마세요.

8. 로깅 - 선택 사항의 경우 로깅이 기본적으로 활성화됩니다. 활성화되면 파일 시스템의 데이터 리포지토리 활동에 대한 실패 및 경고가 Amazon CloudWatch Logs에 기록됩니다. 로깅 버킷 구성에 대한 자세한 내용은 로깅 관리 섹션을 참조하세요.
9. 백업 및 유지 관리 - 선택 사항에서 다음을 수행할 수 있습니다.

   일일 자동 백업의 경우

   - 일일 자동 백업을 비활성화합니다. 이 옵션은 데이터 리포지토리 가져오기/내보내기를 활성화하지 않은 경우 기본적으로 활성화됩니다.
   - 일별 자동 백업 창의 시작 시간을 설정합니다.
   - 자동 보존 기간을 1~35일로 설정합니다.

   자세한 내용은 백업 작업 섹션을 참조하세요.
10. 주간 유지 관리 기간 시작 시간을 설정하거나 기본 설정 없음으로 설정합니다.
11. 루트 스쿼시 (선택 사항) 의 경우 루트 스쿼시는 기본적으로 비활성화됩니다. 루트 스쿼시 활성화 및 구성에 대한 자세한 내용은 을 참조하십시오. 파일 시스템 생성 시 루트 스쿼시를 활성화하려면 (콘솔)
12. 파일 시스템에 적용할 태그를 생성합니다.
13. 다음을 선택하여 파일 시스템 생성 요약 페이지를 표시합니다.
14. Amazon FSx for Lustre 파일 시스템의 설정을 검토하고 파일 시스템 생성을 선택합니다.

이제 파일 시스템을 생성했으니 이후 단계를 위해 정규화된 도메인 이름과 마운트 이름을 기록해 두세요. 캐시 대시보드에서 파일 시스템 이름을 선택한 다음 연결을 선택하면 파일 시스템의 정규화된 도메인 이름과 마운트 이름을 찾을 수 있습니다.

Lustre 클라이언트 설치 및 구성

Amazon EC2 인스턴스에서 Amazon FSx for Lustre 파일 시스템에 액세스하려면 먼저 다음 작업을 수행해야 합니다.

- EC2 인스턴스가 최소 커널 요구 사항을 충족하는지 확인하십시오.
- 필요한 경우 커널을 업데이트하십시오.
- Lustre 클라이언트를 다운로드하고 설치합니다.

커널 버전을 확인하고 Lustre 클라이언트를 다운로드하려면

1. EC2 인스턴스에서 터미널 창을 엽니다.
2. 다음 명령을 실행하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

   ```
   uname -r
   ```

3. 다음 중 하나를 수행합니다.

   - x86 기반 EC2 인스턴스의 경우 명령이 `6.1.79-99.167.amzn2023.x86_64`을 반환하고 Graviton2 기반 EC2 인스턴스의 경우 `6.1.79-99.167.amzn2023.aarch64` 또는 그 이상 버전을 반환하는 경우, 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

     ```
     sudo dnf install -y lustre-client
     ```

   - 명령이 x86 기반 EC2 인스턴스에 대해 `6.1.79-99.167.amzn2023.x86_64`보다 작은 결과를 반환하거나 Graviton2 기반 EC2 인스턴스에 대해 `6.1.79-99.167.amzn2023.aarch64`보다 작은 결과를 반환하는 경우 다음 명령을 실행하여 커널을 업데이트하고 Amazon EC2 인스턴스를 재부팅합니다.

     ```
     sudo dnf -y update kernel && sudo reboot
     ```

     uname -r 명령을 사용해서 커널이 업데이트되었는지 확인합니다. 그런 다음 위에서 설명한 대로 Lustre 클라이언트를 다운로드하고 설치합니다.

다른 Linux 배포판에 Lustre 클라이언트를 설치하는 방법에 대한 자세한 내용은 Lustre 클라이언트 설치 중 섹션을 참조하세요.

파일 시스템을 탑재

파일 시스템을 마운트하려면 탑재 디렉토리 또는 마운트 지점을 만든 다음 파일 시스템을 클라이언트에 마운트하고 클라이언트가 파일 시스템에 액세스할 수 있는지 확인합니다.

파일 시스템 마운트

1. 다음 명령으로 마운트 지점에 대한 디렉터리를 만듭니다.

```
sudo mkdir -p /mnt/fsx
```

2. Amazon FSx for Lustre 파일 시스템을 생성한 디렉터리에 마운트합니다. 다음 명령을 사용하여 다음 항목을 바꿉니다.

   - *file_system_dns_name*을 실제 파일 시스템의 도메인 이름 시스템(DNS) 이름으로 대체합니다.
   - describe-file-systems AWS CLI 명령이나 DescribeFileSystems API 작업을 실행하여 얻을 수 있는 파일 시스템의 마운트 이름으로 *mountname* 바꾸십시오.

```
sudo mount -t lustre -o relatime,flock file_system_dns_name@tcp:/mountname /mnt/fsx
```

   이 명령은 `-o relatime`과 `flock` 같은 두 가지 옵션을 사용하여 파일 시스템을 마운트합니다.

   - `relatime` - `atime` 옵션은 파일에 액세스할 때마다 `atime`(inode 액세스 시간) 데이터를 유지하는 반면, `relatime` 옵션은 `atime` 데이터를 유지하지만 파일에 액세스할 때마다 매번 유지하지는 않습니다. `relatime` 옵션을 활성화하면 `atime` 데이터가 마지막으로 업데이트된 이후 파일이 수정된 경우(`mtime`) 또는 특정 시간 이상 전에 파일을 마지막으로 액세스한 경우(기본값 6시간)에만 `atime` 데이터가 디스크에 기록됩니다. `relatime` 또는 `atime` 옵션 중 하나를 사용하면 파일 릴리스 프로세스가 최적화됩니다.

     > **Note**
     >
     > 정확한 액세스 시간 정확도가 필요한 워크로드의 경우 `atime` 마운트 옵션을 사용하여 마운트할 수 있습니다. 하지만 이렇게 하면 정확한 액세스 시간 값을 유지하는 데 필요한 네트워크 트래픽이 증가하여 워크로드 성능에 영향을 미칠 수 있습니다.
     > 워크로드에 메타데이터 액세스 시간이 필요하지 않은 경우 `noatime` 마운트 옵션을 사용하여 액세스 시간 업데이트를 비활성화하면 성능이 향상될 수 있습니다. 파일 릴리스

나 데이터 유효성 공개와 같은 atime 집중 프로세스는 릴리스에서 정확하지 않을 수 있다는 점에 유의하세요.

- flock - 파일 시스템의 파일 잠금을 활성화합니다. 파일 잠금을 활성화하지 않으려면 flock을 제외한 mount 명령을 사용합니다.

3. 다음 명령을 사용하여 /mnt/fsx 파일 시스템을 마운트한 디렉터리의 내용을 나열하여 마운트 명령이 제대로 실행되었는지 확인합니다.

```
ls /mnt/fsx
import-path  lustre
$
```

다음 df 명령도 사용할 수 있습니다.

```
df
Filesystem                     1K-blocks    Used  Available Use% Mounted on
devtmpf                          1001808       0    1001808   0% /dev
tmpfs                            1019760       0    1019760   0% /dev/shm
tmpfs                            1019760     392    1019368   1% /run
tmpfs                            1019760       0    1019760   0% /sys/fs/cgroup
/dev/xvda1                       8376300 1263180    7113120  16% /
123.456.789.0@tcp:/mountname  3547698816   13824 3547678848   1% /mnt/fsx
tmpfs                             203956       0     203956   0% /run/user/1000
```

결과는 /mnt/fsx에 마운트된 Amazon FSx 파일 시스템을 보여줍니다.

워크로드 실행

이제 파일 시스템이 생성되어 컴퓨팅 인스턴스에 마운트되었으므로 이를 사용하여 고성능 컴퓨팅 워크로드를 실행할 수 있습니다.

데이터 리포지토리 연결을 생성하여 파일 시스템을 Amazon S3 데이터 리포지토리에 연결할 수 있습니다. 자세한 내용은 S3 버킷에 파일 시스템 연결 섹션을 참조하세요.

파일 시스템을 Amazon S3 데이터 리포지토리에 연결한 후 파일 시스템에 기록한 데이터를 언제든지 Amazon S3 버킷으로 다시 내보낼 수 있습니다. 컴퓨팅 인스턴스 중 하나의 터미널에서 다음 명령을 실행하여 Amazon S3 버킷으로 파일을 내보냅니다.

```
sudo lfs hsm_archive file_name
```

폴더 또는 대규모 파일 컬렉션에서 이 명령을 빠르게 실행하는 방법에 대한 자세한 내용은 HSM 명령을 사용하여 파일 내보내기 섹션을 참조하세요.

리소스 정리

이 연습을 완료한 후에는 다음 단계에 따라 리소스를 정리하고 계정을 AWS 보호해야 합니다.

리소스 정리

1. 최종 내보내기를 수행하려면 다음 명령을 실행합니다.

   ```
   nohup find /mnt/fsx -type f -print0 | xargs -0 -n 1 sudo lfs hsm_archive &
   ```

2. Amazon EC2 콘솔에서 인스턴스를 종료합니다. 자세한 내용은 Amazon EC2 사용 설명서의 인스턴스 종료를 참조하십시오.
3. Amazon FSx for Lustre 콘솔에서 다음 절차에 따라 파일 시스템을 삭제합니다.

   a. 탐색 창에서 파일 시스템을 선택합니다.
   b. 대시보드의 파일 시스템 목록에서 삭제하려는 파일 시스템을 선택합니다.
   c. 작업에서 파일 시스템 삭제를 선택합니다.
   d. 표시되는 대화 상자에서 파일 시스템의 최종 백업 생성 여부를 선택합니다. 그런 다음 파일 시스템 ID를 입력하여 삭제를 확인합니다. 파일 시스템 삭제를 선택합니다.
4. 이 연습을 위해 Amazon S3 버킷을 만들었고 내보낸 데이터를 보존하고 싶지 않다면 이제 삭제할 수 있습니다. 자세한 내용은 Amazon Simple Storage Service 콘솔 사용 설명서의 버킷 삭제를 참조하세요.

FSx for Lustre 파일 시스템 배포 옵션

FSx for Lustre는 성능을 극대화하고 병목 현상을 줄이기 위해 여러 네트워크 파일 서버에 데이터를 저장하는 고성능 병렬 파일 시스템을 제공합니다. 이러한 서버에는 디스크가 여러 개 있습니다. 부하를 분산하기 위해 Amazon FSx는 스트라이핑이라는 프로세스를 사용하여 파일 시스템 데이터를 더 작은 청크로 분할하고 디스크와 서버 전체에 분산합니다. FSx for Lustre 데이터 스트라이핑에 대한 자세한 내용은 파일 시스템의 스트라이핑 데이터 섹션을 참조하세요.

Amazon S3에 있는 내구성이 뛰어난 장기 데이터 리포지토리를 FSx for Lustre 고성능 파일 시스템과 연결하는 것이 가장 좋습니다.

이 시나리오에서는 연결된 Amazon S3 데이터 리포지토리에 데이터 세트를 저장합니다. FSx for Lustre 파일 시스템을 생성할 때 이를 S3 데이터 리포지토리에 연결합니다. 이때 S3 버킷의 객체는 FSx 파일 시스템의 파일 및 디렉터리로 나열됩니다. 그러면 Amazon FSx는 Amazon FSx 파일 시스템에서 처음으로 파일에 액세스할 때 S3에서 Lustre 파일 시스템으로 파일 콘텐츠를 자동으로 복사합니다. 컴퓨팅 워크로드가 실행된 후 또는 언제든지 데이터 리포지토리 작업을 사용하여 변경 내용을 S3로 다시 내보낼 수 있습니다. 자세한 내용은 Amazon FSx for Lustre에서 데이터 리포지토리 사용 및 데이터 리포지토리 작업을 사용하여 변경 내용 내보내기 섹션을 참조하세요.

FSx for Lustre의 파일 시스템 배포 옵션

Amazon FSx for Lustre는 스크래치와 퍼시스턴트라는 두 가지 파일 시스템 배포 옵션을 제공합니다.

> **Note**
>
> 두 배포 옵션 모두 솔리드 스테이트 드라이브(SSD) 스토리지를 지원합니다. 그러나 하드 디스크 드라이브(HDD) 스토리지는 영구 배포 유형 중 하나에서만 지원됩니다.

, AWS Command Line Interface (AWS CLI) 또는 Amazon FSx for Lustre API를 사용하여 새 파일 시스템을 생성할 때 파일 시스템 배포 유형을 선택합니다. AWS Management Console자세한 내용은 Amazon FSx API 참조의 CreateFile시스템을 참조하십시오FSx for Lustre 파일 시스템 만들기.

저장 데이터의 암호화는 사용하는 배포 유형과 관계없이 Amazon FSx for Lustre 파일 시스템을 생성할 때 자동으로 활성화됩니다. 스크래치 2 및 영구 파일 시스템은 전송 중 데이터 암호화를 지원하는 Amazon EC2 인스턴스에서 데이터에 액세스할 때 전송 데이터를 자동으로 암호화합니다. 암호화에 대한 자세한 내용은 Amazon FSx for Lustre 의 데이터 암호화 섹션을 참조하세요.

스크래치 파일 시스템

스크래치 파일 시스템은 데이터의 임시 저장 및 단기 처리를 위해 설계되었습니다. 파일 서버에 장애가 발생해도 데이터는 복제되지 않으며 지속되지 않습니다. 스크래치 파일 시스템은 스토리지 용량 TiB당 기본 처리량인 200MBps의 최대 6배에 달하는 높은 버스트 처리량을 제공합니다. 자세한 내용은 파일 시스템 성능 총계 섹션을 참조하세요.

프로세싱이 많은 단기 워크로드를 위한 비용 최적화된 스토리지가 필요한 경우 스크래치 파일 시스템을 사용합니다.

스크래치 파일 시스템에서는 파일 서버에 장애가 발생해도 교체되지 않고 데이터도 복제되지 않습니다. 스크래치 파일 시스템에서 파일 서버나 스토리지 디스크를 사용할 수 없게 되더라도 다른 서버에 저장된 파일은 계속 액세스할 수 있습니다. 클라이언트가 사용할 수 없는 서버나 디스크에 있는 데이터에 액세스하려고 하면 클라이언트에서 즉각적인 I/O 오류가 발생합니다.

다음 표에서는 예제 크기의 스크래치 파일 시스템이 설계된 경우 하루 및 일주일 동안 사용할 수 있는 가용성 또는 내구성을 보여줍니다. 파일 시스템이 클수록 파일 서버와 디스크가 많아지므로 오류가 발생할 확률이 높아집니다.

파일 시스템 크기(TiB)	파일 서버 수	하루 동안의 가용성/내구성	1주일 이상의 가용성/내구성
1.2	2	99.9%	99.4%
2.4	2	99.9%	99.4%
4.8	3	99.8%	99.2%
9.6	5	99.8%	98.6%
50.4	22	99.1%	93.9%

영구 파일 시스템

영구 파일 시스템은 장기 스토리지 및 워크로드를 위해 설계되었습니다. 파일 서버는 가용성이 높으며 파일 시스템이 위치한 동일한 가용 영역 내에 데이터가 자동으로 복제됩니다. 파일 서버에 연결된 데이터 볼륨은 연결된 파일 서버와 독립적으로 복제됩니다.

Amazon FSx는 영구 파일 시스템을 지속적으로 모니터링하여 하드웨어 장애가 있는지 확인하고 장애 발생 시 인프라 구성 요소를 자동으로 교체합니다. 영구 파일 시스템에서는 파일 서버를 사용할 수 없게 되면 장애 발생 후 몇 분 내에 자동으로 교체됩니다. 이 기간 동안 해당 서버의 데이터에 대한 클라이언트 요청은 투명하게 재시도되고 결국 파일 서버가 교체된 후 성공합니다. 영구 파일 시스템에 대한 데이터는 디스크에 복제되고 장애가 발생한 디스크는 자동으로 투명하게 교체됩니다.

장기간 또는 무기한으로 실행되고 가용성 중단에 민감할 수 있는 처리량 중심의 워크로드와 장기 스토리지에는 영구 파일 시스템을 사용합니다.

영구 배포 유형은 전송 중 데이터 암호화를 지원하는 Amazon EC2 인스턴스에서 데이터에 액세스할 때 전송 데이터를 자동으로 암호화합니다.

Amazon FSx for Lustre는 퍼시스턴트_1과 퍼시스턴트_2라는 두 가지 영구 배포 유형을 지원합니다.

퍼시스턴트_2 배포 유형

영구_2는 최신 영구 배포 유형으로, 장기 스토리지가 필요하고 최고 수준의 IOPS 및 처리량을 요구하는 지연 시간에 민감한 워크로드가 있는 사용 사례에 가장 적합합니다. Persistent_2 배포 유형은 Persistent_1 파일 시스템에 비해 유닛 스토리지당 더 높은 수준의 처리량을 지원하며 스토리지 유닛당 처리량 수준 (125, 250, 500 및 1000MB/s/TiB) 의 네 가지 수준을 제공합니다.

Persistent_2 파일 시스템을 생성할 때 메타데이터 구성을 지정하면 파일 시스템의 스토리지 용량과 관계없이 시간이 지남에 따라 메타데이터 성능을 향상시켜 증가하는 성능 요구 사항을 충족하고 더 큰 워크로드를 지원할 수 있습니다.

Amazon FSx 콘솔 및 API를 사용하여 메타데이터 구성 모드로 Persistent_2 파일 시스템을 생성할 수 있습니다. AWS Command Line Interface

퍼시스턴트_1 배포 유형

영구_1 배포 유형은 Lustre 2.10 또는 2.12를 기반으로 구축할 수 있으며 솔리드 스테이트 드라이브(SSD) 및 하드 디스크 드라이브(HDD) 스토리지 유형을 지원합니다. 영구_1 배포 유형은 장기 스토리지가 필요하고 지연 시간에 민감하지 않은 처리량 중심의 워크로드가 있는 사용 사례에 적합합니다.

SSD 스토리지가 있는 영구_1 파일 시스템의 경우 스토리지 단위당 처리량은 테비바이트(TiB) 당 50, 100 또는 200MB/s입니다. HDD 스토리지의 경우 스토리지 유닛당 영구_1 처리량은 TiB당 12 또는 40MB/s입니다.

Persistent_1 배포 유형은 및 AWS CLI Amazon FSx API를 사용해야만 생성할 수 있습니다.

사용 가능한 리전

영구 배포 유형은 다음에서 사용할 수 있습니다. AWS 리전

AWS 리전	영구_1	영구_2
미국 동부(오하이오)	✓	✓
미국 동부(버지니아 북부)	✓	✓
미국 동부 (애틀랜타) 로컬 존		✓ (퍼시스턴트 125 및 250만 해당)
미국 서부(캘리포니아 북부)	✓	
미국 서부 (로스앤젤레스) 로컬 존	✓	
미국 서부(오레곤)	✓	✓
아프리카(케이프타운)	✓	
아시아 태평양(홍콩)	✓	✓
아시아 태평양(하이데라바드)	✓	
아시아 태평양(자카르타)	✓	
아시아 태평양(멜버른)	✓	
아시아 태평양(뭄바이)	✓	✓
아시아 태평양(오사카)	✓	
아시아 태평양(서울)	✓	✓
아시아 태평양(싱가포르)	✓	✓
아시아 태평양(시드니)	✓	✓
아시아 태평양(도쿄)	✓	✓

AWS 리전	영구_1	영구_2
캐나다(중부)	✓	✓
캐나다 서부(캘거리)		✓ (퍼시스턴트 125 및 250만 해당)
유럽(프랑크푸르트)	✓	✓
유럽(아일랜드)	✓	✓
유럽(런던)	✓	✓
유럽(밀라노)	✓	
유럽(파리)	✓	
유럽(스페인)	✓	
유럽(스톡홀름)	✓	✓
유럽(취리히)	✓	
이스라엘(텔아비브)		✓ (퍼시스턴트 125 및 250에만 해당)
중동(바레인)	✓	
중동(UAE)	✓	
남아메리카(상파울루)	✓	
AWS GovCloud (미국 동부)	✓	
AWS GovCloud (미국 서부)	✓	

FSx for Lustre 성능에 대한 자세한 내용은 파일 시스템 성능 총계 섹션을 참조하세요.

Amazon FSx for Lustre에서 데이터 리포지토리 사용

Amazon FSx for Lustre는 빠른 워크로드 처리에 최적화된 고성능 파일 시스템을 제공합니다. 기계 학습, 고성능 컴퓨팅(HPC), 비디오 처리, 금융 모델링, 전자 설계 자동화(EDA)와 같은 워크로드를 지원할 수 있습니다. 이러한 워크로드에서는 일반적으로 데이터 액세스를 위한 확장 가능한 고속 파일 시스템 인터페이스를 사용하여 데이터를 제공해야 합니다. 이러한 워크로드에 사용되는 데이터 세트는 Amazon S3의 장기 데이터 리포지토리에 저장되는 경우가 많습니다. FSx for Lustre는 Amazon S3와 기본적으로 통합되므로 Lustre 파일 시스템으로 데이터 세트를 더 쉽게 처리할 수 있습니다.

> **Note**
>
> 데이터 리포지토리에 연결된 파일 시스템에서는 파일 시스템 백업이 지원되지 않습니다. 자세한 내용은 백업 작업 섹션을 참조하세요.

주제

- 데이터 리포지토리 개요
- 데이터 리포지토리에 대한 POSIX 메타데이터 지원
- S3 버킷에 파일 시스템 연결
- 데이터 리포지토리에서 변경 내용 가져오기
- 데이터 리포지토리로 변경 내용 내보내기
- 데이터 리포지토리 작업
- 파일 릴리스
- 온프레미스 데이터에 Amazon FSx 사용
- 데이터 리포지토리 이벤트 로그
- 이전 배포 유형으로 작업하기

데이터 리포지토리 개요

Amazon FSx for Lustre를 데이터 리포지토리와 함께 사용하는 경우, 자동 가져오기 및 데이터 리포지토리 작업을 사용하여 고성능 파일 시스템에서 대량의 파일 데이터를 수집하고 처리할 수 있습니다. 동시에 데이터 리포지토리 자동 내보내기 또는 내보내기 작업을 사용하여 결과를 데이터 리포지토리에 기록할 수 있습니다. 이러한 기능을 사용하면 데이터 리포지토리에 저장된 최신 데이터를 사용하여 언제든지 워크로드를 다시 시작할 수 있습니다.

> **Note**
>
> FSx for Lustre 2.10 파일 시스템 또는 `Scratch 1` 파일 시스템에서는 데이터 리포지토리 연결, 자동 내보내기 및 다중 데이터 리포지토리 지원을 사용할 수 없습니다.

FSx for Lustre는 Amazon S3와 긴밀하게 통합되어 있습니다. 이러한 통합으로 FSx for Lustre 파일 시스템을 마운트하는 애플리케이션에서 Amazon S3 버킷에 저장된 객체에 원활하게 액세스할 수 있습니다. 또한 AWS 클라우드 의 Amazon EC2 인스턴스에서 컴퓨팅 집약적 워크로드를 실행하고 워크로드가 완료된 후 결과를 데이터 리포지토리로 내보낼 수 있습니다.

Amazon S3 데이터 리포지토리의 객체를 파일 시스템의 파일 및 디렉터리로 액세스하려면 파일 및 디렉터리 메타데이터를 파일 시스템에 로드해야 합니다. 데이터 리포지토리 연결을 생성할 때 연결된 데이터 리포지토리에서 메타데이터를 로드할 수 있습니다.

또한 자동 가져오기 또는 데이터 리포지토리 가져오기 작업을 사용하여 연결된 데이터 리포지토리에서 파일 시스템으로 파일 및 디렉터리 메타데이터를 가져올 수 있습니다. 데이터 리포지토리 연결에 대해 자동 가져오기를 켜면 S3 데이터 리포지토리에서 파일이 생성, 수정 및/또는 삭제될 때 파일 시스템이 파일 메타데이터를 자동으로 가져옵니다. 또는 데이터 리포지토리 가져오기 작업을 사용하여 새 파일 또는 변경된 파일 및 디렉터리의 메타데이터를 가져올 수 있습니다.

> **Note**
>
> 데이터 리포지토리 자동 가져오기 및 가져오기 작업을 파일 시스템에서 동시에 사용할 수 있습니다.

자동 내보내기 또는 데이터 리포지토리 내보내기 작업을 사용하여 파일 시스템의 파일 및 관련 메타데이터를 데이터 리포지토리로 내보낼 수도 있습니다. 데이터 리포지토리 연결에서 자동 내보내기를 켜면 파일이 생성, 수정 또는 삭제될 때 파일 시스템에서 파일 데이터와 메타데이터를 자동으로 내보냅니다. 또는 데이터 리포지토리 내보내기 작업을 사용하여 파일 또는 디렉터리를 내보낼 수도 있습니다. 데이터 리포지토리 내보내기 작업을 사용하면 이러한 마지막 작업 이후 생성되거나 수정된 파일 데이터와 메타데이터를 내보냅니다.

> **Note**
>
> - 파일 시스템에서 데이터 리포지토리 자동 내보내기 및 내보내기 작업을 동시에 사용할 수 없습니다.

- 데이터 리포지토리 연결은 일반 파일, 심볼릭 링크 및 디렉터리만 내보냅니다. 즉, 데이터 리포지토리 자동 내보내기 및 내보내기 작업과 같은 내보내기 프로세스의 일부로 다른 모든 유형의 파일(FIFO 특수 파일, 블록 특수 파일, 문자 특수 파일, 소켓)을 내보낼 수는 없습니다.

또한 FSx for Lustre는 또는 VPN을 사용하여 온프레미스 클라이언트의 데이터를 복사할 수 있도록 함으로써 온프레미스 파일 시스템에서 클라우드 버스팅 워크로드를 지원합니다. AWS Direct Connect

Important

하나 이상의 FSx for Lustre 파일 시스템을 Amazon S3의 데이터 리포지토리에 연결한 경우, 연결된 파일 시스템을 모두 삭제하거나 연결을 해제할 때까지는 Amazon S3 버킷을 삭제하지 마세요.

데이터 리포지토리에 대한 POSIX 메타데이터 지원

Amazon FSx for Lustre는 Amazon S3의 연결된 데이터 리포지토리로 데이터를 가져오거나 Amazon S3의 연결된 데이터 리포지토리에서 데이터를 내보낼 때 파일, 디렉터리 및 심볼릭 링크(symlink)에 대한 휴대용 운영 체제 인터페이스(POSIX) 메타데이터를 자동으로 전송합니다. 파일 시스템의 변경 내용을 연결된 데이터 리포지토리로 내보내는 경우 FSx for Lustre는 POSIX 메타데이터 변경 사항도 S3 객체 메타데이터로 내보냅니다. 즉, 다른 FSx for Lustre 파일 시스템이 S3에서 동일한 파일을 가져오는 경우 파일은 소유권 및 권한을 포함하여 해당 파일 시스템에서 동일한 POSIX 메타데이터를 갖게 됩니다.

FSx for Lustre는 다음과 같이 POSIX 호환 객체 키가 있는 S3 객체만 가져옵니다.

```
mydir/
mydir/myfile1
mydir/mysubdir/
mydir/mysubdir/myfile2.txt
```

FSx for Lustre는 디렉터리와 심볼릭 링크를 S3의 연결된 데이터 리포지토리에 별도의 객체로 저장합니다. 디렉터리의 경우 FSx for Lustre는 다음과 같이 슬래시('/') 로 끝나는 키 이름을 가진 S3 객체를 생성합니다.

- S3 객체 키 `mydir/`은 FSx for Lustre `mydir/` 디렉터리에 매핑됩니다.
- S3 객체 키 `mydir/mysubdir/`은 FSx for Lustre `mydir/mysubdir/` 디렉터리에 매핑됩니다.

심볼릭 링크의 경우 FSx for Lustre는 다음과 같은 Amazon S3 스키마를 사용합니다.

- S3 객체 키 - FSx for Lustre 마운트 디렉터리를 기준으로 한 링크 경로
- S3 객체 데이터 - 이 심볼릭 링크의 대상 경로
- S3 객체 메타데이터 - 심볼릭 링크의 메타데이터

FSx for Lustre는 파일, 디렉터리, 심볼릭 링크에 대한 소유권, 권한, 타임스탬프를 비롯한 POSIX 메타데이터를 다음과 같이 S3 객체에 저장합니다.

- `Content-Type` - 웹 브라우저용 리소스의 미디어 유형을 나타내는 데 사용되는 HTTP 엔티티 헤더입니다.
- `x-amz-meta-file-permissions` - `<octal file type><octal permission mask>` 형식의 파일 유형 및 권한은 Linux stat(2) 매뉴얼 페이지의 `st_mode` 와 일치합니다.

  > **Note**
  >
  > FSx for Lustre는 `setuid` 정보를 가져오거나 보관하지 않습니다.

- `x-amz-meta-file-owner` - 소유자 사용자 ID(UID)는 정수로 표시됩니다.
- `x-amz-meta-file-group` - 그룹 ID(GID)는 정수로 표시됩니다.
- `x-amz-meta-file-atime` - Unix epoch가 시작된 이후 마지막으로 액세스한 시간(나노초) `ns`로 시간 값을 종료합니다. 그렇지 않으면 FSx for Lustre는 값을 밀리초로 해석합니다.
- `x-amz-meta-file-mtime` - Unix epoch가 시작된 이후 마지막으로 수정된 시간(나노초) `ns`로 시간 값을 종료합니다. 그렇지 않으면 FSx for Lustre는 값을 밀리초로 해석합니다.
- `x-amz-meta-user-agent` - 사용자 에이전트로, FSx for Lustre 가져오기 중에는 무시됩니다. 내보내는 동안 FSx for Lustre는 이 값을 `aws-fsx-lustre`로 설정합니다.

연결된 POSIX 권한이 없는 S3에서 객체를 가져올 때 FSx for Lustre가 파일에 할당하는 기본 POSIX 권한은 `755`입니다. 이 권한은 모든 사용자에게 읽기 및 실행 액세스를 허용하고 파일 소유자에게는 쓰기 액세스를 허용합니다.

> **Note**
>
> FSx for Lustre는 S3 객체에 사용자 정의 메타데이터를 보관하지 않습니다.

하드 링크 및 S3로 내보내기

파일 시스템의 DRA에서 자동 내보내기(NEW 및 CHANGED 정책 사용)가 활성화된 경우, DRA에 포함된 각 하드 링크는 각 하드 링크에 대해 별도의 S3 객체로 Amazon S3에 내보내집니다. 파일 시스템에서 하드 링크가 여러 개 있는 파일을 수정하면 파일을 변경할 때 어떤 하드 링크를 사용했는지와 상관없이 S3의 모든 사본이 업데이트됩니다.

데이터 리포지토리 작업(DRT)을 사용하여 하드 링크를 S3로 내보내는 경우 DRT에 지정된 경로에 포함된 각 하드 링크는 각 하드 링크에 대해 별도의 S3 객체로 S3에 내보내집니다. 파일 시스템에서 하드 링크가 여러 개 있는 파일을 수정하면 파일을 변경할 때 어떤 하드 링크를 사용했는지와 상관없이 각 하드 링크를 내보낼 때 S3의 각 사본이 업데이트됩니다.

> **Important**
>
> 새 FSx for Lustre 파일 시스템이 이전에 다른 FSx for Lustre 파일 시스템, AWS DataSync또는 Amazon FSx File Gateway에서 하드 링크를 내보낸 S3 버킷에 연결되면 이후에 하드 링크를 새 파일 시스템에 별도의 파일로 가져옵니다.

하드 링크 및 릴리스된 파일

릴리스된 파일은 메타데이터가 파일 시스템에 있지만 콘텐츠가 S3에만 저장되는 파일입니다. 릴리스된 파일에 대한 자세한 내용은 파일 릴리스 섹션을 참조하세요.

> **Important**
>
> 데이터 리포지토리 연결(DRA)이 있는 파일 시스템에서 하드 링크를 사용하는 경우 다음과 같은 제한이 적용됩니다.
>
> - 여러 개의 하드 링크가 있는 릴리스된 파일을 삭제하고 다시 만들면 모든 하드 링크의 내용을 덮어쓸 수 있습니다.
> - 릴리스된 파일을 삭제하면 데이터 리포지토리 연결 외부에 있는 모든 하드 링크의 콘텐츠가 삭제됩니다.
> - S3 Glacier Flexible Retrieval 또는 S3 Glacier Deep Archive 스토리지 클래스에 있는 릴리스된 파일에 대해 하드 링크를 생성하면 S3에 하드 링크용 새 객체가 생성되지 않습니다.

안내: Amazon S3 버킷으로 객체를 업로드할 때 POSIX 권한 연결

다음 절차는 POSIX 권한을 사용하여 Amazon S3에 객체를 업로드하는 과정을 안내합니다. 이렇게 하면 해당 S3 버킷에 연결된 Amazon FSx 파일 시스템을 생성할 때 POSIX 권한을 가져올 수 있습니다.

POSIX 권한을 가진 객체를 Amazon S3에 업로드

1. 로컬 컴퓨터 또는 시스템에서 다음 예제 명령을 사용하여 S3 버킷에 업로드할 테스트 디렉터리(`s3cptestdir`) 및 파일(`s3cptest.txt`)을 생성합니다.

   ```
   $ mkdir s3cptestdir
   $ echo "S3cp metadata import test" >> s3cptestdir/s3cptest.txt
   $ ls -ld s3cptestdir/ s3cptestdir/s3cptest.txt
   drwxr-xr-x 3 500 500 96 Jan 8 11:29 s3cptestdir/
   -rw-r--r-- 1 500 500 26 Jan 8 11:29 s3cptestdir/s3cptest.txt
   ```

   새로 만든 파일 및 디렉터리의 파일 소유자 사용자 ID(UID)와 그룹 ID(GID)는 500이고 권한은 이전 예와 같습니다.

2. Amazon S3 API를 호출하여 메타데이터 권한이 있는 디렉터리 `s3cptestdir`을 생성합니다. 디렉터리 이름은 뒤에 슬래시(/)를 사용하여 지정해야 합니다. 지원되는 POSIX 메타데이터에 대한 자세한 내용은 데이터 리포지토리에 대한 POSIX 메타데이터 지원 섹션을 참조하세요.

   *bucket_name*을 실제 S3 버킷 이름으로 바꿉니다.

   ```
   $ aws s3api put-object --bucket bucket_name --key s3cptestdir/ --metadata '{"user-
   agent":"aws-fsx-lustre" , \
         "file-atime":"1595002920000000000ns" , "file-owner":"500" , "file-
   permissions":"0100664","file-group":"500" , \
         "file-mtime":"1595002920000000000ns"}'
   ```

3. POSIX 권한이 S3 객체 메타데이터에 태그가 지정되었는지 확인합니다.

   ```
   $ aws s3api head-object --bucket bucket_name --key s3cptestdir/
   {
       "AcceptRanges": "bytes",
       "LastModified": "Fri, 08 Jan 2021 17:32:27 GMT",
       "ContentLength": 0,
       "ETag": "\"d41d8cd98f00b204e9800998ecf8427e\"",
       "VersionId": "bAlhCoWq7aIEjc3R6Myc6UOb8sHHtJkR",
       "ContentType": "binary/octet-stream",
       "Metadata": {
   ```

```
        "user-agent": "aws-fsx-lustre",
        "file-atime": "1595002920000000000ns",
        "file-owner": "500",
        "file-permissions": "0100664",
        "file-group": "500",
        "file-mtime": "1595002920000000000ns"
    }
}
```

4. 컴퓨터에서 메타데이터 권한을 사용하여 S3 버킷으로 테스트 파일(1단계에서 생성)을 업로드합니다.

```
$ aws s3 cp s3cptestdir/s3cptest.txt s3://bucket_name/s3cptestdir/s3cptest.txt \
      --metadata '{"user-agent":"aws-fsx-lustre" , "file-
atime":"1595002920000000000ns" , \
      "file-owner":"500" , "file-permissions":"0100664","file-group":"500" , "file-
mtime":"1595002920000000000ns"}'
```

5. POSIX 권한이 S3 객체 메타데이터에 태그가 지정되었는지 확인합니다.

```
$ aws s3api head-object --bucket bucket_name --key s3cptestdir/s3cptest.txt
{
    "AcceptRanges": "bytes",
    "LastModified": "Fri, 08 Jan 2021 17:33:35 GMT",
    "ContentLength": 26,
    "ETag": "\"eb33f7e1f44a14a8e2f9475ae3fc45d3\"",
    "VersionId": "w9ztRoEhB832m8NC3a_JTlTyIx7Uzql6",
    "ContentType": "text/plain",
    "Metadata": {
        "user-agent": "aws-fsx-lustre",
        "file-atime": "1595002920000000000ns",
        "file-owner": "500",
        "file-permissions": "0100664",
        "file-group": "500",
        "file-mtime": "1595002920000000000ns"
    }
}
```

6. S3 버킷에 연결된 Amazon FSx 파일 시스템에 대한 권한을 확인합니다.

```
$ sudo lfs df -h /fsx
UUID                       bytes        Used   Available Use% Mounted on
3rnxfbmv-MDT0000_UUID       34.4G        6.1M       34.4G   0% /fsx[MDT:0]
```

```
3rnxfbmv-OST0000_UUID        1.1T        4.5M        1.1T   0% /fsx[OST:0]

filesystem_summary:          1.1T        4.5M        1.1T   0% /fsx

$ cd /fsx/s3cptestdir/
$ ls -ld s3cptestdir/
drw-rw-r-- 2 500 500 25600 Jan  8 17:33 s3cptestdir/

$ ls -ld s3cptestdir/s3cptest.txt
-rw-rw-r-- 1 500 500 26 Jan 8 17:33 s3cptestdir/s3cptest.txt
```

`s3cptestdir` 디렉터리와 `s3cptest.txt` 파일 모두 POSIX 권한을 가져왔습니다.

S3 버킷에 파일 시스템 연결

Amazon FSx for Lustre 파일 시스템을 Amazon S3의 데이터 리포지토리에 연결할 수 있습니다. 파일 시스템을 생성할 때 또는 파일 시스템을 생성한 후 언제든지 링크를 생성할 수 있습니다.

파일 시스템의 디렉터리와 S3 버킷 또는 접두사 간의 링크는 데이터 리포지토리 연결(DRA)이라고 부릅니다. FSx for Lustre 파일 시스템에 최대 8개의 데이터 리포지토리 연결을 구성할 수 있습니다. 최대 8개의 DRA 요청을 대기열에 넣을 수 있지만 파일 시스템에 대해 한 번에 하나의 요청만 처리할 수 있습니다. 각 데이터 리포지토리 연결에는 고유한 FSx for Lustre 파일 시스템 디렉터리 및 이와 연결된 고유한 S3 버킷 또는 접두사가 있어야 합니다.

> **Note**
>
> FSx for Lustre 2.10 파일 시스템 또는 `Scratch 1` 파일 시스템에서는 데이터 리포지토리 연결, 자동 내보내기 및 다중 데이터 리포지토리 지원을 사용할 수 없습니다.

파일 시스템의 파일 및 디렉터리로 S3 데이터 리포지토리의 객체에 액세스하려면 파일 및 디렉터리 메타데이터를 파일 시스템에 로드해야 합니다. DRA를 만들 때 연결된 데이터 리포지토리에서 메타데이터를 로드하거나, 나중에 데이터 리포지토리 가져오기 작업을 사용하여 FSx for Lustre 파일 시스템을 사용하여 액세스하려는 파일 및 디렉터리 배치에 대한 메타데이터를 로드하거나, 데이터 리포지토리에서 객체가 추가, 변경 또는 삭제될 때 자동 내보내기를 사용하여 메타데이터를 자동으로 로드할 수 있습니다.

자동 가져오기 전용, 자동 내보내기 전용 또는 두 가지 모두에 대해 데이터 리포지토리 연결을 구성할 수 있습니다. 자동 가져오기와 자동 내보내기로 구성된 데이터 리포지토리 연결은 파일 시스템과 연결

된 S3 버킷 간에 데이터를 양방향으로 전파합니다. S3 데이터 리포지토리의 데이터를 변경하면 FSx for Lustre가 변경 사항을 감지한 다음 변경 사항을 파일 시스템으로 자동으로 가져옵니다. 파일을 생성, 수정 또는 삭제할 때 애플리케이션이 파일 수정을 완료하면 FSx for Lustre는 정의된 변경 사항을 자동으로 Amazon S3에 비동기식으로 내보냅니다.

> **Important**
>
> - 파일 시스템과 S3 버킷 모두에서 동일한 파일을 수정하는 경우 애플리케이션 수준의 조정을 통해 충돌을 방지해야 합니다. FSx for Lustre는 여러 위치에서의 쓰기 충돌을 방지하지 않습니다.
> - 변경할 수 없는 속성으로 표시된 파일의 경우 FSx for Lustre는 FSx for Lustre 파일 시스템과 파일 시스템에 연결된 S3 버킷 간의 변경 사항을 동기화할 수 없습니다. 변경 불가 플래그를 장기간 설정하면 Amazon FSx와 S3 간의 데이터 이동 성능이 저하될 수 있습니다.

데이터 리포지토리 연결을 생성할 때 다음 속성을 구성할 수 있습니다.

- 파일 시스템 경로 - 아래의 지정된 데이터 리포지토리 경로와 매핑될 디렉터리 (예:`/ns1/`) 또는 하위 디렉터리 (예:) 를 가리키는 파일 시스템의 로컬 경로를 입력합니다. `/ns1/subdir/` one-to-one 이름 앞에 슬래시가 있어야 합니다. 두 개의 데이터 리포지토리 연결에 중복되는 파일 시스템 경로가 있을 수 없습니다. 예를 들어, 데이터 리포지토리가 파일 시스템 경로 `/ns1`과 연결된 경우, 또 다른 데이터 리포지토리를 파일 시스템 경로 `/ns1/ns2`와 연결할 수 없습니다.

  > **Note**
  >
  > 슬래시(/)만 파일 시스템 경로로 지정하는 경우 파일 시스템에 하나의 데이터 리포지토리만 연결할 수 있습니다. 파일 시스템과 연결된 첫 번째 데이터 리포지토리의 파일 시스템 경로로 '/'만 지정할 수 있습니다.

- 데이터 리포지토리 경로 - S3 데이터 리포지토리의 경로를 입력합니다. 경로는 `s3://myBucket/myPrefix/` 형식의 S3 버킷 또는 접두사일 수 있습니다. 이 속성은 S3 데이터 리포지토리에서 파일을 가져오거나 내보낼 위치를 지정합니다. 데이터 리포지토리 경로를 제공하지 않으면 FSx for Lustre는 데이터 리포지토리 경로에 후행 “/”를 추가합니다. 예를 들어 데이터 리포지토리 `s3://myBucket/myPrefix` 경로를 제공하면 FSx for Lustre는 이를 `s3://myBucket/myPrefix/`와 같이 해석합니다.

두 개의 데이터 리포지토리 연결에 중복되는 데이터 리포지토리 경로가 있을 수 없습니다. 예를 들어, 데이터 리포지토리가 파일 시스템 경로 `s3://myBucket/myPrefix/`와 연결된 경우, 데이터 리포지토리 경로 `s3://myBucket/myPrefix/mySubPrefix`와 연결되는 또 다른 데이터 리포지토리를 생성할 수 없습니다.

- 리포지토리에서 메타데이터 가져오기 - 이 옵션을 선택하면 데이터 리포지토리 연결을 생성한 후 즉시 전체 데이터 리포지토리에서 메타데이터를 가져올 수 있습니다. 또는 데이터 리포지토리 연결이 생성된 후 언제든지 데이터 리포지토리 가져오기 작업을 실행하여 연결된 데이터 리포지토리의 메타데이터 전체 또는 일부를 파일 시스템으로 로드할 수 있습니다.
- 가져오기 설정 - 연결된 S3 버킷에서 파일 시스템으로 자동으로 가져올 업데이트된 객체(신규, 변경 및 삭제된 객체의 모든 조합)의 유형을 지정하는 가져오기 정책을 선택합니다. 콘솔에서 데이터 리포지토리를 추가하면 자동 가져오기 (신규, 변경, 삭제) 가 기본적으로 활성화되지만, AWS CLI 또는 Amazon FSx API를 사용할 때는 기본적으로 비활성화됩니다.
- 내보내기 설정 - S3 버킷으로 자동으로 내보낼 업데이트된 객체(신규, 변경 및 삭제된 객체의 모든 조합)의 유형을 지정하는 내보내기 정책을 선택합니다. 콘솔에서 데이터 리포지토리를 추가하면 자동 내보내기 (신규, 변경, 삭제) 가 기본적으로 활성화되지만, AWS CLI 또는 Amazon FSx API를 사용할 때는 기본적으로 비활성화됩니다.

파일 시스템 경로 및 데이터 리포지토리 경로 설정은 Amazon FSx의 경로와 S3의 객체 키 간 1:1 매핑을 제공합니다.

연결된 S3 버킷에 대한 리전 및 계정 지원

S3 버킷에 대한 링크를 생성할 때는 다음과 같은 리전 및 계정 지원 제한을 염두에 두세요.

- 자동 내보내기는 크로스 리전 구성을 지원합니다. Amazon FSx 파일 시스템과 연결된 S3 버킷은 AWS 리전 같은 위치에 있거나 다른 위치에 있을 수 있습니다. AWS 리전
- 자동 가져오기는 크로스 리전 구성을 지원하지 않습니다. Amazon FSx 파일 시스템과 연결된 S3 버킷 모두 같은 AWS 리전에 있어야 합니다.
- 자동 내보내기와 자동 가져오기 모두 계정 간 구성을 지원합니다. Amazon FSx 파일 시스템과 연결된 S3 버킷은 AWS 계정 같거나 다른 것에 속할 수 있습니다. AWS 계정

S3 버킷 링크 생성

다음 절차는 및 () 를 사용하여 FSx for Lustre 파일 시스템을 기존 S3 버킷에 대한 데이터 리포지토리 연결을 생성하는 프로세스를 안내합니다. AWS Management Console AWS Command Line

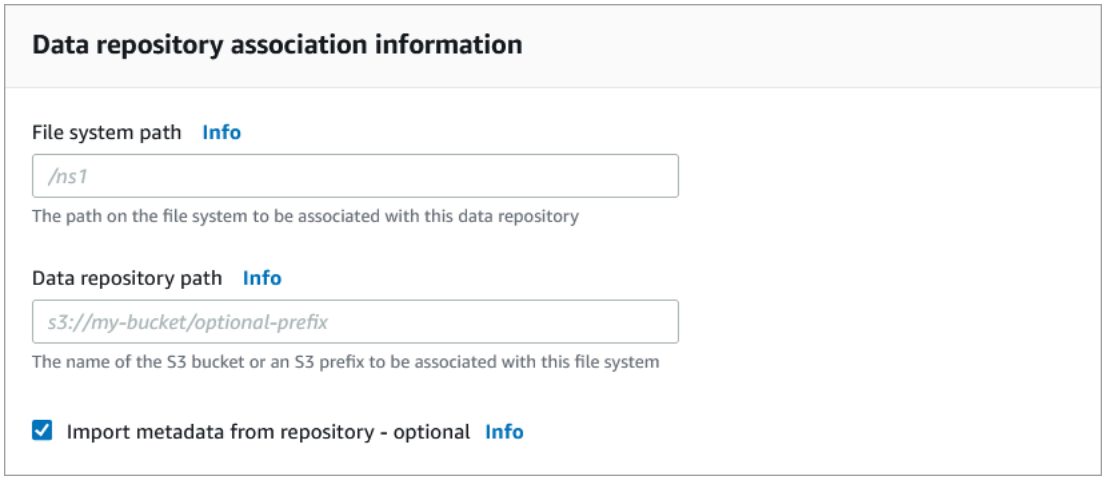
Interface AWS CLI S3 버킷을 파일 시스템에 연결하기 위해 권한을 추가하는 방법에 대한 자세한 내용은 Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가 섹션을 참조하세요.

> **Note**
>
> 데이터 리포지토리는 파일 시스템 백업이 활성화된 파일 시스템에 연결할 수 없습니다. 데이터 리포지토리에 연결하기 전에 백업을 비활성화합니다.

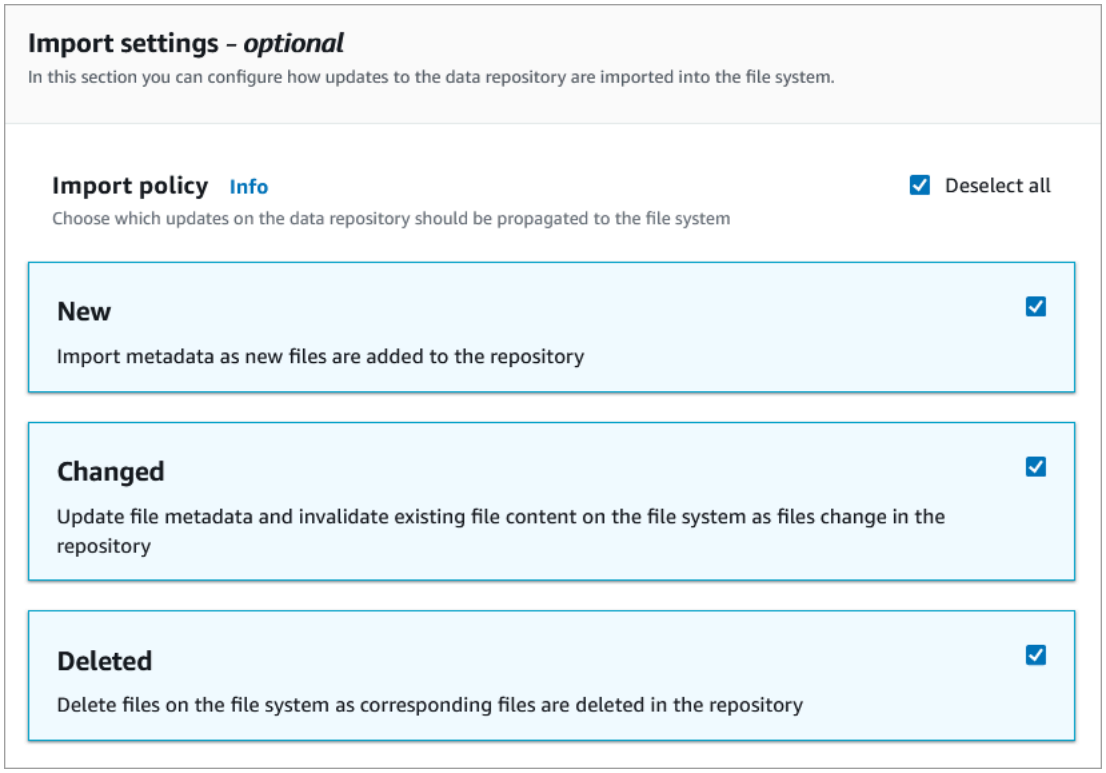
파일 시스템을 생성하는 동안 S3 버킷 연결(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 시작하기 섹션의 FSx for Lustre 파일 시스템 만들기 섹션에 설명된 새 파일 시스템 생성 절차를 따릅니다.
3. 데이터 리포지토리 가져오기/내보내기 - 선택 사항 섹션을 엽니다. 기본적으로 이 기능은 비활성화되어 있습니다.
4. S3에서 데이터 가져오기 및 S3로 데이터 내보내기를 선택합니다.
5. 데이터 리포지토리 연결 정보 대화 상자에서 다음 필드에 대한 정보를 제공합니다.

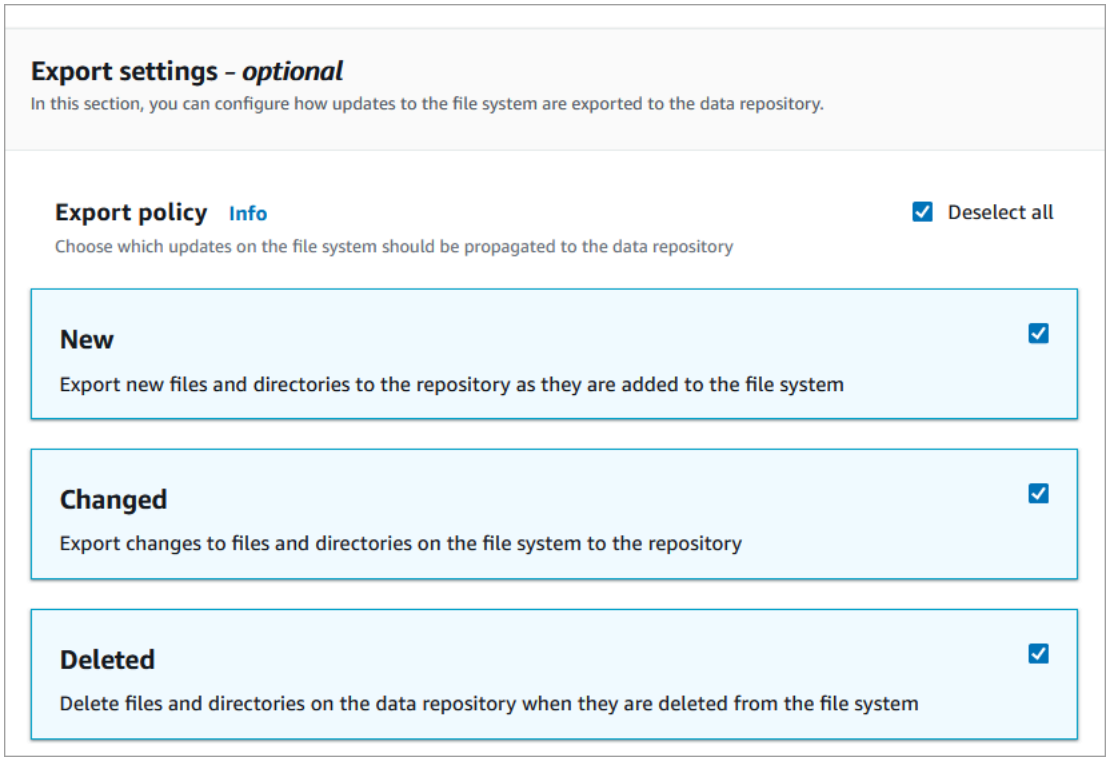
   - 파일 시스템 경로: S3 데이터 리포지토리와 연결할 Amazon FSx 파일 시스템 내의 상위 수준 디렉터리(예: `/ns1`) 또는 하위 디렉터리(예:`/ns1/subdir`)의 이름을 입력합니다. 경로 앞에 슬래시가 있어야 합니다. 두 개의 데이터 리포지토리 연결에 중복되는 파일 시스템 경로가 있을 수 없습니다. 예를 들어, 데이터 리포지토리가 파일 시스템 경로 `/ns1`과 연결된 경우, 파일 시스템 경로 `/ns1/ns2`와 또 다른 데이터 리포지토리를 연결할 수 없습니다. 파일 시스템 경로 설정은 파일 시스템의 모든 데이터 리포지토리 연결에서 고유해야 합니다.
   - 데이터 리포지토리 경로: 파일 시스템에 연결할 기존 S3 버킷 또는 접두사의 경로를 입력합니다(예:`s3://my-bucket/my-prefix/`). 두 개의 데이터 리포지토리 연결에 중복되는 데이터 리포지토리 경로가 있을 수 없습니다. 예를 들어, 데이터 리포지토리가 파일 시스템 경로 `s3://myBucket/myPrefix/`와 연결된 경우, 또 다른 데이터 리포지토리 경로 `s3://myBucket/myPrefix/mySubPrefix`와 연결된 다른 데이터 리포지토리를 생성할 수 없습니다. 데이터 리포지토리 경로 설정은 파일 시스템의 모든 데이터 리포지토리 연결에서 고유해야 합니다.
   - 리포지토리에서 메타데이터 가져오기: 링크를 생성한 후 즉시 메타데이터를 가져오는 데이터 리포지토리 가져오기 작업을 선택적으로 실행하려면 이 속성을 선택합니다.



6. 가져오기 설정 - 선택 사항의 경우 S3 버킷에서 객체를 추가, 변경 또는 삭제할 때 파일 및 디렉터리 목록을 최신 상태로 유지하는 방법을 결정하는 가져오기 정책을 설정합니다. 예를 들어, S3 버킷에서 생성된 새 객체의 메타데이터를 파일 시스템으로 가져오려면 새로 만들기를 선택합니다. 가져오기 정책에 대한 자세한 내용은 S3 버킷에서 업데이트 자동 가져오기 섹션을 참조하세요.



7. 내보내기 정책에서는 파일 시스템에서 객체를 추가, 변경 또는 삭제할 때 연결된 S3 버킷으로 파일을 내보내는 방법을 결정하는 내보내기 정책을 설정합니다. 예를 들어 파일 시스템에서 콘텐츠 또는 메타데이터가 변경된 객체를 내보내려면 변경됨을 선택합니다. 내보내기 정책에 대한 자세한 내용은 업데이트를 S3 버킷으로 자동 내보냅니다. 섹션을 참조하세요.



8. 파일 시스템 생성 마법사의 다음 섹션으로 계속 진행합니다.

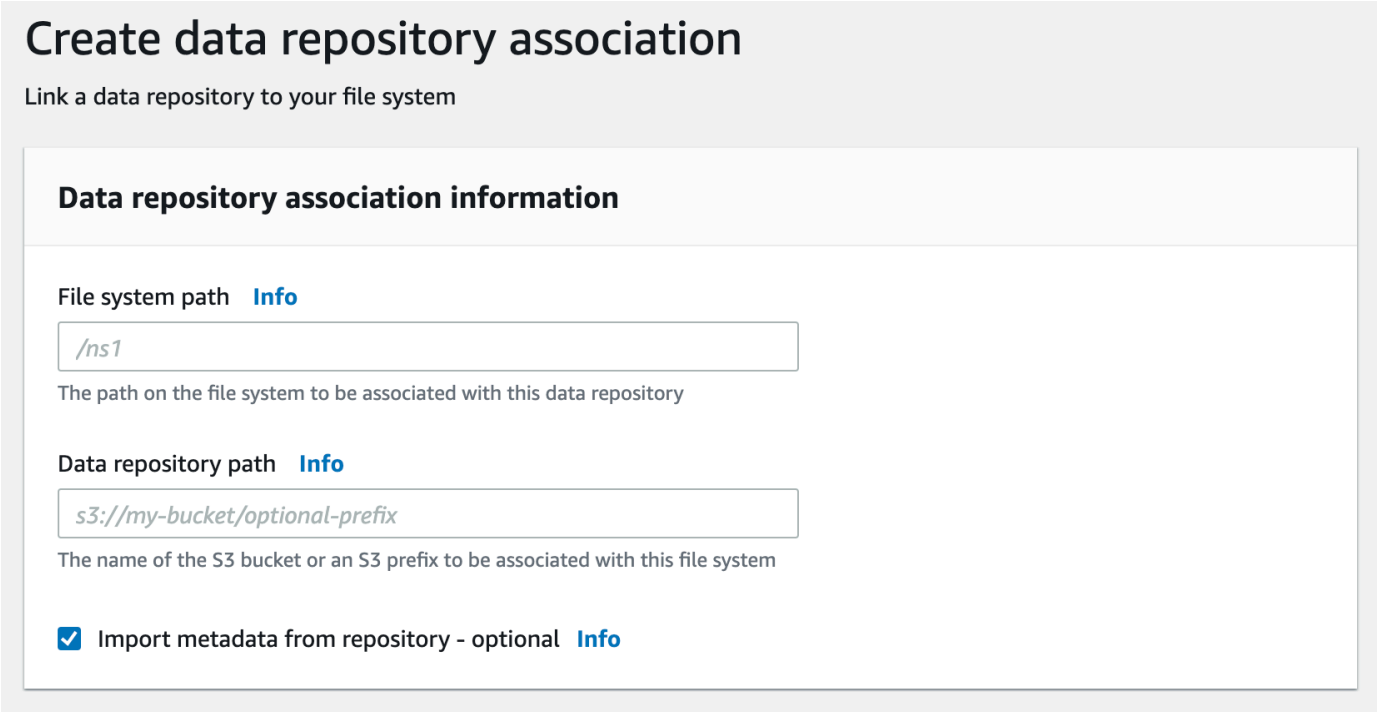
S3 버킷을 기존 파일 시스템에 연결(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템을 선택한 다음 데이터 리포지토리 연결을 생성할 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.
4. 데이터 리포지토리 연결 패널에서 데이터 리포지토리 연결 만들기를 선택합니다.
5. 데이터 리포지토리 연결 정보 대화 상자에서 다음 필드에 대한 정보를 제공합니다.

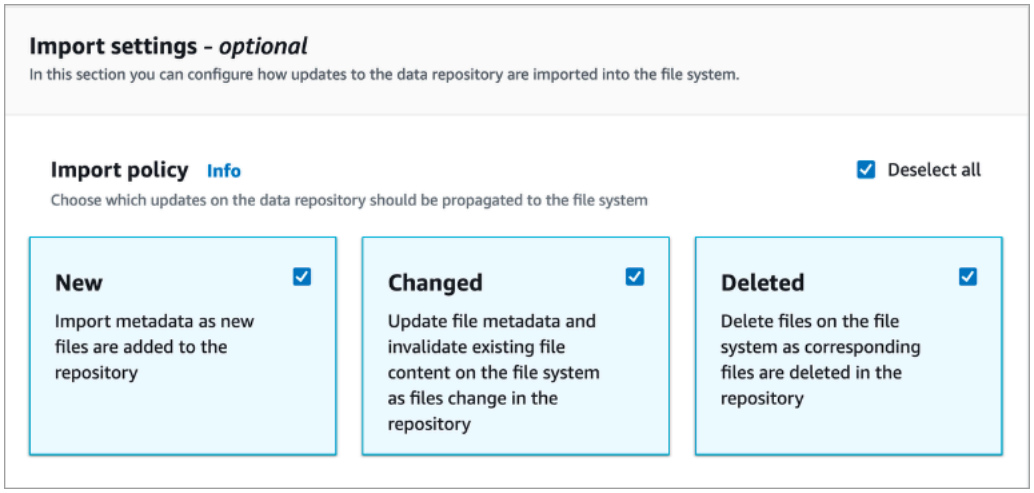
   - 파일 시스템 경로: S3 데이터 리포지토리와 연결할 Amazon FSx 파일 시스템 내의 상위 수준 디렉터리(예: `/ns1`) 또는 하위 디렉터리(예:`/ns1/subdir`)의 이름을 입력합니다. 경로 앞에 슬래시가 있어야 합니다. 두 개의 데이터 리포지토리 연결에 중복되는 파일 시스템 경로가 있을 수 없습니다. 예를 들어, 데이터 리포지토리가 파일 시스템 경로 `/ns1`과 연결된 경우, 파일 시스템 경로 `/ns1/ns2`와 또 다른 데이터 리포지토리를 연결할 수 없습니다. 파일 시스템 경로 설정은 파일 시스템의 모든 데이터 리포지토리 연결에서 고유해야 합니다.
   - 데이터 리포지토리 경로: 파일 시스템에 연결할 기존 S3 버킷 또는 접두사의 경로를 입력합니다(예:`s3://my-bucket/my-prefix/`). 두 개의 데이터 리포지토리 연결에 중복되는 데이

터 리포지토리 경로가 있을 수 없습니다. 예를 들어, 데이터 리포지토리가 파일 시스템 경로 `s3://myBucket/myPrefix/`와 연결된 경우, 데이터 경로로 `s3://myBucket/myPrefix/mySubPrefix`와 또 다른 데이터 리포지토리를 연결할 수 없습니다. 데이터 리포지토리 경로 설정은 파일 시스템의 모든 데이터 리포지토리 연결에서 고유해야 합니다.

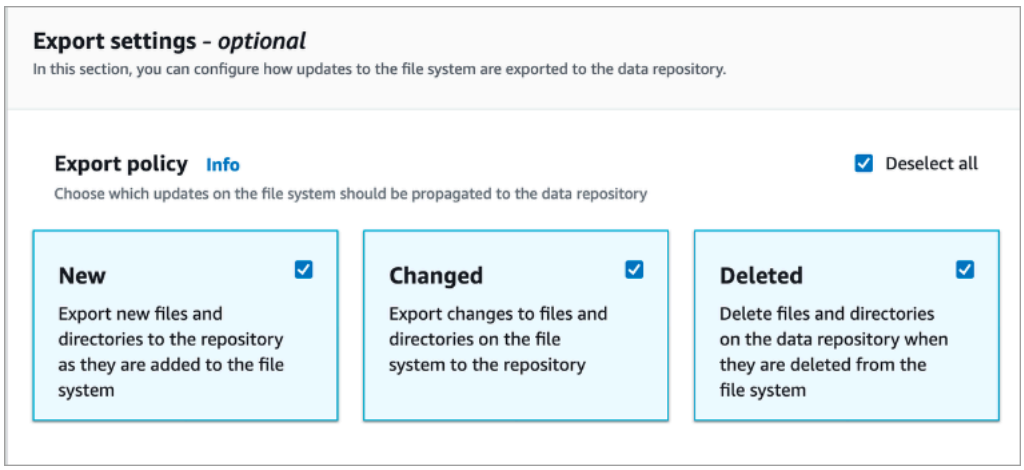
- 리포지토리에서 메타데이터 가져오기: 링크를 생성한 후 즉시 메타데이터를 가져오는 데이터 리포지토리 가져오기 작업을 선택적으로 실행하려면 이 속성을 선택합니다.



6. 가져오기 설정 - 선택 사항의 경우 S3 버킷에서 객체를 추가, 변경 또는 삭제할 때 파일 및 디렉터리 목록을 최신 상태로 유지하는 방법을 결정하는 가져오기 정책을 설정합니다. 예를 들어, S3 버킷에서 생성된 새 객체의 메타데이터를 파일 시스템으로 가져오려면 새로 만들기를 선택합니다. 사용자 정책에 대한 자세한 내용은 S3 버킷에서 업데이트 자동 가져오기 섹션을 참조하세요.



7. 내보내기 정책에서는 파일 시스템에서 객체를 추가, 변경 또는 삭제할 때 연결된 S3 버킷으로 파일을 내보내는 방법을 결정하는 내보내기 정책을 설정합니다. 예를 들어 파일 시스템에서 콘텐츠 또는 메타데이터가 변경된 객체를 내보내려면 변경됨을 선택합니다. 내보내기 정책에 대한 자세한 내용은 업데이트를 S3 버킷으로 자동 내보냅니다. 섹션을 참조하세요.



8. 생성을 선택합니다.

파일 시스템을 S3 버킷에 연결(AWS CLI)

다음 예제는 새 파일이나 변경된 파일을 파일 시스템으로 가져오는 가져오기 정책과 연결된 S3 버킷으로 새 파일, 변경 또는 삭제된 파일을 내보내는 내보내기 정책을 사용하여 Amazon FSx 파일 시스템을 S3 버킷에 연결하는 데이터 리포지토리 연결을 생성합니다.

- 데이터 리포지토리 연결을 생성하려면 다음과 같이 Amazon FSx CLI 명령 `create-data-repository-association`을 사용합니다.

```
$ aws fsx create-data-repository-association \
```

```
      --file-system-id fs-0123456789abcdef0 \
      --file-system-path /ns1/path1/ \
      --data-repository-path s3://mybucket/myprefix/ \
      --s3
"AutoImportPolicy={Events=[NEW,CHANGED,DELETED]},AutoExportPolicy={Events=[NEW,CHANGED,DEL
```

Amazon FSx는 DRA에 대한 JSON 설명을 즉시 반환합니다. DRA는 비동기적으로 생성됩니다.

파일 시스템 생성이 완료되기 전에도 이 명령을 사용하여 데이터 리포지토리 연결을 생성할 수 있습니다. 요청은 대기열에 추가되며 파일 시스템을 사용할 수 있게 되면 데이터 리포지토리 연결이 생성됩니다.

데이터 리포지토리 연결 설정 업데이트

다음 절차에 표시된 대로 AWS Management Console, AWS CLI또는 Amazon FSx API를 사용하여 기존 데이터 리포지토리 연결의 설정을 업데이트할 수 있습니다.

> **Note**
>
> DRA를 생성한 후에는 해당 `File system path` 또는 DRA의 `Data repository path`를 업데이트할 수 없습니다. `File system path` 또는 `Data repository path`를 변경하려면 DRA를 삭제하고 다시 생성해야 합니다.

기존 데이터 리포지토리 연결에 대한 설정 업데이트(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템을 선택한 다음 관리할 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.
4. 데이터 리포지토리 연결 패널에서 변경하려는 데이터 리포지토리 연결을 선택합니다.
5. 업데이트를 선택합니다. 데이터 리포지토리 연결에 대한 편집 대화 상자가 표시됩니다.
6. 가져오기 설정 - 선택 사항의 경우 가져오기 정책을 업데이트할 수 있습니다. 가져오기 정책에 대한 자세한 내용은 S3 버킷에서 업데이트 자동 가져오기 섹션을 참조하세요.
7. 내보내기 설정 - 선택 사항의 경우 내보내기 정책을 업데이트할 수 있습니다. 내보내기 정책에 대한 자세한 내용은 업데이트를 S3 버킷으로 자동 내보냅니다. 섹션을 참조하세요.
8. 업데이트를 선택합니다.

기존 데이터 리포지토리 연결(CLI)에 대한 설정 업데이트

- 데이터 리포지토리 연결을 업데이트하려면 다음과 같이 Amazon FSx CLI 명령 `update-data-repository-association`을 사용합니다.

```
$ aws fsx update-data-repository-association \
      --association-id 'dra-872abab4b4503bfc2' \
      --s3
 "AutoImportPolicy={Events=[NEW,CHANGED,DELETED]},AutoExportPolicy={Events=[NEW,CHANGED,DEL
```

데이터 리포지토리 연결의 가져오기 및 내보내기 정책을 업데이트한 후 Amazon FSx는 업데이트된 데이터 리포지토리 연결의 설명을 JSON으로 반환합니다.

S3 버킷에 대한 연결 삭제

다음 절차는 및 () 를 사용하여 AWS Management Console 기존 Amazon FSx 파일 시스템에서 기존 S3 버킷으로 데이터 리포지토리 연결을 삭제하는 프로세스를 안내합니다. AWS Command Line Interface AWS CLI데이터 리포지토리 연결을 삭제하면 S3 버킷에서 파일 시스템의 연결이 해제됩니다.

파일 시스템에서 S3 버킷으로 연결되는 링크 삭제(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템을 선택한 다음 데이터 리포지토리 연결을 삭제하려는 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.
4. 데이터 리포지토리 연결 창에서 삭제하려는 데이터 리포지토리 연결을 선택합니다.
5. 작업에서 연결 삭제를 선택합니다.
6. (선택 사항) 삭제 대화 상자에서 파일 시스템의 데이터 삭제를 선택하여 데이터 리포지토리 연결에 해당하는 파일 시스템의 데이터를 물리적으로 삭제할 수 있습니다.
7. 삭제를 선택하여 파일 시스템에서 데이터 리포지토리 연결을 제거합니다.

파일 시스템에서 S3 버킷으로 연결되는 링크 삭제(AWS CLI)

다음 예제는 Amazon FSx 파일 시스템을 S3 버킷에 연결하는 데이터 리포지토리 연결을 삭제합니다. `--association-id` 파라미터는 삭제할 데이터 리포지토리 연결의 ID를 지정합니다.

- 데이터 리포지토리 연결을 삭제하려면 다음과 같이 Amazon FSx CLI 명령 `delete-data-repository-association`을 사용합니다.

```
$ aws fsx delete-data-repository-association \
      --association-id dra-872abab4b4503bfc \
      --delete-data-in-file-system false
```

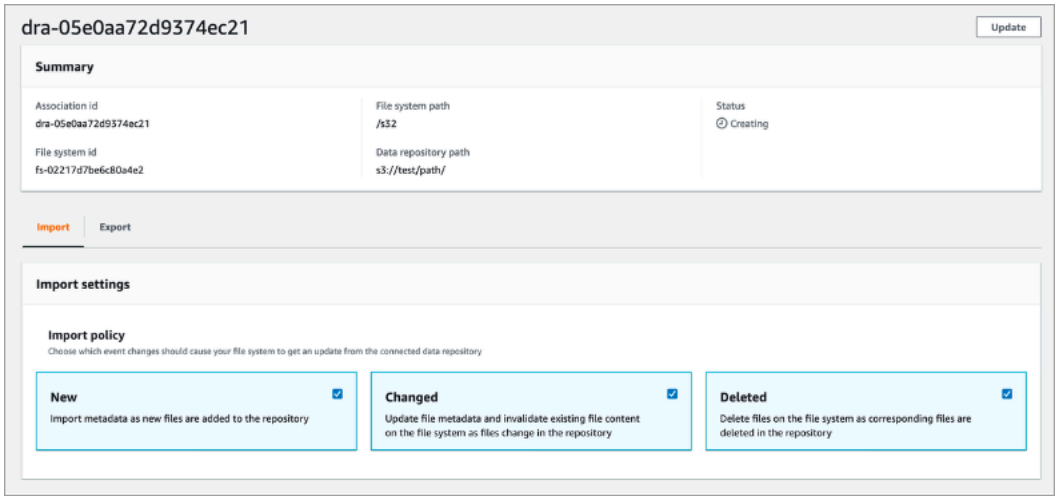
데이터 리포지토리 연결을 삭제한 후 Amazon FSx는 설명을 JSON으로 반환합니다.

데이터 리포지토리 연결 세부 정보 보기

FSx for Lustre 콘솔, 및 API를 사용하여 데이터 리포지토리 연결의 세부 정보를 볼 수 있습니다. AWS CLI세부 정보에는 DRA의 연결 ID, 파일 시스템 경로, 데이터 리포지토리 경로, 가져오기 설정, 내보내기 설정, 상태 및 관련 파일 시스템의 ID가 포함됩니다.

호스트 세부 정보 확인(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템을 선택한 다음 데이터 리포지토리 연결의 세부 정보를 보려는 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.
4. 데이터 리포지토리 연결 창에서 보려는 데이터 리포지토리 연결을 선택합니다. DRA 세부 정보가 표시된 요약 페이지가 표시됩니다.



DRA 세부 정보 확인(CLI)

- 특정 데이터 리포지토리 연결의 세부 정보를 보려면 다음과 같이 Amazon FSx CLI 명령 `describe-data-repository-associations`를 사용합니다.

```
$ aws fsx describe-data-repository-associations \
      --association-ids dra-872abab4b4503bfc2
```

Amazon FSx는 데이터 리포지토리 연결에 대한 설명을 JSON으로 반환합니다.

데이터 리포지토리 연결 수명 주기 상태

데이터 리포지토리 연결 수명 주기 상태는 특정 DRA에 대한 상태 정보를 제공합니다. 데이터 리포지토리 연결에는 다음과 같은 수명 주기 상태가 있을 수 있습니다.

- 생성 - Amazon FSx는 파일 시스템과 연결된 데이터 리포지토리 간의 데이터 리포지토리 연결을 생성합니다. 데이터 리포지토리를 사용할 수 없습니다.
- 사용 가능 - 데이터 리포지토리 연결을 사용할 수 있습니다.
- 업데이트 - 데이터 리포지토리 연결은 고객이 주도한 업데이트를 진행 중이며, 이로 인해 가용성에 영향을 미칠 수 있습니다.
- 삭제 - 고객이 데이터 리포지토리 연결을 삭제하는 중입니다.
- 잘못된 구성 - Amazon FSx는 데이터 리포지토리 연결 구성이 수정될 때까지 S3 버킷에서 업데이트를 자동으로 가져오거나 S3 버킷으로 업데이트를 자동으로 내보낼 수 없습니다.
- 실패 - 데이터 리포지토리 연결이 복구할 수 없는 터미널 상태입니다(예: 파일 시스템 경로가 삭제되거나 S3 버킷이 삭제되었기 때문).

Amazon FSx 콘솔, AWS Command Line Interface및 Amazon FSx API를 사용하여 데이터 리포지토리 연결의 수명 주기 상태를 볼 수 있습니다. 자세한 정보는 데이터 리포지토리 연결 세부 정보 보기을 참조하세요.

서버 측 암호화된 Amazon S3 버킷 사용

FSx for Lustre는 S3에서 관리하는 키 (SSE-S3) 를 사용한 서버 측 암호화와 저장된 키 (SSE-KMS) 를 사용하는 Amazon S3 버킷을 지원합니다. AWS KMS keys AWS Key Management Service

Amazon FSx에서 S3 버킷에 데이터를 쓸 때 데이터를 암호화하도록 하려면 S3 버킷의 기본 암호화를 SSE-S3 또는 SSE-KMS로 설정해야 합니다. 자세한 내용은 Amazon S3 사용자 가이드의 기본 암호화

구성을 참조하세요. S3 버킷에 파일을 작성할 때 Amazon FSx는 S3 버킷의 기본 암호화 정책을 따릅니다.

기본적으로 Amazon FSx는 SSE-S3 방식으로 암호화된 S3 버킷을 지원합니다. Amazon FSx 파일 시스템을 SSE-KMS 암호화를 사용하여 암호화된 S3 버킷에 연결하려면 Amazon FSx가 KMS 키를 사용하여 S3 버킷의 객체를 암호화하고 해독할 수 있도록 허용하는 설명을 고객 관리형 키 정책에 추가해야 합니다.

다음 명령문은 특정 Amazon FSx 파일 시스템이 특정 S3 버킷 *bucket_name*의 객체를 암호화하고 해독할 수 있도록 허용합니다.

```
{
    "Sid": "Allow access through S3 for the FSx SLR to use the KMS key on the objects
 in the given S3 bucket",
    "Effect": "Allow",
    "Principal": {
        "AWS": "arn:aws:iam::aws_account_id:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/AWSServiceRoleForFSxS3Access_fsx_file_system_id"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:CallerAccount": "aws_account_id",
            "kms:ViaService": "s3.bucket-region.amazonaws.com"
        },
        "StringLike": {
            "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket_name/*"
        }
    }
}
```

> **Note**
>
> CMK가 있는 KMS를 사용하여 S3 버킷 키가 활성화된 상태에서 S3 버킷을 암호화하는 경우 다음 예제와 같이 `EncryptionContext`를 객체 ARN이 아닌 버킷 ARN으로 설정합니다.

```
"StringLike": {
    "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket_name"
}
```

다음 정책 설명문은 계정의 모든 Amazon FSx 파일 시스템을 특정 S3 버킷에 연결할 수 있도록 허용합니다.

```
{
    "Sid": "Allow access through S3 for the FSx SLR to use the KMS key on the objects
 in the given S3 bucket",
    "Effect": "Allow",
    "Principal": {
        "AWS": "*"
    },
    "Action": [
        "kms:Encrypt",
        "kms:Decrypt",
        "kms:ReEncrypt*",
        "kms:GenerateDataKey*",
        "kms:DescribeKey"
    ],
    "Resource": "*",
    "Condition": {
        "StringEquals": {
            "kms:CallerAccount": "aws_account_id",
            "kms:ViaService": "s3.bucket-region.amazonaws.com"
        },
        "StringLike": {
            "aws:userid": "*:FSx",
            "kms:EncryptionContext:aws:s3:arn": "arn:aws:s3:::bucket_name/*"
        }
    }
}
```

다른 서버에 있는 암호화된 Amazon S3 버킷에 액세스 AWS 계정

암호화된 Amazon S3 버킷에 연결된 FSx for Lustre 파일 시스템을 생성한 후에는 연결된 S3 버킷에서 데이터를 읽거나 쓰기 전에 S3 버킷을 암호화하는 데 사용되는 KMS 키에 대한

AWSServiceRoleForFSxS3Access_*fs-01234567890* 서비스 연결 역할(SLR) 액세스 권한을 부여해야 합니다. KMS 키에 대한 권한이 이미 있는 IAM 역할을 사용할 수 있습니다.

> **Note**
>
> 이 IAM 역할은 KMS 키/S3 버킷이 속한 계정이 아니라 FSx for Lustre 파일 시스템이 생성된 계정(S3 SLR과 동일한 계정)에 있어야 합니다.

IAM 역할을 사용하여 다음 AWS KMS API를 호출하여 S3 SLR에 대한 권한을 생성하여 SLR이 S3 객체에 대한 권한을 얻도록 합니다. SLR과 연결된 ARN을 찾으려면 파일 시스템 ID를 검색 문자열로 사용하여 IAM 역할을 검색하세요.

```
$ aws kms create-grant --region fs_account_region \
      --key-id arn:aws:kms:s3_bucket_account_region:s3_bucket_account:key/key_id \
      --grantee-principal arn:aws:iam::fs_account_id:role/aws-service-role/s3.data-
source.lustre.fsx.amazonaws.com/AWSServiceRoleForFSxS3Access_file-system-id \
      --operations "Decrypt" "Encrypt" "GenerateDataKey"
 "GenerateDataKeyWithoutPlaintext" "CreateGrant" "DescribeKey" "ReEncryptFrom"
 "ReEncryptTo"
```

서비스 연결 역할에 대한 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요.

데이터 리포지토리에서 변경 내용 가져오기

연결된 데이터 리포지토리에서 Amazon FSx 파일 시스템으로 데이터 및 POSIX 메타데이터에 대한 변경 내용을 가져올 수 있습니다. 관련 POSIX 메타데이터에는 소유권, 권한 및 타임스탬프가 포함됩니다.

파일 시스템에 변경 내용을 가져오려면 다음 방법 중 하나를 사용합니다.

- 연결된 데이터 리포지토리에서 새 파일, 변경 또는 삭제된 파일을 자동으로 가져오도록 파일 시스템을 구성합니다. 자세한 내용은 S3 버킷에서 업데이트 자동 가져오기 섹션을 참조하세요.
- 데이터 리포지토리 연결을 만들 때 메타데이터를 가져오는 옵션을 선택합니다. 이렇게 하면 데이터 리포지토리 연결을 생성한 후 즉시 데이터 리포지토리 가져오기 작업이 시작됩니다.
- 온디맨드 데이터 리포지토리 가져오기 작업을 사용합니다. 자세한 내용은 데이터 리포지토리 작업을 사용하여 변경 내용 가져오기 섹션을 참조하세요.

데이터 리포지토리 자동 가져오기 및 가져오기 작업을 동시에 실행할 수 있습니다.

데이터 리포지토리 연결에 대해 자동 가져오기를 켜면 S3에서 객체가 생성, 수정 또는 삭제될 때 파일 시스템이 파일 메타데이터를 자동으로 업데이트합니다. 데이터 리포지토리 연결을 생성할 때 메타데이터를 가져오는 옵션을 선택하면 파일 시스템이 데이터 리포지토리의 모든 객체에 대한 메타데이터를 가져옵니다. 데이터 리포지토리 가져오기 작업을 사용하여 가져오는 경우 파일 시스템은 마지막 가져오기 이후 생성되거나 수정된 객체의 메타데이터만 가져옵니다.

FSx for Lustre는 애플리케이션이 파일 시스템의 파일에 처음 액세스할 때 데이터 리포지토리에서 파일의 내용을 자동으로 복사하여 파일 시스템으로 로드합니다. 이러한 데이터 이동은 FSx for Lustre에서 관리되며 애플리케이션에 영향을 미치지 않습니다. 이후 이러한 파일에 대한 읽기는 1밀리초 미만의 지연 시간으로 파일 시스템에서 직접 제공됩니다.

전체 파일 시스템 또는 파일 시스템 내의 디렉터리를 미리 로드할 수도 있습니다. 자세한 내용은 파일 시스템에 파일 미리 로드 섹션을 참조하세요. 여러 파일의 사전 로드를 동시에 요청하면 FSx for Lustre는 Amazon S3 데이터 리포지토리에서 파일을 병렬로 로드합니다.

FSx for Lustre는 POSIX 호환 객체 키가 있는 S3 객체만 가져옵니다. 데이터 리포지토리 자동 가져오기 및 가져오기 작업 모두 POSIX 메타데이터를 가져옵니다. 자세한 내용은 데이터 리포지토리에 대한 POSIX 메타데이터 지원 섹션을 참조하세요.

> **Note**
>
> FSx for Lustre는 S3 Glacier Flexible Retrieval 및 S3 Glacier Deep Archive 스토리지 클래스에서 심볼릭 링크(symlink)에 대한 메타데이터 가져오기를 지원하지 않습니다. 심볼릭 링크가 아닌 S3 Glacier Flexiver Flexival 또는 S3 Glacier Deep Archive 객체의 메타데이터를 가져올 수 있습니다. 즉, 올바른 메타데이터를 사용하여 FSx for Lustre 파일 시스템에 inode가 생성됩니다. 그러나 파일 시스템에서 이 데이터를 읽으려면 먼저 S3 Glacier Flexible Retrieval 또는 S3 Glacier Slacier Deep Archive 객체를 복원해야 합니다. S3 Glacier Flexible Retrieval 또는 S3 Glacier Deep Archive 스토리지 클래스에 있는 Amazon S3 객체에서 FSx for Lustre로 파일 데이터를 직접 가져올 수 없습니다.

S3 버킷에서 업데이트 자동 가져오기

객체가 S3 버킷에 추가, 변경 또는 삭제될 때 파일 시스템의 메타데이터를 자동으로 업데이트하도록 FSx for Lustre를 구성할 수 있습니다. FSx for Lustre는 S3의 변경에 따라 파일 및 디렉터리 목록을 생성, 업데이트 또는 삭제합니다. S3 버킷의 변경된 객체가 더 이상 메타데이터를 포함하지 않는 경우 FSx for Lustre는 현재 권한을 포함하여 파일의 현재 메타데이터 값을 유지합니다.

> **Note**
>
> 업데이트를 자동으로 가져오려면 FSx for Lustre 파일 시스템과 연결된 S3 버킷이 같은 AWS 리전 에 위치해야 합니다.

데이터 리포지토리 연결을 생성할 때 자동 가져오기를 구성할 수 있으며, FSx 관리 콘솔, AWS CLI또는 API를 사용하여 언제든지 자동 가져오기 설정을 업데이트할 수 있습니다. AWS

> **Note**
>
> 동일한 데이터 리포지토리 연결에서 자동 가져오기와 자동 내보내기를 모두 구성할 수 있습니다. 이 항목에서는 자동 가져오기 기능에 대해서만 설명합니다.

> **Important**
>
> - 자동 가져오기 정책이 모두 활성화되고 자동 내보내기가 비활성화된 상태에서 S3에서 객체를 수정하면 해당 객체의 내용을 항상 파일 시스템의 해당 파일로 가져옵니다. 파일이 대상 위치에 이미 있는 경우 해당 파일을 덮어씁니다.
> - 모든 자동 가져오기 및 자동 내보내기 정책이 활성화된 상태에서 파일 시스템과 S3에서 파일을 수정하면 파일 시스템의 파일이나 S3의 객체를 다른 사람이 덮어쓸 수 있습니다. 한 위치에서 나중에 편집해도 다른 위치에서 이전에 편집한 내용을 덮어쓴다는 보장은 없습니다. 파일 시스템과 S3 버킷 모두에서 동일한 파일을 수정하는 경우 애플리케이션 수준의 조정을 통해 이러한 충돌을 방지해야 합니다. FSx for Lustre는 여러 위치에서의 쓰기 충돌을 방지하지 않습니다.

가져오기 정책은 연결된 S3 버킷의 콘텐츠가 변경될 때 FSx for Lustre가 파일 시스템을 업데이트하는 방법을 지정합니다. 데이터 리포지토리 연결의 가져오기 정책은 다음 중 하나에 해당될 수 있습니다.

- 신규 - FSx for Lustre는 연결된 S3 데이터 리포지토리에 새 객체가 추가될 때만 파일 및 디렉터리 메타데이터를 자동으로 업데이트합니다.
- 변경됨 - FSx for Lustre는 데이터 리포지토리의 기존 객체가 변경될 때만 파일 및 디렉터리 메타데이터를 자동으로 업데이트합니다.
- 삭제됨 - FSx for Lustre는 데이터 리포지토리의 객체가 삭제될 때만 파일 및 디렉터리 메타데이터를 자동으로 업데이트합니다.

- 신규, 변경됨, 삭제됨의 모든 조합 - FSx for Lustre는 S3 데이터 리포지토리에서 지정된 작업이 발생할 경우 파일 및 디렉터리 메타데이터를 자동으로 업데이트합니다. 예를 들어 객체가 S3 리포지토리에 추가(신규)되거나 제거(삭제됨)될 때 파일 시스템이 업데이트되고, 객체가 변경되면 업데이트되지 않도록 지정할 수 있습니다.
- 정책이 구성되지 않음 - FSx for Lustre는 객체가 S3 데이터 리포지토리에 추가, 변경 또는 삭제될 때 파일 시스템의 파일 및 디렉터리 메타데이터를 업데이트하지 않습니다. 가져오기 정책을 구성하지 않으면 데이터 리포지토리 연결에 대한 자동 가져오기가 비활성화됩니다. 데이터 리포지토리 작업을 사용하여 변경 내용 가져오기에 설명된 대로 여전히 데이터 리포지토리 가져오기 작업을 사용하여 메타데이터 변경 내용을 수동으로 가져올 수 있습니다.

> **Important**
>
> 자동 가져오기는 다음 S3 작업을 연결된 FSx for Lustre 파일 시스템과 동기화하지 않습니다.
>
> - S3 객체 수명 주기 만료를 사용하여 객체 삭제
> - 버전 관리가 활성화된 버킷에서 현재 객체 버전 영구 삭제
> - 버전 관리 활성화 버킷에서 객체 삭제 취소

대부분의 사용 사례에서는 가져오기 정책을 신규, 변경됨, 삭제됨으로 구성하는 것이 좋습니다. 이 정책은 연결된 S3 데이터 리포지토리에서 이루어진 모든 업데이트를 파일 시스템으로 자동으로 가져오도록 합니다.

연결된 S3 데이터 리포지토리의 변경 내용을 기반으로 파일 시스템 파일 및 디렉터리 메타데이터를 업데이트하도록 가져오기 정책을 설정하면 FSx for Lustre는 연결된 S3 버킷에 이벤트 알림 구성을 생성합니다. 이벤트 알림 구성에는 FSx 이름이 지정됩니다. S3 버킷의 FSx 이벤트 알림 구성을 수정하거나 삭제하지 마세요. 수정하거나 삭제하면 업데이트된 파일 및 디렉터리 메타데이터를 파일 시스템으로 자동으로 가져올 수 없습니다.

FSx for Lustre가 연결된 S3 데이터 리포지토리에서 변경된 파일 목록을 업데이트하면 파일이 쓰기 잠겨 있더라도 로컬 파일을 업데이트된 버전으로 덮어씁니다.

FSx for Lustre는 파일 시스템을 업데이트하기 위해 최선을 다합니다. FSx for Lustre는 다음과 같은 상황에서 파일 시스템을 업데이트할 수 없습니다.

- FSx for Lustre에 변경되거나 새 S3 객체를 열 수 있는 권한이 없는 경우. 이 경우 FSx for Lustre는 객체를 건너뛰고 계속합니다. DRA 수명 주기 상태는 영향을 받지 않습니다.

- FSx for Lustre에 버킷 수준 권한이 없는 경우(예: `GetBucketAcl`). 이로 인해 데이터 리포지토리 수명 주기 상태가 잘못 구성될 수 있습니다. 자세한 내용은 데이터 리포지토리 연결 수명 주기 상태 섹션을 참조하세요.
- 연결된 S3 버킷의 FSx 이벤트 알림 구성이 삭제되거나 변경된 경우 이로 인해 데이터 리포지토리 수명 주기 상태가 잘못 구성될 수 있습니다. 자세한 정보는 데이터 리포지토리 연결 수명 주기 상태을 참조하세요.

자동으로 가져올 수 없는 파일이나 디렉터리에 대한 정보를 로깅하려면 CloudWatch 로그 로깅을 켜는 것이 좋습니다. 로그의 경고 및 오류에는 실패 이유에 대한 정보가 포함됩니다. 자세한 내용은 데이터 리포지토리 이벤트 로그 섹션을 참조하세요.

사전 조건

FSx for Lustre가 연결된 S3 버킷에서 새 파일, 변경 또는 삭제된 파일을 자동으로 가져오려면 다음 조건이 필요합니다.

- 파일 시스템과 연결된 S3 버킷은 같은 AWS 리전에 위치합니다.
- S3 버킷에는 잘못 구성된 수명 주기 상태가 없습니다. 자세한 내용은 데이터 리포지토리 연결 수명 주기 상태 섹션을 참조하세요.
- 계정에는 연결된 S3 버킷에서 이벤트 알림을 구성하고 수신하는 데 필요한 권한이 있습니다.

지원되는 파일 변경 유형

FSx for Lustre는 연결된 S3 버킷에서 발생하는 파일 및 디렉터리에 대한 다음과 같은 변경 사항 가져오기를 지원합니다.

- 파일 콘텐츠 변경.
- 파일 또는 디렉터리 메타데이터 변경.
- 심볼릭 링크 대상 또는 메타데이터 변경.
- 파일 및 디렉터리 삭제. 연결된 S3 버킷에서 파일 시스템의 디렉터리에 해당하는 객체(즉, 키 이름이 슬래시로 끝나는 객체)를 삭제하는 경우 FSx for Lustre는 비어 있는 경우에만 파일 시스템에서 해당 디렉터리를 삭제합니다.

가져오기 설정 업데이트

데이터 리포지토리 연결을 생성할 때 연결된 S3 버킷에 대한 파일 시스템의 가져오기 설정을 지정할 수 있습니다. 자세한 내용은 S3 버킷 링크 생성 섹션을 참조하세요.

또한 가져오기 정책을 포함하여 가져오기 설정을 언제든지 업데이트할 수 있습니다. 자세한 내용은 데이터 리포지토리 연결 설정 업데이트 섹션을 참조하세요.

자동 가져오기 모니터링

S3 버킷의 변경 비율이 자동 가져오기에서 이러한 변경 사항을 처리할 수 있는 속도를 초과하는 경우 FSx for Lustre 파일 시스템으로 가져오는 해당 메타데이터 변경 사항이 지연됩니다. 이 경우 `AgeOfOldestQueuedMessage` 지표를 사용하여 자동 가져오기로 처리되기를 기다리는 가장 오래된 변경 사항의 보존 기간을 모니터링할 수 있습니다. 이러한 지표에 대한 자세한 내용은 AutoImport AutoExport 및 지표 섹션을 참조하세요.

메타데이터 변경 사항 가져오기 지연이 14일(`AgeOfOldestQueuedMessage` 지표를 사용하여 측정)을 초과하는 경우, 자동 가져오기로 처리되지 않은 S3 버킷의 변경 사항은 파일 시스템으로 가져오지 않습니다. 또한 데이터 리포지토리 연결 수명 주기가 잘못 구성된 것으로 표시되고 자동 가져오기가 중지됩니다. 자동 내보내기를 활성화한 경우 자동 내보내기는 FSx for Lustre 파일 시스템의 변경 사항을 계속 모니터링합니다. 하지만 추가 변경 사항은 FSx for Lustre 파일 시스템에서 S3로 동기화되지 않습니다.

데이터 리포지토리 연결을 잘못 구성됨 수명 주기 상태에서 사용 가능 수명 주기 상태로 되돌리려면 데이터 리포지토리 연결을 업데이트해야 합니다. update-data-리포지토리 연결 CLI 명령 (또는 해당 API 작업) 을 사용하여 데이터 리포지토리 연결을 업데이트할 수 있습니다. UpdateDataRepositoryAssociation 필요한 유일한 요청 파라미터는 업데이트하려는 데이터 리포지토리 연결의 `AssociationID`입니다.

데이터 리포지토리 연결 수명 주기 상태가 사용 가능으로 변경되면 자동 가져오기(활성화된 경우 자동 내보내기)가 다시 시작됩니다. 다시 시작하면 자동 내보내기가 재개되어 파일 시스템 변경 사항을 S3에 동기화합니다. 가져오지 않았거나 데이터 리포지토리 연결이 잘못 구성된 상태였을 때 가져온 FSx for Lustre 파일 시스템과 S3에 있는 새 객체 및 변경된 객체의 메타데이터를 동기화하려면 데이터 리포지토리 가져오기 작업을 실행합니다. 데이터 리포지토리 가져오기 작업은 S3 버킷의 삭제를 FSx for Lustre 파일 시스템과 동기화하지 않습니다. S3를 파일 시스템과 완전히 동기화(삭제 포함)하려면 파일 시스템을 다시 생성해야 합니다.

메타데이터 변경 사항 가져오기 지연이 14일을 초과하지 않도록 하려면 `AgeOfOldestQueuedMessage` 지표에 경보를 설정하고 `AgeOfOldestQueuedMessage` 지표가 경

보 임계값을 초과할 경우 S3 버킷의 활동을 줄이는 것이 좋습니다. 단일 샤드가 S3로부터 가능한 변경 사항을 지속적으로 전송하는 S3 버킷에 연결된 FSx for Lustre 파일 시스템의 경우, FSx for Lustre 파일 시스템에서 자동 가져오기만 실행되는 경우, 자동 가져오기는 14일 이내에 7시간 동안 S3 변경 사항 백로그를 처리할 수 있습니다.

또한 단일 S3 작업으로 자동 가져오기에서 14일 동안 처리할 수 있는 것보다 더 많은 변경 사항을 생성할 수 있습니다. 이러한 유형의 작업의 예로는 S3로의 AWS Snowball 업로드 및 대규모 삭제가 포함되지만 이에 국한되지는 않습니다. FSx for Lustre 파일 시스템과 동기화하려는 S3 버킷을 대규모로 변경하는 경우, 자동 가져오기 변경 사항이 14일을 초과하지 않도록 하려면 파일 시스템을 삭제하고 S3 변경이 완료되면 다시 생성해야 합니다.

`AgeOfOldestQueuedMessage` 지표가 증가하고 있는 경우 S3 버킷의 `GetRequests`, `PutRequests`, `PostRequests` 및 `DeleteRequests` 지표에서 자동 가져오기로 전송되는 변경 사항의 비율 및/또는 수를 증가시킬 수 있는 활동 변경 사항이 있는지 검토하세요. 사용 가능한 S3 지표에 대한 자세한 내용은 Amazon S3 사용자 가이드의 Amazon S3 모니터링을 참조하세요.

사용 가능한 모든 FSx for Lustre 지표 목록은 아마존을 통한 모니터링 CloudWatch 섹션을 참조하세요.

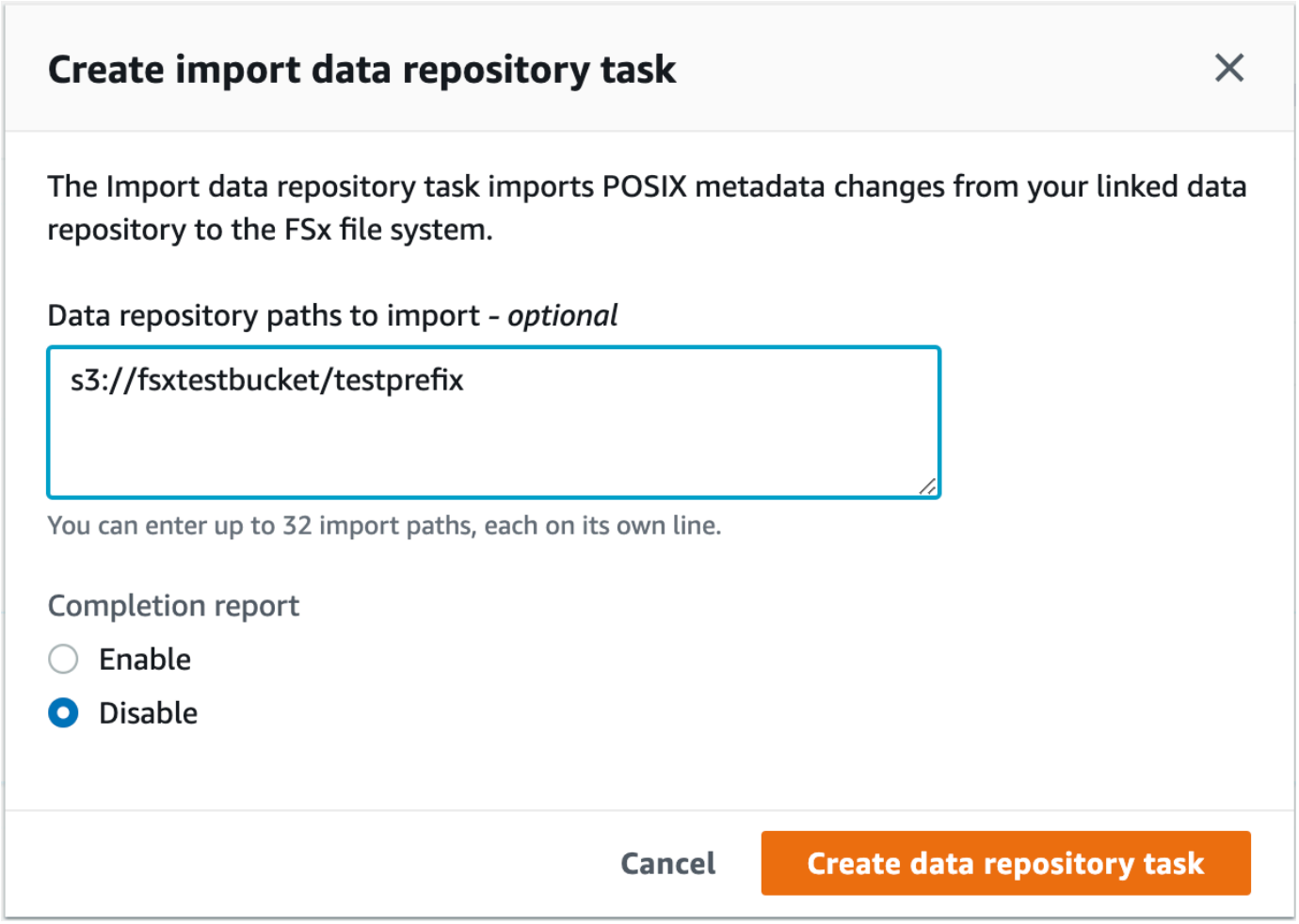
데이터 리포지토리 작업을 사용하여 변경 내용 가져오기

데이터 리포지토리 가져오기 작업은 S3 데이터 리포지토리에서 새로 추가되거나 변경된 객체의 메타데이터를 가져와서 S3 데이터 리포지토리의 모든 새 객체에 대한 새 파일 또는 디렉터리 목록을 생성합니다. 데이터 리포지토리에서 변경된 모든 객체의 경우 해당 파일 또는 디렉터리 목록이 새 메타데이터로 업데이트됩니다. 데이터 리포지토리에서 삭제된 객체에는 아무런 조치가 취해지지 않습니다.

Amazon FSx 콘솔 및 CLI를 사용하여 메타데이터 변경 내용을 가져오려면 다음 절차를 따르세요. 여러 DRA에 대해 하나의 데이터 리포지토리 작업을 사용할 수 있다는 점에 유의하세요.

메타데이터 변경 내용을 가져오기(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 파일 시스템을 선택한 다음 Lustre 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.
4. 데이터 리포지토리 연결 패널에서 가져오기 작업을 만들 데이터 리포지토리 연결을 선택합니다.
5. 작업 메뉴에서 작업 실행을 선택합니다. 파일 시스템이 데이터 리포지토리에 연결되지 않은 경우에는 이 옵션을 사용할 수 없습니다. 가져오기 데이터 리포지토리 만들기 작업 페이지가 표시됩니다.



6. (선택 사항) 가져올 데이터 리포지토리 경로에 해당 디렉터리 또는 파일의 경로를 제공하여 연결된 S3 버킷에서 가져올 디렉터리 또는 파일을 최대 32개까지 지정합니다.

   > **Note**
   >
   > 제공한 경로가 유효하지 않으면 작업이 실패합니다.

7. (선택 사항) 작업 완료 후 작업 완료 보고서를 생성하려면 완료 보고서에서 활성화를 선택합니다. 작업 완료 보고서는 보고서 범위에 제공된 범위를 충족하는 작업으로 처리된 파일에 대한 세부 정보를 제공합니다. Amazon FSx가 보고서를 전송할 위치를 지정하려면 연결된 S3 데이터 리포지토리의 상대 경로를 보고서 경로에 입력합니다.
8. 생성을 선택합니다.

   파일 시스템 페이지 상단의 알림에는 방금 생성한 작업이 진행 중이라는 내용이 표시됩니다.

작업 상태 및 세부 정보를 보려면 파일 시스템의 데이터 리포지토리 탭에서 데이터 리포지토리 작업 창으로 스크롤합니다. 기본 정렬 순서는 목록의 맨 위에 가장 최근 작업을 표시합니다.

이 페이지에서 작업 요약을 보려면 방금 생성한 작업의 작업 ID를 선택합니다. 작업의 요약 페이지가 표시됩니다.

메타데이터 변경 내용을 가져오기(CLI)

- `create-data-repository-task` CLI 명령을 사용하여 FSx for Lustre 파일 시스템에서 메타데이터 변경 내용을 가져올 수 있습니다. 해당 API 작업은 `CreateDataRepositoryTask`입니다.

  ```
  $ aws fsx create-data-repository-task \
      --file-system-id fs-0123456789abcdef0 \
      --type IMPORT_METADATA_FROM_REPOSITORY \
      --paths s3://bucketname1/dir1/path1 \
      --report Enabled=true,Path=s3://bucketname1/dir1/
  path1,Format=REPORT_CSV_20191124,Scope=FAILED_FILES_ONLY
  ```

  데이터 리포지토리 작업을 생성한 후 Amazon FSx는 작업 설명을 JSON으로 반환합니다.

연결된 데이터 리포지토리에서 메타데이터를 가져오는 작업을 생성한 후 데이터 리포지토리 가져오기 작업의 상태를 확인할 수 있습니다. 데이터 리포지토리 보기에 대한 자세한 내용은 데이터 리포지토리 작업 액세스 섹션을 참조하세요.

파일 시스템에 파일 미리 로드

Amazon FSx는 파일에 처음 액세스할 때 Amazon S3 데이터 리포지토리에서 데이터를 복사합니다. 이 접근 방식 때문에 파일에 대한 초기 읽기 또는 쓰기 작업에는 약간의 지연 시간이 발생합니다. 애플리케이션이 이러한 지연 시간에 민감하고 애플리케이션이 액세스해야 하는 파일 또는 디렉터리를 알고 있는 경우 선택적으로 개별 파일 또는 디렉터리의 콘텐츠를 미리 로드할 수 있습니다. 다음과 같이 `hsm_restore` 명령을 사용하여 관리할 수 있습니다.

`hsm_action` 명령(`lfs` 사용자 유틸리티와 함께 실행)을 사용하여 파일 콘텐츠가 파일 시스템에 로드되었는지 확인할 수 있습니다. 반환 값이 `NOOP`이면 파일이 로드되었음을 나타냅니다. 파일 시스템이 마운트된 컴퓨팅 인스턴스에서 다음 명령을 실행합니다. *`path/to/file`*을 파일 시스템에 미리 로드하는 파일의 경로로 바꿉니다.

```
sudo lfs hsm_restore path/to/file
sudo lfs hsm_action path/to/file
```

다음 명령을 사용하여 전체 파일 시스템 또는 파일 시스템 내의 전체 디렉터리를 미리 로드할 수 있습니다. (후행 앰퍼샌드를 사용하면 명령이 백그라운드 프로세스로 실행됩니다.) 여러 파일의 사전 로드를 동시에 요청하는 경우 Amazon FSx는 Amazon S3 데이터 리포지토리에서 파일을 병렬로 로드합니다. 파일이 이미 파일 시스템에 로드된 경우, `hsm_restore` 명령은 파일을 다시 로드하지 않습니다.

```
nohup find local/directory -type f -print0 | xargs -0 -n 1 sudo lfs hsm_restore &
```

> **Note**
>
> 연결된 S3 버킷이 파일 시스템보다 큰 경우 모든 파일 메타데이터를 파일 시스템으로 가져올 수 있어야 합니다. 하지만 파일 시스템의 남은 스토리지 공간에 맞는 만큼의 실제 파일 데이터만 로드할 수 있습니다. 파일 시스템에 스토리지가 더 이상 남아 있지 않을 때 파일 데이터에 액세스하려고 하면 오류가 발생합니다. 이 경우 필요에 따라 스토리지 용량을 늘릴 수 있습니다. 자세한 내용은 스토리지 용량 관리 섹션을 참조하세요.

데이터 리포지토리로 변경 내용 내보내기

FSx for Lustre 파일 시스템의 데이터 및 POSIX 메타데이터 변경 사항을 연결된 데이터 리포지토리로 내보낼 수 있습니다. 관련 POSIX 메타데이터에는 소유권, 권한 및 타임스탬프가 포함됩니다.

파일 시스템에서 변경 내용을 내보내는 방법은 다음 방법 중 하나에 해당될 수 있습니다.

- 새 파일, 변경 파일 또는 삭제된 파일을 연결된 데이터 리포지토리로 자동으로 내보내도록 파일 시스템을 구성합니다. 자세한 내용은 업데이트를 S3 버킷으로 자동 내보냅니다. 섹션을 참조하세요.
- 온디맨드 데이터 리포지토리 내보내기 작업을 사용합니다. 자세한 내용은 데이터 리포지토리 작업을 사용하여 변경 내용 내보내기 섹션을 참조하세요.

데이터 리포지토리 자동 내보내기 및 내보내기 작업은 동시에 실행할 수 없습니다.

> **Important**
>
> 해당 객체가 S3 Glacier Flexieval에 저장되어 있는 경우 자동 내보내기는 파일 시스템의 다음 메타데이터 작업을 S3와 동기화하지 않습니다.
>
> - chmod
> - chown

- 이름 바꾸기

데이터 리포지토리 연결에 대해 자동 내보내기를 켜면 파일이 생성, 수정 또는 삭제될 때 파일 시스템에서 파일 데이터와 메타데이터 변경 내용을 자동으로 내보냅니다. 데이터 리포지토리 내보내기 작업을 사용하여 파일 또는 디렉터리를 내보내는 경우 파일 시스템은 마지막 내보내기 이후 생성되거나 수정된 데이터 파일 및 메타데이터만 내보냅니다.

자동 내보내기 및 데이터 리포지토리 내보내기 작업 모두 POSIX 메타데이터를 내보냅니다. 자세한 내용은 데이터 리포지토리에 대한 POSIX 메타데이터 지원 섹션을 참조하세요.

Important

- FSx for Lustre가 데이터를 S3 버킷으로 내보낼 수 있으려면 데이터를 UTF-8 호환 형식으로 저장해야 합니다.
- S3 객체 키의 최대 길이는 1,024바이트입니다. FSx for Lustre는 해당 S3 객체 키가 1,024바이트를 초과하는 파일을 내보내지 않습니다.

Note

자동 내보내기 및 내보내기 데이터 리포지토리 작업을 통해 생성되는 모든 객체는 S3 Standard 스토리지 클래스를 사용하여 작성됩니다.

주제

- 업데이트를 S3 버킷으로 자동 내보냅니다.
- 데이터 리포지토리 작업을 사용하여 변경 내용 내보내기
- HSM 명령을 사용하여 파일 내보내기

업데이트를 S3 버킷으로 자동 내보냅니다.

파일 시스템에서 파일이 추가, 변경 또는 삭제될 때 연결된 S3 버킷의 콘텐츠를 자동으로 업데이트하도록 FSx for Lustre 파일 시스템을 구성할 수 있습니다. FSx for Lustre는 파일 시스템의 변경에 따라 S3에서 객체를 생성, 업데이트 또는 삭제합니다.

> **Note**
>
> FSx for Lustre 2.10 파일 시스템 또는 `Scratch 1` 파일 시스템에서는 자동 내보내기를 사용할 수 없습니다.

파일 시스템과 같거나 다른 곳에 있는 데이터 리포지토리로 내보낼 수 있습니다. AWS 리전 AWS 리전

FSx 관리 콘솔, AWS CLI및 API를 사용하여 언제든지 데이터 리포지토리 연결을 생성하고 자동 내보내기 설정을 업데이트할 때 자동 내보내기를 구성할 수 있습니다. AWS

> **Note**
>
> 동일한 데이터 리포지토리 연결에서 자동 내보내기와 자동 가져오기를 모두 구성할 수 있습니다. 이 항목에서는 자동 내보내기 기능에 대해서만 설명합니다.

> **Important**
>
> - 자동 내보내기 정책이 모두 활성화되고 자동 가져오기가 비활성화된 상태에서 파일 시스템에서 파일을 수정하면 해당 파일의 내용은 항상 S3의 해당 객체로 내보내집니다. 객체가 대상 위치에 이미 있는 경우 해당 객체를 덮어씁니다.
> - 모든 자동 가져오기 및 자동 내보내기 정책이 활성화된 상태에서 파일 시스템과 S3에서 파일을 수정하면 파일 시스템의 파일이나 S3의 객체를 다른 사람이 덮어쓸 수 있습니다. 한 위치에서 나중에 편집해도 다른 위치에서 이전에 편집한 내용을 덮어쓴다는 보장은 없습니다. 파일 시스템과 S3 버킷 모두에서 동일한 파일을 수정하는 경우 애플리케이션 수준의 조정을 통해 이러한 충돌을 방지해야 합니다. FSx for Lustre는 여러 위치에서의 쓰기 충돌을 방지하지 않습니다.

내보내기 정책은 파일 시스템의 콘텐츠가 변경될 때 FSx for Lustre가 연결된 S3 버킷을 업데이트하는 방법을 지정합니다. 데이터 리포지토리 연결의 자동 내보내기 정책은 다음 중 하나에 해당될 수 있습니다.

- 신규 - FSx for Lustre는 파일 시스템에 새 파일, 디렉터리 또는 심볼릭 링크가 생성되는 경우에만 S3 데이터 리포지토리를 자동으로 업데이트합니다.

- 변경됨 - FSx for Lustre는 파일 시스템의 기존 파일이 변경될 때만 S3 데이터 리포지토리를 자동으로 업데이트합니다. 파일 콘텐츠 변경의 경우 S3 리포지토리로 전파되기 전에 파일을 닫아야 합니다. 작업이 완료되면 메타데이터 변경(이름 변경, 소유권, 권한, 타임스탬프)이 전파됩니다. 이름을 바꾸는 변경(이동 포함)의 경우 기존(미리 이름이 바뀐) S3 객체가 삭제되고 새 이름으로 새 S3 객체가 생성됩니다.
- 삭제됨 - FSx for Lustre는 파일 시스템에서 파일, 디렉터리 또는 심볼릭 링크가 삭제된 경우에만 S3 데이터 리포지토리를 자동으로 업데이트합니다.
- 신규, 변경됨, 삭제됨의 모든 조합 - FSx for Lustre는 파일 시스템에서 지정된 작업이 발생할 경우 S3 데이터 리포지토리를 자동으로 업데이트합니다. 예를 들어, 파일이 파일 시스템에 추가(새로 만들기) 또는 파일 시스템에서 제거(삭제)될 때 S3 리포지토리가 업데이트되지만 파일이 변경될 때는 업데이트되지 않도록 지정할 수 있습니다.
- 정책이 구성되지 않음 - FSx for Lustre는 파일 시스템에서 파일이 추가, 변경 또는 삭제될 때 S3 데이터 리포지토리를 자동으로 업데이트하지 않습니다. 내보내기 정책을 구성하지 않으면 자동 내보내기가 비활성화됩니다. 데이터 리포지토리 작업을 사용하여 변경 내용 내보내기에 설명된 대로 데이터 리포지토리 내보내기 작업을 사용하여 변경 내용을 수동으로 내보낼 수도 있습니다.

대부분의 사용 사례에서는 내보내기 정책을 신규, 변경됨, 삭제됨으로 구성하는 것이 좋습니다. 이 정책을 통해 파일 시스템에서 이루어진 모든 업데이트를 연결된 S3 데이터 리포지토리로 자동으로 내보낼 수 있습니다.

자동으로 내보낼 수 없는 파일이나 디렉터리에 대한 정보를 기록하려면 CloudWatch 로그 로깅을 켜는 것이 좋습니다. 로그의 경고 및 오류에는 실패 이유에 대한 정보가 포함됩니다. 자세한 내용은 데이터 리포지토리 이벤트 로그 섹션을 참조하세요.

내보내기 설정 업데이트

데이터 리포지토리 연결을 생성할 때 파일 시스템의 내보내기 설정을 연결된 S3 버킷으로 설정할 수 있습니다. 자세한 내용은 S3 버킷 링크 생성 섹션을 참조하세요.

또한 내보내기 정책을 포함하여 내보내기 설정을 언제든지 업데이트할 수 있습니다. 자세한 내용은 데이터 리포지토리 연결 설정 업데이트 섹션을 참조하세요.

자동 내보내기 모니터링

Amazon에 게시된 지표 세트를 사용하여 자동 내보내기가 활성화된 데이터 리포지토리 연결을 모니터링할 수 CloudWatch 있습니다. 이 `AgeOfOldestQueuedMessage` 지표는 아직 S3로 내보내지 않은 파일 시스템 업데이트 중 가장 오래된 업데이트 기간을 나타냅니다. `AgeOfOldestQueuedMessage`가 장기간 0보다 큰 경우 메시지 대기열이 줄어들 때까지 파일 시스템

에서 활발하게 변경되는 횟수(특히 디렉터리 이름 변경)를 일시적으로 줄이는 것이 좋습니다. 자세한 내용은 AutoImport AutoExport 및 지표 섹션을 참조하세요.

> **Important**
>
> 자동 내보내기가 활성화된 상태에서 데이터 리포지토리 연결 또는 파일 시스템을 삭제할 때는 먼저 `AgeOfOldestQueuedMessage`가 0인지 확인해야 합니다. 즉, 아직 내보내지 않은 변경 내용이 없음을 의미합니다. 데이터 리포지토리 연결 또는 파일 시스템을 삭제할 때 `AgeOfOldestQueuedMessage`가 0보다 크면 아직 내보내지 않은 변경 내용이 연결된 S3 버킷에 도달하지 않습니다. 이를 방지하려면 `AgeOfOldestQueuedMessage`가 0이 될 때까지 기다렸다가 데이터 리포지토리 연결 또는 파일 시스템을 삭제합니다.

데이터 리포지토리 작업을 사용하여 변경 내용 내보내기

데이터 리포지토리 내보내기 작업은 파일 시스템에서 새로 생성되거나 변경된 파일을 내보냅니다. 파일 시스템의 모든 새 파일에 대해 S3에 새 객체를 생성합니다. 파일 시스템에서 수정되었거나 메타데이터가 수정된 파일의 경우 S3의 해당 객체는 새 데이터 및 메타데이터가 포함된 새 객체로 대체됩니다. 파일 시스템에서 삭제된 파일에 대해서는 조치가 취해지지 않습니다.

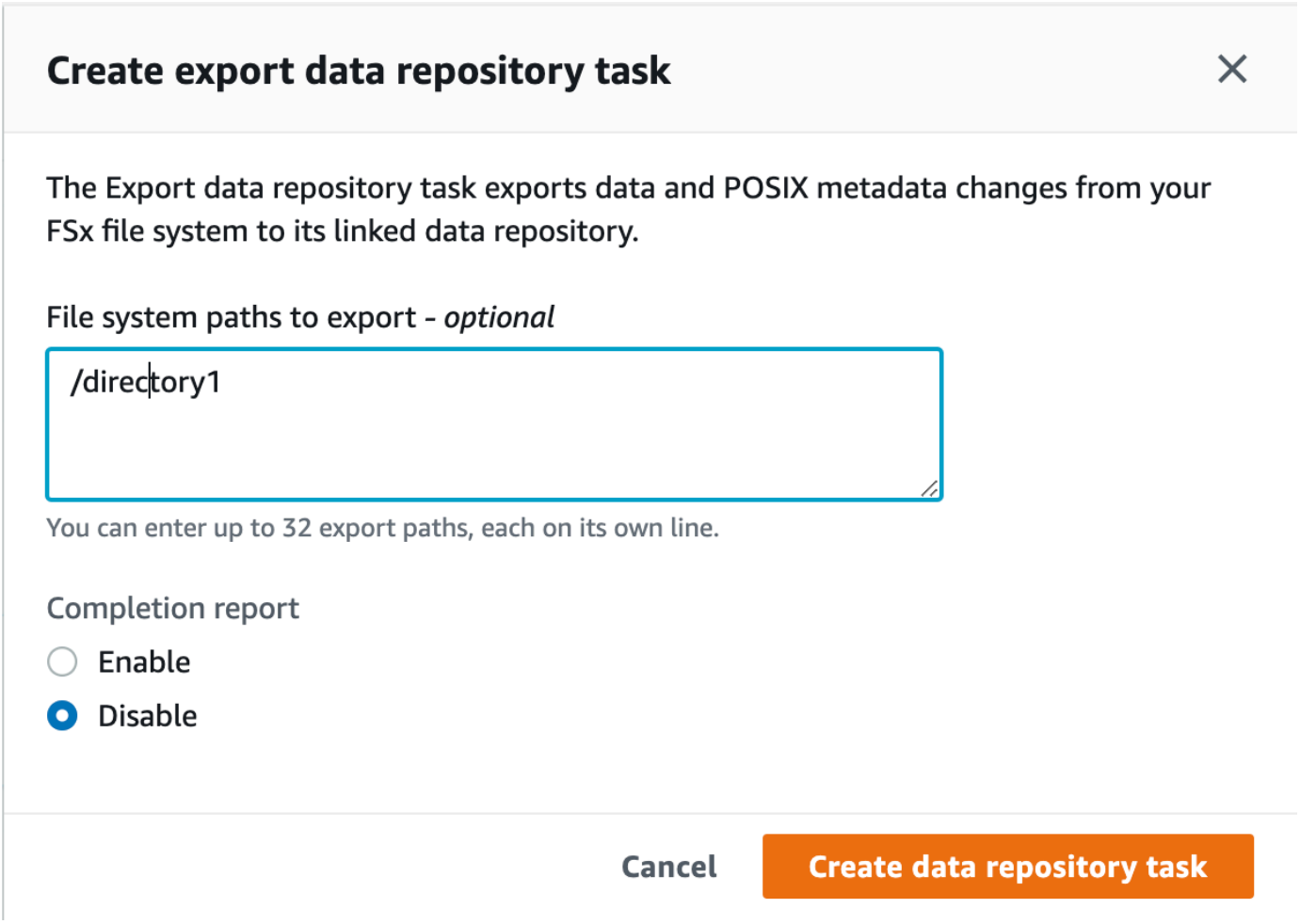
> **Note**
>
> 데이터 리포지토리 내보내기 작업을 사용할 때는 다음 사항에 유의하세요.
>
> - 와일드카드를 사용하여 내보낼 파일을 포함하거나 제외하는 것은 지원되지 않습니다.
> - `mv` 작업을 수행할 때 UID, GID, 권한 또는 콘텐츠 변경이 없더라도 이동된 대상 파일은 S3로 내보내집니다.

Amazon FSx 콘솔 및 CLI를 사용하여 파일 시스템의 데이터 및 메타데이터 변경 사항을 연결된 S3 버킷으로 내보내려면 다음 절차를 따르세요. 여러 DRA에 대해 하나의 데이터 리포지토리 작업을 사용할 수 있다는 점에 유의하세요.

변경 내용 내보내기(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 파일 시스템을 선택한 다음 Lustre 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.

4. 데이터 리포지토리 연결 패널에서 내보내기 작업을 만들 데이터 리포지토리 연결을 선택합니다.
5. 작업에서 작업 내보내기를 선택합니다. 파일 시스템이 S3의 데이터 리포지토리에 연결되지 않은 경우에는 이 옵션을 사용할 수 없습니다. 데이터 리포지토리 내보내기 작업 생성 대화 상자가 표시됩니다.



6. (선택 사항) 내보낼 파일 시스템 경로에 해당 디렉터리 또는 파일의 경로를 제공하여 Amazon FSx 파일 시스템에서 내보낼 디렉터리 또는 파일을 최대 32개까지 지정합니다. 제공하는 경로는 파일 시스템의 마운트 지점을 기준으로 해야 합니다. 마운트 지점이 `/mnt/fsx`이고, `/mnt/fsx/path1`이 내보내려는 파일 시스템의 디렉터리 또는 파일인 경우, 제공할 경로는 `path1`입니다.

> **Note**
>
> 제공한 경로가 유효하지 않으면 작업이 실패합니다.

7. (선택 사항) 작업 완료 후 작업 완료 보고서를 생성하려면 완료 보고서에서 활성화를 선택합니다. 작업 완료 보고서는 보고서 범위에 제공된 범위를 충족하는 작업으로 처리된 파일에 대한 세부 정

보를 제공합니다. Amazon FSx가 보고서를 전송할 위치를 지정하려면 파일 시스템의 연결된 S3 데이터 리포지토리의 보고서 경로에 상대 경로를 입력합니다.

8. 생성을 선택합니다.

   파일 시스템 페이지 상단의 알림에는 방금 생성한 작업이 진행 중이라는 내용이 표시됩니다.

작업 상태 및 세부 정보를 보려면 파일 시스템의 데이터 리포지토리 탭에서 데이터 리포지토리 작업 창으로 스크롤합니다. 기본 정렬 순서는 목록의 맨 위에 가장 최근 작업을 표시합니다.

이 페이지에서 작업 요약을 보려면 방금 생성한 작업의 작업 ID를 선택합니다. 작업의 요약 페이지가 표시됩니다.

변경 내용을 내보내기(CLI)

- `create-data-repository-task` CLI 명령을 사용하여 FSx for Lustre 파일 시스템에서 데이터 및 메타데이터 변경 내용을 내보냅니다. 해당 API 작업은 `CreateDataRepositoryTask`입니다.

  ```
  $ aws fsx create-data-repository-task \
      --file-system-id fs-0123456789abcdef0 \
      --type EXPORT_TO_REPOSITORY \
      --paths path1,path2/file1 \
      --report Enabled=true
  ```

  데이터 리포지토리 작업을 생성한 후 Amazon FSx는 다음 예제와 같이 작업 설명을 JSON으로 반환합니다.

  ```
  {
      "Task": {
          "TaskId": "task-123f8cd8e330c1321",
          "Type": "EXPORT_TO_REPOSITORY",
          "Lifecycle": "PENDING",
          "FileSystemId": "fs-0123456789abcdef0",
          "Paths": ["path1", "path2/file1"],
          "Report": {
              "Path":"s3://dataset-01/reports",
              "Format":"REPORT_CSV_20191124",
              "Enabled":true,
              "Scope":"FAILED_FILES_ONLY"
          },
          "CreationTime": "1545070680.120",
          "ClientRequestToken": "10192019-drt-12",
  ```

```
        "ResourceARN": "arn:aws:fsx:us-
east-1:123456789012:task:task-123f8cd8e330c1321"
    }
}
```

연결된 데이터 리포지토리로 데이터를 내보내는 작업을 생성한 후 데이터 리포지토리 내보내기 작업의 상태를 확인할 수 있습니다. 데이터 리포지토리 보기에 대한 자세한 내용은 데이터 리포지토리 작업 액세스 섹션을 참조하세요.

HSM 명령을 사용하여 파일 내보내기

> **Note**
>
> FSx for Lustre 파일 시스템의 데이터 및 메타데이터에 있는 변경 내용을 Amazon S3의 내구성 있는 데이터 리포지토리로 내보내려면 업데이트를 S3 버킷으로 자동 내보냅니다.에 설명된 자동 내보내기 기능을 사용합니다. 데이터 리포지토리 작업을 사용하여 변경 내용 내보내기에 설명된 대로 데이터 리포지토리 내보내기 작업을 사용할 수도 있습니다.

개별 파일을 데이터 리포지토리로 내보내고 파일이 데이터 리포지토리로 제대로 내보내졌는지 확인하려면 다음과 같은 명령을 실행할 수 있습니다. 반환 값이 `states: (0x00000009) exists archived`이면 파일을 내보냈음을 나타냅니다.

```
sudo lfs hsm_archive path/to/export/file
sudo lfs hsm_state path/to/export/file
```

> **Note**
>
> HSM 명령(예:`hsm_archive`)을 루트 사용자 또는 `sudo`를 사용하여 실행해야 합니다.

전체 파일 시스템 또는 파일 시스템의 전체 디렉터리를 내보내려면 다음 명령을 실행합니다. 여러 파일을 동시에 내보내는 경우 Amazon FSx for Lustre는 파일을 Amazon S3 데이터 리포지토리로 병렬로 내보냅니다.

```
nohup find local/directory -type f -print0 | xargs -0 -n 1 sudo lfs hsm_archive &
```

내보내기가 완료되었는지 확인하려면 다음 명령을 실행합니다.

```
find path/to/export/file -type f -print0 | xargs -0 -n 1 -P 8 sudo lfs hsm_state | awk
 '!/\<archived\>/ || /\<dirty\>/' | wc -l
```

명령을 통해 파일이 하나도 남지 않은 상태가 반환되면 내보내기가 완료된 것입니다.

데이터 리포지토리 작업

데이터 리포지토리 가져오기 및 내보내기 작업을 사용하면 FSx for Lustre 파일 시스템과 Amazon S3의 내구성이 뛰어난 데이터 리포지토리 간에 데이터 및 메타데이터 전송을 관리할 수 있습니다.

데이터 리포지토리 작업은 FSx for Lustre 파일 시스템과 S3의 데이터 리포지토리 간의 데이터 및 메타데이터 전송을 최적화합니다. 이를 수행하는 한 가지 방법은 Amazon FSx 파일 시스템과 연결된 데이터 리포지토리 간의 변경 사항을 추적하는 것입니다. 또한 병렬 전송 기술을 사용하여 최대 수백 GB/s의 속도로 데이터를 전송합니다. Amazon FSx 콘솔, AWS CLI및 Amazon FSx API를 사용하여 데이터 리포지토리 작업을 생성하고 볼 수 있습니다.

데이터 리포지토리 작업은 소유권, 권한, 타임스탬프를 비롯한 파일 시스템의 이동식 운영 체제 인터페이스(POSIX) 메타데이터를 유지 관리합니다. 작업에서 이 메타데이터를 유지 관리하므로 FSx for Lustre 파일 시스템과 연결된 데이터 리포지토리 간에 액세스 제어를 구현하고 유지할 수 있습니다.

데이터 리포지토리 릴리스 작업을 사용하면 Amazon S3로 내보낸 파일을 릴리스하여 새 파일을 위한 파일 시스템 공간을 확보할 수 있습니다. 릴리스된 파일의 콘텐츠는 제거되지만 릴리스된 파일의 메타데이터는 파일 시스템에 남아 있습니다. 사용자와 애플리케이션은 파일을 다시 읽어서 릴리스된 파일에 계속 액세스할 수 있습니다. 사용자 또는 애플리케이션이 릴리스된 파일을 읽으면 FSx for Lustre는 Amazon S3에서 파일 콘텐츠를 투명하게 검색합니다.

데이터 리포지토리 작업 유형

데이터 리포지토리 작업에는 다음과 같은 세 가지 유형이 있습니다:

- 내보내기 데이터 리포지토리 작업이 Lustre 파일 시스템에서 링크된 S3 버킷으로 내보냅니다.
- 가져오기 데이터 리포지토리 작업이 연결된 S3 버킷에서 Lustre 파일 시스템으로 가져옵니다.
- 릴리스 데이터 리포지토리 작업이 Lustre 파일 시스템에서 링크된 S3 버킷으로 내보낸 파일을 릴리스합니다.

자세한 내용은 데이터 리포지토리 작업 생성 섹션을 참조하세요.

주제

- 작업 상태 및 세부 정보 이해
- 데이터 리포지토리 작업 사용
- 작업 완료 보고서 사용
- 데이터 리포지토리 작업 실패 문제 해결

작업 상태 및 세부 정보 이해

데이터 리포지토리 작업은 다음 상태 중 하나를 가질 수 있습니다.

- 보류 Amazon FSx가 작업을 시작하지 않았음을 나타냅니다.
- 실행 Amazon FSx가 작업을 처리 중임을 나타냅니다.
- 실패 Amazon FSx가 작업을 처리하지 못했음을 나타냅니다. 예를 들어, 작업에서 처리하지 못한 파일이 있을 수 있습니다. 명령이 성공하면 아무 것도 반환하지 않습니다. 실패 작업에 대한 자세한 내용은 데이터 리포지토리 작업 실패 문제 해결 섹션을 참조하세요.
- 성공 Amazon FSx가 작업을 완료했음을 나타냅니다.
- 취소 작업이 취소되고 완료되지 않았음을 나타냅니다.
- 취소 중 Amazon FSx가 작업을 취소하는 중임을 나타냅니다.

작업이 생성되면 Amazon FSx 콘솔, CLI 또는 API를 사용하여 데이터 리포지토리 작업에 대한 다음과 같은 세부 정보를 볼 수 있습니다.

- 작업 유형:
  - `EXPORT_TO_REPOSITORY` 내보내기 작업을 나타냅니다.
  - `IMPORT_METADATA_FROM_REPOSITORY` 가져오기 작업을 나타냅니다.
  - `RELEASE_DATA_FROM_FILESYSTEM` 릴리스 작업을 나타냅니다.
- 작업이 실행된 파일 시스템.
- 작업이 생성된 날짜.
- 작업 상태.
- 작업이 처리한 총 파일 수.
- 작업이 처리한 총 파일 수.
- 작업에서 처리하지 못한 총 파일 수. 작업 상태가 실패인 경우 이 값은 0보다 큽니다. 실패한 파일에 대한 자세한 내용은 작업 완료 보고서에서 확인할 수 있습니다. 자세한 내용은 작업 완료 보고서 사용 섹션을 참조하세요.

- 작업이 시작된 시간.
- 작업 상태를 마지막으로 업데이트한 시간. 작업 상태는 30초마다 업데이트됩니다.

현재 데이터 리포지토리 작업 접근에 대한 자세한 내용은 데이터 리포지토리 작업 액세스 섹션을 참조하세요.

데이터 리포지토리 작업 사용

Amazon FSx 콘솔, CLI 또는 API를 사용하여 데이터 리포지토리 작업을 생성하고, 복제하고, 세부 정보를 보고, 취소할 수 있습니다.

주제

- 데이터 리포지토리 작업 생성
- 작업 복제
- 데이터 리포지토리 작업 액세스
- 데이터 리포지토리 작업 취소

데이터 리포지토리 작업 생성

Amazon FSx 콘솔, CLI 또는 API를 사용하여 서비스 연결 역할을 생성할 수 있습니다. 작업을 생성한 후 콘솔, CLI 또는 API를 사용하여 작업의 진행 상황과 상태를 볼 수 있습니다.

세 가지 유형의 데이터 리포지토리 작업을 생성할 수 있습니다.

- 데이터 리포지토리 내보내기 작업은 Lustre 파일 시스템에서 연결된 S3 버킷으로 내보냅니다. 자세한 내용은 데이터 리포지토리 작업을 사용하여 변경 내용 내보내기 섹션을 참조하세요.
- 데이터 리포지토리 가져오기 작업은 연결된 S3 버킷에서 Lustre 파일 시스템으로 가져옵니다. 자세한 내용은 데이터 리포지토리 작업을 사용하여 변경 내용 가져오기 섹션을 참조하세요.
- 데이터 리포지토리 릴리스 작업은 연결된 S3 버킷으로 내보낸 파일을 Lustre 파일 시스템에서 릴리스합니다. 자세한 내용은 데이터 리포지토리 작업을 사용하여 파일을 릴리스합니다. 섹션을 참조하세요.

작업 복제

Amazon FSx 콘솔에서 기존 데이터 리포지토리 작업을 복제할 수 있습니다. 작업을 복제하면 기존 작업의 사본이 데이터 리포지토리 가져오기 작업 생성 또는 데이터 리포지토리 내보내기 작업 생성 페이

지에 표시됩니다. 새 작업을 생성하고 실행하기 전에 필요에 따라 내보내거나 가져올 경로를 변경할 수 있습니다.

> **Note**
>
> 해당 작업의 정확한 사본이 이미 실행 중인 경우 중복 작업 실행 요청은 실패합니다. 이미 실행 중인 작업의 정확한 복사본에는 내보내기 작업의 경우 동일한 파일 시스템 경로 또는 경로가 포함되고 가져오기 작업의 경우 동일한 데이터 리포지토리 경로가 포함됩니다.

작업 세부 정보 보기, 파일 시스템의 데이터 리포지토리 탭에 있는 데이터 리포지토리 작업 창 또는 데이터 리포지토리 작업 페이지에서 작업을 복제할 수 있습니다.

기존 작업 복제

1. 파일 시스템의 데이터 리포지토리 탭에 있는 데이터 리포지토리 작업 창에서 작업을 선택합니다.
2. 작업 복제를 선택합니다. 선택한 작업 유형에 따라 데이터 리포지토리 가져오기 작업 생성 또는 데이터 리포지토리 내보내기 작업 생성 페이지가 표시됩니다. 새 작업의 모든 설정은 복제하려는 작업의 설정과 동일합니다.
3. 가져오거나 내보낼 대상 경로를 변경하거나 추가합니다.
4. 생성을 선택합니다.

데이터 리포지토리 작업 액세스

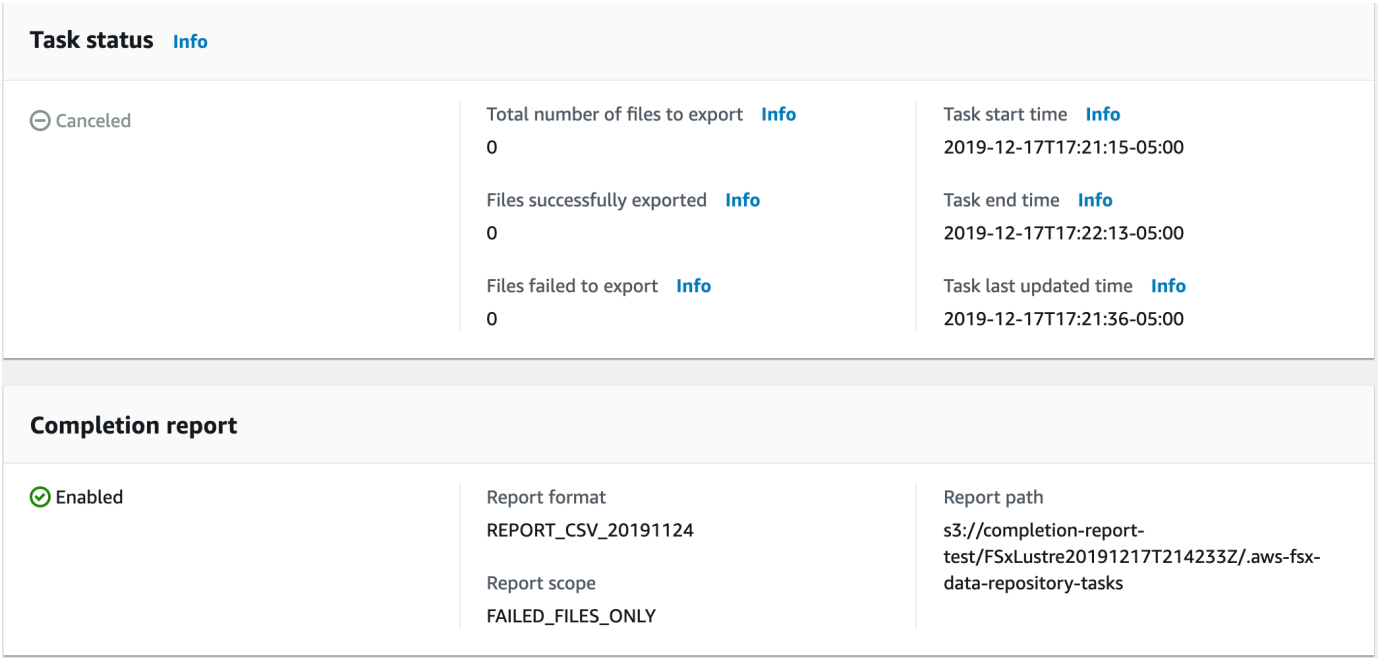
데이터 리포지토리 작업을 생성한 후에는 Amazon FSx 콘솔, CLI 및 API를 사용하여 해당 작업과 계정의 모든 기존 작업에 액세스할 수 있습니다. Amazon FSx는 다음과 같은 세부 작업 정보를 제공합니다.

- 내보내기 작업을 취소할 수 있는 권한을 부여합니다
- 특정 파일 시스템의 모든 작업.
- 특정 데이터 리포지토리 연결에 대한 모든 작업.
- 특정 수명 주기 상태의 모든 작업. 작업 수명 주기 상태에 대한 자세한 내용은 작업 상태 및 세부 정보 이해 섹션을 참조하세요.

다음 설명과 같이 Amazon FSx 콘솔, CLI 또는 API를 사용하여 계정의 모든 기존 데이터 리포지토리 작업에 액세스할 수 있습니다.

데이터 리포지토리 작업 및 작업 세부 정보 확인(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 데이터 리포지토리 작업(Lustre)을 선택합니다. 기존 작업이 표시된 데이터 리포지토리 작업 페이지가 표시됩니다.
3. 작업의 세부 정보를 보려면 데이터 리포지토리 작업 페이지에서 작업 ID 또는 작업 이름을 선택합니다. 정책 세부 정보 페이지가 표시됩니다.



데이터 리포지토리 작업 및 작업 세부 정보 검색(CLI)

Amazon FSx `describe-data-repository-tasks` CLI 명령을 사용하면 계정에서 모든 데이터 리포지토리 작업과 세부 정보를 볼 수 있습니다. `DescribeDataRepositoryTasks`는 동일한 API 명령입니다.

- 다음 명령을 사용하여 계정의 모든 데이터 리포지토리 작업 객체를 볼 수 있습니다.

```
aws fsx describe-data-repository-tasks
```

명령이 성공하면 Amazon FSx는 응답을 JSON 형식으로 반환합니다.

```
{
    "DataRepositoryTasks": [
        {
```

```
            "Lifecycle": "EXECUTING",
            "Paths": [],
            "Report": {
                "Path":"s3://dataset-01/reports",
                "Format":"REPORT_CSV_20191124",
                "Enabled":true,
                "Scope":"FAILED_FILES_ONLY"
            },
            "StartTime": 1591863862.288,
            "EndTime": ,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef3",
            "Status": {
                "SucceededCount": 4255,
                "TotalCount": 4200,
                "FailedCount": 55,
                "LastUpdatedTime": 1571863875.289
            },
            "FileSystemId": "fs-0123456789a7",
            "CreationTime": 1571863850.075,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef3"
        },
        {
            "Lifecycle": "FAILED",
            "Paths": [],
            "Report": {
                "Enabled": false,
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef1",
            "Status": {
                "SucceededCount": 1153,
                "TotalCount": 1156,
                "FailedCount": 3,
                "LastUpdatedTime": 1571863875.289
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1571863850.075,
```

```
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef1"
        },
        {
            "Lifecycle": "SUCCEEDED",
            "Paths": [],
            "Report": {
                "Path":"s3://dataset-04/reports",
                "Format":"REPORT_CSV_20191124",
                "Enabled":true,
                "Scope":"FAILED_FILES_ONLY"
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-04299453935122318",
            "Status": {
                "SucceededCount": 258,
                "TotalCount": 258,
                "FailedCount": 0,
                "LastUpdatedTime": 1771848950.012,
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1771848950.012,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef0"
        }
    ]
}
```


파일 시스템별 작업 보기

다음 설명과 같이 Amazon FSx 콘솔, CLI 또는 API를 사용하여 특정 파일 시스템의 모든 작업을 볼 수 있습니다.

파일 시스템별 작업 보기(콘솔)

1. 탐색 창에서 파일 시스템을 선택합니다. 파일 시스템 페이지가 표시됩니다.
2. 데이터 리포지토리 작업을 볼 파일 시스템을 선택합니다. 파일 시스템 세부 정보 페이지가 표시됩니다.

3. 파일 시스템 세부 정보 페이지에서 데이터 리포지토리 탭을 선택합니다. 이 파일 시스템에 대한 모든 작업은 데이터 리포지토리 작업 패널에 표시됩니다.

파일 시스템별 작업 검색(CLI)

- 다음 명령을 사용하여 파일 시스템 `fs-0123456789abcdef0`의 모든 데이터 리포지토리 작업을 볼 수 있습니다.

```
aws fsx describe-data-repository-tasks \
    --filters Name=file-system-id,Values=fs-0123456789abcdef0
```

명령이 성공하면 Amazon FSx는 응답을 JSON 형식으로 반환합니다.

```
{
    "DataRepositoryTasks": [
        {
            "Lifecycle": "FAILED",
            "Paths": [],
            "Report": {
                "Path":"s3://dataset-04/reports",
                "Format":"REPORT_CSV_20191124",
                "Enabled":true,
                "Scope":"FAILED_FILES_ONLY"
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef1",
            "Status": {
                "SucceededCount": 1153,
                "TotalCount": 1156,
                "FailedCount": 3,
                "LastUpdatedTime": 1571863875.289
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1571863850.075,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef1"
        },
        {
```

```
            "Lifecycle": "SUCCEEDED",
            "Paths": [],
            "Report": {
                "Enabled": false,
            },
            "StartTime": 1571863862.288,
            "EndTime": 1571863905.292,
            "Type": "EXPORT_TO_REPOSITORY",
            "Tags": [],
            "TaskId": "task-0123456789abcdef0",
            "Status": {
                "SucceededCount": 258,
                "TotalCount": 258,
                "FailedCount": 0,
                "LastUpdatedTime": 1771848950.012,
            },
            "FileSystemId": "fs-0123456789abcdef0",
            "CreationTime": 1771848950.012,
            "ResourceARN": "arn:aws:fsx:us-east-1:1234567890:task/
task-0123456789abcdef0"
        }
    ]
}
```


데이터 리포지토리 작업 취소

데이터 리포지토리 작업이 보류 중이거나 실행 중일 때 작업을 취소할 수 있습니다. 작업 취소 시 다음 상황이 발생합니다.

- Amazon FSx가 처리할 대기열에 있는 파일을 처리하지 않습니다.
- Amazon FSx는 현재 처리 중인 모든 파일을 계속 처리합니다.
- Amazon FSx는 작업이 이미 처리한 파일을 되돌리지 않습니다.

데이터 리포지토리 작업 취소(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 데이터 리포지토리 작업을 취소하려는 파일 시스템을 클릭합니다.
3. 데이터 리포지토리 탭을 열고 아래로 스크롤하여 데이터 리포지토리 작업 패널을 확인합니다.
4. 취소하려는 작업의 작업 ID 또는 작업 이름을 선택합니다.

5. 작업을 취소하려면 작업 취소를 선택합니다.
6. 작업 ID를 입력하여 취소 요청을 확인합니다.

데이터 리포지토리 작업 취소(CLI)

작업을 취소하려면 Amazon FSx `cancel-data-repository-task` CLI 명령을 사용합니다. `CancelDataRepositoryTask`는 동일한 API 명령입니다.

- 다음 명령을 사용하여 데이터 리포지토리 작업을 취소합니다.

```
aws fsx cancel-data-repository-task \
    --task-id task-0123456789abcdef0
```

명령이 성공하면 Amazon FSx는 응답을 JSON 형식으로 반환합니다.

```
{
    "Status": "CANCELING",
    "TaskId": "task-0123456789abcdef0"
}
```

작업 완료 보고서 사용

작업 완료 보고서는 데이터 리포지토리 내보내기, 가져오기 또는 릴리스 작업의 결과에 대한 세부 정보를 제공합니다. 보고서에는 보고서 범위와 일치하는 작업에서 처리한 파일의 결과가 포함됩니다. `Enabled` 파라미터를 사용하여 작업에 대한 보고서를 생성할지 여부를 지정할 수 있습니다.

Amazon FSx는 작업을 위해 보고서를 활성화할 때 지정한 경로를 사용하여 Amazon S3에 있는 파일 시스템의 연결된 데이터 리포지토리로 보고서를 전송합니다. 보고서의 파일 이름은 `report.csv`는 가져오기 작업을, `failures.csv`는 내보내기 또는 릴리스 작업을 나타냅니다.

리포트 형식은 쉼표로 구분된 값(CSV) 파일로, `FilePath`, `FileStatus` 및 `ErrorCode`의 세 가지 필드가 있습니다.

보고서는 다음과 같이 RFC-4180 형식 인코딩을 사용하여 인코딩됩니다.

- 다음 문자 중 하나로 시작하는 문자열은 단일 따옴표로 묶습니다. `@ + - =`
- 다음 문자 중 하나 이상을 포함하는 문자열은 큰따옴표로 묶습니다. `" ,`
- 모든 큰따옴표는 추가 큰따옴표로 빠져나갑니다.

다음은 보고서 인코딩의 몇 가지 예입니다.

- `@filename.txt`는 `"""@filename.txt"""`가 됩니다.
- `+filename.txt`는 `"""+filename.txt"""`가 됩니다.
- `file,name.txt`는 `"file,name.txt"`가 됩니다.
- `file"name.txt`는 `"file""name.txt"`가 됩니다.

CSV 형식에 대한 자세한 내용은 IETF 웹사이트의 쉼표로 구분된 값(CSV) 파일에서 RFC 4180 일반 형식 및 MIME 유형을 참조하세요.

다음은 실패한 파일만 포함하는 작업 완료 보고서에 제공되는 정보의 예입니다.

```
myRestrictedFile,failed,S3AccessDenied
dir1/myLargeFile,failed,FileSizeTooLarge
dir2/anotherLargeFile,failed,FileSizeTooLarge
```

가능한 문제 및 해결 방법에 대한 자세한 내용은 데이터 리포지토리 작업 실패 문제 해결 섹션을 참조하세요.

데이터 리포지토리 작업 실패 문제 해결

CloudWatch 로그에 로깅을 켜면 데이터 리포지토리 작업을 사용하여 파일을 가져오거나 내보내는 동안 발생한 모든 실패에 대한 정보를 기록할 수 있습니다. CloudWatch 로그 이벤트 로그에 대한 자세한 내용은 을 참조하십시오데이터 리포지토리 이벤트 로그.

데이터 리포지토리 작업이 실패하는 경우 콘솔의 작업 상태 페이지에서 Amazon FSx가 처리하지 못한 파일 수를 파일 내보내기 실패에서 확인할 수 있습니다. 또는 CLI 또는 API를 사용하여 작업의 `Status: FailedCount` 속성을 볼 수 있습니다. 이 정보에 액세스하는 방법에 대한 자세한 내용은 데이터 리포지토리 작업 액세스 섹션을 참조하세요.

데이터 리포지토리 작업의 경우 Amazon FSx는 완료 보고서에서 실패한 특정 파일 및 디렉터리에 대한 정보도 선택적으로 제공합니다. 작업 완료 보고서에는 장애가 발생한 Lustre 파일 시스템의 파일 또는 디렉터리 경로, 상태 및 실패 이유가 포함됩니다. 자세한 내용은 작업 완료 보고서 사용 섹션을 참조하세요.

데이터 리포지토리 작업은 다음과 같은 여러 가지 이유로 실패할 수 있습니다.

오류 코드	설명
`FileSizeTooLarge`	Amazon S3에서 지원하는 최대 객체 크기는 5TiB입니다.
`InternalError`	Amazon FSx 파일 시스템에서 가져오기, 내보내기 또는 릴리스 작업과 관련하여 오류가 발생했습니다. 일반적으로 이 오류 코드는 실패한 작업이 실행된 Amazon FSx 파일 시스템이 FAILED 수명 주기 상태임을 의미합니다. 이 경우 데이터 손실로 인해 영향을 받은 파일을 복구하지 못할 수 있습니다. 그렇지 않으면 계층적 스토리지 관리(HSM) 명령을 사용하여 파일 및 디렉터리를 S3의 데이터 리포지토리로 내보낼 수 있습니다. 자세한 내용은 HSM 명령을 사용하여 파일 내보내기 섹션을 참조하세요.
`OperationNotPermitted`	Amazon FSx가 연결된 S3 버킷으로 파일을 내보내지 않았기 때문에 파일을 릴리스하지 못했습니다. 파일을 먼저 연결된 Amazon S3 버킷으로 내보내려면 데이터 리포지토리 자동 내보내기 또는 내보내기 작업을 사용해야 합니다.
`PathSizeTooLong`	내보내기 경로가 너무 깁니다. S3에서 지원하는 최대 객체 키 길이는 1,024자입니다.
`ResourceBusy`	Amazon FSx는 파일 시스템의 다른 클라이언트가 액세스하고 있었기 때문에 파일을 내보내거나 릴리스하지 못했습니다. 워크플로에서 파일 쓰기를 DataRepositoryTask 완료한 후에 다시 시도할 수 있습니다.
`S3AccessDenied`	Amazon S3에 대한 데이터 리포지토리 내보내기 또는 가져오기 작업에 대한 액세스가 거부되었습니다. 내보내기 작업의 경우 Amazon FSx 파일 시스템에 S3의 연결된 데이터 리포지토리로 내보내

오류 코드	설명
	기 작업을 수행할 `S3:PutObject` 권한이 있어야 합니다. 이 권한은 `AWSServiceRoleForFSxS3Access_` *`fs-0123456789abcdef0`* 서비스 연결 역할에서 부여됩니다. 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요. 내보내기 작업의 경우 내보내기 작업을 수행하려면 데이터가 파일 시스템의 VPC 외부로 이동해야 하므로 대상 리포지토리에 `aws:SourceVpc` 또는 `aws:SourceVpce` IAM 글로벌 조건 키 중 하나가 포함된 버킷 정책이 있는 경우 이 오류가 발생할 수 있습니다. 가져오기 작업의 경우 Amazon FSx 파일 시스템에 S3의 연결된 데이터 리포지토리에서 가져오기 위한 `S3:HeadObject` 및 `S3:GetObject` 작업을 수행할 권한이 있어야 합니다. 가져오기 작업의 경우 S3 버킷이 AWS Key Management Service (SSE-KMS) 에 고객 관리 키를 저장하여 서버 측 암호화를 사용하는 경우의 정책 구성을 따라야 합니다. 서버 측 암호화된 Amazon S3 버킷 사용 S3 버킷에 파일 시스템에 연결된 S3 버킷 계정과 AWS 계정 다른 계정에서 업로드된 객체가 포함된 경우, 업로드한 계정과 상관없이 데이터 리포지토리 작업이 S3 메타데이터를 수정하거나 S3 객체를 덮어쓸 수 있도록 할 수 있습니다. S3 버킷에 대해 S3 객체 소유권 기능을 활성화하는 것이 좋습니다. 이 기능을 사용하면 업로드할 때 미리 준비된 ACL을 제공하도록 강제함으로써 다른 사람이 버킷에 AWS 계정 업로드한 새 객체의 소유권을 확보할 수 있습니다. `-/-acl bucket-owner-full-control`

오류 코드	설명
	S3 버킷에서 버킷 소유자 선호 옵션을 선택하여 S3 객체 소유권을 활성화합니다. 자세한 내용은 Amazon S3 사용자 가이드의 S3 객체 소유권을 사용하여 업로드된 객체의 소유권 제어를 참조하세요.
`S3Error`	Amazon FSx에서 S3 관련 오류가 발생했지만 `S3AccessDenied` 가 아니었습니다.
`S3FileDeleted`	원본 파일이 데이터 리포지토리에 없기 때문에 Amazon FSx에서 하드 링크 파일을 내보낼 수 없었습니다.
`S3ObjectInUnsupportedTier`	Amazon FSx는 S3 Glacier Flexible Retrievin g 또는 S3 Glacier Deep Archive 스토리지 클래스에서 심볼링크가 아닌 객체를 가져왔습니다. `FileStatus` 가 작업 완료 보고서에 `succeeded with warning` 상태로 표시됩니다. 이 경고는 데이터를 검색하려면 먼저 S3 Glacier Flexible Retrieve 또는 S3 Glacier Deep Archive 객체를 복원한 다음 `hsm_restore` 명령을 사용하여 객체를 가져와야 한다는 의미입니다.
`S3ObjectNotFound`	Amazon FSx는 파일이 데이터 리포지토리에 없기 때문에 파일을 가져오거나 내보낼 수 없었습니다.
`S3ObjectPathNotPosixCompliant`	Amazon S3 객체가 존재하지만 POSIX 호환 객체가 아니므로 가져올 수 없습니다. 지원되는 POSIX 메타데이터에 대한 자세한 내용은 데이터 리포지토리에 대한 POSIX 메타데이터 지원 섹션을 참조하세요.

오류 코드	설명
`S3ObjectUpdateInProgressFromFileRename`	자동 내보내기가 파일 이름 변경을 처리하고 있기 때문에 Amazon FSx에서 파일을 릴리스할 수 없습니다. 자동 내보내기 이름 변경 프로세스를 완료해야 파일을 릴리스할 수 있습니다.
`S3SymlinkInUnsupportedTier`	Amazon FSx는 S3 Glacier Flexible Retrieve 또는 S3 Glacier Deep Archive 스토리지 클래스와 같이 지원되지 않는 Amazon S3 스토리지 클래스에 있으므로 심볼링크 객체를 가져올 수 없습니다. `FileStatus` 가 작업 완료 보고서에 `failed` 상태로 표시됩니다.
`SourceObjectDeletedBeforeReleasing`	파일이 릴리스되기 전에 데이터 리포지토리에서 삭제되었기 때문에 Amazon FSx는 파일 시스템에서 파일을 릴리스하지 못했습니다.

파일 릴리스

릴리스 데이터 리포지토리 작업은 FSx for Lustre 파일 시스템에서 파일 데이터를 릴리스하여 새 파일을 위한 공간을 확보합니다. 파일을 릴리스하면 파일 목록과 메타데이터는 유지되지만 해당 파일 콘텐츠의 로컬 사본은 제거됩니다. 사용자 또는 애플리케이션이 릴리스된 파일에 액세스하는 경우, 데이터는 연결된 Amazon S3 버킷에서 자동으로 투명하게 파일 시스템으로 다시 로드됩니다.

> **Note**
>
> 릴리스 데이터 리포지토리 작업은 FSx for Lustre 2.10 파일 시스템에서 사용할 수 없습니다.

릴리즈할 파일 시스템 경로 및 마지막 액세스 이후 최소 지속 기간 매개변수에 따라 릴리스될 파일이 결정됩니다.

- 릴리스할 파일 시스템 경로: 파일을 릴리스할 경로를 지정합니다.
- 마지막 액세스 이후 최소 기간: 해당 기간 동안 액세스하지 않은 파일은 릴리스되도록 기간 (일) 을 지정합니다. 파일이 마지막으로 액세스된 이후의 기간은 릴리스 작업 생성 시간과 파일에 마지막으로 액세스한 시간 (최대값 `atimemtime`, 및`ctime`) 간의 차이를 취하여 계산합니다.

파일은 S3로 내보낸 후 마지막 액세스 이후 지속 시간이 마지막 액세스 값 이후 최소 기간보다 긴 경우에만 파일 경로를 따라 릴리스됩니다. 마지막 액세스 이후 최소 기간을 `0` 일 수로 지정하면 마지막 액세스 이후의 기간과 관계없이 파일이 릴리스됩니다.

> **Note**
>
> 릴리스할 파일을 포함하거나 제외할 때 와일드카드를 사용하는 것은 지원되지 않습니다.

데이터 리포지토리 해제 작업은 연결된 S3 데이터 리포지토리로 이미 내보낸 파일의 데이터만 릴리스합니다. 자동 내보내기 기능, 데이터 리포지토리 내보내기 작업 또는 HSM 명령을 사용하여 S3로 데이터를 내보낼 수 있습니다. 파일이 데이터 리포지토리로 내보내졌는지 확인하려면 다음 명령을 실행할 수 있습니다. 반환 값이 `states: (0x00000009) exists archived`이면 파일을 내보냈음을 나타냅니다.

```
sudo lfs hsm_state path/to/export/file
```

> **Note**
>
> 루트 사용자 또는 를 사용하여 sudo HSM 명령을 실행해야 합니다.

파일 데이터를 정기적으로 릴리스하려면 Amazon EventBridge Scheduler를 사용하여 데이터 리포지토리 반복 릴리스 작업을 예약할 수 있습니다. 자세한 내용은 Amazon EventBridge 스케줄러 사용 설명서의 EventBridge 스케줄러 시작하기를 참조하십시오.

주제

- 데이터 리포지토리 작업을 사용하여 파일을 릴리스합니다.

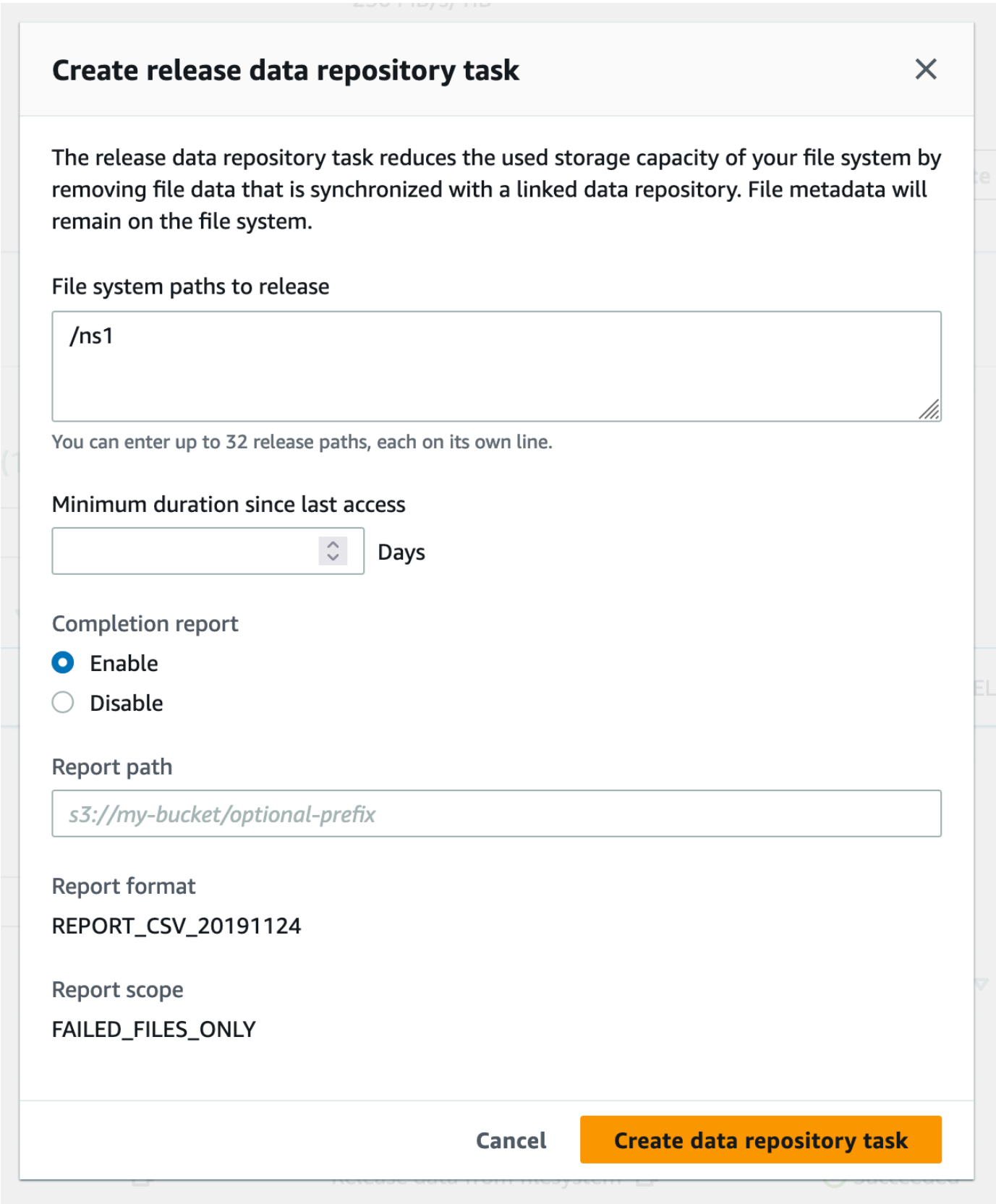
데이터 리포지토리 작업을 사용하여 파일을 릴리스합니다.

Amazon FSx 콘솔 및 CLI를 사용하여 파일 시스템에서 파일을 릴리스하는 작업을 생성하려면 다음 절차를 따르세요 파일을 릴리스하면 파일 목록과 메타데이터는 유지되지만 해당 파일 콘텐츠의 로컬 사본은 제거됩니다.

파일 릴리스(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.

2. 왼쪽 탐색 창에서 파일 시스템을 선택한 다음 Lustre 파일 시스템을 선택합니다.
3. 데이터 리포지토리 탭을 선택합니다.
4. 데이터 리포지토리 연결 창에서 릴리스 작업을 만들 데이터 리포지토리 연결을 선택합니다.
5. 작업에서 릴리스 작업 만들기를 선택합니다. 이 옵션은 파일 시스템이 S3의 데이터 리포지토리에 연결된 경우에만 사용할 수 있습니다. 릴리스 데이터 리포지토리 생성 작업 대화 상자가 표시됩니다.


Create release data repository task
The release data repository task reduces the used storage capacity of your file system by removing file data that is synchronized with a linked data repository. File metadata will remain on the file system.
File system paths to release
/ns1
You can enter up to 32 release paths, each on its own line.
Minimum duration since last access
Days
Completion report
Enable
Disable
Report path
s3://my-bucket/optional-prefix
Report format
REPORT_CSV_20191124
Report scope
FAILED_FILES_ONLY
Cancel
Create data repository task

6. 릴리스할 파일 시스템 경로에서 디렉터리 또는 파일의 경로를 제공하여 Amazon FSx 파일 시스템에서 릴리스할 디렉터리 또는 파일을 최대 32개까지 지정합니다. 제공하는 경로는 파일 시스템의 마운트 지점을 기준으로 해야 합니다. 예를 들어 마운트 지점이 `/mnt/fsx`이고, `/mnt/fsx/path1`가 릴리스하려는 파일 시스템의 파일인 경우 제공할 경로는 `path1`입니다. 파일 시스템의 모든 파일을 릴리스하려면 슬래시(/)를 경로로 지정합니다.

   > **Note**
   >
   > 제공한 경로가 유효하지 않으면 작업이 실패합니다.

7. 마지막 액세스 이후 최소 기간을 일 단위로 지정하여 해당 기간 동안 액세스하지 않은 파일은 릴리스되도록 합니다. 마지막 액세스 시간은 최대값인 `atime`, `mtime` 및 `ctime`를 사용하여 계산됩니다. 마지막 액세스 기간이 마지막 액세스 이후 최소 기간보다 긴 파일(작업 생성 시간 기준)은 릴리스됩니다. 마지막 액세스 기간이 이 일수 미만인 파일은 릴리스할 파일 시스템 경로 필드에 있더라도 릴리스되지 않습니다. 마지막 액세스 이후 기간과 관계없이 파일을 릴리스할 수 있는 기간(`0`일)을 입력합니다.
8. (선택 사항) 완료 보고서에서 활성화를 선택하여 보고서 범위에 제공된 범위를 충족하는 파일에 대한 세부 정보를 제공하는 작업 완료 보고서를 생성합니다. Amazon FSx가 보고서를 전송할 위치를 지정하려면 파일 시스템의 연결된 S3 데이터 리포지토리의 보고서 경로에 상대 경로를 입력합니다.
9. 데이터 리포지토리 작업 생성을 선택합니다.

   파일 시스템 페이지 상단의 알림에는 방금 생성한 작업이 진행 중이라는 내용이 표시됩니다.

작업 상태 및 세부 정보를 보려면 데이터 리포지토리 탭에서 데이터 리포지토리 작업까지 아래로 스크롤합니다. 기본 정렬 순서는 목록의 맨 위에 가장 최근 작업을 표시합니다.

이 페이지에서 작업 요약을 보려면 방금 생성한 작업의 작업 ID를 선택합니다.

파일 릴리스(CLI)

- `create-data-repository-task` CLI 명령을 사용하여 FSx for Lustre 파일 시스템에서 파일을 릴리스하는 작업을 생성합니다. 해당 API 작업은 `CreateDataRepositoryTask`입니다.

  다음 파라미터를 설정합니다.

  - `--file-system-id`를 파일을 릴리스하는 파일 시스템의 ID로 설정합니다.

- `--paths`를 데이터를 릴리스할 파일 시스템의 경로로 설정합니다. 디렉터리를 지정하면 해당 디렉터리 내의 파일이 릴리스됩니다. 파일 경로를 지정하면 해당 파일만 릴리스됩니다. 연결된 S3 버킷으로 내보낸 파일 시스템의 모든 파일을 릴리스하려면 경로에 슬래시(/)를 지정합니다.
- `--type`을 `RELEASE_DATA_FROM_FILESYSTEM`으로 설정합니다.
- `--release-configuration DurationSinceLastAccess` 옵션을 다음과 같이 설정합니다.
  - `Unit` – `DAYS`로 설정합니다.
  - `Value` - 기간(일)을 나타내는 정수로 지정하여 해당 기간 동안 액세스하지 않은 파일은 릴리스해야 합니다. 이 일수 미만의 기간 동안 액세스한 파일은 `--paths` 파라미터에 포함되더라도 릴리스되지 않습니다. 마지막 액세스 이후 기간과 관계없이 파일을 릴리스할 수 있는 기간(`0`일)을 입력합니다.

이 샘플 명령은 연결된 S3 버킷으로 내보내고 `--release-configuration` 기준을 충족하는 파일을 지정된 경로의 디렉터리에서 릴리스하도록 지정합니다.

```
$ aws fsx create-data-repository-task \
    --file-system-id fs-0123456789abcdef0 \
    --type RELEASE_DATA_FROM_FILESYSTEM \
    --paths path1,path2/file1 \
    --release-configuration '{"DurationSinceLastAccess":
{"Unit":"DAYS","Value":10}}' \
    --report Enabled=false
```

데이터 리포지토리 작업을 생성한 후 Amazon FSx는 작업 설명을 JSON으로 반환합니다.

파일을 릴리스하는 작업을 생성한 후 작업 상태를 확인할 수 있습니다. 데이터 리포지토리 보기에 대한 자세한 내용은 데이터 리포지토리 작업 액세스 섹션을 참조하세요.

온프레미스 데이터에 Amazon FSx 사용

FSx for Lustre를 사용하여 클라우드 내 컴퓨팅 인스턴스로 온프레미스 데이터를 처리할 수 있습니다. FSx for Lustre는 VPN을 통한 AWS Direct Connect 액세스를 지원하므로 온프레미스 클라이언트에서 파일 시스템을 마운트할 수 있습니다.

온프레미스 데이터에 FSx for Lustre 사용

1. 파일 시스템을 만듭니다. 자세한 내용은 시작하기 연습의 FSx for Lustre 파일 시스템 만들기 섹션을 참조하세요.
2. 온프레미스 클라이언트에서 파일 시스템을 마운트합니다. 자세한 내용은 온프레미스 또는 피어링된 Amazon VPC에서 Amazon FSx 파일 시스템 마운트 섹션을 참조하세요.
3. 처리하려는 데이터를 FSx for Lustre 파일 시스템에 복사합니다.
4. 파일 시스템을 마운트하는 클라우드 내 Amazon EC2 인스턴스에서 컴퓨팅 집약적 워크로드를 실행합니다.
5. 작업을 마치면 파일 시스템의 최종 결과를 다시 온프레미스 데이터 위치로 복사하고 FSx for Lustre 파일 시스템을 삭제합니다.

데이터 리포지토리 이벤트 로그

CloudWatch 로그에 로깅을 켜면 자동 가져오기, 자동 내보내기 및 데이터 리포지토리 작업을 사용하여 파일을 가져오거나 내보내는 동안 발생한 모든 실패에 대한 정보를 기록할 수 있습니다. 자세한 정보는 Amazon CloudWatch 로그로 로깅을 참조하세요.

> **Note**
>
> 데이터 리포지토리 작업이 실패하면 Amazon FSx는 작업 완료 보고서에도 실패 정보를 기록합니다. 완료 보고서의 실패 정보에 대한 자세한 내용은 데이터 리포지토리 작업 실패 문제 해결 섹션을 확인합니다.

자동 가져오기, 자동 내보내기 및 데이터 리포지토리 작업은 아래 나열된 경우를 포함하여 여러 가지 이유로 실패할 수 있습니다. 로그 확인에 대한 자세한 내용은 로그 보기 섹션을 참조하세요.

이벤트 가져오기

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
`S3ImportListObjectError`	ERROR	`###` 접두사를 사용하여 S3 버킷 *`bucket_name`* 의 S3 객체	Amazon FSx가 S3 버킷의 S3 객체를 나열하지 못했습	N/A

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
		를 나열하지 못했습니다.	니다. 이는 S3 버킷 정책이 Amazon FSx에 충분한 권한을 제공하지 않는 경우 발생할 수 있습니다.	
`S3ImportUnsupportedTierWarning`	WARN	지원되지 않는 계층 *`S3_tier_name`* 의 S3 객체로 인해 S3 버킷 *`bucket_name`* 에 *`key_value`* 이 있는 S3 객체를 가져오지 못했습니다.	Amazon FSx는 S3 Glacier Flexible Retrieve 또는 S3 Glacier Deep Archive 스토리지 클래스와 같이 지원되지 않는 Amazon S3 스토리지 클래스에 있으므로 S3 객체를 가져올 수 없습니다.	`S3ObjectInUnsupportedTier`

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
`S3ImportSymlinkInUnsupportedTierWarning`	WARN	지원되지 않는 계층 *`S3_Tier_name`* 의 S3 심볼릭 링크 객체로 인해 S3 버킷 *`bucket_name`* 에 *`key_value`* 이 있는 S3 객체를 가져오지 못했습니다.	Amazon FSx는 S3 Glacier Flexible Retrieve 또는 S3 Glacier Deep Archive 스토리지 클래스와 같이 지원되지 않는 Amazon S3 스토리지 클래스에 있으므로 심볼릭 링크 객체를 가져올 수 없습니다.	`S3SymlinkInUnsupportedTier`

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
`S3ImportAccessDenied`	ERROR	S3 객체에 대한 액세스가 거부되어 S3 버킷 *`bucket_name`* 에 *`key_value`* 키가 있는 S3 객체를 가져오지 못했습니다.	Amazon S3에 대한 데이터 리포지토리 내보내기 가져오기 작업에 대한 액세스가 거부되었습니다. 가져오기 작업의 경우 Amazon FSx 파일 시스템에 S3의 연결된 데이터 리포지토리에서 가져오기 위한 `s3:HeadObject` 및 `s3:GetObject` 작업을 수행할 권한이 있어야 합니다. 가져오기 작업의 경우 S3 버킷이 고객 관리 키가 저장된 서버 측 암호화 AWS Key Management Service (SSE-KMS) 를 사용하는 경우 의 정책 구성을 따라	`S3AccessDenied`

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
			야 합니다. 서버 측 암호화된 Amazon S3 버킷 사용	
S3ImportDeleteAccessDenied	ERROR	S3 객체에 대한 액세스가 거부되었기 때문에 S3 버킷 *bucket_name* 에 *key_value* 가 있는 S3 객체의 로컬 파일을 삭제하지 못했습니다.	S3 객체에 대한 자동 가져오기 액세스가 거부되었습니다.	N/A
S3ImportObjectPathNotPosixCompliant	ERROR	S3 객체가 POSIX와 호환되지 않기 때문에 S3 버킷 *bucket_name* 에 *key_value* 가 있는 S3 객체를 가져오지 못했습니다.	Amazon S3 객체가 존재하지만 POSIX 호환 객체가 아니므로 가져올 수 없습니다. 지원되는 POSIX 메타데이터에 대한 자세한 내용은 데이터 리포지토리에 대한 POSIX 메타데이터 지원 섹션을 참조하세요.	S3ObjectPathNotPosixCompliant

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
S3ImportObjectTypeMismatch	ERROR	이름이 같은 S3 객체를 이미 파일 시스템으로 가져왔기 때문에 S3 버킷 *bucket_name* 에 *key_value* 가 있는 S3 객체를 가져오지 못했습니다.	가져오는 S3 객체가 파일 시스템에서 이름이 같은 기존 객체와 유형(파일 또는 디렉터리)이 다릅니다.	S3ObjectTypeMismatch
S3ImportDirectoryMetadataUpdateError	ERROR	내부 오류로 인해 로컬 디렉터리 메타데이터를 업데이트하지 못했습니다.	내부 오류로 인해 디렉터리 메타데이터를 가져올 수 없는 경우.	N/A
S3ImportObjectDeleted	ERROR	*key_value* 키가 있는 S3 객체를 S3 버킷 *bucket_name* 에서 찾을 수 없어서 가져오지 못했습니다.	Amazon FSx는 해당 객체가 데이터 리포지토리에 없기 때문에 파일 메타데이터를 가져올 수 없었습니다.	S3FileDeleted

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
S3ImportBucketDoesNotExist	ERROR	버킷이 없어서 S3 버킷 *bucket_name* 에 *key_value* 키가 있는 S3 객체를 가져오지 못했습니다.	S3 버킷이 더 이상 존재하지 않기 때문에 Amazon FSx는 S3 객체를 파일 시스템으로 자동으로 가져올 수 없습니다.	N/A
S3ImportDeleteBucketDoesNotExist	ERROR	버킷이 없어서 S3 버킷 *bucket_name* 에 *key_value* 가 있는 S3 객체의 로컬 파일을 삭제하지 못했습니다.	S3 버킷이 더 이상 존재하지 않기 때문에 Amazon FSx는 파일 시스템의 S3 객체에 연결된 파일을 삭제할 수 없습니다.	N/A
S3ImportDirectoryCreateError	ERROR	내부 오류로 인해 로컬 디렉터리를 만들지 못했습니다.	Amazon FSx가 내부 오류로 인해 파일 시스템에 생성된 디렉터리를 자동으로 가져오지 못했습니다.	N/A

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
NoDiskSpace	ERROR	파일 시스템이 꽉 찼기 때문에 S3 버킷 *bucket_name* 에 *key_value* 키가 있는 S3 객체를 가져오지 못했습니다.	파일 또는 디렉터리를 생성하는 동안 파일 시스템에서 메타데이터 서버의 디스크 공간이 부족했습니다.	N/A

내보내기 이벤트

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
S3ExportInternalError	ERROR	내부 오류로 인해 S3 버킷 *bucket_name* 에 *key_value* 키가 있는 S3 객체를 내보내지 못했습니다.	내부 오류로 인해 객체를 내보내지 못했습니다.	INTERNAL_ERROR
S3ExportAccessDenied	ERROR	S3 버킷 *bucket_name* 에 *key_value* 키가 있는 S3 객체에 대한 액세스가 거부되어 파일을 내보내지 못했습니다.	Amazon S3에 대한 데이터 리포지토리 내보내기 작업에 대한 액세스가 거부되었습니다. 내보내기 작업의 경우 Amazon FSx 파일 시스	S3AccessDenied

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
			템에 S3의 연결된 데이터 리포지토리로 내보내기 작업을 수행할 `s3:PutObject` 권한이 있어야 합니다. 이 권한은 `AWSServiceRoleForFSxS3Access_`*`fs-0123456789abcdef0`* 서비스 연결 역할에서 부여됩니다. 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요. 내보내기 작업을 수행하려면 데이터가 파일 시스템의 VPC 외부로 이동해야 하므로 대상 리포지토리에 `aws:SourceVpc` 또는 `aws:SourceVpce` IAM 글로벌 조건 키 중 하나가 포함된 버	

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
			킷 정책이 있는 경우 이 오류가 발생할 수 있습니다. S3 버킷에 파일 시스템에 연결된 S3 버킷 계정과 AWS 계정 다른 계정에서 업로드된 객체가 포함된 경우, 업로드한 계정과 상관없이 데이터 리포지토리 작업이 S3 메타데이터를 수정하거나 S3 객체를 덮어쓸 수 있도록 할 수 있습니다. S3 버킷에 대해 S3 객체 소유권 기능을 활성화하는 것이 좋습니다. 이 기능을 사용하면 업로드할 때 미리 준비된 ACL을 제공하도록 강제함으로써 다른 사람이 버킷에 AWS 계정 업로드한 새 객체의 소유권을 확보할 수 있습니다. --	

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
			`acl bucket-owner-full-control` S3 버킷에서 버킷 소유자 선호 옵션을 선택하여 S3 객체 소유권을 활성화합니다. 자세한 내용은 Amazon S3 사용자 가이드의 S3 객체 소유권을 사용하여 업로드된 객체의 소유권 제어를 참조하세요.	
`S3ExportPathSizeTooLong`	ERROR	로컬 파일 경로 크기가 S3에서 지원하는 최대 객체 키 길이를 초과하여 파일을 내보내지 못했습니다.	내보내기 경로가 너무 깁니다. S3에서 지원하는 최대 객체 키 길이는 1,024자입니다.	`PathSizeTooLong`
`S3ExportFileSizeTooLarge`	ERROR	파일 크기가 지원되는 최대 S3 객체 크기를 초과하여 파일을 내보내지 못했습니다.	Amazon S3에서 지원하는 최대 객체 크기는 5TiB입니다.	`FileSizeTooLarge`

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
S3ExportKMSKeyNotFound	ERROR	버킷의 KMS 키를 찾을 수 없어서 S3 버킷 *bucket_name* 에 *key_value* 키가 있는 S3 객체의 파일을 내보내지 못했습니다.	파일을 찾을 수 없어서 AWS KMS key Amazon FSx에서 파일을 내보낼 수 없었습니다. S3 AWS 리전 버킷과 동일한 키를 사용해야 합니다. KMS 키 생성에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 키 생성을 참조하십시오.	N/A
S3ExportResourceBusy	ERROR	다른 프로세스에서 사용 중이므로 파일을 내보내지 못했습니다.	파일 시스템의 다른 클라이언트가 파일을 수정하고 있었기 때문에 Amazon FSx에서 파일을 내보낼 수 없었습니다. 워크플로에서 파일 쓰기를 완료한 후 작업을 재시도할 수 있습니다.	ResourceBusy

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
S3ExportLocalObjectReleaseWithoutS3Source	WARN	내보내기 건너뛰기: 로컬 파일이 릴리스된 상태이고 *key_value* 키가 있는 연결된 S3 객체를 버킷 *bucket_name* 에서 찾을 수 없습니다.	Amazon FSx는 파일이 파일 시스템에서 릴리스된 상태였기 때문에 파일을 내보낼 수 없었습니다.	N/A
S3ExportLocalObjectNotMatchDra	WARN	내보내기 건너뛰기: 로컬 파일이 데이터 리포지토리 연결 파일 시스템 경로에 속하지 않습니다.	객체가 데이터 리포지토리에 연결된 파일 시스템 경로에 속하지 않기 때문에 Amazon FSx에서 내보낼 수 없었습니다.	N/A
InternalAutoExportError	ERROR	자동 내보내기에서 파일 시스템 객체를 내보내는 동안 내부 오류가 발생했습니다.	내부(자동 내보내기 또는 Lustre 수준) 오류로 인해 내보내기에 실패했습니다.	N/A
S3CompletionReportUploadFailure	ERROR	데이터 리포지토리 작업 완료 보고서를 *bucket_name* 에 업로드하지 못했습니다.	Amazon FSx가 완료 보고서를 업로드하지 못했습니다.	N/A

오류 코드	로그 수준	로그 메시지	근본 원인:	완료 보고서의 오류 코드
`S3CompletionReportValidateFailure`	ERROR	완료 보고서 경로 *report_path* 가 이 파일 시스템과 연결된 데이터 리포지토리에 속하지 않기 때문에 데이터 리포지토리 작업 완료 보고서를 *bucket_name* 버킷에 업로드하지 못했습니다.	고객이 제공한 S3 경로가 연결된 데이터 리포지토리에 속하지 않기 때문에 Amazon FSx에서 완료 보고서를 업로드하지 못했습니다.	N/A

이전 배포 유형으로 작업하기

이 섹션은 Scratch 1 배포 유형이 있는 파일 시스템과 데이터 리포지토리 연결을 사용하지 않는 `Scratch 2` 또는 `Persistent 1` 배포 유형이 있는 파일 시스템에도 적용됩니다.

주제

- Amazon S3 버킷에 파일 시스템 연결
- S3 버킷에서 업데이트 자동 가져오기

Amazon S3 버킷에 파일 시스템 연결

Amazon FSx for Lustre 파일 시스템을 생성하면 Amazon S3의 내구성 데이터 리포지토리에 연결할 수 있습니다. 파일 시스템을 생성하기 전에 연결하려는 Amazon S3 버킷을 이미 생성했는지 확인합니다. 파일 시스템 생성 마법사의 선택적 데이터 리포지토리 가져오기/내보내기 창에서 다음과 같은 데이터 리포지토리 구성 속성을 설정합니다.

- 파일 시스템이 생성된 후 S3 버킷에 객체를 추가하거나 수정할 때 Amazon FSx가 파일 및 디렉터리 목록을 최신 상태로 유지하는 방법을 선택합니다. 자세한 내용은 S3 버킷에서 업데이트 자동 가져오기 섹션을 참조하세요.
- 가져오기 버킷: 연결된 리포지토리에 사용 중인 S3 버킷의 이름을 입력합니다.

- 가져오기 접두사: S3 버킷의 일부 파일 및 디렉터리 데이터 목록만 파일 시스템으로 가져오려면 선택적 가져오기 접두사를 입력합니다. 가져오기 접두사는 S3 버킷에서 데이터를 가져올 위치를 정의합니다.
- 내보내기 접두사: Amazon FSx가 파일 시스템의 콘텐츠를 연결된 S3 버킷으로 내보내는 위치를 정의합니다.

Amazon FSx가 FSx for Lustre 파일 시스템의 데이터를 가져온 S3 버킷의 동일한 디렉터리로 다시 내보내는 1:1 매핑을 사용할 수 있습니다. 1:1 매핑을 사용하려면 파일 시스템을 생성할 때 접두사 없이 S3 버킷으로의 내보내기 경로를 지정합니다.

- 콘솔을 사용하여 파일 시스템을 생성할 때는 접두사 내보내기 > 지정한 접두사 옵션을 선택하고 접두사 필드를 비워 둡니다.
- AWS CLI 또는 API를 사용하여 파일 시스템을 생성할 때는 내보내기 경로를 추가 접두사 없이 S3 버킷의 이름으로 지정합니다(예:`ExportPath=s3://lustre-export-test-bucket/`).

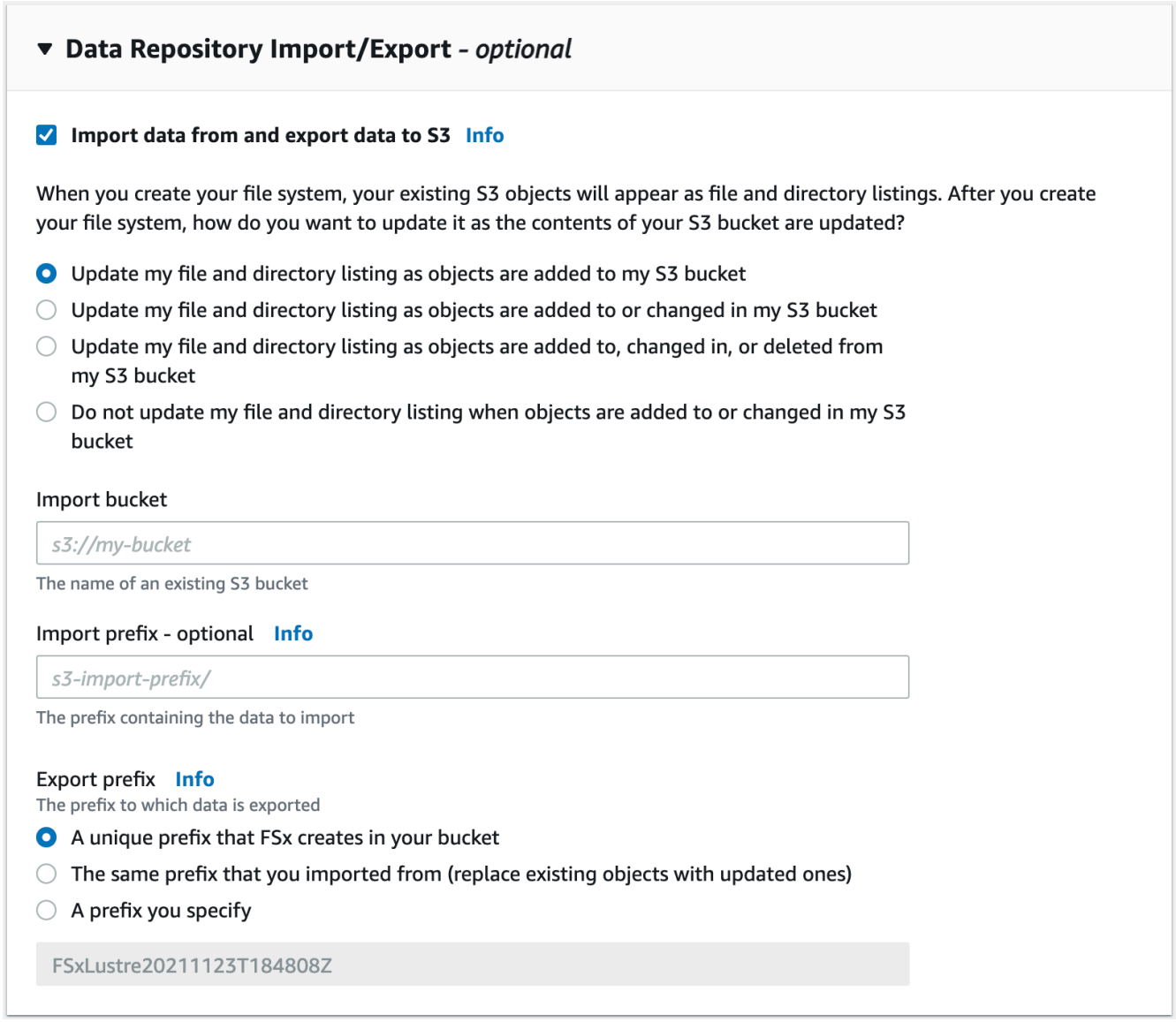
이 방법을 사용하면 가져오기 경로를 지정할 때 가져오기 접두사를 포함할 수 있으며 내보내기의 1:1 매핑에는 영향을 주지 않습니다.

S3 버킷에 연결된 파일 시스템 생성

다음 절차는 AWS 관리 콘솔 및 AWS 명령줄 인터페이스(AWS CLI)를 사용하여 S3 버킷에 연결된 Amazon FSx 파일 시스템을 생성하는 프로세스를 안내합니다.

Console

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템 생성을 선택합니다.
3. 파일 시스템 유형으로 FSx for Lustre를 선택한 후 다음을 선택합니다.
4. 파일 시스템 세부 정보와 네트워크 및 보안 섹션에 필요한 정보를 제공합니다. 자세한 내용은 FSx for Lustre 파일 시스템 만들기 섹션을 참조하세요.
5. 데이터 리포지토리 가져오기/내보내기 패널을 사용하여 Amazon S3의 연결된 데이터 리포지토리를 구성합니다. S3에서 데이터 가져오기 및 S3로 데이터 내보내기를 선택하여 데이터 리포지토리 가져오기/내보내기 섹션을 확장하고 데이터 리포지토리 설정을 구성합니다.



6. 연결된 S3 버킷에서 객체를 추가하거나 수정할 때 Amazon FSx가 파일 및 디렉터리 목록을 최신 상태로 유지하는 방법을 선택합니다. 파일 시스템을 생성하면 기존 S3 객체가 파일 및 디렉터리 목록에 표시됩니다.

   - 객체가 S3 버킷에 추가될 때 파일 및 디렉터리 목록 업데이트: (기본 설정) Amazon FSx는 현재 FSx 파일 시스템에 존재하지 않는 링크된 S3 버킷에 추가된 새 객체의 파일 및 디렉터리 목록을 자동으로 업데이트합니다. Amazon FSx는 S3 버킷에서 변경된 객체의 목록을 업데이트하지 않습니다. Amazon FSx는 S3 버킷에서 삭제된 객체 목록을 삭제하지 않습니다.

> **Note**
>
> CLI 및 API를 사용하여 연결된 S3 버킷에서 데이터를 가져오기 위한 기본 가져오기 기본 설정은 NONE입니다. 콘솔을 사용할 때의 기본 가져오기 기본 설정은 새 객체가 S3 버킷에 추가될 때 Lustre를 업데이트하는 것입니다.

- 내 S3 버킷에 객체가 추가되거나 변경됨에 따라 내 파일 및 디렉터리 목록 업데이트: Amazon FSx는 이 옵션을 선택한 후 S3 버킷에 추가된 새 객체와 S3 버킷에서 변경된 기존 객체의 파일 및 디렉터리 목록을 자동으로 업데이트합니다. Amazon FSx는 S3 버킷에서 삭제된 객체 목록을 삭제하지 않습니다.
- 내 S3 버킷에 오브젝트가 추가, 변경 또는 삭제될 때 내 파일 및 디렉터리 목록 업데이트: Amazon FSx는 이 옵션을 선택한 후 S3 버킷에 추가된 새 객체, S3 버킷에서 변경된 기존 객체 및 S3 버킷에서 삭제된 기존 객체의 파일 및 디렉터리 목록을 자동으로 업데이트합니다.
- 내 파일을 업데이트하지 않고 내 S3 버킷에 객체가 추가, 변경 또는 삭제될 때 직접 나열하지 않음 - Amazon FSx는 파일 시스템을 생성할 때 연결된 S3 버킷과 디렉터리 목록을 업데이트합니다. FSx는 이 옵션을 선택한 후 새로 추가된 객체 또는 변경되거나 삭제된 객체에 대한 파일 및 디렉터리 목록을 업데이트하지 않습니다.

7. S3 버킷에 있는 파일 및 디렉터리 데이터 목록 중 일부만 파일 시스템으로 가져오려면 선택적 가져오기 접두사를 입력합니다. 가져오기 접두사는 S3 버킷에서 데이터를 가져올 위치를 정의합니다. 자세한 내용은 S3 버킷에서 업데이트 자동 가져오기 섹션을 참조하세요.
8. 사용 가능한 내보내기 접두사 옵션 중 하나를 선택합니다.

   - Amazon FSx가 버킷에 작성하는 고유 접두사: FSx for Lustre에서 생성한 접두사를 사용하여 새 객체와 변경된 객체를 내보내려면 이 옵션을 선택합니다. 해당 접두사는 `/FSxLustre`*`file-system-creation- timestamp`*와 같습니다. 타임스탬프는 UTC 형식(예: `FSxLustre20181105T222312Z`)입니다.
   - 가져온 것과 동일한 접두사(기존 객체를 업데이트된 객체로 대체): 기존 객체를 업데이트된 객체로 바꾸려면 이 옵션을 선택합니다.
   - 지정한 접두사: 가져온 데이터를 보존하고 지정한 접두사를 사용하여 새 객체 및 변경된 객체를 내보내려면 이 옵션을 선택합니다. 데이터를 S3 버킷으로 내보낼 때 1:1 매핑을 구현하려면 이 옵션을 선택하고 접두사 필드를 비워 두세요. FSx는 데이터를 가져온 것과 동일한 디렉터리로 데이터를 내보냅니다.

9. (선택 사항) 유지 관리 환경설정을 지정하거나 시스템 기본값을 사용합니다.
10. 다음을 선택하고 설정을 검토합니다. 필요한 경우 변경합니다.

11. 파일 시스템 생성(Create file system)을 선택합니다.

AWS CLI

다음 예제는 파일 시스템이 생성된 후 연결된 데이터 리포지토리의 새 파일, 변경 및 삭제된 파일을 가져오는 가져오기 기본 설정을 사용하여 `lustre-export-test-bucket`에 연결된 Amazon FSx 파일 시스템을 생성합니다.

> **Note**
>
> CLI 및 API를 사용하여 연결된 S3 버킷에서 데이터를 가져오기 위한 기본 가져오기 기본 설정은 `NONE` 이며, 이는 콘솔을 사용할 때의 기본 동작과 다릅니다.

FSx for Lustre 파일 시스템을 생성하려면 아래와 같이 Amazon FSx CLI 명령 `create-file-system`을 사용합니다. 해당 API 작업은 `CreateFileSystem`입니다.

```
$ aws fsx create-file-system \
--client-request-token CRT1234 \
--file-system-type LUSTRE \
--file-system-type-version 2.10 \
--lustre-configuration
AutoImportPolicy=NEW_CHANGED_DELETED,DeploymentType=SCRATCH_1,ImportPath=s
3://lustre-export-test-bucket/,ExportPath=s3://lustre-export-test-bucket/export,
PerUnitStorageThroughput=50 \
--storage-capacity 2400 \
--subnet-ids subnet-123456 \
--tags Key=Name,Value=Lustre-TEST-1 \
--region us-east-2
```

파일 시스템을 생성한 후 Amazon FSx에서는 다음 예에서처럼 파일 시스템 설명을 JSON으로 반환합니다.

```
{
    "FileSystems": [
        {
            "OwnerId": "owner-id-string",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "FileSystemTypeVersion": "2.10",
```

```
            "Lifecycle": "CREATING",
            "StorageCapacity": 2400,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "PERSISTENT_1",
                "DataRepositoryConfiguration": {
                    "AutoImportPolicy": "NEW_CHANGED_DELETED",
                    "Lifecycle": "UPDATING",
                    "ImportPath": "s3://lustre-export-test-bucket/",
                    "ExportPath": "s3://lustre-export-test-bucket/export",
                    "ImportedFileChunkSize": 1024
                },
                "PerUnitStorageThroughput": 50
            }
        }
    ]
}
```


파일 시스템의 내보내기 경로 보기

FSx for Lustre 콘솔, AWS CLI 및 API를 사용하여 파일 시스템의 내보내기 경로를 볼 수 있습니다.

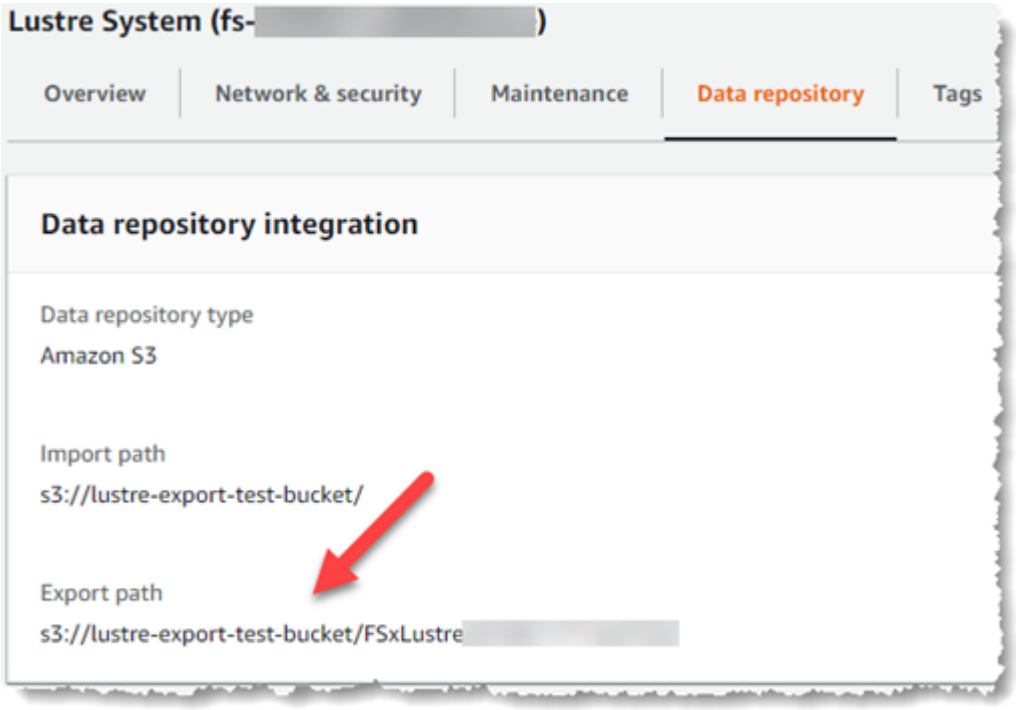
Console

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 내보내기 경로를 보려는 FSx for Lustre 파일 시스템의 파일 시스템 이름 또는 파일 시스템 ID를 선택합니다.

해당 파일 시스템의 파일 시스템 세부 정보 페이지가 표시됩니다.

3. 데이터 리포지토리 탭을 선택합니다.

   데이터 리포지토리 통합 패널이 나타나고 가져오기 및 내보내기 경로가 표시됩니다.



CLI

파일 시스템의 내보내기 경로를 결정하려면 `describe-file-systems` AWS CLI 명령을 사용합니다.

```
aws fsx describe-file-systems
```

응답의 `LustreConfiguration`에서 `ExportPath` 속성을 찾습니다.

```
{
    "OwnerId": "111122223333",
    "CreationTime": 1563382847.014,
    "FileSystemId": "",
    "FileSystemType": "LUSTRE",
    "Lifecycle": "AVAILABLE",
    "StorageCapacity": 2400,
    "VpcId": "vpc-6296a00a",
```

```
    "SubnetIds": [
        "subnet-1111111"
    ],
    "NetworkInterfaceIds": [
        "eni-0c288d5b8cc06c82d",
        "eni-0f38b702442c6918c"
    ],
    "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
    "ResourceARN": "arn:aws:fsx:us-east-2:267731178466:file-system/
fs-0123456789abcdef0",
    "Tags": [
        {
          "Key": "Name",
          "Value": "Lustre System"
        }
    ],
 "LustreConfiguration": {
    "DeploymentType": "SCRATCH_1",
    "DataRepositoryConfiguration": {
    "AutoImportPolicy": " NEW_CHANGED_DELETED",
    "Lifecycle": "AVAILABLE",
    "ImportPath": "s3://lustre-export-test-bucket/",
    "ExportPath": "s3://lustre-export-test-bucket/FSxLustre20190717T164753Z",
    "ImportedFileChunkSize": 1024
    }
  },
  "PerUnitStorageThroughput": 50,
  "WeeklyMaintenanceStartTime": "6:09:30"
}
```

데이터 리포지토리 수명 주기 상태

데이터 리포지토리 수명 주기 상태는 파일 시스템의 연결된 데이터 리포지토리에 대한 상태 정보를 제공합니다. 데이터 리포지토리는 다음과 같은 수명 주기 상태를 가질 수 있습니다.

- 생성: Amazon FSx는 파일 시스템과 연결된 데이터 리포지토리 사이에 데이터 리포지토리 구성을 생성합니다. 데이터 리포지토리를 사용할 수 없습니다.
- 사용 가능: 데이터 리포지토리를 사용할 수 있습니다.
- 업데이트: 데이터 리포지토리 구성이 고객 주도로 업데이트 중이므로 가용성에 영향을 미칠 수 있습니다.

- 잘못된 구성: Amazon FSx는 데이터 리포지토리 구성이 수정될 때까지 S3 버킷에서 업데이트를 자동으로 가져올 수 없습니다. 자세한 내용은 잘못 구성된 연결된 S3 버킷 문제 해결 섹션을 참조하세요.

Amazon FSx 콘솔, AWS 명령줄 인터페이스 및 Amazon FSx API를 사용하여 파일 시스템의 연결된 데이터 리포지토리 수명 주기 상태를 볼 수 있습니다. Amazon FSx 콘솔에서는 파일 시스템의 데이터 리포지토리 탭에 있는 데이터 리포지토리 통합 창에서 데이터 리포지토리 수명 주기 상태에 액세스할 수 있습니다. `Lifecycle` 속성은 `describe-file-systems` CLI 명령(해당하는 API 작업 `DescribeFileSystems`)의 응답으로 `DataRepositoryConfiguration` 객체에 있습니다.

S3 버킷에서 업데이트 자동 가져오기

기본적으로 새 파일 시스템을 생성하면 Amazon FSx는 파일 시스템 생성 시 연결된 S3 버킷에 있는 객체의 파일 메타데이터(이름, 소유권, 타임스탬프, 권한)를 가져옵니다. 파일 시스템 생성 후 S3 버킷에 추가, 변경 또는 삭제된 객체의 메타데이터를 자동으로 가져오도록 FSx for Lustre 파일 시스템을 구성할 수 있습니다. FSx for Lustre는 파일 시스템 생성 시 파일 메타데이터를 가져오는 것과 같은 방식으로 생성 후 변경된 객체의 파일 및 디렉터리 목록을 업데이트합니다. Amazon FSx가 변경된 객체의 파일 및 디렉터리 목록을 업데이트할 때 S3 버킷의 변경된 객체에 더 이상 메타데이터가 포함되지 않는 경우 Amazon FSx는 기본 권한을 사용하지 않고 파일의 현재 메타데이터 값을 유지합니다.

> **Note**
>
> 가져오기 설정은 2020년 7월 23일 오후 3시(동부 표준시) 이후에 생성된 FSx for Lustre 파일 시스템에서 사용할 수 있습니다.

새 파일 시스템을 생성할 때 가져오기 기본 설정을 지정하고 FSx 관리 콘솔, AWS CLI 및 AWS API를 사용하여 기존 파일 시스템의 설정을 업데이트할 수 있습니다. 파일 시스템을 생성하면 기존 S3 객체가 파일 및 디렉터리 목록에 표시됩니다. 파일 시스템을 생성한 후 S3 버킷의 콘텐츠가 업데이트될 때 어떻게 업데이트하고 싶은지 지정합니다. 파일 시스템은 다음 중 하나에 해당될 수 있습니다.

> **Note**
>
> 업데이트를 자동으로 가져오려면 FSx for Lustre 파일 시스템과 연결된 S3 버킷이 같은 AWS 리전에 있어야 합니다.

- 객체가 S3 버킷에 추가될 때 파일 및 디렉터리 목록 업데이트: (기본 설정) Amazon FSx는 현재 FSx 파일 시스템에 존재하지 않는 링크된 S3 버킷에 추가된 새 객체의 파일 및 디렉터리 목록을 자동으로 업데이트합니다. Amazon FSx는 S3 버킷에서 변경된 객체의 목록을 업데이트하지 않습니다. Amazon FSx는 S3 버킷에서 삭제된 객체 목록을 삭제하지 않습니다.

  > **Note**
  >
  > CLI 및 API를 사용하여 연결된 S3 버킷에서 데이터를 가져오기 위한 기본 가져오기 기본 설정은 `NONE`입니다. 콘솔을 사용할 때의 기본 가져오기 기본 설정은 새 객체가 S3 버킷에 추가될 때 Lustre를 업데이트하는 것입니다.

- 내 S3 버킷에 객체가 추가되거나 변경됨에 따라 내 파일 및 디렉터리 목록 업데이트: Amazon FSx는 이 옵션을 선택한 후 S3 버킷에 추가된 새 객체와 S3 버킷에서 변경된 기존 객체의 파일 및 디렉터리 목록을 자동으로 업데이트합니다. Amazon FSx는 S3 버킷에서 삭제된 객체 목록을 삭제하지 않습니다.
- 내 S3 버킷에 오브젝트가 추가, 변경 또는 삭제될 때 내 파일 및 디렉터리 목록 업데이트: Amazon FSx는 이 옵션을 선택한 후 S3 버킷에 추가된 새 객체, S3 버킷에서 변경된 기존 객체 및 S3 버킷에서 삭제된 기존 객체의 파일 및 디렉터리 목록을 자동으로 업데이트합니다.
- 내 파일을 업데이트하지 않고 내 S3 버킷에 객체가 추가, 변경 또는 삭제될 때 직접 나열하지 않음 - Amazon FSx는 파일 시스템을 생성할 때 연결된 S3 버킷과 디렉터리 목록을 업데이트합니다. FSx는 이 옵션을 선택한 후 새로 추가된 객체 또는 변경되거나 삭제된 객체에 대한 파일 및 디렉터리 목록을 업데이트하지 않습니다.

연결된 S3 버킷의 변경 내용을 기반으로 파일 시스템 파일 및 디렉터리 목록을 업데이트하도록 가져오기 기본 설정을 지정하면 Amazon FSx는 연결된 S3 버킷에 `FSx`이라는 이벤트 알림 구성을 생성합니다. S3 버킷의 `FSx` 이벤트 알림 구성을 수정하거나 삭제하지 마세요. 이렇게 하면 새 파일 또는 변경된 파일 및 디렉터리 목록을 파일 시스템으로 자동으로 가져올 수 없습니다.

Amazon FSx가 연결된 S3 버킷에서 변경된 파일 목록을 업데이트하면 파일이 쓰기 잠겨 있더라도 로컬 파일을 업데이트된 버전으로 덮어씁니다. 마찬가지로, 연결된 S3 버킷에서 해당 객체가 삭제되었을 때 Amazon FSx가 파일 목록을 업데이트하면 파일에 쓰기 잠금이 설정되어 있더라도 로컬 파일이 삭제됩니다.

Amazon FSx는 파일 시스템을 업데이트하기 위해 노력합니다. Amazon FSx는 다음과 같은 상황에서 파일 시스템을 변경하여 업데이트할 수 없습니다.

- Amazon FSx에 변경되거나 새 S3 객체를 열 권한이 없는 경우.

- 연결된 S3 버킷의 FSx 이벤트 알림 구성이 삭제되거나 변경된 경우.

이러한 조건 중 하나로 인해 데이터 리포지토리 수명 주기 상태가 잘못 구성될 수 있습니다. 자세한 내용은 데이터 리포지토리 수명 주기 상태 섹션을 참조하세요.

사전 조건

Amazon FSx가 연결된 S3 버킷에서 새 파일, 변경 또는 삭제된 파일을 자동으로 가져오려면 다음 조건이 필요합니다.

- 파일 시스템과 연결된 S3 버킷은 같은 AWS 리전에 있어야 합니다.
- S3 버킷에 잘못 구성된 수명 주기 상태가 없습니다. 자세한 내용은 데이터 리포지토리 수명 주기 상태 섹션을 참조하세요.
- 계정에 연결된 S3 버킷에서 이벤트 알림을 구성하고 수신하는 데 필요한 권한이 있어야 합니다.

지원되는 파일 변경 유형

Amazon FSx는 연결된 S3 버킷에서 발생하는 파일 및 폴더에 대한 다음과 같은 변경 사항 가져오기를 지원합니다.

- 파일 콘텐츠 변경
- 파일 또는 폴더 메타데이터 변경
- 심볼릭 링크 대상 또는 메타데이터 변경

가져오기 기본 설정 업데이트

새 파일 시스템을 생성할 때 파일 시스템의 가져오기 환경설정을 지정할 수 있습니다. 자세한 내용은 S3 버킷에 파일 시스템 연결 섹션을 참조하세요.

다음 절차와 같이 파일 시스템을 생성한 후 AWS 관리 콘솔, AWS CLI 및 Amazon FSx API를 사용하여 파일 시스템의 가져오기 기본 설정을 업데이트할 수도 있습니다.

Console

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 대시보드에서 파일 시스템을 선택합니다.

3. 관리할 파일 시스템을 선택하여 파일 시스템 세부 정보를 표시합니다.
4. 데이터 리포지토리 설정을 보려면 데이터 리포지토리를 선택합니다. 수명 주기 상태가 사용 가능이거나 잘못 구성됨인 경우 가져오기 기본 설정을 수정할 수 있습니다. 자세한 내용은 데이터 리포지토리 수명 주기 상태 섹션을 참조하세요.
5. 작업을 선택한 다음 가져오기 기본 설정 업데이트를 선택하여 가져오기 기본 설정 업데이트 대화 상자를 표시합니다.
6. 새 설정을 선택한 다음 업데이트를 선택하여 변경합니다.

CLI

가져오기 기본 설정을 업데이트하려면 `update-file-system` CLI 명령을 사용합니다. 해당 API 작업은 `UpdateFileSystem`입니다.

파일 시스템의 `AutoImportPolicy`을 업데이트하면 Amazon FSx는 업데이트된 파일 시스템의 설명을 다음과 같이 JSON으로 반환합니다.

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "Lifecycle": "UPDATING",
            "StorageCapacity": 2400,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
```

```
            "LustreConfiguration": {
                "DeploymentType": "SCRATCH_1",
                "DataRepositoryConfiguration": {
                    "AutoImportPolicy": "NEW_CHANGED_DELETED",
                    "Lifecycle": "UPDATING",
                    "ImportPath": "s3://lustre-export-test-bucket/",
                    "ExportPath": "s3://lustre-export-test-bucket/export",
                    "ImportedFileChunkSize": 1024
                }
                "PerUnitStorageThroughput": 50,
                "WeeklyMaintenanceStartTime": "2:04:30"
            }
        }
    ]
}
```

Amazon FSx for Lustre 성능

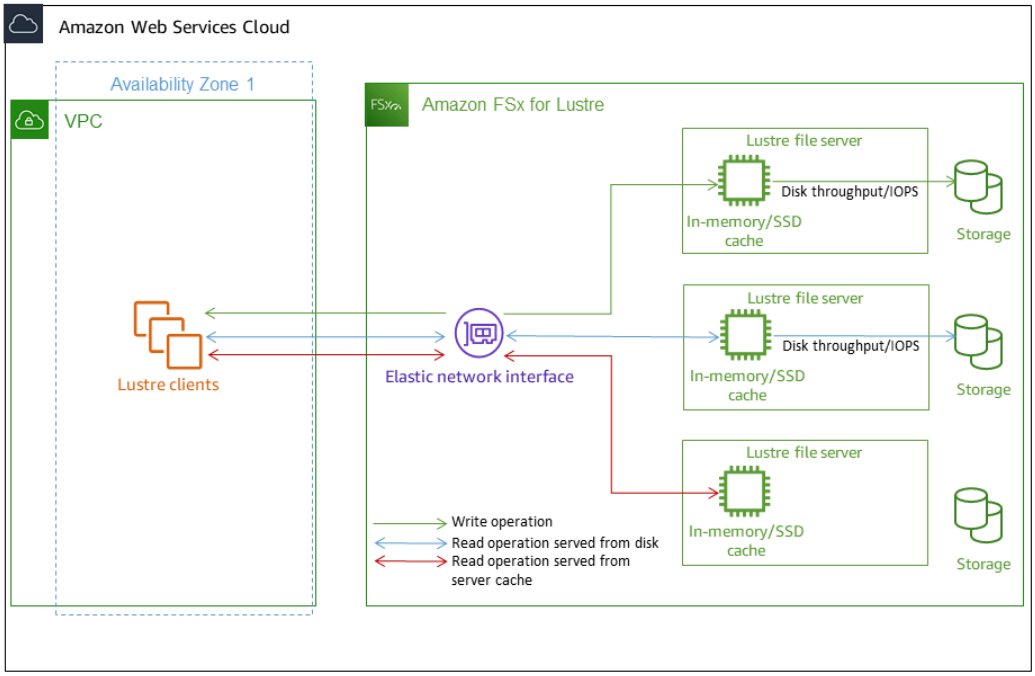
널리 사용되는 고성능 파일 시스템인 Lustre를 기반으로 구축된 Amazon FSx for Lustre는 파일 시스템 크기에 따라 선형적으로 증가하는 스케일 아웃 성능을 제공합니다. Lustre 파일 시스템은 여러 파일 서버와 디스크에서 수평적으로 확장됩니다. 이러한 확장을 통해 각 클라이언트는 각 디스크에 저장된 데이터에 직접 액세스하여 기존 파일 시스템에 존재하는 여러 병목 현상을 없앨 수 있습니다. Amazon FSx for Lustre는 Lustre의 확장 가능한 아키텍처를 기반으로 구축되어 많은 클라이언트에서 높은 수준의 성능을 지원합니다.

주제

- FSx for Lustre 파일 시스템의 작동 방식
- 파일 시스템 성능 총계
- 파일 시스템 메타데이터 성능
- 파일 시스템 스토리지 레이아웃
- 파일 시스템의 스트라이핑 데이터
- 성능 및 사용량 모니터링
- 성능 팁

FSx for Lustre 파일 시스템의 작동 방식

각 FSx for Lustre 파일 시스템은 클라이언트가 통신하는 파일 서버와 데이터를 저장하는 각 파일 서버에 연결된 디스크 세트로 구성됩니다. 각 파일 서버는 고속 인 메모리 캐시를 사용하여 가장 자주 액세스하는 데이터의 성능을 향상시킵니다. HDD 기반 파일 시스템에 SSD 기반 읽기 캐시를 제공하여 가장 자주 액세스하는 데이터의 성능을 더욱 향상시킬 수도 있습니다. 클라이언트가 인메모리 또는 SSD 캐시에 저장된 데이터에 액세스할 때 파일 서버는 디스크에서 데이터를 읽을 필요가 없으므로 지연 시간이 줄어들고 총 처리량을 늘릴 수 있습니다. 다음 다이어그램은 쓰기 작업, 디스크에서 제공되는 읽기 작업, 인 메모리 또는 SSD 캐시에서 제공되는 읽기 작업의 경로를 보여줍니다.



파일 서버의 인 메모리 또는 SSD 캐시에 저장된 데이터를 읽을 때 파일 시스템 성능은 네트워크 처리량에 따라 결정됩니다. 파일 시스템에 데이터를 쓰거나 인 메모리 캐시에 저장되지 않은 데이터를 읽을 때 파일 시스템 성능은 네트워크 처리량과 디스크 처리량 중 낮은 값에 따라 결정됩니다.

SSD 캐시로 HDD Lustre 파일 시스템을 프로비저닝하면 Amazon FSx는 파일 시스템 HDD 스토리지 용량의 20%에 맞게 자동으로 크기가 조정되는 SSD 캐시를 생성합니다. 이렇게 하면 자주 액세스하는 파일에 대해 1밀리초 미만의 지연 시간과 더 높은 IOPS를 제공할 수 있습니다.

파일 시스템 성능 총계

FSx for Lustre 파일 시스템이 지원하는 처리량은 스토리지 용량에 비례합니다. Amazon FSx for Lustre 파일 시스템은 수백 GBps의 처리량과 수백만 IOPS까지 확장할 수 있습니다. Amazon FSx for Lustre는 수천 개의 컴퓨팅 인스턴스에서 동일한 파일 또는 디렉터리에 대한 동시 액세스도 지원합니다. 이 액세스를 통해 애플리케이션 메모리에서 스토리지로 신속하게 데이터를 체크포인팅할 수 있으며, 이는 고성능 컴퓨팅(HPC)에서 흔히 사용되는 기술입니다. 파일 시스템을 생성한 후 언제든지 필요에 따라 스토리지 용량과 처리량을 늘릴 수 있습니다. 자세한 내용은 스토리지 용량 관리 섹션을 참조하세요.

FSx for Lustre 파일 시스템은 네트워크 I/O 크레딧 메커니즘을 이용해 평균 대역폭 활용도를 기준으로 네트워크 대역폭을 할당하는 버스트 읽기 처리량을 제공합니다. 이러한 파일 시스템의 네트워크 대역

폭 사용량이 기준 한도 미만으로 떨어지면 크레딧이 발생하는데, 이 크레딧은 네트워크 데이터를 전송할 때 사용할 수 있습니다.

다음 표에는 FSx for Lustre 배포 옵션이 설계된 성능이 나와 있습니다.

SSD 스토리지 옵션의 파일 시스템 성능

배포 유형	네트워크 처리량(프로비저닝된 스토리지의 MB/s/TiB)		네트워크 IOPS(프로비저닝된 스토리지의 IOPS/TiB)	캐시 스토리지(RAM의 GiB/프로비저닝된 스토리지의 TiB)	파일 작업당 디스크 지연 시간(밀리초, P50)	디스크 처리량(프로비저닝된 스토리지 또는 SSD 캐시의 TiB당 MBps)	
	기준	버스트				기준	버스트
스크래치_2	200	1300	x0,000(기본)	6.7	메타데이터: 밀리초 미만	200(읽기) 100(쓰기)	-
영구-125	320	1300	x00,000 버스트	3.4	데이터: 밀리초 미만	125	500
영구-250	640	1300		6.8		250	500
영구-500	1300	-		13.7		500	-
영구-1000	2600	-		27.3		1000	-

HDD 스토리지 옵션의 파일 시스템 성능

배포 유형	네트워크 처리량(프로비저닝된 스토리지 또는 SSD 캐시의 MB/s/TiB)		네트워크 IOPS(프로비저닝된 스토리지의 IOPS/TiB)	캐시 스토리지(RAM의 GiB/프로비저닝된 스토리지의 TiB)	파일 작업당 디스크 지연 시간(밀리초, P50)	디스크 처리량(프로비저닝된 스토리지 또는 SSD 캐시의 TiB당 MBps)	
	기준	버스트				기준	버스트
영구-12							
HDD 스토리지	40	375*	x0,000(기본) x00,000 버스트	0.4 메모리	메타데이터: 밀리초 미만 데이터: 한 자릿수 ms	12	80(읽기) 50(쓰기)
SSD 읽기 캐시	200	1,900		200 SSD 캐시	데이터: 밀리초 미만	200	-
영구-40							
HDD 스토리지	150	1,300*	x0,000(기본) x00,000 버스트	1.5	메타데이터: 밀리초 미만 데이터: 한 자릿수 ms	40	250(읽기) 150(쓰기)
SSD 읽기 캐시	750	6500		200 SSD 캐시	데이터: 밀리초 미만	200	-

이전 세대 SSD 스토리지 옵션의 파일 시스템 성능

배포 유형	네트워크 처리량(프로비저닝된 스토리지의 TiB당 MB/s)		네트워크 IOPS(프로비저닝된 스토리지의 TiB당 IOPS)	캐시 스토리지(프로비저닝된 스토리지의 TiB당 GiB)	파일 작업당 디스크 지연 시간(밀리초, P50)	디스크 처리량(프로비저닝된 스토리지 또는 SSD 캐시의 TiB당 MB/s)	
	기준	버스트				기준	버스트
영구-50	250	1,300*	x0,000(기본)	2.2 램	메타데이터: 밀리초 미만	50	240
영구-100	500	1,300*		4.4 램		100	240
영구-200	750	1,300*	x00,000 버스트	8.8 램	데이터: 밀리초 미만	200	240

> **Note**
>
> *다음의 영구 파일 시스템은 스토리지 TiB당 최대 530MB/s의 네트워크 버스트를 AWS 리전 제공합니다: 아프리카 (케이프타운), 아시아 태평양 (홍콩), 아시아 태평양 (오사카), 아시아 태평양 (싱가포르), 캐나다 (중부), 유럽 (프랑크푸르트), 유럽 (런던), 유럽 (밀라노), 유럽 (스톡홀름), 중동 (바레인), 남아메리카 (상파울루), 중국 및 미국 서부 (로스앤젤레스).

예제: 기준 및 버스트 처리량 총계

다음 예제는 스토리지 용량과 디스크 처리량이 파일 시스템 성능에 미치는 영향을 보여줍니다.

스토리지 용량이 4.8TiB이고 스토리지 유닛당 처리량이 TiB당 50MB/s인 영구 파일 시스템은 총 기준 디스크 처리량이 240MB/s이고 버스트 디스크 처리량은 1.152GB/s입니다.

파일 시스템 크기와 관계없이 Amazon FSx for Lustre는 파일 작업에 대해 1밀리초 미만의 일관된 지연 시간을 제공합니다.

파일 시스템 메타데이터 성능

파일 시스템 메타데이터 초당 IO 작업 수 (IOPS) 에 따라 초당 생성, 나열, 읽기 및 삭제할 수 있는 파일 및 디렉터리 수가 결정됩니다. 메타데이터 IOPS는 프로비저닝한 스토리지 용량을 기반으로 FSx for Lustre 파일 시스템에서 자동으로 프로비저닝됩니다.

Persistent_2 파일 시스템을 사용하면 스토리지 용량과 관계없이 메타데이터 IOPS를 프로비저닝할 수 있으며 파일 시스템에서 구동되는 메타데이터 IOPS 클라이언트 인스턴스의 수와 유형에 대한 가시성을 높일 수 있습니다.

FSx for Lustre Persistent_2 파일 시스템을 사용하면 프로비저닝하는 메타데이터 IOPS 수와 메타데이터 작업 유형에 따라 파일 시스템이 지원할 수 있는 메타데이터 작업 속도가 결정됩니다. 프로비저닝하는 메타데이터 IOPS 수준에 따라 파일 시스템의 메타데이터 디스크에 프로비저닝되는 IOPS 수가 결정됩니다.

작업 유형	프로비저닝된 각 메타데이터 (IOPS) 에 대해 초당 처리할 수 있는 작업
파일 생성, 열기	2

작업 유형	프로비저닝된 각 메타데이터 (IOPS) 에 대해 초당 처리할 수 있는 작업
파일 삭제	1
디렉토리 생성, 이름 변경	0.1
디렉터리 삭제	0.2

자동 모드 또는 사용자 프로비저닝 모드를 사용하여 메타데이터 IOPS를 프로비저닝하도록 선택할 수 있습니다. 자동 모드에서 Amazon FSx는 아래 표에 따라 파일 시스템의 스토리지 용량을 기반으로 메타데이터 IOPS를 자동으로 프로비저닝합니다.

파일 시스템 스토리지 용량	자동 모드에 포함된 메타데이터 IOPS
1200 GiB	1500
2400 GiB	3000
4800—9600기가바이트	6000
12000—45600 기가바이트	12000
48000 기가바이트 이상	24000 기가바이트당 12000 IOPS

사용자 프로비저닝 모드에서는 프로비저닝할 메타데이터 IOPS 수를 선택적으로 지정할 수 있습니다. 파일 시스템의 기본 메타데이터 IOPS 수보다 많이 프로비저닝된 메타데이터 IOPS에 대한 비용을 지불합니다.

파일 시스템 스토리지 레이아웃

Lustre의 모든 파일 데이터는 오브젝트 스토리지 타겟(OST)이라는 스토리지 볼륨에 저장됩니다. 모든 파일 메타데이터(파일 이름, 타임스탬프, 권한 등 포함)는 메타데이터 대상(MDT)이라는 스토리지 볼륨에 저장됩니다. Amazon FSx for Lustre 파일 시스템은 하나 이상의 MDT와 여러 개의 OST로 구성되어 있습니다. 각 OST의 크기는 파일 시스템의 배포 유형에 따라 약 1~2TiB입니다. Amazon FSx for Lustre는 파일 시스템을 구성하는 OST 전체에 파일 데이터를 분산하여 스토리지 용량과 처리량 및 IOPS 부하의 균형을 유지합니다.

파일 시스템을 구성하는 MDT 및 OST의 스토리지 사용량을 보려면 파일 시스템이 마운트된 클라이언트에서 다음 명령을 실행합니다.

```
lfs df -h mount/path
```

이 명령의 출력은 다음과 같습니다.

Example

```
UUID                       bytes        Used   Available Use% Mounted on
mountname-MDT0000_UUID      68.7G        5.4M       68.7G   0% /fsx[MDT:0]
mountname-OST0000_UUID       1.1T        4.5M        1.1T   0% /fsx[OST:0]
mountname-OST0001_UUID       1.1T        4.5M        1.1T   0% /fsx[OST:1]

filesystem_summary:          2.2T        9.0M        2.2T   0% /fsx
```

파일 시스템의 스트라이핑 데이터

파일 스트라이핑으로 파일 시스템의 처리량 성능을 최적화할 수 있습니다. Amazon FSx for Lustre는 모든 스토리지 서버에서 데이터가 제공되도록 하기 위해 파일을 여러 OST에 자동으로 분산합니다. 파일이 여러 OST에 스트라이핑되는 방식을 구성하여 파일 수준에서 동일한 개념을 적용할 수 있습니다.

스트라이핑이란 파일을 여러 청크로 나눈 다음 여러 OST에 저장할 수 있다는 뜻입니다. 파일이 여러 OST에 스트라이핑되면 파일에 대한 읽기 또는 쓰기 요청이 해당 OST 전체에 분산되어 애플리케이션이 처리할 수 있는 총 처리량 또는 IOPS가 증가합니다.

Amazon FSx for Lustre 파일 시스템의 기본 레이아웃은 다음과 같습니다.

- 2020년 12월 18일 이전에 생성된 파일 시스템의 경우 기본 레이아웃은 스트라이프 수를 1로 지정합니다. 즉, 다른 레이아웃을 지정하지 않는 한 표준 Linux 도구를 사용하여 Amazon FSx for Lustre에서 만든 각 파일은 단일 디스크에 저장됩니다.
- 2020년 12월 18일 이후에 생성된 파일 시스템의 경우 기본 레이아웃은 크기가 1GiB 미만인 파일이 하나의 스트라이프에 저장되는 프로그레시브 파일 레이아웃이고 큰 파일에는 스트라이프 수 5가 할당됩니다.
- 2023년 8월 25일 이후에 생성된 파일 시스템의 경우 기본 레이아웃은 프로그레시브 파일 레이아웃에서 설명하는 4구성 요소 프로그레시브 파일 레이아웃입니다.
- 생성 날짜와 상관없이 모든 파일 시스템에서 Amazon S3에서 가져온 파일은 기본 레이아웃을 사용하지 않고 대신 파일 시스템 `ImportedFileChunkSize` 파라미터의 레이아웃을 사용합니다. S3

에서 가져온 파일 크기가 ImportedFileChunkSize보다 크면 스트라이프 수가 (FileSize / ImportedFileChunksize) + 1인 여러 OST에 저장됩니다. ImportedFileChunkSize의 기본값은 1GiB입니다.

`lfs getstripe` 명령을 사용하여 파일 또는 디렉터리의 레이아웃 구성을 볼 수 있습니다.

```
lfs getstripe path/to/filename
```

이 명령은 파일의 스트라이프 수, 스트라이프 크기 및 스트라이프 오프셋을 보고합니다. 스트라이프 수는 파일이 스트라이핑된 OST 수입니다. 스트라이프 크기는 OST에 저장된 연속 데이터의 양입니다. 스트라이프 오프셋은 파일이 스트라이핑되는 첫 번째 OST의 인덱스입니다.

스트라이핑 구성 수정

파일의 레이아웃 파라미터는 파일이 처음 생성될 때 설정됩니다. `lfs setstripe` 명령을 사용하여 지정된 레이아웃에 비어 있는 새 파일을 생성합니다.

```
lfs setstripe filename --stripe-count number_of_OSTs
```

이 `lfs setstripe` 명령은 새 파일의 레이아웃에만 영향을 줍니다. 파일을 만들기 전에 이 명령을 사용하여 파일의 레이아웃을 지정할 수 있습니다. 디렉터리의 레이아웃을 정의할 수도 있습니다. 디렉터리에 설정된 레이아웃은 해당 디렉터리에 추가되는 모든 새 파일에 적용되지만 기존 파일에는 적용되지 않습니다. 새로 만드는 모든 하위 디렉터리도 새 레이아웃을 상속하며, 이 레이아웃은 해당 하위 디렉터리 내에 새로 만드는 모든 파일 또는 디렉터리에 적용됩니다.

기존 파일의 레이아웃을 수정하려면 `lfs migrate` 명령을 사용합니다. 이 명령은 필요에 따라 파일을 복사하여 명령에서 지정한 레이아웃에 따라 내용을 배포합니다. 예를 들어 파일을 추가하거나 크기를 늘려도 스트라이프 수는 변경되지 않으므로 파일을 마이그레이션하여 파일 레이아웃을 변경해야 합니다. 또는 `lfs setstripe` 명령을 사용하여 새 파일을 만들어 레이아웃을 지정하고 원본 내용을 새 파일에 복사한 다음 새 파일의 이름을 변경하여 원본 파일을 대체할 수 있습니다.

기본 레이아웃 구성이 워크로드에 최적화되지 않는 경우가 있을 수 있습니다. 예를 들어, 수십 개의 OST와 수 기가바이트의 파일이 있는 파일 시스템의 경우, 기본 스트라이프 수 값인 5개 OST를 초과하여 파일을 스트라이핑하면 성능이 향상될 수 있습니다. 스트라이프 수가 적은 대용량 파일을 만들면 I/O 성능 병목 현상이 발생할 수 있으며 OST가 가득 찰 수도 있습니다. 이 경우 이러한 파일에 대해 스트라이프 수가 더 많은 디렉터리를 만들 수 있습니다.

대용량 파일(특히 크기가 1GB보다 큰 파일)의 스트라이프 레이아웃을 설정하는 것은 다음과 같은 이유로 중요합니다.

- 대용량 파일을 읽고 쓸 때 여러 OST 및 관련 서버가 IOPS, 네트워크 대역폭 및 CPU 리소스를 제공할 수 있도록 하여 처리량을 개선합니다.
- 일부 OST가 전체 워크로드 성능을 제한하는 핫스팟이 될 가능성을 줄입니다.
- 대용량 파일 하나가 OST를 가득 채우지 못하여 디스크 전체 오류가 발생할 수 있는 문제를 방지합니다.

모든 사용 사례에 최적화된 단일 레이아웃 구성은 없습니다. 파일 레이아웃에 대한 자세한 지침은 Lustre.org 설명서의 파일 레이아웃(스트라이핑) 및 여유 공간 관리를 참조하세요. 다음은 일반 지침입니다.

- 스트라이프 레이아웃은 대용량 파일, 특히 파일 크기가 일반적으로 수백 메가바이트 이상인 사용 사례에서 가장 중요합니다. 이러한 이유로 새 파일 시스템의 기본 레이아웃에서는 크기가 1GiB를 초과하는 파일에 대해 스트라이프 수를 5개로 지정합니다.
- 스트라이프 수는 대용량 파일을 지원하는 시스템에 맞게 조정해야 하는 레이아웃 파라미터입니다. 스트라이프 수는 스트라이프 파일의 청크를 담을 OST 볼륨 수를 지정합니다. 예를 들어 스트라이프 수가 2이고 스트라이프 크기가 1MiB인 경우 Lustre는 파일의 1MiB 청크를 두 OST 각각에 대체 기록합니다.
- 유효 스트라이프 수는 실제 OST 볼륨 수와 지정한 스트라이프 수 값 중 적은 수입니다. 특수 스트라이프 수의 `-1` 값을 사용하여 모든 OST 볼륨에 스트라이프를 배치하도록 지정할 수 있습니다.
- 파일이 너무 작아서 모든 OST 볼륨의 공간을 차지할 수 없는 경우에도 특정 작업의 경우 Lustre가 레이아웃의 모든 OST로 네트워크 왕복을 해야 하기 때문에 작은 파일에 대해 큰 스트라이프 수를 설정하는 것은 최적이 아닙니다.
- 크기에 따라 파일 레이아웃을 변경할 수 있는 프로그레시브 파일 레이아웃(PFL)을 설정할 수 있습니다. PFL 구성을 사용하면 각 파일에 대해 구성을 명시적으로 설정할 필요 없이 크고 작은 파일이 조합된 파일 시스템을 간편하게 관리할 수 있습니다. 자세한 내용은 프로그레시브 파일 레이아웃 섹션을 참조하세요.
- 스트라이프 크기는 기본적으로 1MiB입니다. 스트라이프 오프셋 설정은 특별한 경우에 유용할 수 있지만 일반적으로 지정하지 않고 기본값을 사용하는 것이 가장 좋습니다.

프로그레시브 파일 레이아웃

디렉터리의 프로그레시브 파일 레이아웃(PFL) 구성을 지정하여 작은 파일과 큰 파일을 채우기 전에 각각 다른 스트라이프 구성을 지정할 수 있습니다. 예를 들어 새 파일 시스템에 데이터를 쓰기 전에 최상위 디렉터리에 PFL을 설정할 수 있습니다.

PFL 구성을 지정하려면 다음 명령과 같이 `lfs setstripe` 명령을 `-E` 옵션과 함께 사용하여 크기가 다른 파일의 레이아웃 구성 요소를 지정합니다.

```
lfs setstripe -E 100M -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32 /mountname/directory
```

이 명령은 네 가지 레이아웃 구성 요소를 설정합니다.

- 첫 번째 구성 요소(`-E 100M -c 1`)는 최대 100MiB 크기의 파일에 대한 스트라이프 수 값 1을 나타냅니다.
- 두 번째 구성 요소(`-E 10G -c 8`)는 최대 10GiB 크기의 파일에 대한 스트라이프 수 8을 나타냅니다.
- 세 번째 구성 요소(`-E 100G -c 16`)는 최대 100GiB 크기의 파일에 대한 스트라이프 수가 16개임을 나타냅니다.
- 네 번째 구성 요소(`-E -1 -c 32`)는 100GiB보다 큰 파일의 스트라이프 수가 32개임을 나타냅니다.

> **Important**
>
> PFL 레이아웃으로 만든 파일에 데이터를 추가하면 해당 레이아웃 구성 요소가 모두 채워집니다. 예를 들어 위에 표시된 4가지 구성 요소 명령을 사용하여 1MiB 파일을 만든 다음 파일 끝에 데이터를 추가하면 파일의 레이아웃이 스트라이프 수 -1, 즉 시스템의 모든 OST를 포함하도록 확장됩니다. 이는 모든 OST에 데이터가 기록된다는 것을 의미하지는 않지만, 파일 길이 읽기와 같은 작업을 수행하면 모든 OST에 병렬로 요청이 전송되므로 파일 시스템에 상당한 네트워크 부하가 가중됩니다.
>
> 따라서 나중에 데이터가 추가될 수 있는 작거나 중간 길이의 파일에서는 스트라이프 수를 제한해야 한다는 점을 유의하세요. 일반적으로 새 레코드가 추가되면 로그 파일이 커지므로 Amazon FSx for Lustre는 상위 디렉터리에 지정된 기본 스트라이프 구성과 상관없이 추가 모드에서 생성되는 모든 파일에 기본 스트라이프 개수 1을 할당합니다.

2023년 8월 25일 이후에 생성된 Amazon FSx for Lustre 파일 시스템의 기본 PFL 구성은 다음 명령으로 설정됩니다.

```
lfs setstripe -E 100M -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32 /mountname
```

중대형 파일에 대한 동시 액세스 빈도가 높은 워크로드를 사용하는 고객은 4개 구성 요소 예제 레이아웃에서 볼 수 있듯이 작은 크기에서 더 많은 스트라이프를 사용하고 가장 큰 파일은 모든 OST에 스트라이핑하는 레이아웃을 사용하는 것이 좋습니다.

성능 및 사용량 모니터링

Amazon FSx for Lustre는 1분마다 각 디스크 (MDT 및 OST) 에 대한 사용량 지표를 Amazon에 내보냅니다. CloudWatch

전체 파일 시스템 사용량 세부 정보를 보려면 각 지표의 합계 통계를 보면 됩니다. 예를 들어, `DataReadBytes` 통계 합계는 파일 시스템의 모든 OST에서 확인한 총 읽기 처리량을 보고합니다. 마찬가지로 `FreeDataStorageCapacity` 통계 합계는 파일 시스템의 파일 데이터에 사용할 수 있는 총 스토리지 용량을 보고합니다.

파일 시스템 성능 모니터링에 대한 자세한 내용은 Amazon FSx for Lustre 모니터링 섹션을 참조하세요.

성능 팁

Amazon FSx for Lustre를 사용할 때는 다음과 같은 성능 관련 팁을 유의하세요. 서비스 제한은 할당량 섹션을 참조하세요.

- 평균 I/O 크기 - Amazon FSx for Lustre는 네트워크 파일 시스템이므로 각 파일 작업은 클라이언트와 Amazon FSx for Lustre를 왕복하여 진행되므로 약간의 지연 시간 오버헤드가 발생합니다. 이렇게 작업당 지연 시간이 있으므로 평균 I/O 크기가 늘어남에 따라 전체 처리량도 대개 증가합니다. 대량의 데이터에 대해 오버헤드가 분할 사용되기 때문입니다.
- 요청 모델 - 파일 시스템에 대한 비동기 쓰기를 활성화하여, 보류 중인 쓰기 작업을 Amazon EC2 인스턴스에서 버퍼링했다가 Amazon FSx for Lustre에 비동기식으로 기록합니다. 비동기 쓰기는 일반적으로 지연 시간이 더 짧습니다. 비동기 쓰기를 수행하는 경우 커널은 캐싱에 추가 메모리를 사용합니다. 동기 쓰기를 활성화한 파일 시스템은 Amazon FSx for Lustre에 동기 요청을 보냅니다. 모든 작업은 클라이언트와 Amazon FSx for Lustre 간을 왕복합니다.

  > **Note**
  >
  > 선택한 요청 모델은 일관성(여러 Amazon EC2 인스턴스를 사용하는 경우)과 속도를 절충합니다.

- 디렉터리 크기 제한 - Persistent_2 FSx for Lustre 파일 시스템에서 최적의 메타데이터 성능을 달성하려면 각 디렉터리를 10만 개 미만의 파일로 제한하십시오. 디렉터리의 파일 수를 제한하면 파일 시스템이 상위 디렉터리를 잠그는 데 필요한 시간을 줄일 수 있습니다.
- Amazon EC2 인스턴스 - 다수의 읽기 및 쓰기 작업을 수행하는 애플리케이션은 그렇지 않은 애플리케이션보다 메모리 또는 컴퓨팅 용량이 훨씬 더 많이 필요합니다. 컴퓨팅 집약적 워크로드를 위한

Amazon EC2 인스턴스를 시작할 때 애플리케이션에 필요한 만큼의 충분한 리소스가 있는 인스턴스 유형을 선택합니다. Amazon FSx for Lustre 파일 시스템의 성능 특성은 EBS 최적화 인스턴스의 사용과 관련이 없습니다.

- 최적의 성능을 위한 권장 클라이언트 인스턴스 조정

  1. 모든 클라이언트 인스턴스 유형 및 크기에 대해 다음 조정을 적용하는 것이 좋습니다.

     ```
     sudo lctl set_param osc.*.max_dirty_mb=64
     ```

  2. 메모리가 64GiB를 초과하는 클라이언트 인스턴스 유형의 경우 다음 조정을 적용하는 것이 좋습니다.

     ```
     lctl set_param ldlm.namespaces.*.lru_max_age=600000
     ```

  3. vCPU 코어가 64개 이상인 클라이언트 인스턴스 유형의 경우 다음 조정을 적용하는 것이 좋습니다.

     ```
     echo "options ptlrpc ptlrpcd_per_cpt_max=32" >> /etc/modprobe.d/modprobe.conf
     echo "options ksocklnd credits=2560" >> /etc/modprobe.d/modprobe.conf

     # reload all kernel modules to apply the above two settings
     sudo reboot
     ```

     클라이언트가 마운트된 후에는 다음 튜닝을 적용해야 합니다.

     ```
     sudo lctl set_param osc.*OST*.max_rpcs_in_flight=32
     sudo lctl set_param mdc.*.max_rpcs_in_flight=64
     sudo lctl set_param mdc.*.max_mod_rpcs_in_flight=50
     ```

  `lctl set_param`은 재부팅하는 경우 지속되지 않는 것으로 알려져 있습니다. 이러한 파라미터는 클라이언트 측에서 영구적으로 설정할 수 없으므로 부팅 크론 작업을 구현하여 권장 튜닝으로 구성을 설정하는 것이 좋습니다.

- OST 간 워크로드 밸런스 - 파일 시스템이 제공할 수 있는 총 처리량(스토리지 TiB당 200MB/s)을 워크로드로 인해 주도하지 못하는 경우도 있습니다. 그렇다면 CloudWatch 지표를 사용하여 워크로드 I/O 패턴의 불균형으로 인해 성능이 영향을 받는지 여부를 해결할 수 있습니다. 이것이 원인인지 확인하려면 Amazon FSx for Lustre의 최대 CloudWatch 측정치를 참조하십시오.

  경우에 따라 이 통계는 처리량(1.2TiB Amazon FSx for Lustre 디스크 한 개의 처리량 용량) 240MBps 이상의 부하를 보여줍니다. 이 경우 워크로드가 디스크 전체에 고르게 분산되지 않습니다.

이 경우 `lfs setstripe` 명령을 사용하여 워크로드에서 가장 자주 액세스하는 파일의 스트라이핑을 수정할 수 있습니다. 성능을 최적화하려면 파일 시스템을 구성하는 모든 OST에서 처리량이 높은 파일을 스트라이핑합니다.

데이터 리포지토리에서 파일을 가져오는 경우 처리량이 높은 파일을 OST 전체에 고르게 스트라이핑하는 다른 방법을 사용할 수 있습니다. 이렇게 하려면 다음 Amazon FSx for Lustre 파일 시스템을 생성할 때 `ImportedFileChunkSize` 파라미터를 수정하면 됩니다.

예를 들어, 워크로드가 7.0TiB 파일 시스템(6x 1.17TiB OST로 구성)을 사용하고 2.4GiB 파일에서 높은 처리량을 구현해야 한다고 가정해 보겠습니다. 이 경우 파일을 파일 시스템의 OST 전체에 균등하게 분산하도록 `ImportedFileChunkSize` 값을 `(2.4 GiB / 6 OSTs) = 400 MiB`로 설정할 수 있습니다.

- 메타데이터 IOPS용 Lustre 클라이언트 - 파일 시스템에 메타데이터 구성이 지정된 경우, 아마존 리눅스 2023, 아마존 리눅스 2, 레드햇/센토스/록키 리눅스 8.9 또는 9.x, 6.2 커널이 설치된 우분투 22 또는 우분투 20 OS 버전 중 하나를 사용하는 Lustre 2.15 클라이언트 또는 Lustre 2.12 클라이언트를 설치하는 것이 좋습니다.

파일 시스템 액세스

Amazon FSx를 사용하면, 또는 VPN을 통해 데이터를 가져와서 온프레미스에서 Amazon Web Services 클라우드로 컴퓨팅 집약적인 워크로드를 버스트할 수 있습니다. AWS Direct Connect 온프레미스에서 Amazon FSx 파일 시스템에 액세스하고, 필요에 따라 데이터를 파일 시스템으로 복사하고, 클라우드 내 인스턴스에서 컴퓨팅 집약적인 워크로드를 실행할 수 있습니다.

다음 섹션에서는 Linux 인스턴스에서 Amazon FSx for Lustre 파일 시스템에 액세스하는 방법을 알아볼 수 있습니다. 또한 파일 `fstab`를 사용하여 시스템을 다시 시작한 후 파일 시스템을 자동으로 다시 마운트하는 방법에 대해서도 알아볼 수 있습니다.

파일 시스템을 탑재하려면 관련 AWS 리소스를 만들고, 구성하고, 시작해야 합니다. 자세한 지침은 Amazon FSx for Lustre 시작하기 섹션을 참조하십시오. 다음으로 컴퓨팅 인스턴스에 Lustre 클라이언트를 설치하고 구성할 수 있습니다.

주제

- Lustre 파일 시스템 및 클라이언트 커널 호환성
- Lustre 클라이언트 설치 중
- Amazon Elastic Compute Cloud 인스턴스에서 마운트
- Amazon Elastic Container Service에 마운트
- 온프레미스 또는 피어링된 Amazon VPC에서 Amazon FSx 파일 시스템 마운트
- Amazon FSx 파일 시스템 자동 마운트
- 특정 파일 세트 마운트
- 파일 시스템 마운트 해제
- Amazon EC2 스팟 인스턴스 작업

Lustre 파일 시스템 및 클라이언트 커널 호환성

클라이언트 인스턴스의 Linux 커널 버전과 호환되는 FSx for Lustre 파일 시스템에는 Lustre 버전을 사용하는 것이 좋습니다.

아마존 리눅스 클라이언트

운영 체제	OS 버전	최소 커널 버전	최대 커널 버전	파일 시스템 버전		
				2.10	2.12	2.15
Amazon Linux 2023	6.1	6.1.79-99.167	6.1.79-99.167+	아니요	예	예
Amazon Linux 2	5.10	5.10.144-127.601	5.10.144-127.601+	예	예	예
			<5.10.144-127.601	예	예	아니요
	5.4	5.4.214-120.368	5.4.214-120.368+	예	예	예
			<5.4.214-120.368	예	예	아니요
	4.14	4.14.294-220.533	4.14.294-220.533+	예	예	예
			<4.14.294-220.533	예	예	아니요

우분투 클라이언트

운영 체제	OS 버전	최소 커널 버전	최대 커널 버전	파일 시스템 버전		
				2.10	2.12	2.15

운영 체제	OS 버전	최소 커널 버전	최대 커널 버전	파일 시스템 버전		
Ubuntu	22	6.2.0.1017.17~22.04	6.2.0. *	아니요	예	예
		5.15.0-1015-aws	5.15.0-1031-aws	예	예	예
	20	5.15.0-1015-aws	5.15.0+	예	예	예
		5.4.0-1011-aws	5.13.0-1031-aws	예	예	아니요

RHEL/센토스/록키 리눅스 클라이언트

운영 체제	OS 버전	아키텍처	최소 커널 버전	최대 커널 버전	파일 시스템 버전		
					2.10	2.12	2.15
RHEL/CentOS/Rocky Lin	9.4	Arm + x86	5.14.0-427.13.1	5.14.0-427.16.1	아니요	예	예
	9.3	Arm + x86	5.14.0-362.18.1	5.14.0-362.18.1	아니요	예	예
	9.0	Arm + x86	5.14.0-70.13.1	5.14.0-70.30.1	아니요	예	예
	8.9	Arm + x86	4.18.0-513*	4.18.0-513*	예	예	예

운영 체제	OS 버전	아키텍처	최소 커널 버전	최대 커널 버전	파일 시스템 버전		
	8.8	Arm + x86	4.18.0-477*	4.18.0-477*	예	예	예
	8.7	Arm + x86	4.18.0-425*	4.18.0-425*	예	예	예
	8.6	Arm + x86	4.18.0-372*	4.18.0-372*	예	예	예
	8.5	Arm + x86	4.18.0-348*	4.18.0-348*	예	예	예
	8.4	Arm + x86	4.18.0-305*	4.18.0-305*	예	예	예
RHEL/CentOS	8.3	Arm + x86	4.18.0-240*	4.18.0-240*	예	예	아니요
	8.2	Arm + x86	4.18.0-193*	4.18.0-193*	예	예	아니요
	7.9	x86	3.10.0-1160*	3.10.0-1160*	예	예	예
	7.8	x86	3.10.0-1127*	3.10.0-1127*	예	예	아니요
	7.7	x86	3.10.0-1062*	3.10.0-1062*	예	예	아니요
CentOS	7.9	Arm	4.18.0-193*	4.18.0-193*	예	예	예
	7.8	Arm	4.18.0-147*	4.18.0-147*	예	예	예

Lustre 클라이언트 설치 중

Linux 인스턴스에서 Amazon FSx for Lustre 파일 시스템을 마운트하려면 먼저 오픈 소스 Lustre 클라이언트를 설치합니다. 그런 다음 운영 체제 버전에 따라 다음 절차 중 하나를 사용합니다. 커널 지원 정보는 을 참조하십시오Lustre 파일 시스템 및 클라이언트 커널 호환성.

컴퓨팅 인스턴스가 설치 지침에 지정된 Linux 커널을 실행하지 않고, 커널을 변경할 수 없는 경우, 자체 Lustre 클라이언트를 구축할 수 있습니다. 자세한 내용은 Lustre Wiki에서 Lustre 컴파일을 참조하세요.

Amazon Linux

아마존 리눅스 2023에 Lustre 클라이언트를 설치하려면

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 실행하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

   ```
   uname -r
   ```

3. 시스템 응답을 검토하고 Amazon Linux 2023에 Lustre 클라이언트를 설치하기 위한 다음과 같은 최소 커널 요구 사항과 비교하십시오.

   - 6.1 커널 최소 요구 사항 - 6.1.79-99.167.amzn2023

   EC2 인스턴스가 최소 커널 요구 사항을 충족하는 경우 단계를 진행하여 lustre 클라이언트를 설치하십시오.

   명령이 커널 최소 요구 사항보다 낮은 결과를 반환하는 경우 다음 명령을 실행하여 커널을 업데이트하고 Amazon EC2 인스턴스를 재부팅합니다.

   ```
   sudo dnf -y update kernel && sudo reboot
   ```

   uname -r 명령을 사용해서 커널이 업데이트되었는지 확인합니다.
4. 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

   ```
   sudo dnf install -y lustre-client
   ```

Amazon Linux 2에 Lustre 클라이언트 설치

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 실행하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

```
uname -r
```

3. 시스템 응답을 검토하고 Amazon Linux 2에 Lustre 클라이언트를 설치하기 위한 다음과 같은 최소 커널 요구 사항과 비교하십시오.

   - 5.10 커널 최소 요구 사항 - 5.10.144-127.601.amzn2
   - 5.4 커널 최소 요구 사항 - 5.4.214-120.368.amzn2
   - 4.14 커널 최소 요구 사항 - 4.14.294-220.533.amzn2

   EC2 인스턴스가 최소 커널 요구 사항을 충족하는 경우 단계를 진행하여 lustre 클라이언트를 설치하십시오.

   명령이 커널 최소 요구 사항보다 낮은 결과를 반환하는 경우 다음 명령을 실행하여 커널을 업데이트하고 Amazon EC2 인스턴스를 재부팅합니다.

```
sudo yum -y update kernel && sudo reboot
```

   uname -r 명령을 사용해서 커널이 업데이트되었는지 확인합니다.
4. 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

```
sudo amazon-linux-extras install -y lustre
```

   커널을 커널 최소 요구 사항 수준으로 업그레이드할 수 없는 경우 다음 명령을 사용하여 레거시 2.10 클라이언트를 설치할 수 있습니다.

```
sudo amazon-linux-extras install -y lustre2.10
```

Amazon Linux에 Lustre 클라이언트 설치

1. 클라이언트에서 터미널을 엽니다.

2. 다음 명령을 실행하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다. Lustre 클라이언트에는 Amazon Linux 커널 `4.14, version 104` 또는 그 이상이 필요합니다.

```
uname -r
```

3. 다음 중 하나를 수행합니다.

   - 명령이 `4.14.104-78.84.amzn1.x86_64` 또는 4.14 이상의 버전을 반환하는 경우 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

     ```
     sudo yum install -y lustre-client
     ```

   - 명령이 `4.14.104-78.84.amzn1.x86_64`보다 작은 결과를 반환하는 경우 다음 명령을 실행하여 커널을 업데이트하고 Amazon EC2 인스턴스를 재부팅합니다.

     ```
     sudo yum -y update kernel && sudo reboot
     ```

     uname -r 명령을 사용해서 커널이 업데이트되었는지 확인합니다. 그런 다음 앞에서 설명한 대로 Lustre 클라이언트를 다운로드하고 설치합니다.

CentOS, Rocky Linux, Red Hat

CentOS, Red Hat 및 록키 리눅스 9.0, 9.3 또는 9.4에 Lustre 클라이언트를 설치하려면

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리에서 Red Hat Enterprise Linux(RHEL), Rocky Linux 및 CentOS와 호환되는 Lustre 클라이언트 패키지를 설치하고 업데이트할 수 있습니다. 이러한 패키지는 다운로드 이전 또는 다운로드 중에 변조되지는 않았는지 확인하는 데 도움이 되도록 서명되었습니다. 시스템에 해당하는 퍼블릭 키를 설치하지 않을 경우 리포지토리 설치가 불가능합니다.

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리 추가

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

```
curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
```

3. 다음 명령을 사용하여 키를 가져옵니다.

```
sudo rpm --import /tmp/fsx-rpm-public-key.asc
```

4. 다음 명령을 사용하여 리포지토리를 추가하고 패키지 관리자를 업데이트합니다.

```
sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/el/9/fsx-lustre-
client.repo -o /etc/yum.repos.d/aws-fsx.repo
```

Amazon FSx Lustre 클라이언트 yum 리포지토리 구성

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리는 기본적으로 지원되는 최신 CentOS, Rocky Linux 및 RHEL 9 릴리스와 함께 처음 제공되는 커널 버전과 호환되는 Lustre 클라이언트를 설치하도록 구성되어 있습니다. 사용 중인 커널 버전과 호환되는 Lustre 클라이언트를 설치하려면 리포지토리 구성 파일을 편집하면 됩니다.

이 섹션에서는 실행 중인 커널을 확인하는 방법, 리포지토리 구성을 편집해야 하는지 여부 및 구성 파일을 편집하는 방법에 대해 설명합니다.

1. 다음 명령을 사용하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

```
uname -r
```

2. 다음 중 하나를 수행합니다.

   - 명령이 `5.14.0-427*` 코드를 반환하는 경우 리포지토리 구성을 수정할 필요가 없습니다. Lustre 클라이언트 설치 절차를 계속 진행합니다.
   - 명령이 반환되면 CentOS`5.14.0-362.18.1`, Rocky Linux 및 RHEL 9.3 릴리스의 Lustre 클라이언트를 가리키도록 리포지토리 구성을 편집해야 합니다.
   - 명령이 반환되면 CentOS`5.14.0-70*`, Rocky Linux 및 RHEL 9.0 릴리스의 Lustre 클라이언트를 가리키도록 리포지토리 구성을 편집해야 합니다.

3. 다음 명령을 사용하여 특정 버전의 RHEL을 표시하도록 리포지토리 구성 파일을 편집합니다. 사용해야 하는 *`specific_RHEL_version`*RHEL 버전으로 교체하십시오.

```
sudo sed -i 's#9#specific_RHEL_version#' /etc/yum.repos.d/aws-fsx.repo
```

   예를 들어, 릴리스 9.3을 가리키려면 다음 예제와 같이 `9.3` 명령에서 를 *`specific_RHEL_version`* 로 대체하십시오.

```
sudo sed -i 's#9#9.3#' /etc/yum.repos.d/aws-fsx.repo
```

4. 다음 명령을 사용하여 yum 캐시를 지웁니다.

```
sudo yum clean all
```

Lustre 클라이언트 설치

- 다음 명령을 사용하여 리포지토리에서 패키지를 설치합니다.

```
sudo yum install -y kmod-lustre-client lustre-client
```

추가 정보 (CentOS, 록키 리눅스, 레드햇 9.0 이상)

앞의 명령은 Amazon FSx 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 패키지를 설치합니다. 리포지토리에는 소스 코드가 포함된 패키지 및 테스트가 포함된 패키지와 같은 추가 Lustre 패키지가 포함되며, 선택적으로 설치할 수 있습니다. 리포지토리에서 사용 가능한 모든 패키지를 나열하려면 다음 명령을 사용합니다.

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

업스트림 소스 코드의 타르볼과 적용한 패치 세트를 포함하는 소스 rpm을 다운로드하려면 다음 명령을 사용합니다.

```
 sudo yumdownloader --source kmod-lustre-client
```

yum 업데이트를 실행하면 가능한 경우 최신 버전의 모듈이 설치되고 기존 버전이 대체됩니다. 업데이트 시 현재 설치된 버전이 제거되지 않도록 하려면, 다음과 같은 라인을 `/etc/yum.conf` 파일에 추가합니다.

```
installonlypkgs=kernel, kernel-PAE, installonlypkg(kernel), installonlypkg(kernel-
module),
              installonlypkg(vm), multiversion(kernel), kmod-lustre-client
```

이 목록에는 `yum.conf` 매뉴얼 페이지에 지정된 기본 설치 전용 패키지와 `kmod-lustre-client` 패키지가 포함됩니다.

CentOS 및 Red Hat 8.2—8.9 또는 록키 리눅스 8.4—8.9에 Lustre 클라이언트를 설치하려면

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리에서 Red Hat Enterprise Linux(RHEL), Rocky Linux 및 CentOS와 호환되는 Lustre 클라이언트 패키지를 설치하고 업데이트할 수 있습니다. 이러한 패키지는 다운로드 이전 또는 다운로드 중에 변조되지는 않았는지 확인하는 데 도움이 되도록 서명되었습니다. 시스템에 해당하는 퍼블릭 키를 설치하지 않을 경우 리포지토리 설치가 불가능합니다.

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리 추가

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

```
curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
```

3. 다음 명령을 사용하여 키를 가져옵니다.

```
sudo rpm --import /tmp/fsx-rpm-public-key.asc
```

4. 다음 명령을 사용하여 리포지토리를 추가하고 패키지 관리자를 업데이트합니다.

```
sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/el/8/fsx-lustre-client.repo -o /etc/yum.repos.d/aws-fsx.repo
```

Amazon FSx Lustre 클라이언트 yum 리포지토리 구성

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리는 기본적으로 지원되는 최신 CentOS, Rocky Linux 및 RHEL 8 릴리스와 함께 처음 제공되는 커널 버전과 호환되는 Lustre 클라이언트를 설치하도록 구성되어 있습니다. 사용 중인 커널 버전과 호환되는 Lustre 클라이언트를 설치하려면 리포지토리 구성 파일을 편집하면 됩니다.

이 섹션에서는 실행 중인 커널을 확인하는 방법, 리포지토리 구성을 편집해야 하는지 여부 및 구성 파일을 편집하는 방법에 대해 설명합니다.

1. 다음 명령을 사용하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

```
uname -r
```

2. 다음 중 하나를 수행합니다.

- 명령이 `4.18.0-513*` 코드를 반환하는 경우 리포지토리 구성을 수정할 필요가 없습니다. Lustre 클라이언트 설치 절차를 계속 진행합니다.
- 명령이 반환되면 CentOS`4.18.0-477*`, Rocky Linux 및 RHEL 8.8 릴리스의 Lustre 클라이언트를 가리키도록 리포지토리 구성을 편집해야 합니다.
- 명령이 `4.18.0-425*` 코드를 반환하는 경우, CentOS, Rocky Linux 및 RHEL 8.7 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.
- 명령이 `4.18.0-372*` 코드를 반환하는 경우, CentOS, Rocky Linux 및 RHEL 8.6 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.
- 명령이 `4.18.0-348*` 코드를 반환하는 경우, CentOS, Rocky Linux 및 RHEL 8.5 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.
- 명령이 `4.18.0-305*` 코드를 반환하는 경우, CentOS, Rocky Linux 및 RHEL 8.4 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.
- 명령이 `4.18.0-240*` 코드를 반환하는 경우, CentOS 및 RHEL 8.3 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.
- 명령이 `4.18.0-193*` 코드를 반환하는 경우, CentOS 및 RHEL 8.2 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.

3. 다음 명령을 사용하여 특정 버전의 RHEL을 표시하도록 리포지토리 구성 파일을 편집합니다.

```
sudo sed -i 's#8#specific_RHEL_version#' /etc/yum.repos.d/aws-fsx.repo
```

예를 들어 릴리스 8.8을 가리키려면 명령에서 를 로 대체하십시오. *specific_RHEL_version* `8.8`

```
sudo sed -i 's#8#8.8#' /etc/yum.repos.d/aws-fsx.repo
```

4. 다음 명령을 사용하여 yum 캐시를 지웁니다.

```
sudo yum clean all
```

Lustre 클라이언트 설치

- 다음 명령을 사용하여 리포지토리에서 패키지를 설치합니다.

```
sudo yum install -y kmod-lustre-client lustre-client
```

추가 정보(CentOS, Rocky Linux, Red Hat 8.2 또는 그 이상)

앞의 명령은 Amazon FSx 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 패키지를 설치합니다. 리포지토리에는 소스 코드가 포함된 패키지 및 테스트가 포함된 패키지와 같은 추가 Lustre 패키지가 포함되며, 선택적으로 설치할 수 있습니다. 리포지토리에서 사용 가능한 모든 패키지를 나열하려면 다음 명령을 사용합니다.

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

업스트림 소스 코드의 타르볼과 적용한 패치 세트를 포함하는 소스 rpm을 다운로드하려면 다음 명령을 사용합니다.

```
 sudo yumdownloader --source kmod-lustre-client
```

yum 업데이트를 실행하면 가능한 경우 최신 버전의 모듈이 설치되고 기존 버전이 대체됩니다. 업데이트 시 현재 설치된 버전이 제거되지 않도록 하려면, 다음과 같은 라인을 `/etc/yum.conf` 파일에 추가합니다.

```
installonlypkgs=kernel, kernel-PAE, installonlypkg(kernel), installonlypkg(kernel-
module),
              installonlypkg(vm), multiversion(kernel), kmod-lustre-client
```

이 목록에는 `yum.conf` 매뉴얼 페이지에 지정된 기본 설치 전용 패키지와 `kmod-lustre-client` 패키지가 포함됩니다.

CentOS 및 Red Hat 7.7, 7.8 또는 7.9(x86_64 인스턴스)에 Lustre 클라이언트 설치

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리에서 Red Hat Enterprise Linux(RHEL) 및 CentOS와 호환되는 Lustre 클라이언트 패키지를 설치하고 업데이트할 수 있습니다. 이러한 패키지는 다운로드 이전 또는 다운로드 중에 변조되지는 않았는지 확인하는 데 도움이 되도록 서명되었습니다. 시스템에 해당하는 퍼블릭 키를 설치하지 않을 경우 리포지토리 설치가 불가능합니다.

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리 추가

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

   ```
   curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-
   key.asc -o /tmp/fsx-rpm-public-key.asc
   ```

3. 다음 명령을 사용하여 키를 가져옵니다.

```
sudo rpm --import /tmp/fsx-rpm-public-key.asc
```

4. 다음 명령을 사용하여 리포지토리를 추가하고 패키지 관리자를 업데이트합니다.

```
sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/el/7/fsx-lustre-
client.repo -o /etc/yum.repos.d/aws-fsx.repo
```

Amazon FSx Lustre 클라이언트 yum 리포지토리 구성

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리는 기본적으로 지원되는 최신 CentOS 및 RHEL 7 릴리스와 함께 처음 제공되는 커널 버전과 호환되는 Lustre 클라이언트를 설치하도록 구성되어 있습니다. 사용 중인 커널 버전과 호환되는 Lustre 클라이언트를 설치하려면 리포지토리 구성 파일을 편집하면 됩니다.

이 섹션에서는 실행 중인 커널을 확인하는 방법, 리포지토리 구성을 편집해야 하는지 여부 및 구성 파일을 편집하는 방법에 대해 설명합니다.

1. 다음 명령을 사용하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

```
uname -r
```

2. 다음 중 하나를 수행합니다.

   - 명령이 `3.10.0-1160*` 코드를 반환하는 경우 리포지토리 구성을 수정할 필요가 없습니다. Lustre 클라이언트 설치 절차를 계속 진행합니다.
   - 명령이 `3.10.0-1127*` 코드를 반환하는 경우 CentOS 및 RHEL 7.8 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.
   - 명령이 `3.10.0-1062*` 코드를 반환하는 경우 CentOS 및 RHEL 7.7 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.

3. 다음 명령을 사용하여 특정 버전의 RHEL을 표시하도록 리포지토리 구성 파일을 편집합니다.

```
sudo sed -i 's#7#specific_RHEL_version#' /etc/yum.repos.d/aws-fsx.repo
```

   릴리스 7.8을 표시하려면 명령에서 *specific_RHEL_version*을 `7.8`로 대체합니다.

```
sudo sed -i 's#7#7.8#' /etc/yum.repos.d/aws-fsx.repo
```

릴리스 7.7을 표시하려면 명령에서 *specific_RHEL_version*을 7.7로 대체합니다.

```
sudo sed -i 's#7#7.7#' /etc/yum.repos.d/aws-fsx.repo
```

4. 다음 명령을 사용하여 yum 캐시를 지웁니다.

```
sudo yum clean all
```

Lustre 클라이언트 설치

- 다음 명령을 사용하여 리포지토리에서 Lustre 클라이언트 패키지를 설치합니다.

```
sudo yum install -y kmod-lustre-client lustre-client
```

추가 정보(CentOS 및 Red Hat 7.7 이상)

앞의 명령은 Amazon FSx 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 패키지를 설치합니다. 리포지토리에는 소스 코드가 포함된 패키지 및 테스트가 포함된 패키지와 같은 추가 Lustre 패키지가 포함되며, 선택적으로 설치할 수 있습니다. 리포지토리에서 사용 가능한 모든 패키지를 나열하려면 다음 명령을 사용합니다.

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

업스트림 소스 코드의 타르볼과 적용한 패치 세트를 포함하는 소스 rpm을 다운로드하려면 다음 명령을 사용합니다.

```
 sudo yumdownloader --source kmod-lustre-client
```

yum 업데이트를 실행하면 가능한 경우 최신 버전의 모듈이 설치되고 기존 버전이 대체됩니다. 업데이트 시 현재 설치된 버전이 제거되지 않도록 하려면 다음과 같은 라인을 /etc/yum.conf 파일에 추가합니다.

```
installonlypkgs=kernel, kernel-big-mem, kernel-enterprise, kernel-smp,
              kernel-debug, kernel-unsupported, kernel-source, kernel-devel, kernel-
PAE,
              kernel-PAE-debug, kmod-lustre-client
```

이 목록에는 `yum.conf` 매뉴얼 페이지에 지정된 기본 설치 전용 패키지와 `kmod-lustre-client` 패키지가 포함됩니다.

CentOS 7.8 또는 7.9에 Lustre 클라이언트를 설치하려면 (ARM 기반 그래비톤 기반 인스턴스) AWS

Arm 기반 AWS Graviton 구동 EC2 인스턴스용 CentOS 7과 호환되는 Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리에서 Lustre 클라이언트 패키지를 설치하고 업데이트할 수 있습니다. 이러한 패키지는 다운로드 이전 또는 다운로드 중에 변조되지는 않았는지 확인하는 데 도움이 되도록 서명되었습니다. 시스템에 해당하는 퍼블릭 키를 설치하지 않을 경우 리포지토리 설치가 불가능합니다.

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리 추가

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

   ```
   curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
   ```

   ```
   curl https://fsx-lustre-client-repo-public-keys.s3.amazonaws.cn/fsx-rpm-public-key.asc -o /tmp/fsx-rpm-public-key.asc
   ```

3. 다음 명령을 사용하여 키를 가져옵니다.

   ```
   sudo rpm --import /tmp/fsx-rpm-public-key.asc
   ```

4. 다음 명령을 사용하여 리포지토리를 추가하고 패키지 관리자를 업데이트합니다.

   ```
   sudo curl https://fsx-lustre-client-repo.s3.amazonaws.com/centos/7/fsx-lustre-client.repo -o /etc/yum.repos.d/aws-fsx.repo
   ```

Amazon FSx Lustre 클라이언트 yum 리포지토리 구성

Amazon FSx Lustre 클라이언트 yum 패키지 리포지토리는 기본적으로 지원되는 최신 CentOS 7 릴리스와 함께 처음 제공된 커널 버전과 호환되는 Lustre 클라이언트를 설치하도록 구성되어 있습니다. 사용 중인 커널 버전과 호환되는 Lustre 클라이언트를 설치하려면 리포지토리 구성 파일을 편집하면 됩니다.

이 섹션에서는 실행 중인 커널을 확인하는 방법, 리포지토리 구성을 편집해야 하는지 여부 및 구성 파일을 편집하는 방법에 대해 설명합니다.

1. 다음 명령을 사용하여 컴퓨팅 인스턴스에서 현재 실행 중인 커널을 확인합니다.

```
uname -r
```

2. 다음 중 하나를 수행합니다.

   - 명령이 `4.18.0-193*` 코드를 반환하는 경우 리포지토리 구성을 수정할 필요가 없습니다. Lustre 클라이언트 설치 절차를 계속 진행합니다.
   - 명령이 `4.18.0-147*` 코드를 반환하는 경우 CentOS 7.8 릴리스의 Lustre 클라이언트를 표시하도록 리포지토리 구성을 편집해야 합니다.

3. 다음 명령을 사용하여 CentOS 7.8 릴리스를 표시하도록 리포지토리 구성 파일을 편집합니다.

```
sudo sed -i 's#7#7.8#' /etc/yum.repos.d/aws-fsx.repo
```

4. 다음 명령을 사용하여 yum 캐시를 지웁니다.

```
sudo yum clean all
```

Lustre 클라이언트 설치

- 다음 명령을 사용하여 리포지토리에서 패키지를 설치합니다.

```
sudo yum install -y kmod-lustre-client lustre-client
```

추가 정보 (ARM AWS 기반 그래비톤 기반 EC2 인스턴스의 경우 CentOS 7.8 또는 7.9)

앞의 명령은 Amazon FSx 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 패키지를 설치합니다. 리포지토리에는 소스 코드가 포함된 패키지 및 테스트가 포함된 패키지와 같은 추가 Lustre 패키지가 포함되며, 선택적으로 설치할 수 있습니다. 리포지토리에서 사용 가능한 모든 패키지를 나열하려면 다음 명령을 사용합니다.

```
yum --disablerepo="*" --enablerepo="aws-fsx" list available
```

업스트림 소스 코드의 타르볼과 적용한 패치 세트를 포함하는 소스 rpm을 다운로드하려면 다음 명령을 사용합니다.

```
 sudo yumdownloader --source kmod-lustre-client
```

yum 업데이트를 실행하면 가능한 경우 최신 버전의 모듈이 설치되고 기존 버전이 대체됩니다. 업데이트 시 현재 설치된 버전이 제거되지 않도록 하려면 다음과 같은 라인을 /etc/yum.conf 파일에 추가합니다.

```
installonlypkgs=kernel, kernel-big-mem, kernel-enterprise, kernel-smp,
              kernel-debug, kernel-unsupported, kernel-source, kernel-devel, kernel-
PAE,
              kernel-PAE-debug, kmod-lustre-client
```

이 목록에는 yum.conf 매뉴얼 페이지에 지정된 기본 설치 전용 패키지와 kmod-lustre-client 패키지가 포함됩니다.

Ubuntu

Ubuntu 22.04에 Lustre 클라이언트 설치

Ubuntu 22.04 Amazon FSx 리포지토리에서 Lustre 패키지를 다운로드할 수 있습니다. 다운로드 전 또는 다운로드 중에 리포지토리의 콘텐츠가 변조되지 않았는지 확인하기 위해 GPG(GNU Privacy Guard) 서명이 리포지토리의 메타데이터에 적용됩니다. 시스템에 올바른 퍼블릭 GPG 키가 설치되어 있지 않으면 리포지토리를 설치할 수 없습니다.

1. 클라이언트에서 터미널을 엽니다.
2. 다음 단계에 따라 Amazon FSx Ubuntu 리포지토리를 추가합니다.

   a. 이전에 클라이언트 인스턴스에 Amazon FSx Ubuntu 리포지토리를 등록하지 않은 경우, 필요한 퍼블릭 키를 다운로드하여 설치합니다. 다음 명령을 사용합니다.

   ```
   wget -O - https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-
   ubuntu-public-key.asc | gpg --dearmor | sudo tee /usr/share/keyrings/fsx-
   ubuntu-public-key.gpg >/dev/null
   ```

   b. 다음 명령을 사용하여 로컬 패키지 관리자에 Amazon FSx 패키지 리포지토리를 추가합니다.

   ```
   sudo bash -c 'echo "deb [signed-by=/usr/share/keyrings/fsx-ubuntu-public-
   key.gpg] https://fsx-lustre-client-repo.s3.amazonaws.com/ubuntu jammy main" > /
   etc/apt/sources.list.d/fsxlustreclientrepo.list && apt-get update'
   ```

3. 클라이언트 인스턴스에서 현재 실행 중인 커널을 확인하고 필요한 경우 업데이트합니다. Ubuntu 22.04의 Lustre 클라이언트에는 x86 기반 EC2 인스턴스와 AWS Graviton 프로세서 구동 ARM 기반 EC2 인스턴스 모두에 커널 5.15.0-1015-aws 이상이 필요합니다.

a. 다음 명령을 실행하여 어떤 커널이 실행 중인지 확인합니다.

```
uname -r
```

b. 다음 명령을 실행하여 최신 Ubuntu 커널과 Lustre 버전으로 업데이트한 다음 재부팅합니다.

```
sudo apt install -y linux-aws lustre-client-modules-aws && sudo reboot
```

커널 버전이 x86 기반 EC2 인스턴스와 Graviton 기반 EC2 인스턴스용 `5.15.0-1015-aws`보다 크고 최신 커널 버전으로 업데이트하지 않으려는 경우, 다음 명령을 사용하여 현재 커널용 Lustre를 설치할 수 있습니다.

```
sudo apt install -y lustre-client-modules-$(uname -r)
```

FSx for Lustre 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 개의 Lustre 패키지가 설치됩니다. 소스 코드가 들어 있는 패키지와 리포지토리에 속하는 테스트가 포함된 패키지와 같은 추가 관련 패키지를 선택적으로 설치할 수 있습니다.

c. 다음 명령을 사용하여 리포지토리 내 사용 가능한 모든 패키지를 나열합니다.

```
sudo apt-cache search ^lustre
```

d. (선택 사항) Lustre 클라이언트 모듈을 항상 업그레이드하도록 시스템을 업그레이드하려면 다음 명령을 사용하여 `lustre-client-modules-aws` 패키지를 설치해야 합니다.

```
sudo apt install -y lustre-client-modules-aws
```

> **Note**
>
> `Module Not Found` 오류가 발생한 경우 누락된 모듈 오류 해결 방법 섹션을 참조하세요.

Ubuntu 20.04에 Lustre 클라이언트 설치

Lustre 2.12 클라이언트는 커널 5.15.0-1015-aws 이상이 설치된 Ubuntu 20.04에서 지원됩니다. Lustre 2.10 클라이언트는 우분투 20.04에서 지원되며 x86 기반 EC2 인스턴스에서는 커널 5.4.0-1011-aws 이상이 지원되며 그라비톤 프로세서 기반 ARM 기반 EC2 인스턴스에서는 커널 5.4.0-1015-aws 이상이 지원됩니다. AWS

Ubuntu 20.04 Amazon FSx 리포지토리에서 Lustre 패키지를 다운로드할 수 있습니다. 다운로드 전 또는 다운로드 중에 리포지토리의 콘텐츠가 변조되지 않았는지 확인하기 위해 GPG(GNU Privacy Guard) 서명이 리포지토리의 메타데이터에 적용됩니다. 시스템에 올바른 퍼블릭 GPG 키가 설치되어 있지 않으면 리포지토리를 설치할 수 없습니다.

1. 클라이언트에서 터미널을 엽니다.
2. 다음 단계에 따라 Amazon FSx Ubuntu 리포지토리를 추가합니다.

   a. 이전에 클라이언트 인스턴스에 Amazon FSx Ubuntu 리포지토리를 등록하지 않은 경우, 필요한 퍼블릭 키를 다운로드하여 설치합니다. 다음 명령을 사용합니다.

   ```
   wget -O - https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-ubuntu-public-key.asc | gpg --dearmor | sudo tee /usr/share/keyrings/fsx-ubuntu-public-key.gpg >/dev/null
   ```

   b. 다음 명령을 사용하여 로컬 패키지 관리자에 Amazon FSx 패키지 리포지토리를 추가합니다.

   ```
   sudo bash -c 'echo "deb [signed-by=/usr/share/keyrings/fsx-ubuntu-public-key.gpg] https://fsx-lustre-client-repo.s3.amazonaws.com/ubuntu focal main" > /etc/apt/sources.list.d/fsxlustreclientrepo.list && apt-get update'
   ```

3. 클라이언트 인스턴스에서 현재 실행 중인 커널을 확인하고 필요한 경우 업데이트합니다.

   a. 다음 명령을 실행하여 어떤 커널이 실행 중인지 확인합니다.

   ```
   uname -r
   ```

   b. 다음 명령을 실행하여 최신 Ubuntu 커널과 Lustre 버전으로 업데이트한 다음 재부팅합니다.

   ```
   sudo apt install -y linux-aws lustre-client-modules-aws && sudo reboot
   ```

   커널 버전이 x86 기반 EC2 인스턴스용 `5.4.0-1011-aws`보다 높거나 Graviton 기반 EC2 인스턴스용 `5.4.0-1015-aws`보다 큰 경우, 최신 커널 버전으로 업데이트하지 않으려는 경우, 다음 명령을 사용하여 현재 커널용 Lustre를 설치할 수 있습니다.

   ```
   sudo apt install -y lustre-client-modules-$(uname -r)
   ```

FSx for Lustre 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 개의 Lustre 패키지가 설치됩니다. 소스 코드가 들어 있는 패키지와 리포지토리에 속하는 테스트가 포함된 패키지와 같은 추가 관련 패키지를 선택적으로 설치할 수 있습니다.

c. 다음 명령을 사용하여 리포지토리 내 사용 가능한 모든 패키지를 나열합니다.

```
sudo apt-cache search ^lustre
```

d. (선택 사항) Lustre 클라이언트 모듈을 항상 업그레이드하도록 시스템을 업그레이드하려면 다음 명령을 사용하여 `lustre-client-modules-aws` 패키지를 설치해야 합니다.

```
sudo apt install -y lustre-client-modules-aws
```

> **Note**
>
> `Module Not Found` 오류가 발생한 경우 누락된 모듈 오류 해결 방법 섹션을 참조하세요.

Ubuntu 18.04에 Lustre 클라이언트 설치

> **Note**
>
> 지원되는 최신 Ubuntu 18 커널 버전은 `5.4.0.1103.aws`입니다.

Ubuntu 18.04 Amazon FSx 리포지토리에서 Lustre 패키지를 다운로드할 수 있습니다. 다운로드 전 또는 다운로드 중에 리포지토리의 콘텐츠가 변조되지 않았는지 확인하기 위해 GPG(GNU Privacy Guard) 서명이 리포지토리의 메타데이터에 적용됩니다. 시스템에 올바른 퍼블릭 GPG 키가 설치되어 있지 않으면 리포지토리를 설치할 수 없습니다.

1. 클라이언트에서 터미널을 엽니다.
2. 다음 단계에 따라 Amazon FSx Ubuntu 리포지토리를 추가합니다.

   a. 이전에 클라이언트 인스턴스에 Amazon FSx Ubuntu 리포지토리를 등록하지 않은 경우, 필요한 퍼블릭 키를 다운로드하여 설치합니다. 다음 명령을 사용합니다.

```
wget -O - https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-
ubuntu-public-key.asc | gpg --dearmor | sudo tee /usr/share/keyrings/fsx-
ubuntu-public-key.gpg >/dev/null
```

b. 다음 명령을 사용하여 로컬 패키지 관리자에 Amazon FSx 패키지 리포지토리를 추가합니다.

```
sudo bash -c 'echo "deb [signed-by=/usr/share/keyrings/fsx-ubuntu-public-
key.gpg] https://fsx-lustre-client-repo.s3.amazonaws.com/ubuntu bionic main"
 > /etc/apt/sources.list.d/fsxlustreclientrepo.list && apt-get update'
```

3. 클라이언트 인스턴스에서 현재 실행 중인 커널을 확인하고 필요한 경우 업데이트합니다. 우분투 18.04의 Lustre 클라이언트에는 x86 기반 EC2 인스턴스의 경우 커널 이상이 필요하고 Graviton 프로세서가 탑재된 ARM 기반 EC2 인스턴스의 경우 커널 이상이 필요합니다. `4.15.0-1054-aws` `5.3.0-1023-aws` AWS

a. 다음 명령을 실행하여 어떤 커널이 실행 중인지 확인합니다.

```
uname -r
```

b. 다음 명령을 실행하여 최신 Ubuntu 커널과 Lustre 버전으로 업데이트한 다음 재부팅합니다.

```
sudo apt install -y linux-aws lustre-client-modules-aws && sudo reboot
```

커널 버전이 x86 기반 EC2 인스턴스용 `4.15.0-1054-aws`보다 높거나 Graviton 기반 EC2 인스턴스용 `5.3.0-1023-aws`보다 큰 경우, 최신 커널 버전으로 업데이트하지 않으려는 경우, 다음 명령을 사용하여 현재 커널용 Lustre를 설치할 수 있습니다.

```
sudo apt install -y lustre-client-modules-$(uname -r)
```

FSx for Lustre 파일 시스템을 마운트하고 상호 작용하는 데 필요한 두 개의 Lustre 패키지가 설치됩니다. 소스 코드가 들어 있는 패키지와 리포지토리에 속하는 테스트가 포함된 패키지와 같은 추가 관련 패키지를 선택적으로 설치할 수 있습니다.

c. 다음 명령을 사용하여 리포지토리 내 사용 가능한 모든 패키지를 나열합니다.

```
sudo apt-cache search ^lustre
```

d. (선택 사항) Lustre 클라이언트 모듈을 항상 업그레이드하도록 시스템을 업그레이드하려면 다음 명령을 사용하여 `lustre-client-modules-aws` 패키지를 설치해야 합니다.

```
sudo apt install -y lustre-client-modules-aws
```

> **Note**
>
> `Module Not Found` 오류가 발생한 경우 누락된 모듈 오류 해결 방법 섹션을 참조하세요.

누락된 모듈 오류 해결 방법

임의의 Ubuntu 버전에서 설치하는 동안 `Module Not Found` 오류가 발생하면 다음을 따르세요.

커널을 지원되는 최신 버전으로 다운그레이드합니다. 패키지의 사용 가능한 버전을 모두 나열하고 해당 커널을 설치합니다. lustre-client-modules 이렇게 하려면 다음 명령을 사용합니다.

```
sudo apt-cache search lustre-client-modules
```

예를 들어 리포지토리에 포함된 최신 버전이 `lustre-client-modules-5.4.0-1011-aws`인 경우 다음을 따르세요.

1. 다음 명령을 사용하여 이 패키지용으로 만들어진 커널을 설치합니다.

   ```
   sudo apt-get install -y linux-image-5.4.0-1011-aws
   ```

   ```
   sudo sed -i 's/GRUB_DEFAULT=.\+/GRUB\_DEFAULT="Advanced options for Ubuntu>Ubuntu,
    with Linux 5.4.0-1011-aws"/' /etc/default/grub
   ```

   ```
   sudo update-grub
   ```

2. 다음 명령을 사용하여 인스턴스를 재부팅합니다.

   ```
   sudo reboot
   ```

3. 다음 명령을 사용하여 Lustre 클라이언트를 설치합니다.

   ```
   sudo apt-get install -y lustre-client-modules-$(uname -r)
   ```

SUSE Linux

SUSE Linux 12 SP3, SP4 또는 SP5에 Lustre 클라이언트 설치

SUSE Linux 12 SP3에 Lustre 클라이언트 설치

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

```
sudo wget https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-sles-public-key.asc
```

3. 다음 명령을 사용하여 키를 가져옵니다.

```
sudo rpm --import fsx-sles-public-key.asc
```

4. 다음 명령을 사용하여 Lustre 클라이언트용 리포지토리를 추가합니다.

```
sudo wget https://fsx-lustre-client-repo.s3.amazonaws.com/suse/sles-12/SLES-12/fsx-lustre-client.repo
```

5. 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo sed -i 's#SLES-12#SP3#' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper refresh
sudo zypper in lustre-client
```

SUSE Linux 12 SP4에 Lustre 클라이언트 설치

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

```
sudo wget https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-sles-public-key.asc
```

3. 다음 명령을 사용하여 키를 가져옵니다.

```
sudo rpm --import fsx-sles-public-key.asc
```

4. 다음 명령을 사용하여 Lustre 클라이언트용 리포지토리를 추가합니다.

```
sudo wget https://fsx-lustre-client-repo.s3.amazonaws.com/suse/sles-12/SLES-12/fsx-lustre-client.repo
```

5. 다음 중 하나를 수행합니다.

   - SP4를 직접 설치한 경우 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo sed -i 's#SLES-12#SP4#' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper refresh
sudo zypper in lustre-client
```

   - SP3에서 SP4로 마이그레이션하고 이전에 SP3용 Amazon FSx 리포지토리를 추가한 경우 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo sed -i 's#SP3#SP4#' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper ref
sudo zypper up --force-resolution lustre-client-kmp-default
```

SUSE Linux 12 SP5에 Lustre 클라이언트 설치

1. 클라이언트에서 터미널을 엽니다.
2. 다음 명령을 사용하여 Amazon FSx rpm 퍼블릭 키를 설치합니다.

```
sudo wget https://fsx-lustre-client-repo-public-keys.s3.amazonaws.com/fsx-sles-public-key.asc
```

3. 다음 명령을 사용하여 키를 가져옵니다.

```
sudo rpm --import fsx-sles-public-key.asc
```

4. 다음 명령을 사용하여 Lustre 클라이언트용 리포지토리를 추가합니다.

```
sudo wget https://fsx-lustre-client-repo.s3.amazonaws.com/suse/sles-12/SLES-12/fsx-lustre-client.repo
```

5. 다음 중 하나를 수행합니다.

- SP5를 직접 설치한 경우 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

```
sudo zypper ar --gpgcheck-strict fsx-lustre-client.repo
sudo zypper refresh
sudo zypper in lustre-client
```

- SP4에서 SP5로 마이그레이션하고 이전에 SP4용 Amazon FSx 리포지토리를 추가한 경우 다음 명령을 사용하여 Lustre 클라이언트를 다운로드하고 설치합니다.

```
sudo sed -i 's#SP4#SLES-12' /etc/zypp/repos.d/aws-fsx.repo
sudo zypper ref
sudo zypper up --force-resolution lustre-client-kmp-default
```

> **Note**
>
> 클라이언트에 설치를 완료하려면 컴퓨팅 인스턴스를 재부팅해야 할 수 있습니다.

Amazon Elastic Compute Cloud 인스턴스에서 마운트

Amazon EC2 인스턴스에서 파일 시스템을 마운트할 수 있습니다.

Amazon EC2에서 파일 시스템 마운트

1. Amazon EC2 인스턴스에 연결합니다.
2. 다음 명령으로 FSx for Lustre 파일 시스템에 마운트 지점에 대한 디렉터리를 만드세요.

```
$ sudo mkdir -p /fsx
```

3. Amazon FSx for Lustre 파일 시스템을 생성한 디렉터리에 마운트합니다. 다음 명령을 사용하여 다음 항목을 바꿉니다.

   - *`file_system_dns_name`*을 실제 파일 시스템의 DNS 이름으로 바꿉니다.
   - *`mountname`*을 파일 시스템의 마운트 이름으로 바꿉니다. 이 마운트 이름은 `CreateFileSystem` API 작업 응답에 반환됩니다. describe-file-systems AWS CLI 명령 및 DescribeFile시스템 API 작업의 응답에서도 반환됩니다.

```
sudo mount -t lustre -o relatime,flock file_system_dns_name@tcp:/mountname /fsx
```

이 명령은 `-o relatime`과 `flock` 같은 두 가지 옵션을 사용하여 파일 시스템을 마운트합니다.

- `relatime` - `atime` 옵션은 파일에 액세스할 때마다 `atime`(inode 액세스 시간) 데이터를 유지하는 반면, `relatime` 옵션은 `atime` 데이터를 유지하지만 파일에 액세스할 때마다 매번 유지하지는 않습니다. `relatime` 옵션을 활성화하면 `atime` 데이터가 마지막으로 업데이트된 이후 파일이 수정된 경우(`mtime`) 또는 특정 시간 이상 전에 파일을 마지막으로 액세스한 경우(기본값 6시간)에만 `atime` 데이터가 디스크에 기록됩니다. `relatime` 또는 `atime` 옵션 중 하나를 사용하면 파일 릴리스 프로세스가 최적화됩니다.

  > **Note**
  >
  > 정확한 액세스 시간 정확도가 필요한 워크로드의 경우 `atime` 마운트 옵션을 사용하여 마운트할 수 있습니다. 하지만 이렇게 하면 정확한 액세스 시간 값을 유지하는 데 필요한 네트워크 트래픽이 증가하여 워크로드 성능에 영향을 미칠 수 있습니다.
  > 워크로드에 메타데이터 액세스 시간이 필요하지 않은 경우 `noatime` 마운트 옵션을 사용하여 액세스 시간 업데이트를 비활성화하면 성능이 향상될 수 있습니다. 파일 릴리스나 데이터 유효성 공개와 같은 `atime` 집중 프로세스는 릴리스에서 정확하지 않을 수 있다는 점에 유의하세요.

- `flock` - 파일 시스템의 파일 잠금을 활성화합니다. 파일 잠금을 활성화하지 않으려면 `flock`을 제외한 `mount` 명령을 사용합니다.

4. 다음 명령을 사용하여 파일 시스템을 마운트한 디렉터리인 /mnt/fsx의 내용을 나열하여 마운트 명령이 제대로 실행되었는지 확인합니다.

```
$ ls /fsx
import-path  lustre
$
```

다음 `df` 명령도 사용할 수 있습니다.

```
$ df
Filesystem                   1K-blocks    Used  Available Use% Mounted on
devtmpfs                       1001808       0    1001808   0% /dev
tmpfs                          1019760       0    1019760   0% /dev/shm
```

```
tmpfs                              1019760     392    1019368   1% /run
tmpfs                              1019760       0    1019760   0% /sys/fs/cgroup
/dev/xvda1                         8376300 1263180    7113120  16% /
123.456.789.0@tcp:/mountname 3547698816   13824 3547678848   1% /fsx
tmpfs                               203956       0     203956   0% /run/user/1000
```

결과는 /fsx에 마운트된 Amazon FSx 파일 시스템을 보여줍니다.

Amazon Elastic Container Service에 마운트

Amazon EC2 인스턴스의 Amazon Elastic Container Service(Amazon ECS) 도커 컨테이너에서 FSx for Lustre 파일 시스템에 액세스할 수 있습니다. 다음 두 가지 옵션 중 하나를 사용할 수 있습니다.

1. Amazon ECS 작업을 호스팅하는 Amazon EC2 인스턴스에서 FSx for Lustre 파일 시스템을 마운트함으로써 해당 마운트 포인트를 컨테이너로 내보냅니다.
2. 파일 시스템을 작업 컨테이너 내에 직접 마운트합니다.

자세한 내용은 Amazon Elastic Container Service 개발자 안내서 내 Amazon Elastic Container Service란 무엇인가요? 섹션을 참조하세요.

옵션 1(Amazon ECS 작업을 호스팅하는 Amazon EC2 인스턴스에서 마운트)을 사용하는 것이 좋습니다. 동일한 EC2 인스턴스에서 여러 컨테이너(5개 이상)를 시작하거나 작업의 수명이 짧을 경우(5분 미만) 더 나은 리소스 사용량을 제공합니다.

EC2 인스턴스를 구성할 수 없거나 애플리케이션에 컨테이너의 유연성이 필요한 경우 옵션 2(도커 컨테이너에서 마운트)를 사용합니다.

> **Note**
>
> Fargate 시작 유형에서 FSx for Lustre를 AWS 마운트하는 것은 지원되지 않습니다.

다음 섹션에서는 Amazon ECS 컨테이너에서 FSx for Lustre 파일 시스템을 마운트하기 위한 각 옵션의 절차를 설명합니다.

주제

- Amazon ECS 작업을 호스팅하는 Amazon EC2 인스턴스에서 마운트
- 도커 컨테이너에서 마운트

Amazon ECS 작업을 호스팅하는 Amazon EC2 인스턴스에서 마운트

이 절차는 FSx for Lustre 파일 시스템을 로컬 마운트하도록 EC2 인스턴스 기반 Amazon ECS를 구성하는 방법을 보여줍니다. 이 절차는 `volumes`와 `mountPoints` 컨테이너 속성을 사용하여 리소스를 공유하고 로컬 실행 작업에서 이 파일 시스템에 액세스할 수 있도록 합니다. 자세한 내용은 Amazon Elastic Container Service 개발자 안내서의 Amazon ECS 컨테이너 인스턴스 시작 섹션을 참조하세요.

이 절차는 Amazon ECS 최적화 Amazon Linux 2 AMI용입니다. 다른 Linux 배포판을 사용하는 경우 Lustre 클라이언트 설치 중 섹션을 참조하세요.

Amazon ECS에서 EC2 인스턴스에 파일 시스템 마운트

1. 수동으로 또는 오토 스케일링 그룹을 사용하여 Amazon ECS 인스턴스를 시작할 때는 다음 코드 예제의 라인을 사용자 데이터 필드 끝에 추가합니다. 예제의 다음 항목들을 바꿉니다.

   - *`file_system_dns_name`*을 실제 파일 시스템의 DNS 이름으로 바꿉니다.
   - *`mountname`*을 파일 시스템의 마운트 이름으로 바꿉니다.
   - *`mountpoint`*를 생성해야 하는 파일 시스템의 마운트 지점으로 바꿉니다.

   ```
   #!/bin/bash

   ...<existing user data>...

   fsx_dnsname=file_system_dns_name
   fsx_mountname=mountname
   fsx_mountpoint=mountpoint
   amazon-linux-extras install -y lustre
   mkdir -p "$fsx_mountpoint"
   mount -t lustre ${fsx_dnsname}@tcp:/${fsx_mountname} ${fsx_mountpoint} -o
    relatime,flock
   ```

2. Amazon ECS 작업을 생성할 때 JSON 정의에 다음 `volumes` 및 `mountPoints` 컨테이너 속성을 추가합니다. *`mountpoint`*를 파일 시스템의 마운트 포인트(예:`/mnt/fsx`)로 대체합니다.

   ```
   {
       "volumes": [
              {
                    "host": {
                         "sourcePath": "mountpoint"
                    },
   ```

```
                "name": "Lustre"
            }
    ],
    "mountPoints": [
            {
                "containerPath": "mountpoint",
                "sourceVolume": "Lustre"
            }
    ],
}
```

도커 컨테이너에서 마운트

다음 절차는 lustre-client 패키지를 설치하고 FSx for Lustre 파일 시스템을 마운트하도록 Amazon ECS 작업 컨테이너를 구성하는 방법을 보여줍니다. 이 절차에서는 Amazon Linux(amazonlinux) 도커 이미지를 사용하지만 다른 배포판에서도 비슷한 접근 방식을 사용할 수 있습니다.

도커 컨테이너에서 파일 시스템 마운트

1. 도커 컨테이너에서 lustre-client 패키지를 설치하고 해당 속성을 사용하여 command 속성에 FSx for Lustre 파일 시스템을 마운트합니다. 예제의 다음 항목들을 바꿉니다.
   - *file_system_dns_name*을 실제 파일 시스템의 DNS 이름으로 바꿉니다.
   - *mountname*을 파일 시스템의 마운트 이름으로 바꿉니다.
   - *mountpoint*를 파일 시스템의 마운트 지점으로 바꿉니다.

```
"command": [
  "/bin/sh -c \"amazon-linux-extras install -y lustre; mount -t
 lustre file_system_dns_name@tcp:/mountname mountpoint -o relatime,flock;\""
],
```

2. linuxParameters 속성을 사용하여 FSx for Lustre 파일 시스템을 마운트하도록 승인하는 SYS_ADMIN 기능을 컨테이너에 추가합니다.

```
"linuxParameters": {
  "capabilities": {
      "add": [
        "SYS_ADMIN"
```

```
        ]
    }
}
```

온프레미스 또는 피어링된 Amazon VPC에서 Amazon FSx 파일 시스템 마운트

다음 두 가지 방법으로 Amazon FSx 파일 시스템에 액세스할 수 있습니다. 하나는 파일 시스템의 VPC에 피어링된 Amazon VPC에 위치한 Amazon EC2 인스턴스에서 가져오는 방법입니다. 다른 하나는 또는 VPN을 AWS Direct Connect 사용하여 파일 시스템의 VPC에 연결된 온프레미스 클라이언트에서 가져온 것입니다.

VPC 피어링 연결 또는 VPC 전송 게이트웨이를 사용하여 클라이언트의 VPC와 Amazon FSx 파일 시스템의 VPC를 연결합니다. VPC 피어링 연결 또는 전송 게이트웨이를 사용하여 VPC를 연결하면 VPC가 다른 계정에 속해 있더라도 하나의 VPC에 있는 Amazon EC2 인스턴스가 다른 VPC의 Amazon FSx 파일 시스템에 액세스할 수 있습니다.

다음 절차를 사용하기 전에 VPC 피어링 연결 또는 VPC 전송 게이트웨이를 설정해야 합니다.

전송 게이트웨이는 VPC와 온프레미스 네트워크를 상호 연결하는 데 사용할 수 있는 네트워크 전송 허브입니다. VPC 전송 게이트웨이 사용에 대한 자세한 내용은 Amazon VPC Transit Gateway 가이드의 전송 게이트웨이 시작하기 섹션을 참조하세요.

VPC 피어링 연결은 두 VPC 간의 네트워킹 연결입니다. 이러한 유형의 연결을 사용하면 프라이빗 Internet Protocol version 4(IPv4) 또는 Internet Protocol version 6(IPv6) 주소를 사용하여 이들 간의 트래픽을 라우팅할 수 있습니다. VPC 피어링을 사용하여 동일한 AWS 지역 내 또는 지역 간에 VPC를 연결할 수 있습니다. AWS 자세한 내용은 Amazon VPC 피어링 가이드의 VPC 피어링이란? 섹션을 참조하세요.

기본 네트워크 인터페이스의 IP 주소를 사용하여 VPC 외부에서 파일 시스템을 마운트할 수 있습니다. 기본 네트워크 인터페이스는 명령을 실행할 때 반환되는 첫 번째 네트워크 인터페이스입니다. `aws fsx describe-file-systems` AWS CLI Amazon Web Services Management Console에서도 IP 주소를 얻을 수 있습니다.

다음 표는 파일 시스템의 VPC 외부에 있는 클라이언트를 사용하여 Amazon FSx 파일 시스템에 액세스하기 위한 IP 주소 요구 사항을 보여줍니다.

클라이언트 위치	2020년 12월 17일 이전에 생성된 파일 시스템에 대한 액세스	2020년 12월 17일 이후에 생성된 파일 시스템에 대한 액세스
VPC 피어링을 사용하는 피어링된 VPC 또는 AWS Transit Gateway	RFC 1918 프라이빗 IP 주소 범위에 속하는 IP 주소를 가진 클라이언트	✓
또는 를 사용하는 AWS Direct Connect 피어링된 네트워크 AWS VPN	• 10.0.0.0/8 • 172.16.0.0/12 • 192.168.0.0/16	✓

프라이빗 IP 주소 범위를 사용하여 2020년 12월 17일 이전에 생성된 Amazon FSx 파일 시스템에 액세스해야 하는 경우, 파일 시스템의 백업을 복원하여 새 파일 시스템을 생성할 수 있습니다. 자세한 정보는 백업 작업을 참조하세요.

파일 시스템의 기본 네트워크 인터페이스의 IP 주소 검색

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 파일 시스템을 선택합니다.
3. 대시보드에서 파일 시스템을 선택합니다.
4. 파일 시스템 세부 정보 페이지에서 네트워크 및 보안을 선택합니다.
5. 네트워크 인터페이스의 경우 기본 탄력적 네트워크 인터페이스의 ID를 선택합니다. 이렇게 하면 Amazon EC2 콘솔로 이동합니다.
6. 세부 정보 탭에서 기본 프라이빗 IPv4 IP를 찾습니다. 해당 IP가 기본 네트워크 인터페이스의 IP 주소입니다.

> **Note**
>
> Amazon FSx 파일 시스템을 연결된 VPC 외부에서 마운트할 때는 도메인 이름 시스템(DNS) 이름 확인을 사용할 수 없습니다.

Amazon FSx 파일 시스템 자동 마운트

인스턴스에 처음 연결한 후 Amazon EC2 인스턴스에서 `/etc/fstab` 파일을 업데이트하여 재부팅할 때마다 Amazon FSx 파일 시스템을 마운트하도록 할 수 있습니다.

/etc/fstab 사용하여 FSx for Lustre 자동으로 마운트

Amazon EC2 인스턴스가 재부팅될 때마다 Amazon FSx 파일 시스템 디렉터리를 자동으로 마운트하도록 `fstab` 파일을 사용할 수 있습니다. `fstab` 파일에는 파일 시스템에 대한 정보가 들어 있습니다. 인스턴스 시작 중에 실행되는 `mount -a` 명령은 `fstab` 파일에 나열된 파일 시스템을 마운트합니다.

> **Note**
>
> EC2 인스턴스의 `/etc/fstab` 파일을 업데이트하려면 Amazon FSx 파일 시스템을 미리 만들어 두어야 합니다. 자세한 내용은 시작하기 연습의 FSx for Lustre 파일 시스템 만들기 섹션을 참조하세요.

EC2 인스턴스의 /etc/fstab 파일 업데이트

1. EC2 인스턴스를 연결하고 편집기에서 `/etc/fstab` 파일을 엽니다.
2. `/etc/fstab` 파일에 다음 줄을 추가합니다.

   Amazon FSx for Lustre 파일 시스템을 생성한 디렉터리에 마운트합니다. 다음 명령을 사용하여 다음을 대체합니다.

   - Amazon FSx 파일 시스템을 마운트하려는 디렉터리로 */fsx*을 대체합니다.
   - *file_system_dns_name*을 실제 파일 시스템의 DNS 이름으로 바꿉니다.
   - *mountname*을 파일 시스템의 마운트 이름으로 바꿉니다. 이 마운트 이름은 `CreateFileSystem` API 작업 응답에 반환됩니다. describe-file-systems AWS CLI 명령 및 API 작업의 응답에서도 반환됩니다. `DescribeFileSystems`

```
file_system_dns_name@tcp:/mountname /fsx lustre defaults,relatime,flock,_netdev,x-
systemd.automount,x-systemd.requires=network.service 0 0
```

> **Warning**
>
> 파일 시스템을 자동으로 마운트하는 경우 네트워크 파일 시스템 식별에 사용하는 _netdev 옵션을 사용합니다. _netdev이 빠진 경우 EC2 인스턴스가 응답을 중지합니다. 컴퓨팅 인스턴스가 네트워킹을 시작한 후 네트워크 파일 시스템의 초기화를 완료해야 하기 때문입니다. 자세한 정보는 자동 마운트 실패 및 인스턴스 무응답을 참조하세요.

3. 파일 변경 사항을 저장합니다.

이제 EC2 인스턴스가 다시 시작될 때마다 Amazon FSx 파일 시스템을 마운트하도록 구성되었습니다.

> **Note**
>
> 경우에 따라 마운트된 Amazon FSx 파일 시스템의 상태와 관계없이 Amazon EC2 인스턴스를 시작해야 할 수 있습니다. 이 경우 /etc/fstab 파일의 파일 시스템 항목에 nofail 옵션을 추가합니다.

/etc/fstab 파일에 추가한 코드 줄은 다음 작업을 수행합니다.

필드	설명
file_system_dns_name @tcp:/	파일 시스템을 식별하는 Amazon FSx 파일 시스템의 DNS 이름입니다. 이 이름은 콘솔에서 가져오거나 또는 AWS SDK에서 프로그래밍 방식으로 가져올 수 있습니다. AWS CLI
mountname	파일 시스템에 대한 마운트 이름입니다. 이 이름은 콘솔에서 가져오거나 describe-file-systems 명령을 사용하여 프로그래밍 방식으로 가져오거나 작업을 사용하는 AWS API 또는 AWS CLI SDK를 사용하여 프로그래밍 방식으로 가져올 수 있습니다. DescribeFileSystems
/fsx	EC2 인스턴스의 Amazon FSx 파일 시스템 마운트 지점
lustre	Amazon FSx 파일 시스템 유형
mount options	파일 시스템의 마운트 옵션은 쉼표로 구분된 다음 옵션 목록에 표시됩니다.

필드	설명
	• `defaults` - 이 값은 운영 체제에 기본 마운트 옵션을 사용하도록 지시합니다. `mount` 명령의 출력을 확인해서 파일 시스템을 마운트한 후 기본 마운트 옵션을 나열할 수 있습니다. • `relatime` - 이 옵션은 `atime`(inode 액세스 시간) 데이터를 유지하지만 파일을 액세스할 때마다 매번 유지하는 것은 아닙니다. 이 옵션을 활성화하면 `atime` 데이터가 마지막으로 업데이트된 이후 파일이 수정된 경우(`mtime`) 또는 특정 시간 이상 전에 파일을 마지막으로 액세스한 경우(기본값 1일)에만 `atime` 데이터가 디스크에 기록됩니다. inode 액세스 시간 업데이트를 끄려면 `noatime` 마운트 옵션을 대신 사용합니다. • `flock` - 파일 잠금을 활성화한 상태로 파일 시스템을 마운트합니다. 파일 잠금을 활성화하지 않으려면 `noflock` 마운트 옵션을 대신 사용하십시오. • `_netdev` – 운영 체제에 파일 시스템을 네트워크 액세스를 요구하는 장치에 위치시키라고 명령하는 값입니다. 클라이언트에서 네트워크가 활성화되기 전에 인스턴스가 파일 시스템을 마운트하는 것을 방지하는 옵션입니다.
`x-systemd.automount,x-systemd.requires=network.service`	이러한 옵션을 사용하면 네트워크 연결이 온라인 상태가 될 때까지 자동 마운터가 실행되지 않습니다. **Note** Amazon Linux 2023과 우분투 22.04의 경우 `x-systemd.requires=systemd-networkd-wait-online.service` 옵션 대신 옵션을 사용하십시오. `x-systemd.requires=network.service`
`0`	파일 시스템을 dump으로 백업해야 하는지 나타내는 값. Amazon FSx의 경우 이 값은 `0`이어야 합니다.
`0`	부팅 시 `fsck`가 파일을 체크하는 순서를 나타내는 값. Amazon FSx 파일 시스템의 경우 이 값을 `0`으로 하여 시작 시 `fsck`가 실행되지 않도록 해야 합니다.

특정 파일 세트 마운트

Lustre 파일 세트 기능을 사용하면 파일 시스템 네임스페이스의 일부만 마운트할 수 있으며, 이를 파일 세트라고 합니다. 파일 시스템의 파일 세트를 마운트하려면 클라이언트에서 파일 시스템 이름 뒤에 하위 디렉터리 경로를 지정합니다. 파일 세트 마운트(하위 디렉터리 마운트라고도 함)는 특정 클라이언트의 파일 시스템 네임스페이스 가시성을 제한합니다.

예 - Lustre 파일세트 마운트

1. 다음 디렉터리가 있는 FSx for Lustre 파일 시스템이 있다고 가정해 보겠습니다.

```
team1/dataset1/
team2/dataset2/
```

2. `team1/dataset1` 파일 세트만 마운트하면 파일 시스템의 이 부분만 클라이언트에 로컬로 표시됩니다. 다음 명령을 사용하여 다음 항목을 바꿉니다.

   - *`file_system_dns_name`*을 실제 파일 시스템의 DNS 이름으로 바꿉니다.
   - *`mountname`*을 파일 시스템의 마운트 이름으로 바꿉니다. 이 마운트 이름은 `CreateFileSystem` API 작업 응답에 반환됩니다. describe-file-systems AWS CLI 명령 및 DescribeFile시스템 API 작업의 응답에서도 반환됩니다.

```
mount -t lustre file_system_dns_name@tcp:/mountname/team1/dataset1 /fsx
```

Lustre 파일 세트 기능을 사용할 때 다음 사항에 유의하세요.

- 클라이언트가 다른 파일 세트를 사용하거나 파일 세트를 전혀 사용하지 않고 파일 시스템을 다시 마운트하지 못하도록 하는 제약 조건은 없습니다.
- 파일 세트를 사용할 경우 `lfs fid2path` 명령과 같은 `.lustre/` 디렉터리 액세스가 필요한 일부 Lustre 관리 명령이 작동하지 않을 수 있습니다.
- 동일한 파일 시스템의 여러 하위 디렉터리를 동일한 호스트에 마운트하려는 경우, 이 경우 단일 마운트 지점보다 많은 리소스가 소모되므로 파일 시스템 루트 디렉터리를 한 번만 마운트하는 것이 더 효율적일 수 있다는 점을 유의하세요.

Lustre 파일 세트 기능에 대한 자세한 내용은 Lustre 설명서 웹 사이트의 Lustre 운영 매뉴얼을 참조하세요.

파일 시스템 마운트 해제

파일 시스템을 삭제하기 전에 파일 시스템이 연결된 모든 Amazon EC2 인스턴스에서 파일 시스템을 탑재 해제하는 것이 좋습니다. 인스턴스 자체에서 umount 명령을 실행하여 Amazon EC2 인스턴스에서 파일 시스템의 탑재를 해제할 수 있습니다. , AWS Management Console또는 SDK를 통해 AWS CLI Amazon FSx 파일 시스템을 마운트 해제할 수 없습니다. AWS Linux를 실행하는 Amazon EC2 인스턴스에 연결된 Amazon FSx 파일 시스템을 마운트 해제하려면 다음과 같이 umount 명령을 사용합니다.

```
umount /mnt/fsx
```

다른 umount 옵션은 지정하지 않는 것이 좋습니다. 기본값과 다른 그 밖의 umount 옵션은 설정하지 않는 것이 좋습니다.

df 명령을 실행하여 Amazon FSx 파일 시스템이 마운트 해제되었는지 확인할 수 있습니다. 이 명령은 Linux 기반 Amazon EC2 인스턴스에 현재 마운트된 파일 시스템에 대한 디스크 사용량 통계를 표시합니다. 마운트 해제하려는 Amazon FSx 파일 시스템이 df 명령 출력에 나열되어 있지 않으면 그 파일 시스템은 마운트 해제된 것입니다.

Example - Amazon FSx 파일 시스템의 마운트 상태 식별 및 마운트 해제

```
$ df -T
Filesystem Type 1K-blocks Used Available Use% Mounted on
file-system-id.fsx.aws-region.amazonaws.com@tcp:/mountname /fsx 3547708416 61440
 3547622400 1% /fsx
      /dev/sda1 ext4 8123812 1138920 6884644 15% /
```

```
$ umount /fsx
```

```
$ df -T
```

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

Amazon EC2 스팟 인스턴스 작업

FSx for Lustre를 EC2 스팟 인스턴스와 함께 사용하면 Amazon EC2 비용을 크게 낮출 수 있습니다. 스팟 인스턴스는 온디맨드 가격보다 저렴한 비용으로 제공되는 미사용 EC2 인스턴스입니다. Amazon

EC2는 스팟 인스턴스에 대한 수요가 증가하거나 스팟 인스턴스의 공급이 감소하거나 스팟 가격이 최고가를 초과하는 경우 스팟 인스턴스를 중단할 수 있습니다.

Amazon EC2는 스팟 인스턴스를 중단할 때 스팟 인스턴스 중단 공지를 보내 중단 2분 전에 이를 인스턴스에 경고합니다. 자세한 내용은 Amazon EC2 사용 설명서의 스팟 인스턴스를 참조하십시오.

Amazon FSx 파일 시스템이 EC2 스팟 인스턴스 중단의 영향을 받지 않도록 EC2 스팟 인스턴스를 종료하거나 최대 절전 모드로 전환하기 전에 Amazon FSx 파일 시스템을 마운트 해제하는 것이 좋습니다. 자세한 내용은 파일 시스템 마운트 해제 섹션을 참조하세요.

Amazon EC2 스팟 인스턴스 중단 처리

FSx for Lustre는 서버와 클라이언트 인스턴스가 협력하여 성능이 뛰어나고 안정적인 파일 시스템을 제공하는 분산 파일 시스템입니다. 클라이언트와 서버 인스턴스 모두에서 분산되고 일관된 상태를 유지합니다. FSx for Lustre 서버는 클라이언트가 I/O를 활발히 수행하고 파일 시스템 데이터를 캐싱하는 동안 임시 액세스 권한을 클라이언트에 위임합니다. 서버가 임시 액세스 권한을 취소하도록 요청하면 클라이언트는 짧은 시간 내에 응답할 것으로 예상됩니다. 오작동하는 클라이언트로부터 파일 시스템을 보호하기 위해 서버는 몇 분 후에도 응답하지 않는 Lustre 클라이언트를 제거할 수 있습니다. 응답하지 않는 클라이언트가 서버 요청에 응답할 때까지 몇 분 동안 기다릴 필요가 없도록 하려면 EC2 스팟 인스턴스를 종료하기 전에 Lustre 클라이언트를 완전히 마운트 해제하는 것이 중요합니다.

EC2 스팟은 인스턴스를 종료하기 2분 전에 종료 알림을 보냅니다. EC2 스팟 인스턴스를 종료하기 전에 Lustre 클라이언트를 완전히 마운트 해제하는 프로세스를 자동화하는 것이 좋습니다.

Example - 종료된 EC2 스팟 인스턴스를 완전히 마운트 해제하기 위한 스크립트

이 예제 스크립트는 다음을 수행하여 종료된 EC2 스팟 인스턴스를 완전히 마운트 해제합니다.

- 스팟 종료 알림에 유의하세요.
- 종료 통지를 받는 경우
  - 파일 시스템에 액세스하는 애플리케이션을 중지합니다.
  - 인스턴스가 종료되기 전에 파일 시스템을 마운트 해제합니다.

특히 애플리케이션을 정상적으로 종료하는 경우 필요에 따라 스크립트를 조정할 수 있습니다. 스팟 인스턴스 중단 처리 모범 사례에 대한 자세한 내용은 EC2 스팟 인스턴스 중단 처리 모범 사례를 참조하세요.

```
#!/bin/bash
```

```
# TODO: Specify below the FSx mount point you are using
*FSXPATH=/fsx*

cd /

TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 21600")
if [ "$?" -ne 0 ]; then
    echo "Error running 'curl' command" >&2
    exit 1
fi

# Periodically check for termination
while sleep 5
do

    HTTP_CODE=$(curl -H "X-aws-ec2-metadata-token: $TOKEN" -s -w %{http_code} -o /dev/
null http://169.254.169.254/latest/meta-data/instance-action)

    if [[ "$HTTP_CODE" -eq 401 ]] ; then
        # Refreshing Authentication Token
        TOKEN=$(curl -s -X PUT "http://169.254.169.254/latest/api/token" -H "X-aws-ec2-
metadata-token-ttl-seconds: 30")
        continue
    elif [[ "$HTTP_CODE" -ne 200 ]] ; then
        # If the return code is not 200, the instance is not going to be interrupted
        continue
    fi

    echo "Instance is getting terminated. Clean and unmount '$FSXPATH' ..."
    curl -H "X-aws-ec2-metadata-token: $TOKEN" -s http://169.254.169.254/latest/meta-
data/instance-action
    echo

    # Gracefully stop applications accessing the filesystem
    #
    # TODO*: Replace with the proper command to stop your application if possible*

    # Kill every process still accessing Lustre filesystem
    echo "Kill every process still accessing Lustre filesystem..."
    fuser -kMm -TERM "${FSXPATH}"; sleep 2
    fuser -kMm -KILL "${FSXPATH}"; sleep 2

    # Unmount FSx For Lustre filesystem
```

```
    if ! umount -c "${FSXPATH}"; then
        echo "Error unmounting '$FSXPATH'. Processes accessing it:" >&2
        lsof "${FSXPATH}"

        echo "Retrying..."
        continue
    fi

    # Start a graceful shutdown of the host
    shutdown now

done
```

파일 시스템 관리

FSx for Lustre는 관리 작업의 성능을 단순화하는 일련의 기능을 제공합니다. 여기에는 point-in-time 백업, 파일 시스템 스토리지 할당량 관리, 스토리지 및 처리 용량 관리, 데이터 압축 관리, 시스템의 정기 소프트웨어 패치 수행을 위한 유지 관리 기간 설정 등이 포함됩니다.

Amazon FSx 관리 콘솔 AWS Command Line Interface , (), Amazon FSx API 또는 SDK를 사용하여 FSx for Lustre 파일 시스템을 관리할 수 있습니다.AWS CLI AWS

주제

- 백업 작업
- 스토리지 할당량
- 스토리지 용량 관리
- 메타데이터 성능 관리
- 처리량 용량 관리
- Lustre 데이터 압축
- Lustre 루트 스쿼시
- FSx for Lustre 파일 시스템 상태
- Amazon FSx 리소스 태그 지정
- Amazon FSx for Lustre 유지 관리 기간
- 파일 시스템 삭제

백업 작업

Amazon FSx for Lustre를 사용하면 Amazon S3의 내구성 있는 데이터 리포지토리에 연결되지 않은 영구 파일 시스템의 자동 일일 백업 및 사용자 시작 백업을 수행할 수 있습니다. Amazon FSx file-system-consistent 백업은 내구성이 뛰어나고 점진적입니다. Amazon FSx for Lustre는 99.999999999%(9 11개)의 높은 내구성을 보장하기 위해 Amazon Simple Storage Service(Amazon S3)에 백업을 저장합니다.

FSx for Lustre 파일 시스템 백업은 자동 일일 백업이든 사용자가 시작한 백업이든 여부와 관계없이 증분식입니다. 즉, 백업을 수행할 때 Amazon FSx는 파일 시스템의 데이터를 블록 수준에서 이전 백업과 비교합니다. 그러면 Amazon FSx가 모든 블록 수준 변경 사항의 사본을 새 백업에 저장합니다. 이전 백

업이 새 백업에 저장되지 않아 변경되지 않은 블록 수준 데이터. 백업 프로세스 기간은 마지막 백업을 수행한 이후 변경된 데이터의 양에 따라 달라지며 파일 시스템의 스토리지 용량과는 무관합니다. 다음 목록은 다양한 상황에서의 백업 시간을 보여줍니다.

- 데이터가 거의 없는 새로운 파일 시스템의 초기 백업을 완료하는 데 몇 분이 걸립니다.
- TB 용량의 데이터를 로드한 후 새로 만든 파일 시스템의 초기 백업은 완료하는 데 몇 시간이 걸립니다.
- 블록 수준 데이터에 대한 변경을 최소화하면서 TB의 데이터가 포함된 파일 시스템의 두 번째 백업(생성/수정 횟수가 비교적 적음)을 완료하는 데 몇 초가 걸립니다.
- 대량의 데이터를 추가하고 수정한 후 동일한 파일 시스템을 세 번째로 백업하는 경우 완료하는 데 몇 시간이 걸립니다.

백업을 삭제하면 해당 백업의 고유한 데이터만 제거됩니다. 각 FSx for Lustre 백업에는 백업에서 새 파일 시스템을 생성하는 데 필요한 모든 정보가 포함되어 있어 파일 시스템의 스냅샷을 효과적으로 point-in-time 복원합니다.

정기적으로 파일 시스템의 백업을 생성하는 것은 Amazon FSx for Lustre의 파일 시스템 복제를 보완하는 모범 사례입니다. Amazon FSx 백업은 백업 보존 및 규정 준수 요구 사항을 지원하는 데 도움이 됩니다. Amazon FSx for Lustre 백업 작업은 백업 생성, 백업 복사, 백업의 파일 시스템 복원, 백업 삭제와 관계없이 쉽습니다.

스크래치 파일 시스템은 임시 스토리지 및 단기 데이터 처리를 위해 설계되었으므로 백업이 지원되지 않습니다. Amazon S3 버킷에 연결된 파일 시스템에서는 백업이 지원되지 않습니다. S3 버킷이 기본 데이터 리포지토리 역할을 하고 Lustre 파일 시스템이 특정 시점에 반드시 전체 데이터 세트를 포함하지는 않기 때문입니다.

주제

- FSx for Lustre의 백업 지원
- 자동 일일 백업 작업
- 사용자 시작 백업 작업
- 아마존 AWS Backup FSx와 함께 사용
- 백업 복사
- 동일한 내에서 백업 복사 AWS 계정
- 백업 복원
- 백업 삭제

FSx for Lustre의 백업 지원

Amazon S3 데이터 리포지토리에 연결되지 않은 FSx for Lustre 영구 파일 시스템에서만 백업이 지원됩니다.

Amazon FSx는 스크래치 파일 시스템의 백업을 지원하지 않습니다. 임시 스토리지 및 단기 데이터 처리를 위해 설계되었기 때문입니다. Amazon FSx는 Amazon S3 버킷에 연결된 파일 시스템에서의 백업을 지원하지 않습니다. S3 버킷이 기본 데이터 리포지토리 역할을 하고 파일 시스템이 특정 시점에 반드시 전체 데이터 세트를 포함하지는 않기 때문입니다. 자세한 내용은 파일 시스템 배포 옵션 및 데이터 리포지토리 사용 섹션을 참조하세요.

자동 일일 백업 작업

Amazon FSx for Lustre는 파일 시스템을 매일 자동으로 백업할 수 있습니다. 이러한 자동 일일 백업은 파일 시스템을 생성할 때 설정한 일일 백업 기간 중에 발생합니다. 일일 백업 기간 중에 백업 프로세스가 시작될 때 스토리지 I/O가 일시적으로 중단될 수 있습니다(일반적으로 몇 초 이하). 일일 백업 기간을 선택할 때는 하루 중 편리한 시간을 선택하는 것이 좋습니다. 파일 시스템을 사용하는 애플리케이션의 정상 작동 시간을 벗어나는 것이 이상적입니다.

자동 일일 백업은 보존 기간이라고 하는 특정 기간 동안 보관됩니다. 백업 보존 기간은 0~90일로 설정할 수 있습니다. 보존 기간을 0일로 설정하면 자동 일일 백업이 꺼집니다. 자동 일일 백업의 기본 보존 기간은 0일입니다. 파일 시스템이 삭제되면 자동 일일 백업도 삭제됩니다.

> **Note**
>
> 보존 기간을 0일로 설정하면 파일 시스템이 자동으로 백업되지 않습니다. 어떤 수준이든 중요 기능이 관련된 파일 시스템에 대해서는 자동 일일 백업을 사용하는 것이 좋습니다.

AWS SDK AWS CLI 또는 둘 중 하나를 사용하여 파일 시스템의 백업 기간과 백업 보존 기간을 변경할 수 있습니다. `UpdateFileSystem` API 작업 또는 `update-file-system` CLI 명령을 사용합니다.

사용자 시작 백업 작업

Amazon FSx for Lustre로 언제든지 파일 시스템을 수동으로 백업할 수 있습니다. Amazon FSx for Lustre 콘솔, API 또는 (CLI) 를 사용하여 이 작업을 수행할 수 있습니다. AWS Command Line Interface 사용자가 시작한 Amazon FSx 파일 시스템 백업은 절대 만료되지 않으며, 원하는 시간만큼 유지할 수 있습니다. 사용자가 시작한 백업은 백업된 파일 시스템을 삭제한 후에도 보존됩니다. 사용자가 시작한

백업은 Amazon FSx for Lustre 콘솔, API 또는 CLI를 사용해야만 삭제할 수 있으며 Amazon FSx에서 자동으로 삭제하지는 않습니다. 자세한 내용은 백업 삭제 섹션을 참조하세요.

사용자 시작 백업 생성

다음 절차는 Amazon FSx 콘솔에서 기존 파일 시스템의 사용자 시작 백업을 생성하는 방법을 안내합니다.

파일 시스템의 사용자 시작 백업 생성

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx for Lustre 콘솔을 엽니다.
2. 콘솔 대시보드에서 백업하려는 파일 시스템의 이름을 선택합니다.
3. 작업에서 백업 생성을 선택합니다.
4. 열리는 백업 생성 대화 상자에서 백업 이름을 입력합니다. 백업 이름은 문자, 공백, 숫자 및 특수 문자 + - = _:/를 포함한 최대 256자의 유니코드 문자입니다.
5. 백업 생성을 선택합니다.

이제 파일 시스템 백업을 생성했습니다. 왼쪽 탐색 메뉴에서 백업을 선택하여 Amazon FSx for Lustre 콘솔의 모든 백업 테이블을 확인할 수 있습니다. 백업에 지정한 이름을 검색할 수 있으며, 테이블은 일치하는 결과만 표시하도록 필터링됩니다.

이 절차에서 설명한 대로 사용자가 시작한 백업을 생성하면 `USER_INITIATED` 유형이 지정되며 Amazon FSx에서 백업을 생성하는 동안 해당 백업은 생성 중 상태가 됩니다. 백업이 Amazon S3로 전송되면 완전히 사용할 수 있을 때까지 상태가 전송 중으로 변경됩니다.

아마존 AWS Backup FSx와 함께 사용

AWS Backup Amazon FSx 파일 시스템을 백업하여 데이터를 보호하는 간단하고 비용 효율적인 방법입니다. AWS Backup 는 백업의 생성, 복사, 복원 및 삭제를 단순화하는 동시에 향상된 보고 및 감사를 제공하도록 설계된 통합 백업 서비스입니다. AWS Backup 법률, 규정 및 전문 규정 준수를 위한 중앙 집중식 백업 전략을 보다 쉽게 개발할 수 있습니다. AWS Backup 또한 다음을 수행할 수 있는 중앙 위치를 제공하여 AWS 스토리지 볼륨, 데이터베이스 및 파일 시스템을 더 간단하게 보호할 수 있습니다.

- 백업하려는 AWS 리소스를 구성하고 감사하십시오.
- 백업 예약을 자동화합니다.
- 보존 정책을 설정합니다.
- AWS 지역 간 및 AWS 계정 간에 백업을 복사합니다.

- 최근 백업 및 복원 활동을 모두 모니터링합니다.

AWS Backup Amazon FSx의 내장된 백업 기능을 사용합니다. AWS Backup 콘솔에서 생성된 백업은 Amazon FSx 콘솔을 통해 생성된 백업과 동일한 수준의 파일 시스템 일관성과 성능, 동일한 복원 옵션을 제공합니다. 를 AWS Backup 사용하여 이러한 백업을 관리하면 무제한 보존 옵션과 한 시간마다 예약 백업을 생성할 수 있는 기능과 같은 추가 기능을 사용할 수 있습니다. 또한 소스 파일 시스템이 AWS Backup 삭제된 후에도 변경할 수 없는 백업을 보존합니다. 이렇게 하면 실수로 삭제되거나 악의적으로 삭제되는 것을 방지하는 데 도움이 됩니다.

에서 수행한 AWS Backup 백업은 사용자가 시작한 백업으로 간주되며 Amazon FSx의 사용자 시작 백업 할당량에 포함됩니다. Amazon FSx 콘솔, CLI 및 AWS Backup API에서 만든 백업을 보고 복원할 수 있습니다. 에서 생성한 AWS Backup 백업에는 백업 유형이 있습니다. `AWS_BACKUP` 하지만 Amazon FSx 콘솔, CLI 또는 AWS Backup API에서 만든 백업은 삭제할 수 없습니다. Amazon FSx 파일 시스템을 AWS Backup 백업하는 데 사용하는 방법에 대한 자세한 내용은 개발자 안내서의 Amazon FSx 파일 시스템 사용을 참조하십시오.AWS Backup

백업 복사

Amazon FSx를 사용하여 AWS 동일한 계정 내의 백업을 AWS 다른 지역 (지역 간 사본) 또는 동일한 지역 내 (지역 내 사본) 에 AWS 수동으로 복사할 수 있습니다. 동일한 파티션 내에서만 지역 간 사본을 만들 수 있습니다. AWS Amazon FSx 콘솔 또는 API를 사용하여 사용자가 시작한 백업 사본을 생성할 수 있습니다. AWS CLI사용자 시작 백업 사본에는 다음과 같이 `USER_INITIATED` 유형이 있습니다.

또한 를 사용하여 AWS 지역 간 및 계정 간에 백업을 AWS Backup 복사할 수 있습니다. AWS AWS Backup 정책 기반 백업 계획을 위한 중앙 인터페이스를 제공하는 완전 관리형 백업 관리 서비스입니다. 교차 계정 관리를 사용하면, 백업 정책을 사용하여 조직 내의 계정 전체에 걸쳐 백업 계획을 자동으로 적용할 수 있습니다.

크로스 리전 백업 복사본은 크로스 리전 재해 복구에 특히 유용합니다. 백업을 만들어 다른 AWS 지역으로 복사하면 주 지역에 재해가 발생할 경우 다른 AWS AWS 지역의 백업 및 복구 가용성을 신속하게 복원할 수 있습니다. 백업 복사본을 사용하여 파일 데이터세트를 다른 AWS 지역이나 같은 AWS 지역 내에 복제할 수도 있습니다. Amazon FSx 콘솔 또는 Amazon FSx for Lustre API를 사용하여 동일한 AWS 계정 (지역 간 또는 지역 내) 내에서 백업 사본을 만들 수 있습니다. AWS CLI또한 AWS Backup으로 온디맨드 또는 정책 기반으로 백업 복사를 수행하는 데에도 사용할 수 있습니다.

계정 간 백업 복사는 격리된 계정에 백업을 복사할 때 규정 준수 요구 사항을 충족하는 데 유용합니다. 또한 우발적 또는 악의적인 백업 삭제, 자격 증명 손실 또는 키 손상을 방지하는 데 도움이 되는 추가 데이터 보호 계층을 제공합니다. AWS KMS 교차 계정 백업은 팬인(여러 기본 계정의 백업을 하나의 격리

된 백업 사본 계정으로 복사) 및 팬아웃(하나의 기본 계정에서 여러 격리된 백업 사본 계정으로 백업 복사)을 지원합니다.

지원 부서와 AWS Backup 함께 AWS Organizations 사용하면 계정 간 백업 복사본을 만들 수 있습니다. 계정 간 복사본의 계정 한도는 정책에 따라 AWS Organizations 정의됩니다. 계정 간 백업 복사본을 만드는 AWS Backup 데 사용하는 방법에 대한 자세한 내용은 AWS Backup 개발자 안내서의 백업 복사본 만들기를 참조하세요. AWS 계정

백업 사본 제한 사항

다음은 백업을 복사할 때 적용되는 몇몇 제한 사항입니다.

- 지역 간 백업 복사본은 두 상업용 지역 간 AWS 리전, 중국 (베이징) 과 중국 (닝샤) 지역 간, (미국 동부) 및 AWS GovCloud AWS GovCloud (미국 서부) 지역 간에만 지원되며 해당 지역 간에는 지원되지 않습니다.
- 크로스 리전 백업 복사본은 옵트인 리전에서 지원되지 않습니다.
- 모든 지역 내에서 지역 내 백업 복사본을 만들 수 있습니다. AWS
- 원본 백업이 `AVAILABLE` 상태여야만 복사할 수 있습니다.
- 복사 중인 소스 백업은 삭제할 수 없습니다. 대상 백업을 사용할 수 있게 되는 시점과 소스 백업을 삭제할 수 있는 시점 사이에는 약간의 지연이 있을 수 있습니다. 소스 백업을 다시 삭제하려고 할 때는 지연을 염두에 두어야 합니다.
- 계정당 단일 대상 AWS 지역으로 최대 5개의 백업 사본 요청을 진행할 수 있습니다.

크로스 리전 백업 복사본 권한

IAM 정책 설명을 사용하여 백업 복사 작업을 수행할 권한을 부여합니다. 소스 AWS 리전과 통신하여 리전 간 백업 사본을 요청하려면 요청자 (IAM 역할 또는 IAM 사용자) 가 소스 백업 및 소스 리전에 액세스할 수 있어야 합니다. AWS

정책을 사용하여 백업 복사 작업에 대한 `CopyBackup` 작업 권한을 부여합니다. 다음 예제와 같이 정책의 `Action` 필드에서 작업을 지정하고, 정책의 `Resource` 필드에서 리소스 값을 지정합니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": "fsx:CopyBackup",
```

```
            "Resource": "arn:aws:fsx:*:111122223333:backup/*"
        }
    ]
}
```

IAM 정책에 대한 자세한 내용은 IAM 사용 설명서의 IAM의 정책 및 권한을 참조하세요.

전체 및 증분 복사

백업을 원본 백업과 다른 AWS 리전 곳에 복사하는 경우 첫 번째 사본은 전체 백업 사본입니다. 해당 지역에서 이전에 복사한 백업을 모두 삭제하지 않고 동일한 키를 사용한 경우, 첫 번째 백업 복사본 이후 동일한 AWS 계정 내의 동일한 대상 지역에 대한 모든 후속 백업 사본은 증분 백업입니다. AWS KMS 두 조건 중 하나라도 충족되지 않은 상태에서 복사 작업을 수행하면 증분이 아닌 전체 백업 사본이 생성됩니다.

동일한 내에서 백업 복사 AWS 계정

다음 절차에 설명된 대로, CLI 및 API를 AWS Management Console사용하여 FSx for Lustre 파일 시스템의 백업을 복사할 수 있습니다.

콘솔을 사용하여 동일한 계정(크로스 리전 또는 리전 내) 내에서 백업 복사

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 백업을 선택합니다.
3. 백업 테이블에서 복사할 백업을 선택한 다음 백업 복사를 선택합니다.
4. 설정 섹션에서 다음을 수행합니다.
   - 대상 지역 목록에서 백업을 복사할 대상 AWS 지역을 선택합니다. 대상은 다른 지역에 있을 수도 있고 (AWS 지역 간 복사) 같은 지역 내에 있을 수도 있고 (AWS 지역 내 복사) 내에 있을 수도 있습니다.
   - (선택 사항) 소스 백업에서 대상 백업으로 태그를 복사하려면 태그 복사를 선택합니다. 6단계에서 태그 복사를 선택하고 태그도 추가하면 모든 태그가 병합됩니다.
5. 암호화에서는 복사한 AWS KMS 백업을 암호화할 암호화 키를 선택합니다.
6. 태그 - 선택 사항의 경우 키와 값을 입력하여 태그를 복사된 백업에 추가합니다. 여기에 태그를 추가하고 4단계에서 태그 복사를 선택하면 모든 태그가 병합됩니다.
7. 백업 복사를 선택합니다.

   백업은 동일한 파일 내에서 선택한 AWS 계정 AWS 리전백업에 복사됩니다.

CLI를 사용하여 동일한 계정(크로스 리전 또는 리전 내) 내에서 백업 복사

- copy-backupCLI 명령 또는 CopyBackupAPI 작업을 사용하여 리전 또는 AWS 리전 내에서 동일한 AWS 계정 내에서 백업을 복사합니다. AWS

  다음 명령은 us-east-1 리전에서 ID가 backup-0abc123456789cba7인 백업을 복사합니다.

  ```
  aws fsx copy-backup \
    --source-backup-id backup-0abc123456789cba7 \
    --source-region us-east-1
  ```

  응답에는 복사된 백업의 설명이 표시됩니다.

  Amazon FSx 콘솔에서 또는 CLI 명령 또는 API 작업을 사용하여 describe-backups 프로그래밍 방식으로 백업을 볼 수 있습니다. DescribeBackups

백업 복원

사용 가능한 백업을 사용하여 새 파일 시스템을 생성하여 다른 파일 시스템의 point-in-time 스냅샷을 효과적으로 복원할 수 있습니다. 콘솔 또는 AWS SDK 중 하나를 사용하여 백업을 복원할 수 있습니다. AWS CLI백업을 새 파일 시스템으로 복원하는 데는 새 파일 시스템을 만드는 시간과 동일한 시간이 걸립니다. 백업에서 복원된 데이터는 파일 시스템에 지연 로드되고, 로딩되는 동안 지연 시간이 약간 더 길어집니다.

다음 절차는 콘솔을 사용하여 백업을 복원하고 새 파일 시스템을 만드는 방법을 안내합니다.

> **Note**
>
> 백업은 원본과 동일한 Lustre 버전 유형, 배포 유형, 스토리지 단위당 처리량, 스토리지 용량, 데이터 압축 유형 및 AWS 지역의 파일 시스템에만 복원할 수 있습니다. 복원된 파일 시스템이 사용할 수 있는 상태가 되면 파일 시스템의 스토리지 용량을 늘릴 수 있습니다. 자세한 내용은 스토리지 용량 관리 섹션을 참조하세요.

백업에서 파일 시스템 복원

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx for Lustre 콘솔을 엽니다.
2. 콘솔 대시보드의 왼쪽 탐색 메뉴에서 백업을 선택합니다.
3. 백업 테이블에서 복원할 백업을 선택한 다음 백업 복원을 선택합니다.

그러면 파일 시스템 생성 마법사가 열립니다. 이 마법사는 파일 시스템 구성(예: 배포 유형, 스토리지 단위당 처리량)을 제외하면 표준 파일 시스템 생성 마법사와 동일합니다. 그러나 연결된 VPC 및 백업 설정을 변경할 수 있습니다.

4. 새 파일 시스템을 생성할 때와 마찬가지로 마법사를 완료합니다.
5. 검토 및 생성을 선택합니다.
6. Amazon FSx for Lustre 파일 시스템의 선택한 설정을 검토한 다음 파일 시스템 생성을 선택합니다.

백업에서 복원하여 이제 새 파일 시스템이 생성됩니다. 파일 시스템이 `AVAILABLE` 상태로 변경되면 정상적으로 사용할 수 있습니다.

백업 삭제

백업 삭제는 영구적이고 복구할 수 없는 작업입니다. 삭제된 백업의 모든 데이터도 삭제됩니다. 나중에 해당 백업이 다시 필요하지 않을 것이라는 확신이 들지 않으면 백업을 삭제하지 마세요. Amazon FSx 콘솔, CLI 또는 AWS Backup API에서 만든 백업은 삭제할 수 없습니다.

백업 삭제

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx for Lustre 콘솔을 엽니다.
2. 콘솔 대시보드의 왼쪽 탐색 메뉴에서 백업을 선택합니다.
3. 백업 테이블에서 삭제하려는 백업을 선택한 다음 백업 삭제를 선택합니다.
4. 열린 백업 삭제 대화 상자에서 백업 ID가 삭제하려는 백업을 식별하는지 확인합니다.
5. 삭제할 백업의 확인란이 선택되어 있는지 확인합니다.
6. 백업 삭제를 선택합니다.

이제 백업과 포함된 모든 데이터가 영구적으로 삭제되어 복구할 수 없습니다.

스토리지 할당량

FSx for Lustre 파일 시스템에서 사용자, 그룹 및 프로젝트에 대한 스토리지 할당량을 생성할 수 있습니다. 스토리지 할당량을 사용하면 사용자, 그룹 또는 프로젝트가 사용할 수 있는 디스크 공간과 파일 수를 제한할 수 있습니다. 스토리지 할당량은 사용자 수준, 그룹 수준, 프로젝트 수준 사용량을 자동으로 추적하므로 스토리지 한도 설정 여부와 관계없이 사용량을 모니터링할 수 있습니다.

Amazon FSx는 할당량을 적용하고 할당량을 초과한 사용자가 스토리지 공간에 더 저장할 수 없도록 합니다. 사용자가 할당량을 초과할 경우 할당량 한도 미만까지 충분한 파일을 삭제해야 파일 시스템에 다시 저장할 수 있습니다.

주제

- 할당량 적용
- 할당량 유형
- 할당량 제한 및 유예 기간
- 할당량 설정 및 보기
- 할당량 및 Amazon S3 연결 버킷
- 할당량 및 백업 복원

할당량 적용

사용자, 그룹 및 프로젝트 할당량 적용은 모든 FSx for Lustre 파일 시스템에서 자동으로 활성화됩니다. 할당량 적용을 비활성화할 수 없습니다.

할당량 유형

AWS 계정 루트 사용자 자격 증명이 있는 시스템 관리자는 다음과 같은 유형의 할당량을 생성할 수 있습니다.

- 사용자 할당량은 개별 사용자에게 적용됩니다. 특정 사용자의 사용자 할당량은 다른 사용자의 할당량과 다를 수 있습니다.
- 그룹 할당량은 특정 그룹의 구성원인 모든 사용자에게 적용됩니다.
- 프로젝트 할당량은 프로젝트와 관련된 모든 파일 또는 디렉터리에 적용됩니다. 프로젝트에는 파일 시스템 내의 다른 디렉터리에 있는 다수의 디렉터리 또는 개별 파일이 포함될 수 있습니다.

> **Note**
>
> 프로젝트 할당량은 FSx for Lustre 파일 시스템의 Lustre 버전 2.15에서만 지원됩니다.

- 블록 할당량은 사용자, 그룹 또는 프로젝트가 사용할 수 있는 디스크 공간을 제한합니다. 스토리지 크기를 킬로바이트 단위로 구성합니다.
- inode 할당량은 사용자, 그룹 또는 프로젝트가 만들 수 있는 파일 또는 디렉터리 수를 제한합니다. 최대 inode 수를 정수로 구성합니다.

> **Note**
>
> 기본 할당량은 지원되지 않습니다.

특정 사용자 및 그룹에 할당량을 설정하고 사용자가 해당 그룹의 구성원인 경우 사용자의 데이터 사용량은 두 할당량 모두에 적용됩니다. 또한 두 할당량 모두에 의해 제한됩니다. 할당량 한도 중 하나에 도달하면 사용자는 파일 시스템에 더 이상 저장할 수 없습니다.

> **Note**
>
> 루트 사용자에 대해 설정된 할당량은 적용되지 않습니다. 마찬가지로 sudo 명령을 사용하여 루트 사용자로 데이터를 쓰면 할당량 적용을 우회할 수 있습니다.

할당량 제한 및 유예 기간

Amazon FSx는 사용자, 그룹 및 프로젝트 할당량에 엄격한 한도를 적용하거나 구성 가능한 유예 기간이 있는 완화된 한도를 적용합니다.

엄격한 한도는 절대 한도입니다. 사용자가 엄격한 한도를 초과하면 디스크 할당량 초과 메시지와 함께 블록 또는 inode 할당에 실패합니다. 엄격한 할당량 제한에 도달한 사용자는 할당량 한도 미만이 될 만큼 충분한 파일이나 디렉터리를 삭제해야 파일 시스템에 다시 저장할 수 있습니다. 유예 기간을 설정한 경우 사용자가 유예 기간 내에 엄격한 한도 이하까지는 유예 기간 내에 완화된 한도를 초과할 수 있습니다.

완화된 한도의 경우 유예 기간을 초 단위로 구성합니다. 완화된 한도는 엄격한 한도보다 작아야 합니다.

inode와 블록 할당량에 대해 서로 다른 유예 기간을 설정할 수 있습니다. 사용자 할당량, 그룹 할당량, 프로젝트 할당량에 대해 서로 다른 유예 기간을 설정할 수도 있습니다. 사용자, 그룹 및 프로젝트 할당량의 유예 기간이 서로 다른 경우 이러한 할당량의 유예 기간이 경과하면 완화된 한도가 엄격한 한도로 전환됩니다.

사용자가 완화된 한도를 초과하는 경우 Amazon FSx는 유예 기간이 경과하거나 엄격한 한도에 도달할 때까지 계속해서 할당량을 초과할 수 있도록 허용합니다. 유예 기간이 끝나면 완화된 한도가 엄격한 한도로 전환되고 스토리지 사용량이 정의된 블록 할당량 또는 inode 할당량 한도 아래로 돌아올 때까지 사용자는 추가 쓰기 작업을 할 수 없습니다. 유예 기간이 시작되어도 사용자는 알림이나 경고를 받지 않습니다.

할당량 설정 및 보기

Linux 터미널에서 Lustre 파일 시스템 `lfs` 명령을 사용하여 스토리지 할당량을 설정합니다. `lfs setquota` 명령은 할당량 제한을 설정하고 `lfs quota` 명령은 할당량 정보를 표시합니다.

Lustre 할당량 명령에 대한 자세한 내용은 Lustre 설명서 웹 사이트의 Lustre 운영 매뉴얼을 참조하세요.

사용자, 그룹, 프로젝트 할당량 설정

사용자, 그룹 또는 프로젝트 할당량을 설정하는 `setquota` 명령 구문은 다음과 같습니다.

```
lfs setquota {-u|--user|-g|--group|-p|--project} username|groupname|projectid
             [-b block_softlimit] [-B block_hardlimit]
             [-i inode_softlimit] [-I inode_hardlimit]
             /mount_point
```

위치:

- `-u` 또는 `--user` 할당량을 설정할 사용자를 지정합니다.
- `-g` 또는 `--group` 할당량을 설정할 그룹을 지정합니다.
- `-p` 또는 `--project` 할당량을 설정할 프로젝트를 지정합니다.
- `-b` 코드는 완화된 한도로 블록 할당량을 설정합니다. `-B`는 엄격한 한도로 블록 할당량을 설정합니다. *`block_softlimit`*와 *`block_hardlimit`*는 모두 킬로바이트로 표시되며 최소값은 1024KB입니다.
- `-i` 코드는 완화된 한도로 inode 할당량을 설정합니다. `-I`는 엄격한 한도로 inode 할당량을 설정합니다. *`inode_softlimit`*와 *`inode_hardlimit`*는 모두 inode 수로 표시되며 최소값은 1024개의 inode입니다.
- *`mount_point`*는 파일 시스템이 마운트된 디렉터리입니다.

사용자 할당량 예제: 다음 명령은 `/mnt/fsx`에 마운트된 파일 시스템의 `user1`에 대해 5,000KB 완화된 블록 제한, 8,000KB 엄격한 블록 제한, 2,000 완화된 inode 제한 및 3,000개의 엄격한 inode 제한 할당량을 설정합니다.

```
sudo lfs setquota -u user1 -b 5000 -B 8000 -i 2000 -I 3000 /mnt/fsx
```

그룹 할당량 예제: 다음 명령은 `/mnt/fsx`에 마운트된 파일 시스템에 이름이 지정된 `group1` 그룹에 대해 100,000KB의 엄격한 블록 제한을 설정합니다.

```
sudo lfs setquota -g group1 -B 100000 /mnt/fsx
```

프로젝트 할당량 예제: 먼저 `project` 명령을 사용하여 원하는 파일 및 디렉터리를 프로젝트에 연결했는지 확인합니다. 예를 들어 다음 명령은 `/mnt/fsxfs/dir1` 디렉터리의 모든 파일 및 하위 디렉터리를 프로젝트 ID가 `100`인 프로젝트와 연결합니다.

```
sudo lfs project -p 100 -r -s /mnt/fsxfs/dir1
```

그런 다음 `setquota` 명령어를 사용하여 프로젝트 할당량을 설정합니다. 다음 명령은 `/mnt/fsx`에 마운트된 파일 시스템의 `250` 프로젝트에 대해 307,200KB 완화된 블록 제한, 309,200KB 엄격한 블록 제한, 10,000개의 완화된 inode 제한 및 11,000개의 엄격한 inode 제한 할당량을 설정합니다.

```
sudo lfs setquota -p 250 -b 307200 -B 309200 -i 10000 -I 11000 /mnt/fsx
```

유예 기간 설정

기본 유예 기간은 1주일입니다. 다음 구문을 사용하여 사용자, 그룹 또는 프로젝트의 기본 유예 기간을 조정할 수 있습니다.

```
lfs setquota -t {-u|-g|-p}
             [-b block_grace]
             [-i inode_grace]
             /mount_point
```

위치:

- `-t` 유예 기간이 설정됨을 나타냅니다.
- `-u` 모든 사용자에게 유예 기간을 설정합니다.
- `-g` 모든 그룹에 유예 기간을 설정합니다.
- `-p` 모든 프로젝트에 유예 기간을 설정합니다.
- `-b` 블록 할당량의 유예 기간을 설정합니다. `-i` inode 할당량의 유예 기간을 설정합니다. *block_grace*와 *inode_grace*는 모두 정수(초) 또는 `XXwXXdXXhXXmXXs` 형식으로 표현됩니다.
- *mount_point*는 파일 시스템이 마운트된 디렉터리입니다.

다음 명령은 사용자 블록 할당량의 유예 기간을 1,000초, 사용자 inode 할당량의 경우 1주 4일로 설정합니다.

```
sudo lfs setquota -t -u -b 1000 -i 1w4d /mnt/fsx
```

할당량 보기

quota 명령은 사용자 할당량, 그룹 할당량, 프로젝트 할당량, 유예 기간에 대한 정보를 표시합니다.

할당량 명령 보기	할당량 정보 표시
`lfs quota /mount_point`	명령을 실행하는 사용자 및 사용자의 주 그룹에 대한 일반 할당량 정보(디스크 사용량 및 한도)
`lfs quota -u username /mount_point`	특정 사용자에 대한 일반 할당량 정보. AWS 계정 루트 사용자 자격 증명이 있는 사용자는 모든 사용자에 대해 이 명령을 실행할 수 있지만 루트가 아닌 사용자는 이 명령을 실행하여 다른 사용자에 대한 할당량 정보를 가져올 수 없습니다.
`lfs quota -u username -v /mount_point`	특정 사용자에 대한 일반 할당량 정보와 각 OST(오브젝트 스토리지 대상) 및 메타데이터 대상(MDT)에 대한 세부 할당량 통계. AWS 계정 루트 사용자 자격 증명이 있는 사용자는 모든 사용자에 대해 이 명령을 실행할 수 있지만 루트가 아닌 사용자는 이 명령을 실행하여 다른 사용자에 대한 할당량 정보를 가져올 수 없습니다.
`lfs quota -g groupname  /mount_point`	특정 그룹의 일반 할당량 정보.
`lfs quota -p projectid  /mount_point`	특정 프로젝트의 일반 할당량 정보.

할당량 명령 보기	할당량 정보 표시
`lfs quota -t -u /mount_point`	사용자 할당량의 블록 및 inode 유예 시간
`lfs quota -t -g /mount_point`	그룹 할당량의 블록 및 inode 유예 시간
`lfs quota -t -p /mount_point`	프로젝트 할당량의 블록 및 inode 유예 시간

할당량 및 Amazon S3 연결 버킷

FSx for Lustre 파일 시스템을 Amazon S3 데이터 리포지토리에 연결할 수 있습니다. 자세한 내용은 S3 버킷에 파일 시스템 연결 섹션을 참조하세요.

연결된 S3 버킷 내의 특정 폴더나 접두사를 파일 시스템의 가져오기 경로로 선택할 수도 있습니다. Amazon S3의 폴더를 지정하고 S3에서 파일 시스템으로 가져오면 해당 폴더의 데이터만 할당량에 적용됩니다. 전체 버킷의 데이터는 할당량 한도에 포함되지 않습니다.

연결된 S3 버킷의 파일 메타데이터는 Amazon S3에서 가져온 폴더와 구조가 일치하는 폴더로 가져옵니다. 이러한 파일은 파일을 소유한 사용자 및 그룹의 inode 할당량에 포함됩니다.

사용자가 파일을 `hsm_restore` 또는 지연 로드를 수행하면 파일의 전체 크기가 파일 소유자와 관련된 블록 할당량에 포함됩니다. 예를 들어 사용자 A가 사용자 B가 소유한 파일을 지연 로드하면 스토리지와 inode 사용량이 사용자 B의 할당량에 포함됩니다. 마찬가지로, 사용자가 Amazon FSx API를 사용하여 파일을 릴리스하면 파일을 소유한 사용자 또는 그룹의 블록 할당량에서 데이터가 비워집니다.

HSM 복원 및 지연 로딩은 루트 액세스를 통해 수행되므로 할당량 적용을 우회합니다. 데이터를 가져오면 S3에 설정된 소유권을 기준으로 사용자 또는 그룹에 포함되므로 사용자 또는 그룹이 블록 한도를 초과할 수 있습니다. 이 경우 파일 시스템에 다시 쓸 수 있도록 파일을 비워야 합니다.

마찬가지로, 자동 가져오기가 활성화된 파일 시스템은 S3에 추가된 객체에 대해 새 inode를 자동으로 생성합니다. 이러한 새 inode는 루트 액세스 권한으로 생성되며 생성되는 동안 할당량 적용을 우회합니다. 이러한 새 inode는 S3에서 객체를 소유한 사람을 기준으로 사용자와 그룹에 포함됩니다. 해당 사용자와 그룹이 자동 가져오기 활동에 따른 inode 할당량을 초과할 경우, 추가 용량을 확보하고 할당량 한도 미만으로 확보하기 위해 파일을 삭제해야 합니다.

할당량 및 백업 복원

백업을 복원하면 원래 파일 시스템의 할당량 설정이 복원된 파일 시스템에 구현됩니다. 예를 들어 파일 시스템 A에 할당량이 설정되어 있고 파일 시스템 A의 백업에서 파일 시스템 B가 생성되면 파일 시스템 A의 할당량이 파일 시스템 B에 적용됩니다.

스토리지 용량 관리

추가 스토리지 및 처리량이 필요한 경우 FSx for Lustre 파일 시스템에 구성된 스토리지 용량을 늘릴 수 있습니다. FSx for Lustre 파일 시스템의 처리량은 스토리지 용량에 따라 선형적으로 확장되므로 처리량 용량도 그에 비례하여 증가합니다. 스토리지 용량을 늘리려면 Amazon FSx 콘솔,AWS CLI() 또는 Amazon FSx AWS Command Line Interface API를 사용할 수 있습니다.

파일 시스템의 스토리지 용량 업데이트를 요청하면 Amazon FSx가 자동으로 새 네트워크 파일 서버를 추가하고 메타데이터 서버를 확장합니다. 스토리지 용량을 확장하는 동안 파일 시스템을 사용하지 못할 수 있습니다. 파일 시스템을 사용할 수 없는 상태에서 클라이언트가 실행한 파일 작업은 명백히 재시도되며 스토리지 확장이 완료된 후 실행 완료됩니다. 파일 시스템을 사용할 수 없는 동안에는 파일 시스템 상태가 `UPDATING`으로 설정됩니다. 스토리지 확장이 완료되면 파일 시스템 상태가 `AVAILABLE`으로 설정됩니다.

그런 다음 Amazon FSx는 기존 및 새로 추가된 파일 서버 전반에서 데이터를 명백하게 재조정하는 스토리지 최적화 프로세스를 실행합니다. 재조정은 파일 시스템 가용성에 영향을 주지 않고 백그라운드에서 수행됩니다. 재조정 중에는 데이터 이동에 리소스가 소비되므로 파일 시스템 성능이 저하될 수 있습니다. 대부분의 파일 시스템에서 스토리지 최적화는 몇 시간에서 며칠까지 걸립니다. 최적화 단계에서 파일 시스템에 액세스하여 사용할 수 있습니다.

Amazon FSx 콘솔, CLI 및 API를 사용하여 언제든지 스토리지 최적화 진행 상황을 추적할 수 있습니다. 자세한 내용은 스토리지 용량 증가 모니터링 섹션을 참조하세요.

주제

- 스토리지 용량 증가 시 고려 사항
- 스토리지 용량을 늘려야 하는 경우
- 동시 스토리지 크기 조정 및 백업 요청을 처리하는 방법
- 스토리지 용량을 늘리는 방법
- 스토리지 용량 증가 모니터링

스토리지 용량 증가 시 고려 사항

스토리지 용량을 늘릴 때 고려해야 할 몇 가지 중요한 항목은 다음과 같습니다.

- 증가만 - 파일 시스템의 스토리지 용량을 늘릴 수만 있고 스토리지 용량을 줄일 수는 없습니다.
- 증분 증가 - 스토리지 용량을 늘릴 때는 스토리지 용량 증가 대화 상자에 나열된 증분을 사용합니다.
- 증가 사이 경과 시간 - 마지막 증가 요청 후 6시간 또는 스토리지 최적화 프로세스가 완료될 때까지 (둘 중 더 긴 시간이 경과할 때까지) 파일 시스템의 스토리지 용량을 추가로 늘릴 수 없습니다.
- 처리량 용량 - 스토리지 용량을 늘리면 처리량 용량이 자동으로 증가합니다. SSD 캐시가 있는 영구 HDD 파일 시스템의 경우 읽기 캐시 스토리지 용량도 마찬가지로 증가하여 SSD 캐시를 HDD 스토리지 용량의 20% 로 유지합니다. Amazon FSx는 스토리지 및 처리량 용량 단위의 새 값을 계산하여 스토리지 용량 증가 대화 상자에 나열합니다.

  > **Note**
  >
  > 파일 시스템의 스토리지 용량을 업데이트하지 않고도 영구 SSD 기반 파일 시스템의 처리량 용량을 독립적으로 수정할 수 있습니다. 자세한 내용은 처리량 용량 관리 섹션을 참조하세요.

- 배포 유형 - 스크래치 1 파일 시스템을 제외한 모든 배포 유형의 스토리지 용량을 늘릴 수 있습니다. 스크래치 1 파일 시스템이 있는 경우에는 스토리지 용량이 더 큰 새 파일 시스템을 만들 수 있습니다.

스토리지 용량을 늘려야 하는 경우

여유 스토리지 용량이 부족할 경우 파일 시스템의 스토리지 용량을 늘립니다. `FreeStorageCapacity` CloudWatch 지표를 사용하여 파일 시스템에서 사용 가능한 무료 스토리지의 양을 모니터링할 수 있습니다. 이 지표에 Amazon CloudWatch 경보를 생성하고 특정 임계값 아래로 떨어지면 알림을 받을 수 있습니다. 자세한 정보는 아마존을 통한 모니터링 CloudWatch을 참조하세요.

CloudWatch 지표를 사용하여 파일 시스템의 지속적인 처리량 사용 수준을 모니터링할 수 있습니다. 파일 시스템에 더 높은 처리량 용량이 필요하다고 판단되면 지표 정보를 사용하여 스토리지 용량을 얼마나 늘릴지 결정할 수 있습니다. 파일 시스템의 현재 처리량을 확인하는 방법에 대한 자세한 내용은 Amazon FSx for Lustre 지표 사용 방법 섹션을 참조하세요. 스토리지 용량이 처리량 용량에 미치는 영향에 대한 자세한 내용은 Amazon FSx for Lustre 성능 섹션을 참조하세요.

또한 파일 시스템 세부 정보 페이지의 요약 패널에서 파일 시스템의 스토리지 용량과 총 처리량을 볼 수 있습니다.

동시 스토리지 크기 조정 및 백업 요청을 처리하는 방법

스토리지 크기 조정 워크플로가 시작되기 직전이나 진행 중에 백업을 요청할 수 있습니다. Amazon FSx가 두 요청을 처리하는 순서는 다음과 같습니다.

- 스토리지 확장 워크플로가 진행 중이고(스토리지 확장 상태는 `IN_PROGRESS` 이고 파일 시스템 상태는`UPDATING`) 백업을 요청하면 백업 요청이 대기열에 추가됩니다. 스토리지 확장이 스토리지 최적화 단계에 있을 때 백업 작업이 시작됩니다(스토리지 크기 조정 상태는 `UPDATED_OPTIMIZING` 이고 파일 시스템 상태는`AVAILABLE`).
- 백업이 진행 중이고(백업 상태는 `CREATING`) 스토리지 스케일링을 요청하면 스토리지 스케일링 요청이 대기열에 추가됩니다. 스토리지 확장 워크플로는 Amazon FSx가 백업을 Amazon S3로 전송할 때 시작됩니다(백업 상태는 `TRANSFERRING`).

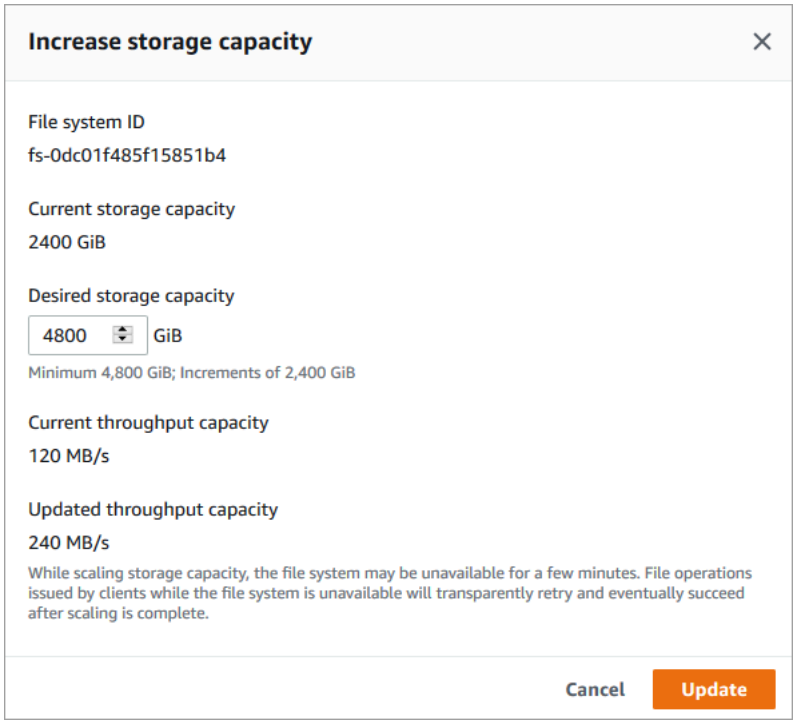
스토리지 확장 요청이 보류 중이고 파일 시스템 백업 요청도 보류 중인 경우 백업 작업의 우선 순위가 더 높습니다. 스토리지 조정 작업은 백업 작업이 완료될 때까지 시작되지 않습니다.

스토리지 용량을 늘리는 방법

Amazon FSx 콘솔, AWS CLI또는 Amazon FSx API를 사용하여 파일 시스템의 스토리지 용량을 늘릴 수 있습니다.

파일 시스템의 스토리지 용량 증가(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 파일 시스템으로 이동하여 스토리지 용량을 늘리려는 Lustre 파일 시스템을 선택합니다.
3. 작업에서 스토리지 용량 업데이트를 선택합니다. 또는 요약 패널에서 파일 시스템의 스토리지 용량 옆에 있는 업데이트를 선택하여 스토리지 용량 증가 대화 상자를 선택합니다.



4. 원하는 스토리지 용량의 경우 파일 시스템의 현재 스토리지 용량보다 큰 새 스토리지 용량을 GiB 단위로 입력합니다.

   - 영구 SSD 또는 스크래치 2 파일 시스템의 경우 이 값은 2,400GiB의 배수 단위여야 합니다.
   - 영구 HDD 파일 시스템의 경우 이 값은 12MB/s/TiB 파일 시스템의 경우 6,000GiB의 배수, 40MB/s/TiB 파일 시스템의 경우 1,800GiB의 배수 단위여야 합니다.

   > **Note**
   >
   > 스크래치 1 파일 시스템의 스토리지 용량을 늘릴 수 없습니다.

5. 업데이트를 선택하여 스토리지 용량 업데이트를 시작합니다.
6. 업데이트 탭의 파일 시스템 세부 정보 페이지에서 업데이트 진행 상황을 모니터링할 수 있습니다.

파일 시스템의 스토리지 용량 증가(CLI)

1. FSx for Lustre 파일 시스템의 스토리지 용량을 늘리려면 명령을 사용합니다. AWS CLI update-file-system 다음 파라미터를 설정합니다.

   업데이트하려는 파일 시스템 ID를 `--file-system-id`로 설정합니다.

`--storage-capacity`을 스토리지 용량 증가량(GiB)을 나타내는 정수 값으로 설정합니다. 영구 SSD 또는 스크래치 2 파일 시스템의 경우 이 값은 2,400의 배수여야 합니다. 영구 HDD 파일 시스템의 경우 이 값은 12MB/s/TiB 파일 시스템의 경우 6,000의 배수, 40MB/s/TiB 파일 시스템의 경우 1,800의 배수 단위여야 합니다. 새 대상 값은 파일 시스템의 현재 스토리지 용량보다 커야 합니다.

이 명령은 영구 SSD 또는 스크래치 2 파일 시스템의 스토리지 용량 대상 값을 9,600GiB로 지정합니다.

```
$ aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --storage-capacity 9600
```

2. 명령을 사용하여 업데이트 진행 상황을 모니터링할 수 있습니다. AWS CLI describe-file-systems 출력에서 `administrative-actions`를 찾습니다.

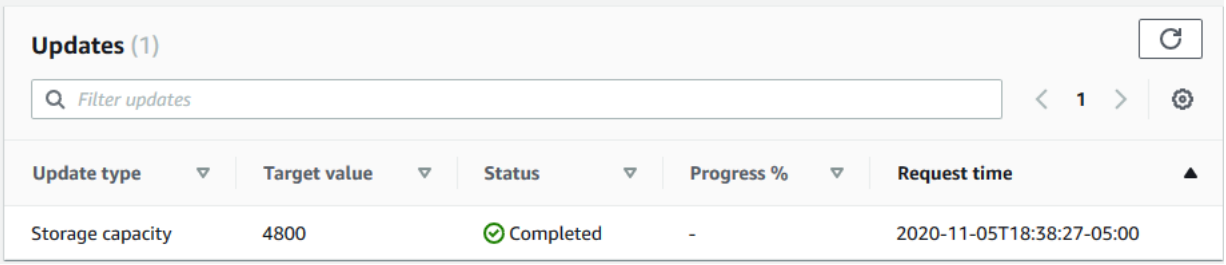
   자세한 내용은 을 참조하십시오 AdministrativeAction.

스토리지 용량 증가 모니터링

Amazon FSx 콘솔, API 및 AWS CLI를 사용하여 스토리지 용량 증가 진행 상황을 모니터링할 수 있습니다.

콘솔에서 증가 모니터링

파일 시스템 세부 정보 페이지의 업데이트 탭에서 각 업데이트 유형에 대한 최신 업데이트 10개를 볼 수 있습니다.



다음 정보를 볼 수 있습니다.

업데이트 유형

지원되는 유형은 스토리지 용량 및 스토리지 최적화입니다.

대상 값

파일 시스템의 스토리지 용량을 업데이트하려는 적정 값입니다.

상태

스토리지 용량의 현재 상태입니다. 가능한 값은 다음과 같습니다.

- 보류 중 – Amazon FSx가 업데이트 요청을 받았지만 처리를 시작하지 않았습니다.
- 진행 중 – Amazon FSx에서 업데이트 요청을 처리하고 있습니다.
- 업데이트 후 최적화 중 - Amazon FSx가 파일 시스템의 스토리지 용량을 늘렸습니다. 스토리지 최적화 프로세스에서 파일 서버 전반의 데이터를 재조정하고 있습니다.
- 완료 - 스토리지 용량 증가가 완료되었습니다.
- 실패 - 스토리지 용량 증가에 실패했습니다. 스토리지 업데이트가 실패한 자세한 이유를 보려면 ?를 선택합니다.

진행률(%)

스토리지 최적화 프로세스의 진행률을 완료율로 표시합니다.

요청 시간

Amazon FSx가 업데이트 작업 요청을 받은 시간입니다.

AWS CLI 및 API를 사용하면 모니터링이 증가합니다.

describe-file-systems AWS CLI 명령과 DescribeFileSystemsAPI 작업을 사용하여 파일 시스템 스토리지 용량 증가 요청을 보고 모니터링할 수 있습니다. `AdministrativeActions` 배열에 각 관리 작업 유형에 대한 가장 최근의 업데이트 작업 10개가 나열됩니다. 파일 시스템의 스토리지 용량을 늘리면 `FILE_SYSTEM_UPDATE` 및 `STORAGE_OPTIMIZATION`이라는 두 개의 `AdministrativeActions` 작업이 생성됩니다.

다음 예제는 describe-file-systems CLI 명령의 응답 발췌문을 보여줍니다. 파일 시스템의 스토리지 용량은 4,800GB이며, 스토리지 용량을 9,600GB로 늘리기 위한 관리 작업이 보류 중입니다.

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            .
```

```
            .
            .
            "StorageCapacity": 4800,
            "AdministrativeActions": [
                {
                     "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
                     "RequestTime": 1581694764.757,
                     "Status": "PENDING",
                     "TargetFileSystemValues": {
                         "StorageCapacity": 9600
                     }
                },
                {
                    "AdministrativeActionType": "STORAGE_OPTIMIZATION",
                    "RequestTime": 1581694764.757,
                    "Status": "PENDING",
                }
            ]
```

Amazon FSx는 FILE_SYSTEM_UPDATE 작업을 먼저 처리하여 파일 시스템에 새 파일 서버를 추가합니다. 파일 시스템에서 새 스토리지를 사용할 수 있게 되면 FILE_SYSTEM_UPDATE 상태가 UPDATED_OPTIMIZING으로 변경됩니다. 스토리지 용량은 더 큰 새로운 값을 보여주며, Amazon FSx는 STORAGE_OPTIMIZATION 관리 작업을 처리하기 시작합니다. 이는 describe-file-systems CLI 명령의 다음 응답 발췌문에 나와 있습니다.

ProgressPercent 속성은 스토리지 최적화 프로세스의 진행 상황을 표시합니다. 스토리지 최적화 프로세스가 완료되면 FILE_SYSTEM_UPDATE 작업 상태가 COMPLETED로 변경되고 STORAGE_OPTIMIZATION 작업이 더 이상 표시되지 않습니다.

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            .
            .
            .
            "StorageCapacity": 9600,
            "AdministrativeActions": [
                {
                    "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
                    "RequestTime": 1581694764.757,
                    "Status": "UPDATED_OPTIMIZING",
```

```
                    "TargetFileSystemValues": {
                        "StorageCapacity": 9600
                }
                },
                {
                    "AdministrativeActionType": "STORAGE_OPTIMIZATION",
                    "RequestTime": 1581694764.757,
                    "Status": "IN_PROGRESS",
                    "ProgressPercent": 50,
                }
            ]
```

스토리지 용량 증가에 실패하면 FILE_SYSTEM_UPDATE 작업 상태가 FAILED로 변경됩니다. 이 FailureDetails 속성은 다음 예제와 같이 실패에 대한 정보를 제공합니다.

```
{
    "FileSystems": [
        {
            "OwnerId": "111122223333",
            .
            .
            .
            "StorageCapacity": 4800,
            "AdministrativeActions": [
                {
                    "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
                    "FailureDetails": {
                        "Message": "string"
                    },
                    "RequestTime": 1581694764.757,
                    "Status": "FAILED",
                    "TargetFileSystemValues":
                        "StorageCapacity": 9600
                }
            ]
```

메타데이터 성능 관리

Amazon FSx 콘솔, Amazon FSx API 또는 () 를 사용하여 최종 사용자나 애플리케이션에 영향을 주지 않고 FSx for Lustre 파일 시스템의 메타데이터 구성을 업데이트할 수 있습니다. AWS Command Line

Interface AWS CLI업데이트 절차는 파일 시스템에 프로비저닝된 메타데이터 IOPS의 수를 증가시킵니다.

> **Note**
>
> Persistent_2 배포 유형 및 지정된 메타데이터 구성으로 생성된 FSx for Lustre 파일 시스템에서만 메타데이터 성능을 높일 수 있습니다.

파일 시스템의 향상된 메타데이터 성능을 몇 분 내에 사용할 수 있습니다. 메타데이터 성능 향상 요청이 최소 6시간 간격으로 이루어지면 언제든지 메타데이터 성능을 업데이트할 수 있습니다. 메타데이터 성능을 조정하는 동안 파일 시스템을 몇 분 동안 사용하지 못할 수 있습니다. 파일 시스템을 사용할 수 없을 때 클라이언트가 실행한 파일 작업은 투명하게 재시도되며 메타데이터 성능 확장이 완료된 후 결국 성공합니다. 새 메타데이터 성능 향상을 사용할 수 있게 되면 그에 대한 요금이 청구됩니다.

Amazon FSx 콘솔, CLI 및 API를 사용하여 언제든지 메타데이터 성능 증가 진행 상황을 추적할 수 있습니다. 자세한 정보는 메타데이터 구성 업데이트 모니터링을 참조하세요.

주제

- Lustre 메타데이터 성능 구성
- 메타데이터 성능 향상 시 고려 사항
- 메타데이터 성능을 향상시켜야 하는 경우
- 메타데이터 성능을 높이는 방법
- 메타데이터 구성 모드 변경
- 메타데이터 구성 업데이트 모니터링

Lustre 메타데이터 성능 구성

프로비저닝된 메타데이터 IOPS 수에 따라 파일 시스템에서 지원할 수 있는 메타데이터 작업의 최대 속도가 결정됩니다.

파일 시스템을 생성할 때는 자동 또는 사용자 프로비저닝의 두 가지 메타데이터 구성 모드 중 하나를 선택합니다.

- 자동 모드에서 Amazon FSx는 파일 시스템 스토리지 용량을 기반으로 파일 시스템의 메타데이터 IOPS 수를 자동으로 프로비저닝하고 조정합니다.

- 사용자 프로비저닝 모드에서는 파일 시스템에 프로비저닝할 메타데이터 IOPS 수를 지정합니다.

언제든지 자동 모드에서 사용자 프로비저닝 모드로 전환할 수 있습니다. 또한 파일 시스템에 프로비저닝된 메타데이터 IOPS 수가 자동 모드에서 프로비저닝된 기본 메타데이터 IOPS 수와 일치하는 경우 사용자 프로비저닝 모드에서 자동 모드로 전환할 수 있습니다.

유효한 메타데이터 IOPS 값은 1500, 3000, 6000, 12000이며 12000의 배수에서 최대 192000까지입니다. 각 12000 메타데이터 IOPS 값에는 파일 시스템이 상주하는 서브넷 내에 하나의 IP 주소가 필요합니다.

자동 모드에서 프로비저닝되는 메타데이터 IOPS의 기본 수는 파일 시스템 용량에 따라 달라집니다. 파일 시스템 스토리지 용량을 기반으로 프로비저닝되는 기본 메타데이터 IOPS 수에 대한 자세한 내용은 이 표를 참조하십시오.

워크로드의 메타데이터 성능이 자동 모드에서 프로비저닝된 메타데이터 IOPS 수를 초과하는 경우 사용자 프로비저닝 모드를 사용하여 파일 시스템의 메타데이터 IOPS 값을 높일 수 있습니다.

다음과 같이 파일 시스템 메타데이터 서버 구성의 현재 값을 볼 수 있습니다.

- 콘솔 사용 - 파일 시스템 세부 정보 페이지의 요약 패널에서 메타데이터 IOPS 필드에는 프로비저닝된 메타데이터 IOPS의 현재 값과 파일 시스템의 현재 메타데이터 구성 모드 (자동 또는 사용자 프로비저닝) 가 표시됩니다.
- CLI 또는 API 사용 - 파일 시스템 설명 CLI 명령 또는 시스템 API 작업을 사용하여 속성을 DescribeFile찾습니다. `MetadataConfiguration`

메타데이터 성능 향상 시 고려 사항

메타데이터 성능을 높일 때 고려해야 할 몇 가지 중요한 사항은 다음과 같습니다.

- 메타데이터 성능만 향상 - 파일 시스템의 메타데이터 IOPS 수만 늘릴 수 있고 메타데이터 IOPS 수는 줄일 수 없습니다.
- 자동 모드에서 메타데이터 IOPS 지정이 지원되지 않음 - 자동 모드인 파일 시스템에서는 메타데이터 IOPS 수를 지정할 수 없습니다. 사용자 프로비저닝 모드로 전환한 다음 요청을 해야 합니다. 자세한 정보는 메타데이터 구성 모드 변경을 참조하세요.
- 증가 사이의 시간 — 마지막 증가가 요청된 후 6시간이 지나야 파일 시스템에서 메타데이터 성능을 추가로 높일 수 있습니다.
- 동시 메타데이터 성능 및 SSD 스토리지 증가 - 메타데이터 성능과 파일 시스템 스토리지 용량을 동시에 확장할 수는 없습니다.

메타데이터 성능을 향상시켜야 하는 경우

파일 시스템에 기본적으로 프로비저닝되는 것보다 더 높은 수준의 메타데이터 성능이 필요한 워크로드를 실행해야 하는 경우 메타데이터 IOPS 수를 늘리십시오. 파일 시스템에서 사용하고 있는 프로비저닝된 AWS Management Console 메타데이터 서버 성능의 비율을 보여주는 `Metadata IOPS Utilization` 그래프를 사용하여 메타데이터 성능을 모니터링할 수 있습니다.

더 세분화된 CloudWatch 지표를 사용하여 메타데이터 성능을 모니터링할 수도 있습니다. CloudWatch 지표에는 디스크 IO가 필요한 메타데이터 서버 작업의 양을 제공하는 `DiskReadOperations` 및`DiskWriteOperations`, 파일 및 디렉터리 생성, 통계, 읽기 및 삭제를 비롯한 메타데이터 작업에 대한 세분화된 지표가 포함됩니다. 자세한 정보는 파일 시스템 메타데이터 메트릭을 참조하세요.

메타데이터 성능을 높이는 방법

Amazon FSx 콘솔, AWS CLI또는 Amazon FSx API를 사용하여 파일 시스템의 메타데이터 성능을 향상시킬 수 있습니다.

파일 시스템의 메타데이터 성능을 높이려면 (콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 왼쪽 탐색 열에서 파일 시스템을 선택합니다. 파일 시스템 목록에서 메타데이터 성능을 높이려는 FSx for Lustre 파일 시스템을 선택합니다.
3. [액션] 에서 [메타데이터 IOPS 업데이트] 를 선택합니다. 또는 요약 패널에서 파일 시스템의 메타데이터 IOPS 필드 옆에 있는 업데이트를 선택합니다.

   메타데이터 IOPS 업데이트 대화 상자가 나타납니다.
4. 사용자 프로비저닝을 선택합니다.
5. 원하는 메타데이터 IOPS의 경우 새 메타데이터 IOPS 값을 선택합니다. 유효한 값은`1500`,`3000`, `600012000`, 및 `12000` 최대 의 배수입니다. `192000` 입력하는 값은 현재 메타데이터 IOPS 값보다 크거나 같아야 합니다.
6. 업데이트를 선택합니다.

파일 시스템 (CLI) 의 메타데이터 성능을 높이려면

FSx for Lustre 파일 시스템의 메타데이터 성능을 높이려면 UpdateFileSystem update-file-system 명령을 AWS CLI 사용하십시오 (동일한 API 작업임). 다음 파라미터를 설정합니다.

- `--file-system-id`를 업데이트하려는 파일 시스템의 ID로 설정합니다.

- 메타데이터 성능을 높이려면 속성을 사용하세요. `--lustre-configuration MetadataConfiguration` 이 속성에는 두 개의 매개 변수와 가 `Iops` 있습니다. `Mode`
  1. 파일 시스템이 USER_PROVISIONED 모드인 경우 사용은 선택 `Mode` 사항입니다 (사용하는 경우로 설정). `Mode` `USER_PROVISIONED`

     파일 시스템이 자동 모드인 경우 `USER_PROVISIONED` (`Mode`로 설정하면 메타데이터 IOPS 값이 증가할 뿐만 아니라 파일 시스템 모드가 USER_PROVISIONED로 전환됨) 로 설정합니다.
  2. `1500`,, `3000` `600012000`, 또는 최대값의 `12000` 배수 `Iops` 값으로 설정합니다. `192000` 입력하는 값은 현재 메타데이터 IOPS 값보다 크거나 같아야 합니다.

다음 예에서는 프로비저닝된 메타데이터 IOPS를 96000으로 업데이트합니다.

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration 'MetadataConfiguration={Mode=USER_PROVISIONED,Iops=96000}'
```

메타데이터 구성 모드 변경

다음 절차에 설명된 대로 AWS 콘솔과 CLI를 사용하여 기존 파일 시스템의 메타데이터 구성 모드를 변경할 수 있습니다.

자동 모드에서 사용자 프로비저닝 모드로 전환할 때는 현재 파일 시스템 메타데이터 IOPS 값보다 크거나 같은 메타데이터 IOPS 값을 제공해야 합니다.

사용자 프로비저닝 모드에서 자동 모드로 전환을 요청하고 현재 메타데이터 IOPS 값이 자동 기본값보다 큰 경우, 메타데이터 IOPS 다운스케일링이 지원되지 않기 때문에 Amazon FSx는 요청을 거부합니다. 모드 전환의 차단을 해제하려면 자동 모드의 현재 메타데이터 IOPS와 일치하도록 스토리지 용량을 늘려야 모드 전환을 다시 활성화할 수 있습니다.

Amazon FSx 콘솔, AWS CLI또는 Amazon FSx API를 사용하여 파일 시스템의 메타데이터 구성 모드를 변경할 수 있습니다.

파일 시스템의 메타데이터 구성 모드를 변경하려면 (콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 왼쪽 탐색 열에서 파일 시스템을 선택합니다. 파일 시스템 목록에서 메타데이터 구성 모드를 변경할 FSx for Lustre 파일 시스템을 선택합니다.
3. [액션] 에서 [메타데이터 IOPS 업데이트] 를 선택합니다. 또는 요약 패널에서 파일 시스템의 메타데이터 IOPS 필드 옆에 있는 업데이트를 선택합니다.

메타데이터 IOPS 업데이트 대화 상자가 나타납니다.

4. 다음 중 하나를 수행하세요.

   - 사용자 프로비저닝 모드에서 자동 모드로 전환하려면 자동을 선택합니다.
   - 자동 모드에서 사용자 제공 모드로 전환하려면 사용자 프로비저닝을 선택합니다. 그런 다음 원하는 메타데이터 IOPS에 대해 현재 파일 시스템 메타데이터 IOPS 값보다 크거나 같은 메타데이터 IOPS 값을 제공하십시오.

5. 업데이트를 선택합니다.

파일 시스템 (CLI) 의 메타데이터 구성 모드를 변경하려면

FSx for Lustre 파일 시스템의 메타데이터 구성 모드를 변경하려면 UpdateFileSystem update-file-system 명령 (동일한 API 작업) 을 AWS CLI 사용합니다. 다음 파라미터를 설정합니다.

- `--file-system-id`를 업데이트하려는 파일 시스템의 ID로 설정합니다.
- 메타데이터 구성 모드를 변경하려면 속성을 사용하십시오. `--lustre-configuration MetadataConfiguration` 이 속성에는 두 개의 매개 변수와 가 `Iops` 있습니다. `Mode`
  - 자동 모드에서 USER_PROVISIONED 모드로 `Mode` 전환하려면 `USER_PROVISIONED` 메타데이터 IOPS 값을 현재 파일 시스템 메타데이터 IOPS 값보다 크거나 같은 값으로 설정하거나 그 값으로 설정하십시오. `Iops` 예:

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration
 'MetadataConfiguration={Mode=USER_PROVISIONED,Iops=96000}'
```

  - USER_PROVISIONED 모드에서 자동 모드로 전환하려면 파라미터를 사용하도록 설정하고 사용하지 마십시오. `Mode AUTOMATIC Iops` 예:

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration 'MetadataConfiguration={Mode=AUTOMATIC}'
```

메타데이터 구성 업데이트 모니터링

Amazon FSx 콘솔, API 또는 를 사용하여 메타데이터 구성 업데이트의 진행 상황을 모니터링할 수 있습니다. AWS CLI

메타데이터 구성 업데이트 모니터링 (콘솔)

파일 시스템 세부 정보 페이지의 업데이트 탭에서 메타데이터 구성 업데이트를 모니터링할 수 있습니다.

메타데이터 구성 업데이트의 경우 다음 정보를 볼 수 있습니다.

업데이트 유형

지원되는 유형은 메타데이터 IOPS 및 메타데이터 구성 모드입니다.

대상 값

파일 시스템의 메타데이터 IOPS 또는 메타데이터 구성 모드의 업데이트된 값입니다.

상태

업데이트의 현재 상태입니다. 가능한 값은 다음과 같습니다.

- 보류 중 – Amazon FSx가 업데이트 요청을 받았지만 처리를 시작하지 않았습니다.
- 진행 중 – Amazon FSx에서 업데이트 요청을 처리하고 있습니다.
- 완료 - 업데이트가 완료되었습니다.
- 실패 - 업데이트 요청이 실패했습니다. 물음표 (? 요청이 실패한 이유에 대한 세부 정보를 보려면 여기를 클릭하십시오.

요청 시간

Amazon FSx가 업데이트 작업 요청을 받은 시간입니다.

메타데이터 구성 업데이트 모니터링 (CLI)

describe-file-systems AWS CLI 명령과 시스템 API 작업을 사용하여 메타데이터 구성 업데이트 요청을 보고 모니터링할 수 있습니다. DescribeFile `AdministrativeActions` 배열에 각 관리 작업 유형에 대한 가장 최근의 업데이트 작업 10개가 나열됩니다. 파일 시스템의 메타데이터 성능 또는 메타데이터 구성 모드를 업데이트하면 a가 생성됩니다. `FILE_SYSTEM_UPDATE` `AdministrativeActions`

다음 예제는 describe-file-systems CLI 명령의 응답 발췌문을 보여줍니다. 파일 시스템에는 메타데이터 IOPS를 96000으로 늘리고 메타데이터 구성 모드를 USER_PROVISIONED로 늘리기 위한 관리 작업이 보류 중입니다.

```
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1678840205.853,
        "Status": "PENDING",
        "TargetFileSystemValues": {
            "LustreConfiguration": {
                "MetadataConfiguration": {
                    "Iops": 96000,
                    "Mode": USER_PROVISIONED
                }
            }
        }
    }
]
```

Amazon FSx는 파일 시스템의 메타데이터 FILE_SYSTEM_UPDATE IOPS와 메타데이터 구성 모드를 수정하여 작업을 처리합니다. 파일 시스템에서 새 메타데이터 리소스를 사용할 수 있게 되면 상태가 로 변경됩니다. FILE_SYSTEM_UPDATE COMPLETED

메타데이터 구성 업데이트 요청이 실패하면 다음 예와 같이 FILE_SYSTEM_UPDATE 작업 상태가 로 FAILED 변경됩니다. FailureDetails 속성은 실패에 대한 정보를 제공합니다.

```
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1678840205.853,
        "Status": "FAILED",
        "TargetFileSystemValues": {
            "LustreConfiguration": {
                "MetadataConfiguration": {
                    "Iops": 96000,
                    "Mode": USER_PROVISIONED
                }
            }
        },
        "FailureDetails": {
            "Message": "failure-message"
```

```
        }
    }
]
```

처리량 용량 관리

FSx for Lustre 파일 시스템에는 파일 시스템을 생성할 때 구성된 처리량 용량이 있습니다. FSx for Lustre 파일 시스템의 처리량은 테비바이트당 초당 메가바이트 단위로 측정됩니다(MB/초/TiB). 처리량 용량은 파일 시스템을 호스팅하는 파일 서버의 파일 데이터 서비스 제공 속도를 결정하는 요소 중 하나입니다. 처리량 용량이 높을수록 초당 I/O 작업 수(IOPS) 및 파일 서버의 데이터 캐싱을 위한 메모리 용량 또한 많아집니다. 자세한 내용은 Amazon FSx for Lustre 성능 섹션을 참조하세요.

스토리지 단위당 파일 시스템 처리량 값을 늘리거나 줄여 영구 SSD 기반 파일 시스템의 처리량 계층을 수정할 수 있습니다. 유효한 값은 다음과 같이 파일 시스템의 배포 유형에 따라 달라집니다.

- 영구_1 SSD 기반 배포 유형의 경우 유효한 값은 초당 50, 100, 200MB/TiB
- 영구_2 SSD 기반 배포 유형의 경우 유효한 값은 초당 125, 250, 500, 1,000MB/TiB

다음과 같이 스토리지 단위당 파일 시스템 처리량의 현재 값을 볼 수 있습니다.

- 콘솔 사용 - 파일 시스템 세부 정보 페이지의 요약 패널에서 스토리지 단위당 처리량 필드에 현재 값이 표시됩니다.
- CLI 또는 API 사용 - describe-file-systemsCLI 명령 또는 DescribeFileSystemsAPI 작업을 사용하여 속성을 찾습니다. `PerUnitStorageThroughput`

파일 시스템의 처리량 용량을 수정하면 Amazon FSx가 파일 시스템의 파일 서버를 백그라운드에서 교체합니다. 처리량 용량 조정 중에 몇 분 동안 파일 시스템을 사용할 수 없게 됩니다. 새 처리량 용량을 파일 시스템에서 사용할 수 있게 되면 요금이 청구됩니다.

주제

- 처리량 용량 업데이트 시 고려 사항
- 처리량 용량을 수정해야 하는 경우
- 처리량 용량을 수정하는 방법
- 처리량 용량 변화 모니터링

처리량 용량 업데이트 시 고려 사항

처리량 용량을 업데이트할 때 고려해야 할 몇 가지 중요한 항목은 다음과 같습니다.

- 증가 또는 감소 - 파일 시스템의 처리량 용량을 늘리거나 줄일 수 있습니다.
- 업데이트 증분 - 처리량 용량을 수정할 때는 처리량 계층 업데이트 대화 상자에 나열된 증분을 사용하십시오.
- 증가 사이 경과 시간 - 마지막 요청 후 6시간 또는 처리량 최적화 프로세스가 완료될 때까지(둘 중 더 긴 시간이 경과할 때까지) 파일 시스템의 처리량 용량을 추가로 변경할 수 없습니다.
- 배포 유형 - 영구 SSD 기반 배포 유형의 처리량 용량만 업데이트할 수 있습니다.

처리량 용량을 수정해야 하는 경우

Amazon FSx는 CloudWatch Amazon과 통합되므로 파일 시스템의 지속적인 처리량 사용 수준을 모니터링할 수 있습니다. 파일 시스템을 통해 구동할 수 있는 성능(처리량 및 IOPS)은 파일 시스템의 처리량 용량, 스토리지 용량, 스토리지 유형뿐 아니라 특정 워크로드의 특성에 따라 달라집니다. 파일 시스템의 현재 처리량을 확인하는 방법에 대한 자세한 내용은 Amazon FSx for Lustre 지표 사용 방법 섹션을 참조하세요. CloudWatch 지표에 대한 자세한 내용은 을 참조하십시오. 아마존을 통한 모니터링 CloudWatch

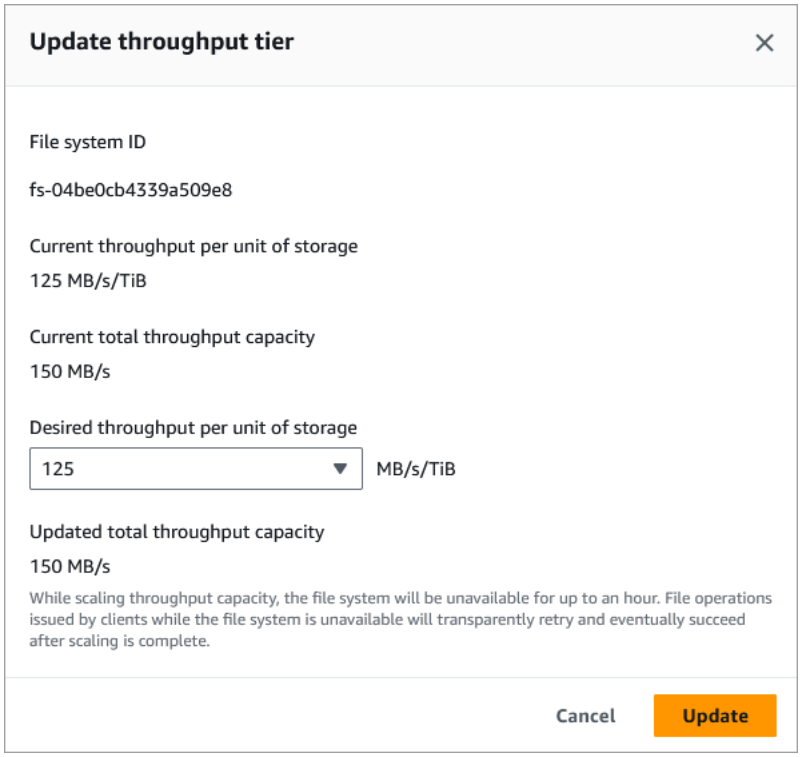
처리량 용량을 수정하는 방법

Amazon FSx 콘솔, AWS Command Line Interface (AWS CLI) 또는 Amazon FSx API를 사용하여 파일 시스템의 처리량 용량을 수정할 수 있습니다.

파일 시스템의 처리량 용량 수정(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 파일 시스템으로 이동하여 처리량 용량을 늘리려는 FSx for Lustre 파일 시스템을 선택합니다.
3. 작업에서 처리량 계층 업데이트를 선택합니다. 또는 요약 패널에서 파일 시스템의 스토리지 유닛당 처리량 옆에 있는 업데이트를 선택합니다.

   처리량 계층 업데이트 창이 나타납니다.
4. 목록에서 스토리지 단위당 원하는 처리량의 새 값을 선택합니다.



5. 업데이트를 선택하여 처리량 용량 업데이트를 시작합니다.

> **Note**
>
> 파일 시스템은 업데이트 중에 매우 짧은 기간 동안 사용할 수 없게 됩니다.

파일 시스템의 처리량 용량 수정(CLI)

- 파일 시스템의 처리 용량을 수정하려면 update-file-systemCLI 명령 (또는 이에 상응하는 UpdateFileSystemAPI 작업) 을 사용합니다. 다음 파라미터를 설정합니다.
  - `--file-system-id`를 업데이트하려는 파일 시스템의 ID로 설정합니다.
  - 영구_1 SSD 파일 시스템의 경우 `--lustre-configuration PerUnitStorageThroughput`을 `50`, `100` 또는 `200`MB/s/TiB의 값으로 설정하고, 영구_2 SSD 파일 시스템의 경우, `125`, `250`, `500` 또는 `1000`MB/s/TiB의 값으로 설정합니다.

  이 명령은 파일 시스템의 처리량 용량을 1,000MB/s/TiB로 설정하도록 지정합니다.

```
aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
```

```
    --lustre-configuration PerUnitStorageThroughput=1000
```

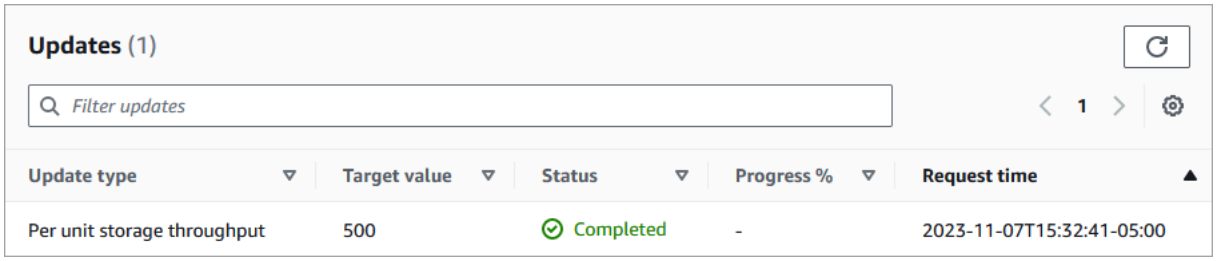
처리량 용량 변화 모니터링

Amazon FSx 콘솔, API 및 AWS CLI를 사용하여 처리량 용량 수정 진행 상황을 모니터링할 수 있습니다.

처리량 용량 변경 모니터링 (콘솔)

https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.

- 파일 시스템 세부 정보 창의 업데이트 탭에서 각 업데이트 작업 유형에 대한 최신 업데이트 작업 10개를 볼 수 있습니다.



  처리량 용량 업데이트 작업에서 다음 정보를 볼 수 있습니다.

  업데이트 유형

    지원되는 유형은 단위당 스토리지 처리량입니다.

  대상 값

    파일 시스템의 스토리지 유닛당 처리량을 변경할 원하는 값입니다.

  상태

    업데이트의 현재 상태입니다. 처리량 용량 업데이트에 사용할 수 있는 값은 다음과 같습니다.

    - 보류 중 – Amazon FSx가 업데이트 요청을 받았지만 처리를 시작하지 않았습니다.
    - 진행 중 – Amazon FSx에서 업데이트 요청을 처리하고 있습니다.
    - 업데이트 후 최적화 중 - Amazon FSx가 파일 시스템의 네트워크 I/O, CPU 및 메모리 리소스를 업데이트했습니다. 새로운 디스크 I/O 성능 수준을 쓰기 작업에 사용할 수 있습니다. 파일 시스템이 더 이상 이 상태가 아닐 때까지 읽기 작업에서는 이전 수준과 새 수준 사이의 디스크 I/O 성능을 확인할 수 있습니다.
    - 완료됨 – 처리량 용량 업데이트가 완료되었습니다.

- 실패 - 처리량 용량 업데이트에 실패했습니다. 처리량 업데이트가 실패한 자세한 이유를 보려면 물음표(?)를 선택합니다.

요청 시간

Amazon FSx가 업데이트 요청을 받은 시간입니다.

파일 시스템 업데이트 모니터링 (CLI)

- describe-file-systemsCLI 명령과 DescribeFileSystemsAPI 작업을 사용하여 파일 시스템 처리 용량 수정 요청을 보고 모니터링할 수 있습니다. `AdministrativeActions` 배열에 각 관리 작업 유형에 대한 가장 최근의 업데이트 작업 10개가 나열됩니다. 파일 시스템의 처리량 용량을 수정하면 `FILE_SYSTEM_UPDATE` 관리 작업이 생성됩니다.

  다음 예제는 `describe-file-systems` CLI 명령의 응답 발췌문을 보여줍니다. 파일 시스템의 스토리지 단위당 목표 처리량은 500MB/s/TiB입니다.

```
.
.
.
"AdministrativeActions": [
    {
        "AdministrativeActionType": "FILE_SYSTEM_UPDATE",
        "RequestTime": 1581694764.757,
        "Status": "PENDING",
        "TargetFileSystemValues": {
          "LustreConfiguration": {
            "PerUnitStorageThroughput": 500
          }
        }
    }
]
```

  Amazon FSx가 작업을 처리하면 상태가 `COMPLETED`로 변경됩니다. 그러면 파일 시스템에서 새 처리량 용량을 사용할 수 있으며 `PerUnitStorageThroughput` 속성에 표시됩니다.

  처리량 용량 수정에 실패하면 상태가 `FAILED`로 변경되고 `FailureDetails` 속성이 실패에 대한 정보를 제공합니다.

Lustre 데이터 압축

Lustre 데이터 압축 기능을 사용하면 고성능 Amazon FSx for Lustre 파일 시스템 및 백업 스토리지에서 비용을 절감할 수 있습니다. 데이터 압축이 활성화되면 Amazon FSx for Lustre는 새로 작성된 파일을 디스크에 쓰기 전에 자동으로 압축하고, 읽을 때 자동으로 압축을 해제합니다.

데이터 압축은 파일 시스템 성능에 부정적인 영향을 주지 않으면서 높은 수준의 압축을 제공하도록 최적화된 LZ4 알고리즘을 사용합니다. LZ4는 커뮤니티에서 신뢰하는 성능 지향적인 Lustre 알고리즘으로, 압축 속도와 압축된 파일 크기 사이의 균형을 제공합니다. 일반적으로 데이터 압축을 활성화해도 지연 시간에 측정 가능한 영향을 미치지는 않습니다.

데이터 압축은 Amazon FSx for Lustre 파일 서버와 스토리지 간에 전송되는 데이터의 양을 줄입니다. 아직 압축된 파일 형식을 사용하고 있지 않은 경우, 데이터 압축을 사용하면 전체 파일 시스템 처리량 용량이 증가하는 것을 확인할 수 있습니다. 데이터 압축과 관련된 처리량 용량 증가는 프런트 엔드 네트워크 인터페이스 카드를 가득 채운 후에 제한됩니다.

예를 들어 파일 시스템이 PERSISTENT-50 SSD 배포 유형인 경우 네트워크 처리량은 스토리지 TiB당 기준 250MB/s입니다. 디스크 처리량의 기준은 TiB당 50MB/s입니다. 데이터 압축을 사용하면 디스크 처리량이 기본 네트워크 처리량 한도인 TiB당 50MB/s에서 TiB당 최대 250MB/s까지 증가할 수 있습니다. 네트워크 및 디스크 처리량 제한에 대한 자세한 내용은 파일 시스템 성능 총계의 파일 시스템 성능 표를 참조하세요. 데이터 압축 성능에 대한 자세한 내용은 AWS 스토리지 블로그의 Amazon FSx for Lustre 데이터 압축으로 성능을 향상시키면서 비용을 절감하는 방법 게시물을 참조하세요.

주제

- 데이터 압축 관리
- 이전에 작성한 파일 압축
- 파일 크기 보기
- 지표 사용 CloudWatch

데이터 압축 관리

Amazon FSx for Lustre 파일 시스템을 생성할 때 데이터 압축을 켜거나 끌 수 있습니다. 콘솔 또는 API에서 Amazon FSx for Lustre 파일 시스템을 생성하면 데이터 압축이 기본적으로 해제됩니다. AWS CLI

파일 시스템을 생성할 때 데이터 압축 켜기(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.

2. 시작하기 섹션의 FSx for Lustre 파일 시스템 만들기 섹션에 설명된 새 파일 시스템 생성 절차를 따릅니다.
3. 파일 시스템 세부 정보 섹션에서 데이터 압축 유형으로 LZ4를 선택합니다.
4. 새 파일 시스템을 생성할 때와 마찬가지로 마법사를 완료합니다.
5. 검토 및 생성을 선택합니다.
6. Amazon FSx for Lustre 파일 시스템의 선택한 설정을 검토한 다음 파일 시스템 생성을 선택합니다.

파일 시스템을 사용할 수 있게 되면 데이터 압축이 켜집니다.

파일 시스템을 생성할 때 데이터 압축 켜기(CLI)

- 데이터 압축이 활성화된 상태에서 FSx for Lustre 파일 시스템을 생성하려면 다음 그림과 같이 Amazon FSx CLI 명령 `create-file-system`을 `DataCompressionType` 파라미터와 함께 사용합니다. 해당 API 작업은 CreateFileSystem입니다.

```
$ aws fsx create-file-system \
      --client-request-token CRT1234 \
      --file-system-type LUSTRE \
      --file-system-type-version 2.12 \
      --lustre-configuration
 DeploymentType=PERSISTENT_1,PerUnitStorageThroughput=50,DataCompressionType=LZ4 \
      --storage-capacity 3600 \
      --subnet-ids subnet-123456 \
      --tags Key=Name,Value=Lustre-TEST-1 \
      --region us-east-2
```

파일 시스템을 생성한 후 Amazon FSx에서는 다음 예에서처럼 파일 시스템 설명을 JSON으로 반환합니다.

```
{

    "FileSystems": [
        {
            "OwnerId": "111122223333",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
```

```
            "FileSystemTypeVersion": "2.12",
            "Lifecycle": "CREATING",
            "StorageCapacity": 3600,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "PERSISTENT_1",
                "DataCompressionType": "LZ4",
                "PerUnitStorageThroughput": 50
            }
        }
    ]
}
```

기존 파일 시스템의 데이터 압축 구성을 변경할 수도 있습니다. 기존 파일 시스템의 데이터 압축을 켜면 새로 작성된 파일만 압축되고 기존 파일은 압축되지 않습니다. 자세한 내용은 이전에 작성한 파일 압축 섹션을 참조하세요.

기존 파일 시스템에서 데이터 압축 업데이트(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 파일 시스템으로 이동하여 데이터 압축을 관리할 Lustre 파일 시스템을 선택합니다.
3. 작업에서 데이터 압축 유형 업데이트를 선택합니다.
4. 데이터 압축 유형 업데이트 대화 상자에서 데이터 압축을 켜려면 LZ4를 선택하고, 끄려면 없음을 선택합니다.
5. 업데이트를 선택합니다.
6. 업데이트 탭의 파일 시스템 세부 정보 페이지에서 업데이트 진행 상황을 모니터링할 수 있습니다.

기존 파일 시스템에서 데이터 압축 업데이트(CLI)

기존 FSx for Lustre 파일 시스템의 데이터 압축 구성을 업데이트하려면 명령을 사용합니다. AWS CLI update-file-system 다음 파라미터를 설정합니다.

- `--file-system-id`를 업데이트하려는 파일 시스템의 ID로 설정합니다.
- `--lustre-configuration DataCompressionType`을 `NONE`으로 설정하여 데이터 압축을 끄거나 `LZ4`를 LZ4 알고리즘을 사용하여 데이터 압축을 켜도록 설정합니다.

이 명령은 데이터 압축이 LZ4 알고리즘으로 켜지도록 지정합니다.

```
$ aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration DataCompressionType=LZ4
```

백업에서 파일 시스템을 생성할 때의 데이터 압축 구성

사용 가능한 백업을 사용하여 새로운 Amazon FSx for Lustre 파일 시스템을 생성할 수 있습니다. 백업에서 새 파일 시스템을 생성할 때는 백업의 `DataCompressionType` 설정을 사용하여 설정이 적용되므로 `DataCompressionType`을 지정할 필요가 없습니다. 백업에서 생성할 때 `DataCompressionType`을 지정하도록 선택한 경우 값이 백업 `DataCompressionType` 설정과 일치해야 합니다.

백업에 대한 설정을 보려면 Amazon FSx 콘솔의 백업 탭에서 해당 설정을 선택합니다. 백업의 세부 정보는 백업의 요약 페이지에 나열됩니다. `describe-backups` AWS CLI 명령을 실행할 수도 있습니다(해당하는 API 작업은 다음과 같습니다). `DescribeBackups`

이전에 작성한 파일 압축

Amazon FSx for Lustre 파일 시스템에서 데이터 압축이 해제되었을 때 생성된 파일은 압축되지 않습니다. 데이터 압축을 켜도 기존의 압축되지 않은 데이터는 자동으로 압축되지 않습니다.

Lustre 클라이언트 설치의 일부로 설치된 `lfs_migrate` 명령을 사용하여 기존 파일을 압축할 수 있습니다. 예제는 에서 사용할 수 있는 FSXL 압축을 참조하십시오. GitHub

파일 크기 보기

다음 명령을 사용하여 파일 및 디렉터리의 압축되지 않은 크기 및 압축된 크기를 볼 수 있습니다.

- du 압축된 크기를 표시합니다.

- `du --apparent-size` 압축되지 않은 크기를 표시합니다.
- `ls -l` 압축되지 않은 크기를 표시합니다.

다음 예제는 동일한 파일이 있는 각 명령의 출력을 보여줍니다.

```
$ du -sh samplefile
272M samplefile
$ du -sh --apparent-size samplefile
1.0G samplefile
$ ls -lh samplefile
-rw-r--r-- 1 root root 1.0G May 10 21:16 samplefile
```

이 `-h` 옵션은 사람이 읽을 수 있는 형식으로 크기를 인쇄하므로 이러한 명령에 유용합니다.

지표 사용 CloudWatch

Amazon CloudWatch Logs 지표를 사용하여 파일 시스템 사용량을 확인할 수 있습니다. `LogicalDiskUsage` 지표는 총 논리적 디스크 사용량(압축 제외)을 `PhysicalDiskUsage` 지표는 총 물리적 디스크 사용량(압축 포함)을 보여줍니다. 이 두 지표는 파일 시스템에서 데이터 압축을 활성화했거나 이전에 활성화한 경우에만 사용할 수 있습니다.

`LogicalDiskUsage` 통계의 `Sum`을 `PhysicalDiskUsage` 통계의 `Sum`으로 나누어 파일 시스템의 압축률을 확인할 수 있습니다. 지표 수학을 사용하여 이 비율을 계산하는 방법에 대한 자세한 내용은 지표 수학: 데이터 압축률 섹션을 참조하세요.

파일 시스템의 성능 모니터링에 대한 자세한 내용은 Amazon FSx for Lustre 모니터링 섹션을 참조하세요.

Lustre 루트 스쿼시

루트 스쿼시는 현재 네트워크 기반 액세스 제어 및 POSIX 파일 권한 위에 파일 액세스 제어 계층을 추가하는 관리 기능입니다. 루트 스쿼시 기능을 사용하면 FSx for Lustre 파일 시스템에 루트로 액세스하려는 클라이언트의 루트 수준 액세스를 제한할 수 있습니다.

FSx for Lustre 파일 시스템의 권한 관리와 같은 관리 작업을 수행하려면 루트 사용자 권한이 필요합니다. 그러나 루트 액세스는 사용자에게 무제한 액세스를 제공하므로 사용자는 권한 검사를 우회하여 파일 시스템 객체를 액세스, 수정 또는 삭제할 수 있습니다. 루트 스쿼시 기능을 사용하면 파일 시스템에 루트가 아닌 사용자 ID(UID)와 그룹 ID(GID)를 지정하여 데이터에 대한 무단 액세스 또는 삭제를 방지

할 수 있습니다. 파일 시스템에 액세스하는 루트 사용자는 스토리지 관리자가 설정한 제한된 권한을 가진 지정된 권한이 없는 사용자/그룹으로 자동 변환됩니다.

또한 루트 스쿼시 기능을 사용하면 루트 스쿼시 설정의 영향을 받지 않는 클라이언트 목록을 제공할 수도 있습니다. 이러한 클라이언트는 무제한 권한으로 루트로 파일 시스템에 액세스할 수 있습니다.

주제

- 루트 스쿼시 작동 방식
- 루트 스쿼시 관리

루트 스쿼시 작동 방식

루트 스쿼시 기능은 루트 사용자의 사용자 ID(UID)와 그룹 ID(GID)를 Lustre 시스템 관리자가 지정한 UID와 GID에 다시 매핑하여 작동합니다. 또한 루트 스쿼시 기능을 사용하면 UID/GID 재매핑이 적용되지 않는 클라이언트 세트를 선택적으로 지정할 수 있습니다.

새 FSx for Lustre 파일 시스템을 만들 때 루트 스쿼시는 기본적으로 비활성화되어 있습니다. FSx for Lustre 파일 시스템의 UID 및 GID 루트 스쿼시 설정을 구성하여 루트 스쿼시를 활성화합니다. UID 및 GID 값은 `0`에서 `4294967294` 범위의 정수입니다.

- UID 및 GID 값이 0이 아닌 경우 루트 스쿼시가 활성화됩니다. UID 및 GID 값은 다를 수 있지만 각각 0이 아닌 값이어야 합니다.
- UID 및 GID 값이 `0`(0)이면 루트를 나타내므로 루트 스쿼시가 비활성화됩니다.

파일 시스템 생성 중에 Amazon FSx 콘솔을 사용하여 다음과 같이 루트 스쿼시 속성에 루트 스쿼시 UID 및 GID 값을 제공할 수 있습니다. 파일 시스템 생성 시 루트 스쿼시를 활성화하려면 (콘솔) 와 같이 AWS CLI 또는 API와 함께 `RootSquash` 파라미터를 사용하여 UID 및 GID 값을 제공할 수도 있습니다. 파일 시스템(CLI) 생성 시 루트 스쿼시 활성화 방법

선택적으로 루트 스쿼시가 적용되지 않는 클라이언트의 NID 목록을 지정할 수도 있습니다. 클라이언트 NID는 클라이언트를 고유하게 식별하는 데 사용되는 Lustre 네트워크 식별자입니다. NID를 단일 주소 또는 주소 범위로 지정할 수 있습니다.

- 단일 주소는 클라이언트의 IP 주소 다음에 Lustre 네트워크 ID(예: `10.0.1.6@tcp`)를 지정하여 표준 Lustre NID 형식으로 설명됩니다.
- 주소 범위는 대시를 사용하여 범위를 구분하여 설명합니다(예: `10.0.[2-10].[1-255]@tcp`).
- 클라이언트 NID를 지정하지 않는 경우 루트 스쿼시에는 예외가 없습니다.

파일 시스템을 생성하거나 업데이트할 때 Amazon FSx 콘솔의 루트 스쿼시 예외 속성을 사용하여 클라이언트 NID 목록을 제공할 수 있습니다. AWS CLI 또는 API에서 파라미터를 사용합니다. `NoSquashNids` 자세한 내용은 의 절차를 참조하십시오루트 스쿼시 관리.

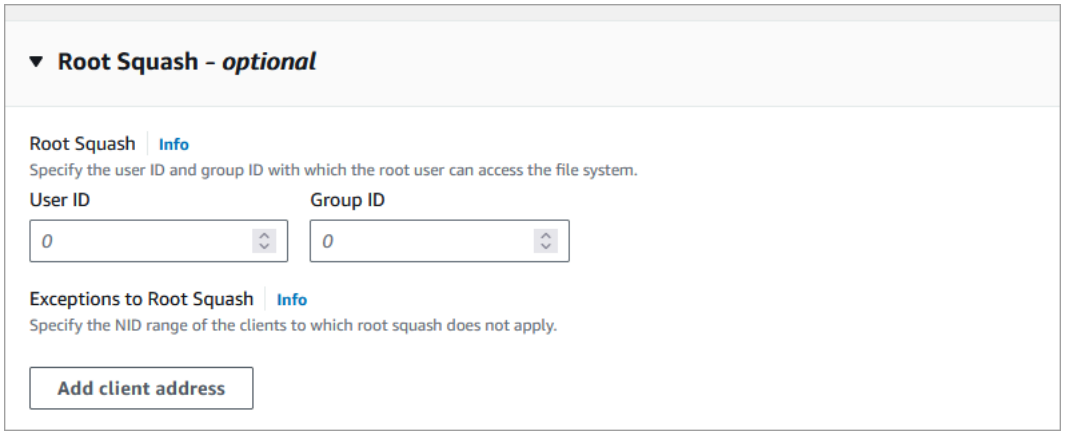
> **Note**
>
> 루트 스쿼시는 백업 및 복원에는 지원되지 않습니다. 백업 및 복원을 사용하려면 AWS CLI 또는 API를 사용하여 `RootSquash` 파라미터를 로 설정하고, Amazon FSx 콘솔의 루트 스쿼시 설정 업데이트 대화 상자에서 비활성화를 선택하여 루트 스쿼시를 비활성화해야 합니다. `0:0 NoSquashNids []`

루트 스쿼시 관리

파일 시스템 생성 중에는 루트 스쿼시가 기본적으로 비활성화됩니다. Amazon FSx 콘솔 또는 API에서 Lustre용 Amazon FSx 파일 시스템을 새로 생성할 때 루트 스쿼시를 활성화할 수 있습니다. AWS CLI

파일 시스템 생성 시 루트 스쿼시를 활성화하려면 (콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 시작하기 섹션의 FSx for Lustre 파일 시스템 만들기 섹션에 설명된 새 파일 시스템 생성 절차를 따릅니다.
3. 루트 스쿼시 - 옵션 섹션을 엽니다.



4. 루트 스쿼시의 경우 루트 사용자가 파일 시스템에 액세스할 수 있는 사용자 및 그룹 ID를 제공하십시오. 14294967294—의 범위 내에서 모든 정수를 지정할 수 있습니다.

   1. 사용자 ID의 경우 루트 사용자가 사용할 사용자 ID를 지정합니다.
   2. 그룹 ID의 경우 루트 사용자가 사용할 그룹 ID를 지정합니다.

5. (선택 사항) 루트 스쿼시 예외의 경우 다음을 수행하십시오.

   1. 클라이언트 주소 추가를 선택합니다.
   2. 클라이언트 주소 필드에서 루트 스쿼시가 적용되지 않는 클라이언트의 IP 주소를 지정합니다. IP 주소 형식에 대한 자세한 내용은 을 참조하십시오루트 스쿼시 작동 방식.
   3. 필요에 따라 반복하여 클라이언트 IP 주소를 더 추가합니다.
6. 새 파일 시스템을 생성할 때와 마찬가지로 마법사를 완료합니다.
7. 검토 및 생성을 선택합니다.
8. Amazon FSx for Lustre 파일 시스템의 선택한 설정을 검토한 다음 파일 시스템 생성을 선택합니다.

파일 시스템을 사용할 수 있게 되면 루트 스쿼시가 활성화됩니다.

파일 시스템(CLI) 생성 시 루트 스쿼시 활성화 방법

- 루트 스쿼시가 활성화된 상태에서 FSx for Lustre 파일 시스템을 생성하려면 Amazon FSx CLI 명령을 `RootSquashConfiguration` 파라미터와 함께 `create-file-system`을 사용합니다. 해당 API 작업은 CreateFileSystem입니다.

  `RootSquashConfiguration` 파라미터에 대해 다음 옵션 중 하나를 선택합니다.

  - `RootSquash` - 루트 사용자가 사용할 사용자 ID와 그룹 ID를 지정하는 콜론으로 구분된 UID:GID 값입니다. 각 ID에 대해 `0` - `4294967294`(0은 루트) 범위 내의 모든 정수를 지정할 수 있습니다(예: `65534:65534`).
  - `NoSquashNids` - 루트 스쿼시가 적용되지 않는 클라이언트의 Lustre 네트워크 식별자(NID)를 지정합니다. 클라이언트 NID 형식에 대한 자세한 내용은 루트 스쿼시 작동 방식 섹션을 참조하세요.

  다음 예제에서는 루트 스쿼시가 활성화된 FSx for Lustre 파일 시스템을 생성합니다.

```
$ aws fsx create-file-system \
      --client-request-token CRT1234 \
      --file-system-type LUSTRE \
      --file-system-type-version 2.15 \
      --lustre-configuration
 "DeploymentType=PERSISTENT_2,PerUnitStorageThroughput=250,DataCompressionType=LZ4,
\
          RootSquashConfiguration={RootSquash="65534:65534",\
```

```
            NoSquashNids=["10.216.123.47@tcp", "10.216.12.176@tcp"]}" \
      --storage-capacity 2400 \
      --subnet-ids subnet-123456 \
      --tags Key=Name,Value=Lustre-TEST-1 \
      --region us-east-2
```

파일 시스템을 생성한 후 Amazon FSx에서는 다음 예에서처럼 파일 시스템 설명을 JSON으로 반환합니다.

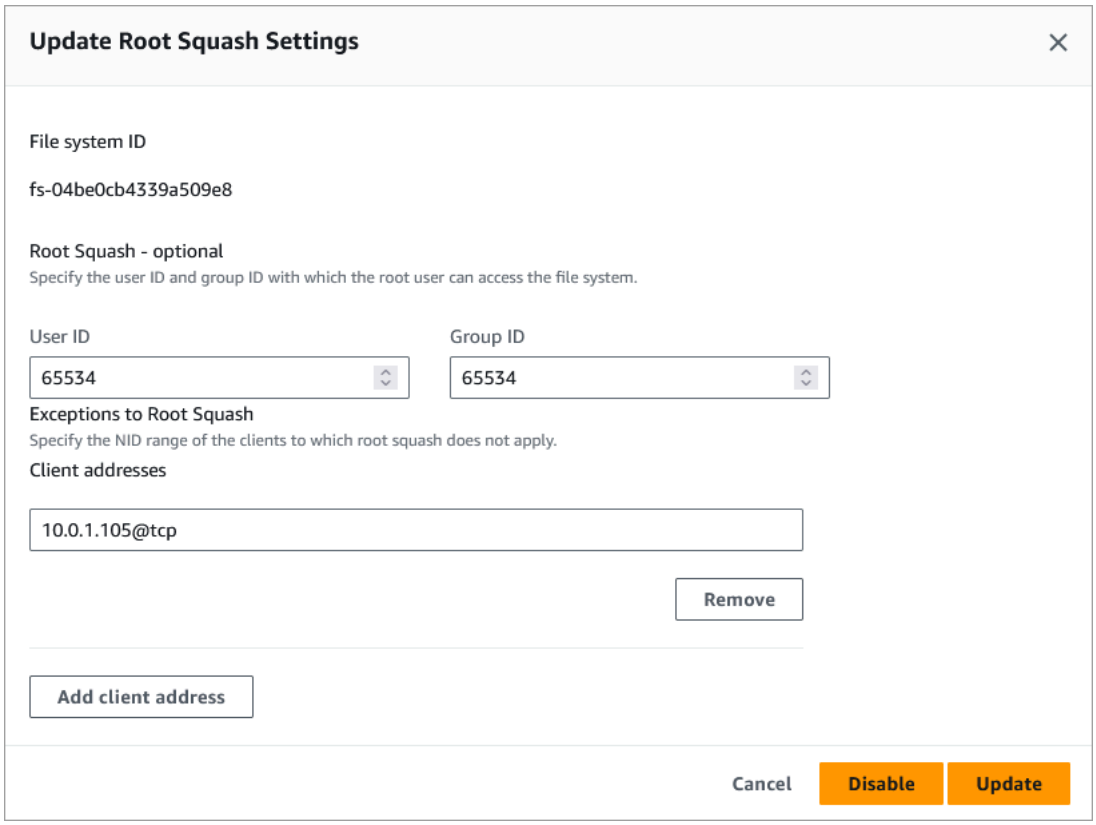
```
{

    "FileSystems": [
        {
            "OwnerId": "111122223333",
            "CreationTime": 1549310341.483,
            "FileSystemId": "fs-0123456789abcdef0",
            "FileSystemType": "LUSTRE",
            "FileSystemTypeVersion": "2.15",
            "Lifecycle": "CREATING",
            "StorageCapacity": 2400,
            "VpcId": "vpc-123456",
            "SubnetIds": [
                "subnet-123456"
            ],
            "NetworkInterfaceIds": [
                "eni-039fcf55123456789"
            ],
            "DNSName": "fs-0123456789abcdef0.fsx.us-east-2.amazonaws.com",
            "ResourceARN": "arn:aws:fsx:us-east-2:123456:file-system/
fs-0123456789abcdef0",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "Lustre-TEST-1"
                }
            ],
            "LustreConfiguration": {
                "DeploymentType": "PERSISTENT_2",
                "DataCompressionType": "LZ4",
                "PerUnitStorageThroughput": 250,
                "RootSquashConfiguration": {
                    "RootSquash": "65534:65534",
```

```
                    "NoSquashNids": "10.216.123.47@tcp 10.216.29.176@tcp"
                }
            }
        ]
}
```

Amazon FSx 콘솔 AWS CLI또는 API를 사용하여 기존 파일 시스템의 루트 스쿼시 설정을 업데이트할 수도 있습니다. 예를 들어 루트 스쿼시 UID 및 GID 값을 변경하거나, 클라이언트 NID를 추가 또는 제거하거나, 루트 스쿼시를 비활성화할 수 있습니다.

기존 파일 시스템에서 루트 스쿼시 설정을 업데이트하려면 (콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 파일 시스템으로 이동하여 루트 스쿼시를 관리할 Lustre 파일 시스템을 선택합니다.
3. Actions에서 루트 스쿼시 업데이트를 선택합니다. 또는 요약 패널에서 파일 시스템의 루트 스쿼시 필드 옆에 있는 업데이트를 선택하여 루트 스쿼시 설정 업데이트 대화 상자를 표시합니다.



4. 루트 스쿼시의 경우 루트 사용자가 파일 시스템에 액세스할 수 있는 사용자 및 그룹 ID를 업데이트하십시오. 04294967294—의 범위 내에서 임의의 정수를 지정할 수 있습니다. 루트 스쿼시를 비활성화하려면 두 ID 모두에 0 (0) 을 지정하십시오.

1. 사용자 ID의 경우 루트 사용자가 사용할 사용자 ID를 지정합니다.
2. 그룹 ID의 경우 루트 사용자가 사용할 그룹 ID를 지정합니다.

5. 루트 스쿼시 예외의 경우 다음을 수행하십시오.
   1. 클라이언트 주소 추가를 선택합니다.
   2. 클라이언트 주소 필드에서 루트 스쿼시가 적용되지 않는 클라이언트의 IP 주소를 지정합니다.
   3. 필요에 따라 반복하여 클라이언트 IP 주소를 더 추가합니다.
6. 업데이트를 선택합니다.

> **Note**
>
> 루트 스쿼시가 활성화되어 있는데 비활성화하려면 4-6단계를 수행하는 대신 비활성화를 선택합니다.

업데이트 탭의 파일 시스템 세부 정보 페이지에서 업데이트 진행 상황을 모니터링할 수 있습니다.

기존 파일 시스템(CLI)에서 루트 스쿼시 설정 업데이트

기존 FSx for Lustre 파일 시스템의 루트 스쿼시 설정을 업데이트하려면 명령을 사용합니다. AWS CLI update-file-system 해당 API 작업은 UpdateFileSystem입니다.

다음 파라미터를 설정합니다.

- `--file-system-id`를 업데이트하려는 파일 시스템의 ID로 설정합니다.
- `--lustre-configuration RootSquashConfiguration` 옵션을 다음과 같이 설정합니다.
  - `RootSquash` - 루트 사용자가 사용할 사용자 ID 및 그룹 ID를 지정하는 콜론으로 구분된 UID:GID 값을 설정합니다. 각 ID에 대해 `0` - `4294967294`(0은 루트) 범위 내의 모든 정수를 지정할 수 있습니다. 루트 스쿼시를 비활성화하려면 `0:0`에 대한 UID:GID 값을 지정합니다.
  - `NoSquashNids` - 루트 스쿼시가 적용되지 않는 클라이언트의 Lustre 네트워크 식별자(NID)를 지정합니다. `[]`를 모든 클라이언트 NID를 제거하는 데 사용합니다. 즉, 루트 스쿼시에는 예외가 없습니다.

이 명령은 루트 사용자의 사용자 ID 및 그룹 ID 값으로 `65534`를 사용하여 루트 스쿼시를 활성화하도록 지정합니다.

```
$ aws fsx update-file-system \
    --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration RootSquashConfiguration={RootSquash="65534:65534", \
          NoSquashNids=["10.216.123.47@tcp", "10.216.12.176@tcp"]}
```

명령이 성공하면 Amazon FSx for Lustre는 응답을 JSON 형식으로 반환합니다.

Amazon FSx 콘솔의 파일 시스템 세부 정보 페이지에 있는 요약 패널 또는 describe-file-systemsCLI 명령 (해당하는 API 작업) 의 응답에서 파일 시스템의 루트 스쿼시 설정을 볼 수 있습니다. DescribeFileSystems

FSx for Lustre 파일 시스템 상태

Amazon FSx 콘솔, AWS CLI 파일 시스템 설명 또는 API 운영 시스템을 사용하여 Amazon FSx 파일 시스템의 상태를 볼 수 있습니다. DescribeFile

파일 시스템 상태	설명
사용 가능	파일 시스템이 정상 상태이며 접속하여 사용할 수 있습니다.
생성 중	Amazon FSx가 새 파일 시스템을 생성하고 있습니다.
삭제 중	Amazon FSx가 기존 파일 시스템을 삭제하고 있습니다.
업데이트 중	파일 시스템이 고객이 시작한 업데이트를 진행 중입니다.
잘못 구성됨	파일 시스템이 실패했지만 복구 가능한 상태입니다.
실패함	이 상태는 다음 중 하나를 의미할 수 있습니다. • 파일 시스템에 오류가 발생하여 Amazon FSx가 복구할 수 없습니다. • 새 파일 시스템을 생성할 때 Amazon FSx가 파일 시스템을 생성하지 못했습니다.

Amazon FSx 리소스 태그 지정

파일 시스템 및 기타 Amazon FSx for Lustre 리소스 관리를 돕기 위해 태그 형식으로 각 리소스에 고유한 메타데이터를 할당할 수 있습니다. 태그를 사용하면 용도, 소유자 또는 환경 등 다양한 방식으로 AWS 리소스를 분류할 수 있습니다. 이 기능은 동일 유형의 리소스가 많을 때 유용합니다. 지정한 태그에 따라 특정 리소스를 빠르게 식별할 수 있습니다. 이 주제에서는 태그를 설명하고 태그를 생성하는 방법을 보여줍니다.

주제

- 태그 기본 사항
- 리소스 태그 지정
- 태그 제한
- 권한 및 태그

태그 기본 사항

태그는 AWS 리소스에 할당하는 레이블입니다. 각 태그는 사용자가 정의하는 키와 선택적 값으로 구성됩니다.

태그를 사용하면 용도, 소유자 또는 환경 등 다양한 방식으로 AWS 리소스를 분류할 수 있습니다. 예를 들어, 계정의 Amazon FSx for Lustre 파일 시스템에 대해 각 인스턴스의 소유자나 스택 수준을 추적하는 데 도움이 되는 태그 세트를 정의할 수 있습니다.

각 리소스 유형에 대한 요건을 충족하는 태그 키 세트를 고안하는 것이 좋습니다. 일관된 태그 키 세트를 사용하면 리소스를 보다 쉽게 관리할 수 있습니다. 추가하는 태그에 따라 리소스를 검색하고 필터링할 수 있습니다.

태그는 Amazon FSx에는 의미가 없으며 엄격하게 문자열로 해석됩니다. 또한 태그는 리소스에 자동으로 배정되지 않습니다. 태그 키와 값을 편집할 수 있으며 언제든지 리소스에서 태그를 제거할 수 있습니다. 태그의 값을 빈 문자열로 설정할 수 있지만 태그의 값을 Null로 설정할 수는 없습니다. 해당 리소스에 대해 키가 기존 태그와 동일한 태그를 추가하는 경우 새 값이 이전 값을 덮어씁니다. 리소스를 삭제하면 리소스 태그도 삭제됩니다.

Amazon FSx for Lustre API, AWS CLI AWS 또는 SDK를 사용하는 경우 API 작업을 사용하여 기존 리소스에 태그를 적용할 수 있습니다`TagResource`. 또한 일부 리소스 생성 작업에서는 리소스 생성 시 리소스의 태그를 지정할 수 있습니다. 리소스 생성 도중 태그를 적용할 수 없는 경우, 리소스 생성 프로

세스가 롤백됩니다. 이는 태그를 사용하여 리소스가 생성되거나 아예 리소스가 생성되지 않도록 하고 언제든 태그 지정되지 않은 리소스가 남지 않게 합니다. 생성 시 리소스에 태그를 지정하면 리소스 생성 후 사용자 지정 태그 지정 스크립트를 실행할 필요가 없습니다. 사용자가 생성 시 리소스 태그를 지정할 수 있도록 하는 방법에 대한 자세한 내용은 생성 시 리소스 태그 지정에 대한 권한 부여 섹션을 참조하세요.

리소스 태그 지정

계정에 존재하는 대부분의 Amazon FSx for Lustre 리소스에 태그를 지정할 수 있습니다. Amazon FSx 콘솔을 사용하는 경우, 관련 리소스 화면에서 태그 탭을 사용하여 리소스에 태그를 적용할 수 있습니다. 리소스를 생성할 때 Name 키를 값과 함께 적용할 수 있으며, 새 파일 시스템을 생성할 때 원하는 태그를 적용할 수 있습니다. 콘솔은 Name 태그에 따라 리소스를 조직할 수 있지만 이 태그는 Amazon FSx for Lustre 서비스에 대한 의미가 없습니다.

생성 시 태그를 지원하는 Amazon FSx for Lustre API 작업에 IAM 정책의 태그 기반 리소스 수준 권한을 적용하여 생성 시 리소스에 태그를 지정할 수 있는 사용자와 그룹을 세밀하게 제어할 수 있습니다. 리소스를 생성하면 태그가 즉시 적용되기 때문에 생성 단계부터 리소스를 적절하게 보호할 수 있습니다. 따라서 태그를 기반으로 리소스 사용을 제어하는 리소스 권한이 즉시 발효됩니다. 이에 따라 더욱 정확한 리소스 추적 및 보고가 가능합니다. 새 리소스에서 태그 지정 사용을 적용하고 리소스에서 어떤 태그 키와 값이 설정되는지 제어할 수 있습니다.

IAM 정책에서 `TagResource` 및 `UntagResource` Amazon FSx for Lustre API 작업에 리소스 수준 권한을 적용하여 기존 리소스에서 어떤 태그 키와 값이 설정되는지 제어할 수도 있습니다.

결제를 위한 리소스 태그 지정에 대한 자세한 내용은 AWS Billing 사용 설명서에서 비용 할당 태그 사용을 참조하세요.

태그 제한

태그에 적용되는 기본 제한은 다음과 같습니다.

- 리소스당 최대 태그 수 – 50개
- 각 리소스에 대해 각 태그 키는 고유하며 하나의 값만 가질 수 있습니다.
- 최대 키 길이 - UTF-8 형식의 유니코드 문자 128자
- 최대 값 길이 - UTF-8 형식의 유니코드 문자 256자
- Amazon FSx for Lustre 태그에서 허용되는 문자는 UTF-8로 표현할 수 있는 문자, 숫자 및 공백과 특수 문자 + - = . _ : / @입니다.

- 태그 키와 값은 대/소문자를 구분합니다.
- 접두사는 사용하도록 예약되어 있습니다. `aws:` AWS 태그에 이 접두사가 있는 태그 키가 있는 경우 태그의 키 또는 값을 편집하거나 삭제할 수 없습니다. `aws:` 접두사가 지정된 태그는 리소스당 태그 수 제한에 포함되지 않습니다.

태그에만 기초하여 리소스를 삭제할 수 없습니다. 리소스 식별자를 지정해야 합니다. 예를 들어 `DeleteMe`라는 태그 키로 태그를 지정한 파일 시스템을 삭제하려면 해당 파일 시스템 리소스 식별자(예: fs-1234567890abcdef0)를 지정하여 `DeleteFileSystem` 작업을 사용해야 합니다.

공개 또는 공유 리소스에 태그를 지정하면 할당한 태그는 본인만 사용할 수 있으며 다른 AWS 계정 사람은 해당 태그에 액세스할 수 없습니다 AWS 계정. 공유 리소스에 대한 태그 기반 액세스 제어의 경우 각 리소스에 대한 액세스를 제어하는 자체 태그 세트를 AWS 계정 할당해야 합니다.

권한 및 태그

생성 시 Amazon FSx 리소스에 태그를 지정하는 데 필요한 권한에 대한 정보는 생성 시 리소스 태그 지정에 대한 권한 부여 섹션을 참조하세요. IAM 정책에서 태그를 사용하여 Amazon FSx 리소스에 대한 액세스를 제한하는 방법에 대한 자세한 내용은 태그를 사용하여 Amazon FSx 리소스에 대한 액세스 제어 섹션을 참조하세요.

Amazon FSx for Lustre 유지 관리 기간

Amazon FSx for Lustre는 관리하는 Lustre 소프트웨어에 대해 정기적인 소프트웨어 패치를 수행합니다. 유지 관리 기간을 통해 이 소프트웨어 패치가 적용되는 요일과 시간을 제어할 수 있습니다.

패치 적용에는 30분 유지 관리 기간 중 극히 일부만 필요합니다. 이 몇 분 동안에는 파일 시스템을 일시적으로 사용할 수 없습니다. 파일 시스템 생성 중에 유지 관리 기간을 선택합니다. 원하는 시간이 없는 경우 30분의 기본 기간이 지정됩니다.

FSx for Lustre를 사용하면 워크로드 및 운영 요구 사항에 맞게 필요에 따라 유지 관리 기간을 조정할 수 있습니다. 유지 관리 기간이 14일에 한 번 이상으로 예약되어 있는 경우, 필요에 따라 유지 관리 기간을 수시로 이동할 수 있습니다. 패치가 릴리스되고 14일 이내에 유지 관리 기간을 예약하지 않은 경우 FSx for Lustre는 파일 시스템의 보안 및 신뢰성을 보장하기 위해 파일 시스템에 대한 유지 관리를 진행합니다.

Amazon FSx 관리 콘솔 AWS CLI AWS , API 또는 SDK 중 AWS 하나를 사용하여 파일 시스템의 유지 관리 기간을 변경할 수 있습니다.

콘솔을 사용하여 유지 관리 기간 변경

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 탐색 창에서 파일 시스템을 선택합니다.
3. 유지 관리 기간을 변경하려는 파일 시스템을 선택합니다. 파일 시스템 세부 정보 페이지가 표시됩니다.
4. 유지 관리 탭을 선택합니다. 유지 관리 기간 설정 패널이 표시됩니다.
5. 편집을 선택하고 유지 관리 기간을 시작하려는 새 날짜 및 시간을 선택합니다.
6. 저장을 선택하여 변경 사항을 저장합니다. 새 유지 관리 시작 시간이 설정 패널에 표시됩니다.

update-file-systemCLI 명령을 사용하여 파일 시스템의 유지 관리 기간을 변경할 수 있습니다. 다음 명령을 실행하여 파일 시스템 ID를 파일 시스템의 ID로 바꾸고 기간을 시작하려는 날짜 및 시간으로 바꿉니다.

```
aws fsx update-file-system --file-system-id fs-01234567890123456 --lustre-configuration
 WeeklyMaintenanceStartTime=1:01:30
```

파일 시스템 삭제

Amazon FSx 콘솔, 및 Amazon FSx API를 사용하여 Lustre용 Amazon FSx AWS CLI파일 시스템을 삭제할 수 있습니다. FSx for Lustre 파일 시스템을 삭제하기 전에 연결된 모든 Amazon EC2 인스턴스에서 해당 파일 시스템을 마운트 해제해야 합니다. S3 연결 파일 시스템에서는 파일 시스템을 삭제하기 전에 모든 데이터가 S3에 다시 기록되도록 AgeOfOldestQueued메시지 지표가 0이 되도록 모니터링하거나 (자동 내보내기를 사용하는 경우) 데이터 리포지토리 내보내기 작업을 실행할 수 있습니다. 자동 내보내기를 활성화한 상태에서 데이터 리포지토리 내보내기 작업을 사용하려면 데이터 리포지토리 내보내기 작업을 실행하기 전에 자동 내보내기를 비활성화해야 합니다.

모든 Amazon EC2 인스턴스에서 마운트를 해제한 후 파일 시스템 삭제

- 콘솔 사용 - 리소스 정리에 설명된 절차를 따릅니다.
- API 또는 CLI 사용 - DeleteFile시스템 API 작업 또는 파일 시스템 삭제 CLI 명령을 사용합니다.

를 사용하여 Amazon FSx for Lustre로 마이그레이션하기 AWS DataSync

를 AWS DataSync 사용하여 FSx for Lustre 파일 시스템 간에 데이터를 전송할 수 있습니다. DataSync 인터넷을 통해 자체 관리형 스토리지 시스템과 스토리지 서비스 간에 데이터를 이동 및 복제하는 작업을 간소화, 자동화 및 가속화하는 데이터 전송 서비스입니다. AWS AWS Direct Connect DataSync 소유권, 타임스탬프, 액세스 권한 등의 파일 시스템 데이터와 메타데이터를 전송할 수 있습니다.

AWS DataSync을(를) 사용하여 기존 파일을 FSx for Lustre로 마이그레이션하는 방법

FSx for Lustre 파일 시스템과 DataSync 함께 사용하여 일회성 데이터 마이그레이션을 수행하고, 분산 워크로드를 위해 데이터를 주기적으로 수집하고, 데이터 보호 및 복구를 위한 복제 일정을 예약할 수 있습니다. 특정 전송 시나리오에 대한 자세한 내용은 AWS DataSync 사용 설명서에서 어디로 데이터를 전송할 수 있나요?를 참조하세요.

필수 조건

데이터를 FSx for Lustre 설정으로 마이그레이션하려면 요구 사항을 충족하는 서버와 네트워크가 필요합니다. DataSync 자세한 내용은 사용 설명서의 요구 사항을 참조하십시오 DataSync.AWS DataSync

- FSx for Lustre 파일 시스템용 대상을 생성했습니다. 자세한 내용은 FSx for Lustre 파일 시스템 만들기 섹션을 참조하세요.
- 소스 파일 시스템과 대상 파일 시스템이 동일한 Virtual Private Cloud(VPC)에 연결되어 있습니다. 소스 파일 시스템은 온프레미스 또는 다른 Amazon VPC에 위치할 수 있지만 Amazon VPC AWS 계정 피어링, Transit Gateway AWS 리전또는 를 사용하는 대상 파일 시스템과 피어링된 네트워크에 있어야 합니다. AWS Direct Connect AWS VPN자세한 내용은 Amazon VPC 피어링 가이드의 VPC 피어링이란? 섹션을 참조하세요.

> **Note**
>
> DataSync 다른 전송 위치가 Amazon S3인 경우에만 FSx for AWS 계정 Lustre로 또는 FSx for Lustre에서 데이터를 주고받을 수 있습니다.

를 사용하여 파일을 마이그레이션하는 기본 단계 DataSync

를 사용하여 소스에서 대상으로 파일을 DataSync 전송하려면 다음과 같은 기본 단계가 필요합니다.

- 사용자 환경에 에이전트를 다운로드하여 배포하고 활성화합니다 (에이전트를 다른 곳으로 전송하는 경우에는 필요 없음 AWS 서비스).
- 소스 및 대상 위치 생성.
- 작업 생성.
- 작업을 실행하여 소스에서 대상으로 파일 전송.

자세한 내용은 AWS DataSync 사용 설명서의 다음 항목을 참조하십시오.

- 온프레미스 스토리지 간 전송 및 AWS
- Amazon FSx for Lustre를 사용하여 AWS DataSync 전송을 구성하는 방법은 사용 설명서를 참조하십시오.AWS DataSync
- Amazon EC2에 에이전트 배포

Amazon FSx for Lustre 모니터링

다음과 같은 자동 모니터링 도구를 사용하여 Amazon FSx for Lustre를 관찰하고 문제 발생 시 보고할 수 있습니다.

- Amazon을 사용한 모니터링 CloudWatch — Amazon FSx for Lustre에서 원시 데이터를 CloudWatch 수집하여 읽기 쉽고 실시간에 가까운 지표로 처리합니다. CloudWatch 경보 상태가 변경될 때 Amazon SNS 메시지를 보내는 경보를 생성할 수 있습니다.
- Lustre 로깅을 사용한 모니터링 - 파일 시스템에 대해 활성화된 로깅 이벤트를 모니터링할 수 있습니다. Lustre 로깅은 이러한 이벤트를 Amazon CloudWatch Logs에 기록합니다.
- AWS CloudTrail 로그 모니터링 — 계정 간에 로그 파일을 공유하고, CloudTrail 로그 파일을 CloudWatch Logs로 전송하여 실시간으로 모니터링하고, Java로 로그 처리 애플리케이션을 작성하고, 전송 후 로그 파일이 변경되지 않았는지 확인합니다. CloudTrail

주제

- 아마존을 통한 모니터링 CloudWatch
- Amazon CloudWatch 로그로 로깅
- 를 사용하여 FSx for Lustre API 호출을 로깅합니다. AWS CloudTrail

아마존을 통한 모니터링 CloudWatch

Amazon CloudWatch FSx for Lustre의 원시 데이터를 수집하여 읽기 쉬운 실시간에 가까운 지표로 처리하는 Amazon을 사용하여 파일 시스템을 모니터링할 수 있습니다. 이러한 통계는 15개월간 보관되므로 기록 정보를 보고 웹 애플리케이션이나 서비스가 어떻게 실행되고 있는지 전체적으로 더 잘 파악할 수 있습니다. 기본적으로 Amazon FSx for Lustre 지표 데이터는 1분 간격으로 자동으로 전송됩니다. CloudWatch 에 대한 CloudWatch 자세한 내용은 Amazon이란 무엇입니까 CloudWatch? 를 참조하십시오. Amazon CloudWatch 사용 설명서에서 확인할 수 있습니다.

CloudWatch 지표는 원시 바이트로 보고됩니다. 바이트는 단위의 십진수나 이진수에 반올림되지 않습니다.

파일 시스템 지표

FSx for Lustre는 다음 메트릭을 의 네임스페이스에 게시합니다. FSx CloudWatch FSx for Lustre는 각 지표에 대해 분당 디스크당 데이터 포인트를 생성합니다. Sum 통계를 사용하여 전체 파일 시스템 세부

정보를 볼 수 있습니다. FSx for Lustre 파일 시스템 뒤에 있는 파일 서버는 여러 디스크에 분산되어 있다는 점에 유의하세요.

지표	설명
DataReadBytes	각 파일 시스템 읽기 작업의 바이트 수 Sum 통계는 기간 동안 읽기 작업과 연결된 총 바이트 수입니다. Minimum 통계는 단일 디스크 내 읽기 작업과 연결된 최소 바이트 수입니다. Maximum 통계는 디스크 내 읽기 작업과 연결된 최대 바이트 수입니다. Average 통계는 디스크 당 읽기 작업과 연결된 평균 바이트 수입니다. SampleCount 통계는 디스크 수입니다. 일정 기간 평균 처리량(바이트/초)을 계산하려면 Sum 통계를 초 단위의 해당 기간으로 나누면 됩니다. 단위: • Sum, Minimum, Maximum 및 Average 통계일 때 바이트 • SampleCount 통계일 때 수 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount
DataWriteBytes	각 파일 쓰기 작업의 바이트 수 Sum 통계는 쓰기 작업과 연결된 총 바이트 수입니다. Minimum 통계는 단일 디스크 내 쓰기 작업과 연결된 최소 바이트 수입니다. Maximum 통계는 디스크 내 쓰기 작업과 연결된 최대 바이트 수입니다. Average 통계는 디스크 당 쓰기 작업과 연결된 평균 바이트 수입니다. SampleCount 통계는 디스크 수입니다. 일정 기간 평균 처리량(바이트/초)을 계산하려면 Sum 통계를 초 단위의 해당 기간으로 나누면 됩니다. 단위: • Sum, Minimum, Maximum 및 Average 통계일 때 바이트 • SampleCount 통계일 때 수

지표	설명
	유효한 통계: Sum, Minimum, Maximum, Average, SampleCount
DataReadOperations	읽기 작업 수입니다. Sum 통계는 총 읽기 작업 수입니다. Minimum 통계는 단일 디스크의 최소 읽기 작업 수입니다. Maximum 통계는 디스크의 최대 읽기 작업 수입니다. Average 통계는 디스크 당 읽기 작업의 평균 크기입니다. SampleCount 통계는 디스크 수입니다. 일정 기간 평균 읽기 작업 수(초당 작업 수)를 계산하려면 Sum 통계를 초 단위의 해당 기간으로 나누면 됩니다. 단위: • Sum, Minimum, Maximum 및 Average 통계일 때 바이트 • SampleCount 통계일 때 수 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount
DataWriteOperations	초당 쓰기 작업 수 Sum 통계는 총 쓰기 작업 수입니다. Minimum 통계는 단일 디스크의 최소 쓰기 작업 수입니다. Maximum통계는 디스크의 최대 쓰기 작업 수입니다. Average 통계는 디스크 당 쓰기 작업의 평균 수입니다. SampleCount 통계는 디스크 수입니다. 일정 기간 평균 쓰기 작업 수(초당 작업 수)를 계산하려면 Sum 통계를 초 단위의 해당 기간으로 나누면 됩니다. 단위: • Sum, Minimum, Maximum 및 Average 통계일 때 바이트 • SampleCount 통계일 때 수 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount

지표	설명
MetadataOperations	메타데이터 작업 수. Sum 통계는 메타데이터 작업의 수를 제공합니다. Minimum 통계는 디스크당 최소 메타데이터 작업 수입니다. Maximum 통계는 디스크당 최대 메타데이터 작업 수입니다. Average 통계는 디스크당 평균 메타데이터 작업 수입니다. SampleCount 통계는 디스크 수입니다. 일정 기간 평균 메타데이터 작업 수(초당 작업 수)를 계산하려면 Sum 통계를 초 단위의 해당 기간으로 나누면 됩니다. 단위: • Sum, Minimum, Maximum, Average 및 SampleCount 를 셉니다. 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount
FreeDataStorageCapacity	사용 가능한 스토리지 용량 크기입니다. Sum 통계는 파일 시스템에서 사용할 수 있는 총 바이트 수입니다. Minimum 통계는 가장 꽉 찬 디스크에서 사용할 수 있는 총 바이트 수입니다. Maximum 통계는 사용 가능한 스토리지가 가장 많이 남아 있는 디스크에서 사용 가능한 총 바이트 수입니다. Average 통계는 디스크당 사용 가능한 평균 바이트 수입니다. SampleCount 통계는 디스크 수입니다. 단위: • Sum, Minimum, Maximum의 바이트 • SampleCount 통계일 때 수 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount

지표	설명
LogicalDiskUsage	저장된(압축되지 않음) 논리 데이터의 양. Sum 통계는 파일 시스템에 저장된 총 논리적 바이트 수입니다. Minimum 통계는 파일 시스템의 디스크에 저장된 최소 논리적 바이트 수입니다. Maximum 통계는 파일 시스템의 디스크에 저장된 최대 논리적 바이트 수입니다. Average 통계는 디스크당 저장된 논리적 바이트의 평균 수입니다. SampleCount 통계는 디스크 수입니다. 단위: • Sum, Minimum, Maximum의 바이트 • SampleCount 통계일 때 수 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount
PhysicalDiskUsage	파일 시스템 데이터(압축됨)가 물리적으로 차지하는 스토리지의 양. Sum 통계는 파일 시스템에서 디스크가 차지하는 총 바이트 수입니다. Minimum 통계는 가장 비어 있는 디스크에서 차지하는 총 바이트 수입니다. Maximum 통계는 가장 꽉 찬 디스크에서 차지하는 총 바이트 수입니다. Average 통계는 디스크당 점유한 평균 바이트 수입니다. SampleCount 통계는 디스크 수입니다. 단위: • Sum, Minimum, Maximum의 바이트 • SampleCount 통계일 때 수 유효한 통계: Sum, Minimum, Maximum, Average, SampleCount

파일 시스템 메타데이터 메트릭

FSx for Lustre는 다음 파일 시스템 메타데이터 메트릭을 의 네임스페이스에 게시합니다. FSx CloudWatch 이러한 지표는 차원을 사용하여 메타데이터 데이터를 더 세밀하게 측정할 수 있습니다.

모든 메타데이터 지표에는 FileSystemId 및 StorageTargetId 측정기준이 있습니다. 파일 시스템 메타데이터 지표는 파일 시스템에 메타데이터 구성이 지정된 경우에만 노출됩니다.

지표	설명
DiskReadOperations	스토리지 볼륨에 액세스하는 파일 서버의 읽기 작업 수입니다. 이 지표에는 백그라운드 작업을 포함한 모든 트래픽이 고려됩니다. 각 파일 시스템의 스토리지 볼륨에 대해 1분마다 하나의 지표가 내보내집니다. Sum통계는 지정된 기간 동안 지정된 스토리지 볼륨에서 수행한 총 읽기 작업 수입니다. Average통계는 지정된 기간 동안 지정된 스토리지 볼륨에서 분당 수행한 평균 읽기 작업 수입니다. Minimum통계는 지정된 기간 동안 지정된 스토리지 볼륨에서 분당 수행한 최소 읽기 작업 수입니다. Maximum통계는 지정된 기간 동안 지정된 스토리지 볼륨에서 분당 수행한 최대 읽기 작업 수입니다. 해당 기간의 평균 메타데이터 디스크 IOPS를 계산하려면 Average 통계를 사용하고 결과를 60 (초) 으로 나눕니다. 단위: 개 유효한 통계:Sum,Average, 및 Minimum Maximum
DiskWriteOperations	스토리지 볼륨에 액세스하는 파일 서버의 쓰기 작업 수입니다. 이 스토리지 볼륨에 대한 쓰기 작업 수입니다. 이 지표에는 백그라운드 작업을 포함한 모든

지표	설명
	트래픽이 고려됩니다. 각 파일 시스템의 스토리지 볼륨에 대해 1분마다 하나의 지표가 내보내집니다. Sum통계는 지정된 기간 동안 지정된 스토리지 볼륨에서 수행한 총 쓰기 작업 수입니다. Average통계는 지정된 기간 동안 지정된 스토리지 볼륨에서 분당 수행한 평균 쓰기 작업 수입니다. 해당 기간의 평균 메타데이터 디스크 IOPS를 계산하려면 Average 통계를 사용하고 결과를 60 (초) 으로 나눕니다. 단위: 개 유효한 통계: 및 Sum Average
FileCreateOperations	총 파일 생성 작업 수입니다. 단위: 수
FileOpenOperations	총 파일 열기 작업 수입니다. 단위: 수
FileDeleteOperations	총 파일 삭제 작업 수입니다. 단위: 수
StatOperations	총 통계 작업 수입니다. 단위: 수

지표	설명
RenameOperations	전체 디렉터리 이름 변경 또는 디렉터리 간 이름 변경 여부에 관계없이 총 디렉터리 이름 변경 수입니다. 단위: 수

AutoImport AutoExport 및 지표

FSx for Lustre는 다음 (자동 가져오기AutoExport) 및 (자동 내보내기) AutoImport 메트릭을 의 네임스페이스에 게시합니다. FSx CloudWatch 이러한 지표는 측정기준을 사용하여 데이터를 더 세밀하게 측정할 수 있습니다. 모든 AutoImport 및 AutoExport 지표는 FileSystemId 및 Publisher 측정기준을 가집니다.

지표	설명
AgeOfOldestQueuedMessage 차원: AutoExport	내보내기 대기 중인 가장 오래된 메시지의 보존 기간(초). Average 통계는 내보내기 대기 중인 가장 오래된 메시지의 평균 보존 기간입니다. Maximum 통계는 메시지가 내보내기 대기열에 남아 있는 최대 시간(초)입니다. Minimum 통계는 메시지가 내보내기 대기열에 남아 있는 최소 시간(초)입니다. 값이 0이면 내보내기를 기다리는 메시지가 없음을 나타냅니다. 단위: 초 유효한 통계: Average, Minimum, Maximum
RepositoryRenameOperations 측정기준: AutoExport	대규모 디렉터리 이름 변경에 대한 응답으로 파일 시스템에서 처리한 이름 변경 횟수입니다. Sum 통계는 디렉터리 이름 변경으로 인한 이름 바꾸기 작업의 총 수입니다. Average 통

지표	설명
	계는 파일 시스템의 평균 이름 바꾸기 작업 수입니다. Maximum 통계는 파일 시스템의 디렉터리 이름 바꾸기와 관련된 이름 바꾸기 작업의 최대 수입니다. Minimum 통계는 파일 시스템의 디렉터리 이름 변경과 관련된 최소 이름 변경 횟수입니다. 단위: 개 유효한 통계: Sum, Minimum, Maximum, Average
AgeOfOldestQueuedMessage 측정기준: AutoImport	가져오기 대기 중인 가장 오래된 메시지의 보존 기간(초). Average 통계는 가져오기를 기다리는 가장 오래된 메시지의 평균 보존 기간입니다. Maximum 통계는 메시지가 가져오기 대기열에 남아 있는 최대 시간(초)입니다. Minimum 통계는 메시지가 가져오기 대기열에 남아 있는 최소 시간(초)입니다. 값이 0이면 가져오기를 기다리는 메시지가 없음을 나타냅니다. 단위: 초 유효한 통계: Average, Minimum, Maximum

Amazon FSx for Lustre 측정기준

Amazon FSx for Lustre 지표는 FSx 네임스페이스를 사용하며 FileSystemId 측정기준의 지표를 제공합니다. describe-file-systems AWS CLI *### #### ## #### ID# ## # ###, # ID# fs-01234567890123456# ### ####.*

StorageTargetId차원은 파일 시스템 메타데이터 메트릭을 게시한 MDT (메타데이터 대상) CloudWatch 를 나타내는 데 사용할 수 있습니다. A는 MDTxxxx (예:) 의 형식을 StorageTargetId 취합니다. MDT0001

Publisher차원은 에서 사용할 수 CloudWatch 있으며, 및 AWS CLI 지표에는 AutoImport AutoImport 지표를 게시한 서비스를 나타내는 데 사용할 수 있습니다.

Amazon FSx for Lustre 지표 사용 방법

Amazon FSx for Lustre에서 보고하는 지표는 다양한 방법으로 분석이 기능한 정보를 제공합니다. 다음은 몇 가지 일반적인 지표 사용 사례입니다. 모든 사용 사례를 망라한 것은 아니지만 시작하는 데 참고가 될 것입니다.

결정하는 방법	관련 지표(측정기준 | 지표)
내 파일 시스템의 처리량은?	합계 (DataReadBytes + DataWriteBytes) /기간 (초)
내 파일 시스템의 IOPS는?	총 IOPS = 합계 (DataReadOperations DataWriteOperations + MetadataOperations) /기간 (초)
내 파일 시스템의 데이터 압축률은?	합계 (LogicalDisk사용량)/합계 (PhysicalDisk사용량)
파일 시스템 업데이트가 S3 버킷과 동기화된 경우 어떻게 되나요?	AutoExport | AgeOfOldestQueuedMessage
S3 버킷 업데이트가 파일 시스템과 동기화된 경우 어떻게 되나요?	AutoImport | AgeOfOldestQueuedMessage

지표 수학: 데이터 압축률

메트릭 수학을 사용하면 여러 CloudWatch 메트릭을 쿼리하고 수학 식을 사용하여 이러한 메트릭을 기반으로 새 시계열을 만들 수 있습니다. CloudWatch 콘솔에서 결과 시계열을 시각화하고 대시보드에 추가할 수 있습니다. 지표 수학에 대한 자세한 내용은 Amazon 사용 CloudWatch 설명서의 지표 수학 사용을 참조하십시오.

이 지표 수학식은 Lustre Amazon FSx for Lustre 파일 시스템의 데이터 압축 비율을 계산합니다. 이 비율을 계산하려면 먼저 LogicalDiskUsage 지표에서 제공하는 총 논리적 디스크 사용량(압축 제외)의 합계 통계를 구합니다. 그런 다음 이 수치를 PhysicalDiskUsage 지표에서 제공한 총 물리적 디스크 사용량(압축 포함)의 합계 통계로 나눕니다.

따라서 해당 논리는 다음과 같습니다. 합계 `LogicalDiskUsage` ÷ 합계 `PhysicalDiskUsage`

그러면 CloudWatch 측정치 정보는 다음과 같습니다.

ID	사용 가능한 지표	통계	기간
m1	`LogicalDiskUsage`	Sum	1분
m2	`PhysicalDiskUsage`	Sum	1분

지표 수식 ID와 표현식은 다음과 같습니다.

ID	표현식
e1	`m1/m2`

`e1`은 데이터 압축률입니다.

CloudWatch 지표 액세스

Amazon FSx for CloudWatch Lustre 메트릭은 여러 가지 방법으로 확인할 수 있습니다. CloudWatch콘솔을 통해 보거나 CloudWatch CLI 또는 API를 사용하여 액세스할 수 있습니다. CloudWatch 다음의 절차는 다양한 도구를 사용하여 지표에 액세스하는 방법을 설명합니다.

콘솔을 CloudWatch 사용하여 지표를 보려면

1. CloudWatch 콘솔을 엽니다.
2. 탐색 창에서 지표를 선택합니다.
3. FSx 네임스페이스를 선택합니다.
4. (선택 사항) 지표를 보려면 검색 필드에 이름을 입력합니다.
5. (선택 사항) 측정기준별로 필터링하려면 FileSystemId를 선택합니다.

에서 지표에 액세스하려면 AWS CLI

- `list-metrics` 명령과 `--namespace "AWS/FSx"` 네임스페이스를 사용합니다. 자세한 내용은 AWS CLI 명령 참조를 참조하세요.

CloudWatch API에서 지표에 액세스하려면

- `GetMetricStatistics`을 호출합니다. 자세한 내용은 Amazon CloudWatch API 레퍼런스를 참조하십시오.

Amazon FSx for Lustre를 모니터링하기 위한 CloudWatch 경보 생성

CloudWatch 경보 상태가 변경될 때 Amazon SNS 메시지를 보내는 경보를 생성할 수 있습니다. 경보는 지정한 기간에 단일 지표를 감시하고 여러 기간에 지정된 임계값에 대한 지표 값을 기준으로 작업을 하나 이상 수행합니다. 이 작업은 Amazon SNS 주제 또는 Auto Scaling 정책으로 전송되는 알림입니다.

경보는 지속적인 상태 변경에 대한 조치만 호출합니다. CloudWatch 경보는 단순히 특정 상태에 있다는 이유만으로 작업을 호출하지 않습니다. 상태가 변경되고 지정된 기간 동안 유지되어야 합니다.

다음 절차에서는 Amazon FSx for Lustre에 대한 경보를 만드는 방법을 간략하게 설명합니다.

콘솔을 사용하여 경보를 설정하려면 CloudWatch

1. https://console.aws.amazon.com/cloudwatch/ 에서 AWS Management Console 로그인하고 CloudWatch 콘솔을 엽니다.
2. 경보 생성을 선택합니다. 그러면 경보 생성 마법사가 시작됩니다.
3. FSx 지표를 선택하고 Amazon FSx for Lustre 지표를 스크롤하여 경보를 생성할 지표를 찾습니다. 이 대화 상자에서 Amazon FSx for Lustre 지표만 표시하려면 파일 시스템의 파일 시스템 ID를 검색합니다. 지표를 선택하여 경보를 생성한 다음 다음을 선택합니다.
4. 조건 섹션에서 경보에 적용할 조건을 선택한 후 다음을 선택합니다.

> **Note**
>
> 파일 시스템 유지 관리 중에는 지표가 게시되지 않을 수 있습니다. 불필요하고 오해의 소지가 있는 경보 조건 변경을 방지하고 누락된 데이터 포인트에 대해 복원력을 갖도록 경보

를 구성하려면 Amazon User Guide의 CloudWatch 경보가 누락된 데이터를 처리하는 방법 구성을 참조하십시오. CloudWatch

5. 경보 상태에 도달했을 때 이메일을 CloudWatch 보내려면 이 경보가 발생할 때마다 [State is ALARM] 을 선택하십시오. 다음 주소로 알림 전송에서 기존 SNS 주제를 선택합니다. 주제 생성을 선택한 경우 새 이메일 구독 목록에 대한 이름 및 이메일 주소를 설정할 수 있습니다. 이 목록은 향후 경보를 위해 박스에 저장되고 표시됩니다.

> **Note**
>
> 새 Amazon SNS 주제를 생성하기 위해 주제 생성을 사용할 경우 이메일 주소는 알림을 보내기 전에 검증합니다. 이메일은 경보가 경보 상태에 입력될 때만 전송됩니다. 이러한 경보 상태 변경이 이메일이 검증되기 전에 발생할 경우에는 알림을 받지 못합니다.

6. 경보 미리 보기 영역에서 생성할 경보를 미리 확인할 수 있습니다. 예상대로 표시되면 경보 생성을 선택합니다.

를 사용하여 알람을 설정하려면 AWS CLI

- `put-metric-alarm`을 호출합니다. 자세한 내용은 AWS CLI 명령 참조를 참조하세요.

CloudWatch API를 사용하여 알람을 설정하려면

- `PutMetricAlarm`을 호출합니다. 자세한 내용은 Amazon CloudWatch API 레퍼런스를 참조하십시오.

Amazon CloudWatch 로그로 로깅

FSx for Lustre는 파일 시스템과 연결된 데이터 리포지토리의 오류 및 경고 이벤트를 Amazon Logs에 기록하는 것을 지원합니다. CloudWatch

> **Note**
>
> Amazon Logs를 사용한 CloudWatch 로깅은 2021년 11월 30일 오후 3시 (태평양 표준시) 이후에 생성된 Amazon FSx for Lustre 파일 시스템에서만 사용할 수 있습니다.

주제

- 로깅 개요
- 로그 대상
- 로깅 관리
- 로그 보기

로깅 개요

FSx for Lustre 파일 시스템에 연결된 데이터 리포지토리가 있는 경우 Amazon Logs에 데이터 리포지토리 이벤트 로깅을 활성화할 수 있습니다. CloudWatch 다음 데이터 리포지토리 작업에서 오류 및 경고 이벤트를 기록할 수 있습니다.

- 자동 내보내기
- 데이터 리포지토리 작업

이러한 작업 및 데이터 리포지토리 연결에 대한 자세한 내용은 Amazon FSx for Lustre에서 데이터 리포지토리 사용 섹션을 참조하세요.

Amazon FSx가 기록하는 로그 수준을 구성할 수 있습니다. 즉, Amazon FSx가 오류 이벤트만 기록할지, 경고 이벤트만 기록할지 또는 오류 및 경고 이벤트를 모두 기록할지 여부를 구성할 수 있습니다. 또한 언제든지 이벤트 로그아웃을 해제할 수도 있습니다.

> **Note**
>
> 어떤 수준이든 중요 기능이 관련된 파일 시스템에 대해서는 파일 시스템 로그를 활성화하는 것이 좋습니다.

로그 대상

로깅이 활성화된 경우 FSx for Lustre를 Amazon Logs 대상으로 구성해야 합니다. CloudWatch 이벤트 로그 대상은 Amazon CloudWatch Logs 로그 그룹이며, Amazon FSx는 이 로그 그룹 내에 파일 시스템에 대한 로그 스트림을 생성합니다. CloudWatch 로그를 사용하면 Amazon CloudWatch 콘솔에서 감사 이벤트 로그를 저장, 확인 및 검색하고, Logs Insights를 사용하여 CloudWatch 로그에 대한 쿼리를 실행하고, CloudWatch 경보 또는 Lambda 함수를 트리거할 수 있습니다.

FSx for Lustre 파일 시스템을 생성할 때 또는 나중에 업데이트할 때 로그 대상을 선택합니다. 자세한 정보는 로깅 관리을 참조하세요.

기본적으로 Amazon FSx는 사용자 계정의 CloudWatch 기본 로그 로그 그룹을 생성하여 이벤트 로그 대상으로 사용합니다. 사용자 지정 로그 로그 그룹을 이벤트 CloudWatch 로그 대상으로 사용하려는 경우 이벤트 로그 대상의 이름 및 위치에 대한 요구 사항은 다음과 같습니다.

- 로그 CloudWatch 로그 그룹 이름은 `/aws/fsx/` 접두사로 시작해야 합니다.
- 콘솔에서 파일 시스템을 생성하거나 업데이트할 때 기존 CloudWatch 로그 로그 그룹이 없는 경우 Amazon FSx for Lustre는 로그 로그 그룹에 기본 로그 스트림을 생성하고 사용할 수 있습니다. CloudWatch `/aws/fsx/lustre` 로그 스트림은 `datarepo_`*`file_system_id`* 형식(예: `datarepo_fs-0123456789abcdef0`)으로 생성됩니다.
- 기본 로그 그룹을 사용하지 않으려는 경우 콘솔에서 파일 시스템을 생성하거나 업데이트할 때 구성 UI를 사용하여 CloudWatch 로그 로그 그룹을 생성할 수 있습니다.
- 대상 로그 CloudWatch 로그 그룹은 Amazon FSx for Lustre 파일 AWS 리전시스템과 동일한 AWS 파티션에 있어야 하고 AWS 계정 동일한 파티션에 있어야 합니다.

이벤트 로그 대상은 언제든지 변경할 수 있습니다. 이렇게 하면 새 이벤트 로그가 새 대상으로만 전송됩니다.

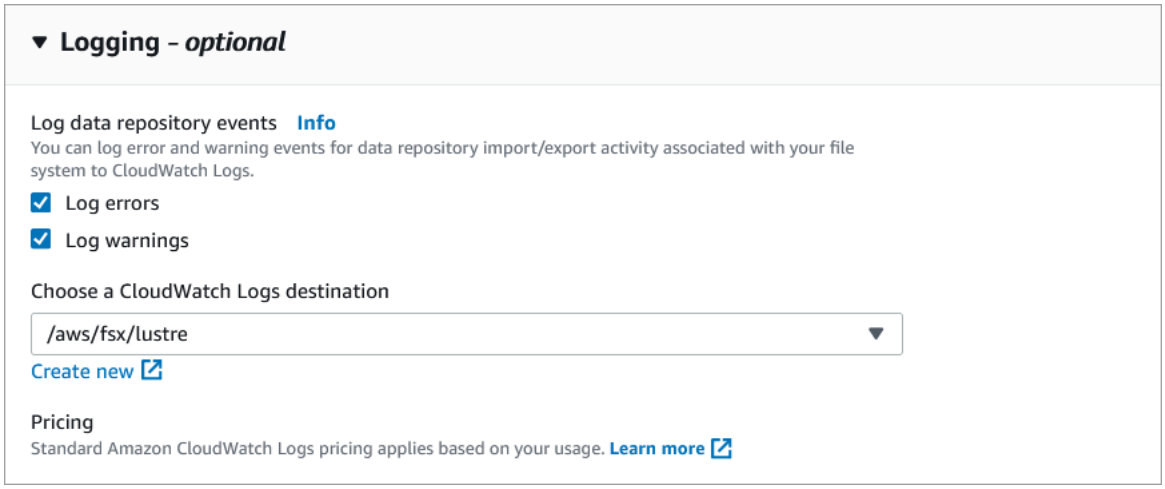
로깅 관리

새 FSx for Lustre 파일 시스템을 생성할 때 또는 나중에 업데이트하여 로깅을 활성화할 수 있습니다. Amazon FSx 콘솔에서 파일 시스템을 생성할 때 기본적으로 로깅이 활성화됩니다. 하지만 Amazon FSx API를 사용하여 파일 시스템을 생성할 때는 기본적으로 로깅이 AWS CLI 해제됩니다.

로깅이 활성화된 기존 파일 시스템에서는 이벤트를 기록할 로그 수준 및 로그 대상을 비롯한 이벤트 로깅 설정을 변경할 수 있습니다. Amazon FSx 콘솔 AWS CLI또는 Amazon FSx API를 사용하여 이러한 작업을 수행할 수 있습니다.

파일 시스템을 생성할 때 로깅 활성화(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 시작하기 섹션의 FSx for Lustre 파일 시스템 만들기 섹션에 설명된 새 파일 시스템 생성 절차를 따릅니다.
3. 로깅 - 선택 섹션을 엽니다. 로깅은 기본적으로 활성화됩니다.



4. 파일 시스템 생성 마법사의 다음 섹션으로 계속 진행합니다.

파일 시스템이 사용 가능해지면 로깅이 활성화됩니다.

파일 시스템을 생성할 때 로깅 활성화(CLI)

1. 새 파일 시스템을 생성할 때 시스템 작업과 함께 `LogConfiguration` 속성을 사용하여 새 파일 CreateFile시스템에 대한 로깅을 활성화하십시오.

```
create-file-system --file-system-type LUSTRE \
    --storage-capacity 1200 --subnet-id subnet-08b31917a72b548a9  \
    --lustre-configuration "LogConfiguration={Level=WARN_ERROR, \
        Destination="arn:aws:logs:us-east-1:234567890123:log-group:/aws/fsx/
testEventLogging"}"
```

2. 파일 시스템이 사용 가능 상태가 되면 로깅 기능이 활성화됩니다.

로깅 구성 변경(콘솔)

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 파일 시스템으로 이동하여 로깅을 관리할 Lustre 파일 시스템을 선택합니다.
3. 모니터링 탭을 선택합니다.
4. 로깅 패널에서 업데이트를 선택합니다.
5. 로깅 구성 업데이트 대화 상자에서 원하는 설정을 변경합니다.

   a. 오류 이벤트만 기록하려면 오류 기록을 선택하고 경고 이벤트만 기록하려면 경고 기록을 선택하거나 또는 둘 다를 선택합니다. 선택하지 않으면 로깅이 비활성화됩니다.

b. 기존 로그 CloudWatch 로그 대상을 선택하거나 새 로그 대상을 생성합니다.

6. 저장을 선택합니다.

로깅 구성 변경(CLI)

- `update-file-system` CLI 명령 또는 이에 상응하는 `UpdateFileSystem` API 작업을 사용합니다.

```
update-file-system --file-system-id fs-0123456789abcdef0 \
    --lustre-configuration "LogConfiguration={Level=WARN_ERROR, \
        Destination="arn:aws:logs:us-east-1:234567890123:log-group:/aws/fsx/
testEventLogging"}"
```

로그 보기

Amazon FSx에서 로그를 생성하기 시작한 후에 감사 이벤트 로그를 볼 수 있습니다. 다음 로그를 볼 수도 있습니다.

- Amazon CloudWatch 콘솔로 이동하여 이벤트 로그를 전송할 로그 그룹과 로그 스트림을 선택하여 로그를 볼 수 있습니다. 자세한 내용은 Amazon Logs 사용 설명서의 CloudWatch Logs로 전송된 CloudWatch 로그 데이터 보기를 참조하십시오.
- CloudWatch Logs Insights를 사용하여 대화형 방식으로 로그 데이터를 검색하고 분석할 수 있습니다. 자세한 내용은 Amazon Logs 사용 설명서의 CloudWatch Logs Insights를 사용한 CloudWatch 로그 데이터 분석을 참조하십시오.
- 또한 로그를 Amazon S3로 내보낼 수 있습니다. 자세한 내용은 Amazon Logs 사용 설명서의 Amazon S3로 CloudWatch 로그 데이터 내보내기를 참조하십시오.

실패 사유에 대한 자세한 내용은 데이터 리포지토리 이벤트 로그 섹션을 참조하세요.

를 사용하여 FSx for Lustre API 호출을 로깅합니다. AWS CloudTrail

Amazon FSx for Lustre는 Amazon FSx for Lustre에서 사용자, AWS 역할 또는 서비스가 수행한 작업에 대한 기록을 제공하는 서비스와 AWS CloudTrail통합되어 있습니다. CloudTrail Amazon FSx for Lustre에 대한 모든 API 호출을 이벤트로 캡처합니다. 캡처되는 호출에는 Amazon FSx for Lustre 콘솔로부터의 호출과 Amazon FSx for Lustre API 작업에 대한 코드 호출이 포함됩니다.

트레일을 생성하면 Amazon FSx for Lustre에 대한 CloudTrail 이벤트를 포함하여 Amazon S3 버킷으로 이벤트를 지속적으로 전송할 수 있습니다. 트레일을 구성하지 않아도 CloudTrail 콘솔의 이벤트 기록에서 가장 최근 이벤트를 계속 볼 수 있습니다. 에서 수집한 CloudTrail 정보를 사용하여 Amazon FSx for Lustre에 이루어진 요청을 확인할 수 있습니다. 또한 어떤 IP 주소에서 요청했는지, 누가 언제 요청했는지 등의 추가 세부 정보도 확인할 수 있습니다.

자세한 CloudTrail 내용은 사용 설명서를 참조하십시오.AWS CloudTrail

Amazon FSx for Lustre 정보는 다음에서 확인할 수 있습니다. CloudTrail

CloudTrail AWS 계정을 생성하면 계정에서 활성화됩니다. Amazon FSx for Lustre에서 API 활동이 발생하면 해당 활동이 이벤트 CloudTrail 기록의 다른 서비스 이벤트와 함께 AWS 이벤트에 기록됩니다. 계정에서 최근 이벤트를 보고, 검색하고, 다운로드할 수 있습니다. AWS 자세한 내용은 이벤트 기록으로 CloudTrail 이벤트 보기를 참조하십시오.

Amazon FSx for Lustre의 이벤트를 포함하여 AWS 계정에서 진행 중인 이벤트 기록을 보려면 트레일을 생성하십시오. 트레일을 사용하면 CloudTrail Amazon S3 버킷에 로그 파일을 전송할 수 있습니다. 기본적으로 콘솔에서 트레일을 생성하면 트레일이 모든 AWS 지역에 적용됩니다. 트레일은 AWS 파티션에 있는 모든 AWS 지역의 이벤트를 기록하고 지정한 Amazon S3 버킷으로 로그 파일을 전송합니다. 또한 CloudTrail 로그에서 수집된 이벤트 데이터를 추가로 분석하고 이에 따라 조치를 취하도록 다른 AWS 서비스를 구성할 수 있습니다. 자세한 내용은AWS CloudTrail 사용 설명서에서 다음 주제를 참조하세요.

- 추적 생성 개요
- CloudTrail 지원되는 서비스 및 통합
- 에 대한 Amazon SNS 알림 구성 CloudTrail
- 여러 지역에서 CloudTrail 로그 파일 수신 및 여러 계정으로부터 CloudTrail 로그 파일 수신

모든 Amazon FSx for Lustre API 호출은 에 의해 기록됩니다. CloudTrail 예를 들어, `CreateFileSystem` 및 `TagResource` 작업에 대한 호출은 로그 파일에 항목을 생성합니다. CloudTrail

모든 이벤트 및 로그 항목에는 요청을 생성한 사용자에 대한 정보가 들어 있습니다. 신원 정보를 이용하면 다음을 쉽게 알아볼 수 있습니다.

- 요청이 루트 또는 AWS Identity and Access Management (IAM) 사용자 자격 증명으로 이루어졌는지 여부.
- 역할 또는 연동 사용자를 위한 임시 보안 인증으로 요청을 생성하였는지.

- 다른 AWS 서비스에서 요청했는지 여부.

자세한 내용은 CloudTrail 사용자 안내서의 UserIdentity 요소를 참조하십시오.AWS CloudTrail

Amazon FSx for Lustre 로그 파일 항목 이해

트레일은 지정한 Amazon S3 버킷에 이벤트를 로그 파일로 전송할 수 있는 구성입니다. CloudTrail 로그 파일에는 하나 이상의 로그 항목이 포함되어 있습니다. 이벤트는 모든 소스의 단일 요청을 나타내며 요청된 작업, 작업 날짜 및 시간, 요청 매개 변수 등에 대한 정보를 포함합니다. CloudTrail 로그 파일은 공개 API 호출의 정렬된 스택 트레이스가 아니므로 특정 순서로 표시되지 않습니다.

다음 예제는 콘솔에서 파일 시스템용 태그를 생성할 때의 `TagResource` 작업을 보여주는 CloudTrail 로그 항목을 보여줍니다.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": “Root”,
        "principalId": “111122223333”,
        "arn": "arn:aws:sts::111122223333:root”,
        "accountId": “111122223333”,
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE”,
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-11-14T22:36:07Z"
            }
        }
    },
    "eventTime": "2018-11-14T22:36:07Z",
    "eventSource": "fsx.amazonaws.com",
    "eventName": "TagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": “192.0.2.0”,
    "userAgent": “console.amazonaws.com”,
    "requestParameters": {
        "resourceARN": "arn:aws:fsx:us-east-1:111122223333:file-system/fs-
ab12cd34ef56gh789”
    },
    "responseElements": null,
    "requestID": “aEXAMPLE-abcd-1234-56ef-b4cEXAMPLE51”,
    "eventID": “bEXAMPLE-gl12-3f5h-3sh4-ab6EXAMPLE9p”,
```

```
    "eventType": "AwsApiCall",
    "apiVersion": "2018-03-01",
    "recipientAccountId": “111122223333”
}
```

다음 예제는 콘솔에서 파일 시스템의 태그가 삭제될 때의 UntagResource 작업을 보여주는 CloudTrail 로그 항목을 보여줍니다.

```
{
    "eventVersion": "1.05",
    "userIdentity": {
        "type": “Root”,
        "principalId": "111122223333",
        "arn": "arn:aws:sts::111122223333:root",
        "accountId": "111122223333",
        "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
        "sessionContext": {
            "attributes": {
                "mfaAuthenticated": "false",
                "creationDate": "2018-11-14T23:40:54Z"
            }
        }
    },
    "eventTime": "2018-11-14T23:40:54Z",
    "eventSource": "fsx.amazonaws.com",
    "eventName": "UntagResource",
    "awsRegion": "us-east-1",
    "sourceIPAddress": "192.0.2.0",
    "userAgent": "console.amazonaws.com",
    "requestParameters": {
        "resourceARN": "arn:aws:fsx:us-east-1:111122223333:file-system/fs-
ab12cd34ef56gh789"
    },
    "responseElements": null,
    "requestID": "aEXAMPLE-abcd-1234-56ef-b4cEXAMPLE51",
    "eventID": "bEXAMPLE-gl12-3f5h-3sh4-ab6EXAMPLE9p",
    "eventType": "AwsApiCall",
    "apiVersion": "2018-03-01",
    "recipientAccountId": "111122223333"
}
```

FSx for Lustre 보안

클라우드 AWS 보안이 최우선 과제입니다. AWS 고객은 가장 보안에 민감한 조직의 요구 사항을 충족하도록 구축된 데이터 센터 및 네트워크 아키텍처를 활용할 수 있습니다.

보안은 기업과 기업 간의 공동 책임입니다. AWS 공동 책임 모델은 이 사항을 클라우드 내 보안 및 클라우드의 보안으로 설명합니다.

- 클라우드 보안 — AWS Amazon Web AWS Services 클라우드에서 서비스를 실행하는 인프라를 보호하는 역할을 합니다. AWS 또한 안전하게 사용할 수 있는 서비스를 제공합니다. 서드 파티 감사자는 정기적으로 AWS 규정 준수 프로그램의 일환으로 보안 효과를 테스트하고 검증합니다. Amazon FSx for Lustre에 적용되는 규정 준수 프로그램에 대한 자세한 내용은 AWS 규정 준수 프로그램 제공 범위 내 서비스를 참조하세요.
- 클라우드에서의 보안 — 귀하의 책임은 사용하는 AWS 서비스에 따라 결정됩니다. 또한 귀하는 귀사의 데이터의 민감도, 귀사의 요구 사항, 관련 법률 및 규정을 비롯한 기타 요소에 대해서도 책임이 있습니다.

이 설명서는 Amazon FSx for Lustre를 사용할 때 공동 책임 모델을 적용하는 방법을 이해하는 데 도움이 됩니다. 다음 주제에서는 보안 및 규정 준수 목적에 맞게 Amazon FSx를 구성하는 방법을 보여줍니다. 또한 Amazon FSx for Lustre 리소스를 모니터링하고 보호하는 데 도움이 되는 다른 Amazon 서비스를 사용하는 방법을 배우게 됩니다.

아래에서 Amazon FSx 작업을 위한 보안 고려 사항에 대한 설명을 확인할 수 있습니다.

주제

- Amazon FSx for Lustre 데이터 보호
- Amazon FSx for Lustre용 ID 및 액세스 관리
- Amazon VPC를 사용한 파일 시스템 액세스 제어
- Amazon VPC 네트워크 ACL
- Amazon FSx for Lustre에 대한 규정 준수 확인
- Amazon FSx for Lustre 및 인터페이스 VPC 엔드포인트(AWS PrivateLink)

Amazon FSx for Lustre 데이터 보호

AWS 공동 책임 모델 Amazon FSx for Lustre의 데이터 보호에 적용됩니다. 이 모델에 설명된 대로 AWS 은 모든 모델을 실행하는 글로벌 인프라를 보호하는 역할을 합니다. AWS 클라우드사용자는 인프라에서 호스팅되는 콘텐츠를 관리해야 합니다. 사용하는 AWS 서비스 의 보안 구성과 관리 작업에 대한 책임도 사용자에게 있습니다. 데이터 프라이버시에 대한 자세한 내용은 데이터 프라이버시 FAQ를 참조하세요. 유럽의 데이터 보호에 대한 자세한 내용은 AWS 보안 블로그의 AWS 공동 책임 모델 및 GDPR 블로그 게시물을 참조하세요.

데이터 보호를 위해 AWS 계정 자격 증명을 보호하고 AWS IAM Identity Center OR AWS Identity and Access Management (IAM) 을 사용하여 개별 사용자를 설정하는 것이 좋습니다. 이렇게 하면 개별 사용자에게 자신의 직무를 충실히 이행하는 데 필요한 권한만 부여됩니다. 또한 다음과 같은 방법으로 데이터를 보호하는 것이 좋습니다.

- 각 계정에 멀티 팩터 인증 설정(MFA)을 사용하세요.
- SSL/TLS를 사용하여 리소스와 통신하세요. AWS TLS 1.2는 필수이며 TLS 1.3를 권장합니다.
- 를 사용하여 API 및 사용자 활동 로깅을 설정합니다. AWS CloudTrail
- 포함된 모든 기본 보안 제어와 함께 AWS 암호화 솔루션을 사용하십시오 AWS 서비스.
- Amazon S3에 저장된 민감한 데이터를 검색하고 보호하는 데 도움이 되는 Amazon Macie와 같은 고급 관리형 보안 서비스를 사용하세요.
- 명령줄 인터페이스 또는 API를 AWS 통해 액세스할 때 FIPS 140-2로 검증된 암호화 모듈이 필요한 경우 FIPS 엔드포인트를 사용하십시오. 사용 가능한 FIPS 엔드포인트에 대한 자세한 내용은 FIPS(Federal Information Processing Standard) 140-2를 참조하세요.

고객의 이메일 주소와 같은 기밀 정보나 중요한 정보는 태그나 이름 필드와 같은 자유 양식 필드에 입력하지 않는 것이 좋습니다. 여기에는 Amazon FSx 또는 AWS 서비스 기타 콘솔 AWS CLI, API 또는 SDK를 사용하여 작업하는 경우가 포함됩니다. AWS 이름에 사용되는 태그 또는 자유 형식 텍스트 필드에 입력하는 모든 데이터는 청구 또는 진단 로그에 사용될 수 있습니다. 외부 서버에 URL을 제공할 때 해당 서버에 대한 요청을 검증하기 위해 보안 인증 정보를 URL에 포함시켜서는 안 됩니다.

주제

- Amazon FSx for Lustre 의 데이터 암호화
- 인터네트워크 트래픽 개인 정보

Amazon FSx for Lustre 의 데이터 암호화

Amazon FSx for Lustre는 저장 데이터 암호화와 전송 중 데이터 암호화라는 두 가지 파일 시스템 암호화를 지원합니다. Amazon FSx 파일 시스템을 생성할 때 저장 데이터 암호화가 자동으로 활성화됩니다. 전송 중 데이터 암호화는 이 기능을 지원하는 Amazon EC2 인스턴스에서 또는 Amazon FSx 파일 시스템에 액세스할 때 자동으로 활성화됩니다.

암호화를 사용해야 하는 경우

조직이 저장 상태의 데이터 및 메타데이터 암호화를 요구하는 기업 정책이나 규제 정책을 준수해야 하는 경우 전송 중 데이터 암호화를 사용하여 파일 시스템을 마운트하는 암호화된 파일 시스템을 생성하는 것이 좋습니다.

콘솔을 사용하여 유휴 상태에서 암호화된 파일 시스템을 생성하는 방법에 대한 자세한 내용은 Amazon FSx for Lustre 파일 시스템 생성을 참조하세요.

주제

- 저장 데이터 암호화
- 전송 중 데이터 암호화

저장 데이터 암호화

저장 데이터의 암호화는 Amazon FSx API 또는 SDK 중 하나를 AWS Management Console통해, 또는 프로그래밍 방식으로 Amazon FSx for Lustre 파일 시스템을 생성할 때 자동으로 활성화됩니다. AWS CLI AWS 조직에서 특정 분류를 충족하거나 특정 애플리케이션이나 워크로드, 환경과 연결된 모든 데이터를 암호화해야 할 수 있습니다. 영구 파일 시스템을 생성하는 경우 데이터를 암호화하는 데 사용할 키를 지정할 수 있습니다. AWS KMS 스크래치 파일 시스템을 생성하는 경우 Amazon FSx에서 관리하는 키를 사용하여 데이터가 암호화됩니다. 콘솔을 사용하여 유휴 상태에서 암호화된 파일 시스템을 생성하는 방법에 대한 자세한 내용은 Amazon FSx for Lustre 파일 시스템 생성을 참조하세요.

> **Note**
>
> AWS 키 관리 인프라는 연방 정보 처리 표준 (FIPS) 140-2 승인 암호화 알고리즘을 사용합니다. 이 인프라는 미국 국립 표준 기술 연구소(NIST) 800-57 표준의 권장 사항에 부합됩니다.

FSx for Lustre의 사용 방법에 대한 자세한 내용은 을 참조하십시오. AWS KMSAmazon FSx for Lustre가 사용하는 방법 AWS KMS

유휴 암호화 작동 방식

암호화가 적용된 파일 시스템의 경우, 데이터와 메타데이터가 자동으로 암호화된 후 파일 시스템에 기록됩니다. 유사하게 데이터와 메타데이터를 읽을 때, 자동으로 암호화를 해제한 후 애플리케이션으로 전달됩니다. Amazon FSx for Lustre는 해당 프로세스를 투명하게 처리하기 때문에 애플리케이션을 수정할 필요가 없습니다.

Amazon FSx for Lustre는 업계 표준 AES-256 암호화 알고리즘을 사용하여 저장된 파일 시스템 데이터를 암호화합니다. 자세한 내용은 AWS Key Management Service 개발자 안내서의 암호화 기본 사항을 참조하세요.

Amazon FSx for Lustre가 사용하는 방법 AWS KMS

Amazon FSx for Lustre는 파일 시스템에 데이터를 쓰기 전에 데이터를 자동으로 암호화하고 데이터를 읽을 때 자동으로 복호화합니다. 데이터는 XTS-AES-256 블록 암호를 사용하여 암호화됩니다. 모든 스크래치 FSx for Lustre 파일 시스템은 유휴 상태에서 암호화되며 에서 관리하는 키를 사용합니다. AWS KMS Amazon FSx for AWS KMS Lustre는 키 관리를 위해 와 통합됩니다. 저장된 스크래치 파일 시스템을 암호화하는 데 사용되는 키는 파일 시스템별로 고유하며 파일 시스템이 삭제된 후에는 삭제됩니다. 영구 파일 시스템의 경우 데이터를 암호화하고 해독하는 데 사용되는 KMS 키를 선택합니다. 영구 파일 시스템을 생성할 때 사용할 키를 지정합니다. KMS 키에 대한 권한을 활성화, 비활성화, 취소할 수 있습니다. KMS 키는 다음 두 가지 유형 중 하나가 될 수 있습니다.

- AWS 관리형 키 Amazon FSx의 경우 — 기본 KMS 키입니다. KMS 키를 생성하고 저장하는 데 요금이 부과되는 것은 아니지만 사용 요금이 있습니다. 자세한 내용은 AWS Key Management Service 요금을 참조하세요.
- 고객 관리형 키 – 여러 사용자나 서비스에 대한 키 정책 및 권한을 구성할 수 있는 가장 유연한 KMS 키입니다. 고객 관리 키 생성에 대한 자세한 내용은 개발자 안내서의 키 생성을 참조하십시오. AWS Key Management Service

고객 관리형 키를 데이터 암호화 및 암호화 해제의 KMS 키로 사용하면 키 교체를 활성화할 수 있습니다. 키 순환을 활성화하면 1년에 한 번 AWS KMS 자동으로 키를 교체합니다. 또한 고객 관리형 키를 사용하면 고객 관리형 키에 대한 액세스를 비활성화, 재활성화, 삭제, 취소하는 시기를 선택할 수 있습니다.

> **Important**
>
> Amazon FSx는 대칭 암호화 KMS 키만 승인합니다. Amazon FSx에서는 비대칭 KMS 키를 사용할 수 없습니다.

다음에 대한 아마존 FSx 주요 정책 AWS KMS

키 정책은 KMS 키에 대한 액세스를 제어하는 기본 방법입니다. 키 정책에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 AWS KMS의 키 정책 사용을 참조하세요. 다음은 Amazon FSx가 암호화된 저장 파일 시스템을 위해 지원하는 모든 AWS KMS관련 권한을 나열한 목록입니다.

- kms:Encrypt – (선택 사항) 일반 텍스트를 사이퍼텍스트로 암호화합니다. 이 권한은 기본 키 정책에 포함되어 있습니다.
- kms:Decrypt – (필수 사항) 사이퍼텍스트를 암호화 해제합니다. 사이퍼텍스트는 이전에 암호화한 일반 텍스트입니다. 이 권한은 기본 키 정책에 포함되어 있습니다.
- kms: ReEncrypt — (선택 사항) 클라이언트 측 데이터의 일반 텍스트를 노출하지 않고 새 KMS 키로 서버 측 데이터를 암호화합니다. 먼저 데이터를 복호화한 후 다시 암호화합니다. 이 권한은 기본 키 정책에 포함되어 있습니다.
- kms: GenerateData KeyWithout 일반 텍스트 — (필수) KMS 키로 암호화된 데이터 암호화 키를 반환합니다. 이 권한은 kms: Key*의 기본 키 정책에 포함됩니다. GenerateData
- kms: CreateGrant - (필수) 누가 어떤 조건에서 키를 사용할 수 있는지 지정하는 권한 부여를 키에 추가합니다. 이런 권한 부여는 키 정책을 대체하는 권한 메커니즘입니다. 권한 부여에 대한 자세한 내용은 AWS Key Management Service 개발자 안내서의 권한 부여 사용을 참조하세요. 이 권한은 기본 키 정책에 포함되어 있습니다.
- kms: DescribeKey — (필수) 지정된 KMS 키에 대한 세부 정보를 제공합니다. 이 권한은 기본 키 정책에 포함되어 있습니다.
- kms: ListAliases - (선택 사항) 계정의 모든 키 별칭을 나열합니다. 콘솔을 사용해 암호화된 파일 시스템을 생성하는 경우, 이 권한이 KMS 마스터 키 선택 목록을 채웁니다. 최상의 사용자 경험을 제공하기 위해 이 권한을 사용하는 것이 좋습니다. 이 권한은 기본 키 정책에 포함되어 있습니다.

전송 중 데이터 암호화

스크래치 2와 영구 파일 시스템은 전송 중 데이터를 자동으로 암호화할 수 있습니다. 다음 표에서 해당 배포 유형의 셀에 체크 표시가 있는 경우 전송 중 암호화를 지원하는 Amazon EC2 인스턴스에서 파일 시스템에 액세스하면 전송 중에 데이터가 암호화되고 AWS 리전파일 시스템 내의 호스트 간 모든 통신도 암호화됩니다. 전송 중 암호화를 지원하는 EC2 인스턴스에 대해 알아보려면 Amazon EC2 사용 설명서의 전송 중 암호화를 참조하십시오.

스크래치 2 및 영구 파일 시스템의 전송 중 데이터 암호화는 다음에서 사용할 수 있습니다. AWS 리전

AWS 리전	스크래치_2	영구_1	영구_2
미국 동부(오하이오)	✓	✓	✓
미국 동부(버지니아 북부)	✓	✓	✓
미국 동부 (애틀랜타) 로컬 존*			✓
미국 서부(오레곤)	✓	✓	✓
미국 서부(캘리포니아 북부)*	✓	✓	
미국 서부 (로스앤젤레스) 로컬 존	✓	✓	
AWS GovCloud (미국 동부) *	✓	✓	
AWS GovCloud (미국 서부)	✓	✓	
캐나다(중부)*	✓	✓	✓
캐나다 서부 (캘거리) *			✓
유럽(아일랜드)	✓	✓	✓
유럽(밀라노)	✓	✓	
유럽(프랑크푸르트)	✓	✓	✓
유럽(파리)	✓	✓	
유럽(런던)	✓	✓	✓
유럽(스톡홀름)*	✓	✓	✓
아시아 태평양(서울)	✓	✓	✓
아시아 태평양(싱가포르)	✓	✓	✓
아시아 태평양(도쿄)*	✓	✓	✓
아시아 태평양(뭄바이)*	✓	✓	✓

AWS 리전	스크래치_2	영구_1	영구_2
아시아 태평양(홍콩)*	✓	✓	✓
아시아 태평양(시드니)*	✓	✓	✓
이스라엘(텔아비브)*	✓		✓
남아메리카(상파울루)*	✓	✓	

> **Note**
>
> * 전송 중 데이터 암호화는 2021년 4월 11일 이후에 생성된 파일 시스템에 사용할 수 있습니다.

인터네트워크 트래픽 개인 정보

이 주제에서는 Amazon FSx가 해당 서비스에서 다른 위치로 향하는 연결을 보호하는 방법을 설명합니다.

Amazon FSx 및 온프레미스 클라이언트 간 트래픽

사설 네트워크와 다음 두 가지 연결 옵션이 있습니다. AWS

- AWS Site-to-Site VPN 연결. 자세한 내용은 AWS Site-to-Site VPN무엇입니까를 참조하십시오.
- AWS Direct Connect 연결. 자세한 내용은 AWS Direct Connect무엇입니까를 참조하십시오.

네트워크를 통해 FSx for Lustre에 액세스하여 관리 작업을 수행하기 위해 게시된 API 작업에 도달하고 파일 시스템과 상호 AWS작용하기 위한 Lustre 포트에 액세스할 수 있습니다.

API 트래픽 암호화

AWS게시된 API 작업에 액세스하려면 클라이언트가 TLS (전송 계층 보안) 1.2 이상을 지원해야 합니다. TLS 1.2는 필수이며 TLS 1.3를 권장합니다. 클라이언트는 DHE(Ephemeral Diffie-Hellman) 또는 ECDHE(Elliptic Curve Diffie-Hellman Ephemeral)와 같은 PFS(전달 완전 보안)가 포함된 암호 제품군도 지원해야 합니다. Java 7 이상의 최신 시스템은 대부분 이러한 모드를 지원합니다. 또한 요청은 액세스 키 자격 증명 및 IAM 보안 주체와 관련된 비밀 액세스 키를 사용하여 서명해야 합니다. 또는 AWS

Security Token Service (STS)를 사용하여 임시 보안 인증 정보를 생성하여 요청에 서명할 수 있습니다.

데이터 트래픽 암호화

전송 중 데이터의 암호화는 지원되는 EC2 인스턴스에서 AWS 클라우드내 파일 시스템에 액세스하는 경우 활성화됩니다. 자세한 정보는 전송 중 데이터 암호화을 참조하세요. FSx for Lustre는 온프레미스 클라이언트와 파일 시스템 간 전송 시 암호화를 기본적으로 제공하지 않습니다.

Amazon FSx for Lustre용 ID 및 액세스 관리

AWS Identity and Access Management (IAM) 은 관리자가 리소스에 대한 액세스를 안전하게 제어할 수 AWS 서비스 있도록 AWS 도와줍니다. IAM 관리자는 어떤 사용자가 Amazon FSx 리소스를 사용할 수 있도록 인증(로그인)되고 권한이 부여(권한 있음)될 수 있는지 제어합니다. IAM은 추가 AWS 서비스 비용 없이 사용할 수 있습니다.

주제

- 고객
- 보안 인증 정보를 통한 인증
- 정책을 사용한 액세스 관리
- Amazon FSx for Lustre가 IAM과 작동하는 방식
- Amazon FSx for Lustre의 자격 증명 기반 정책 예
- AWS 아마존 FSx에 대한 관리형 정책
- Amazon FSx for Lustre 자격 증명 및 액세스 문제 해결
- Amazon FSx에서 태그 사용
- Amazon FSx에 대해 서비스 연결 역할 사용

고객

사용 방법 AWS Identity and Access Management (IAM) 은 Amazon FSx에서 수행하는 작업에 따라 다릅니다.

서비스 사용자 – Amazon FSx 서비스를 사용하여 작업을 수행하는 경우 필요한 보안 인증 정보와 권한을 관리자가 제공합니다. 더 많은 Amazon FSx 기능을 사용하여 작업을 수행한다면 추가 권한이 필

요할 수 있습니다. 액세스 권한 관리 방식을 이해하면 적절한 권한을 관리자에게 요청할 수 있습니다. Amazon FSx의 기능에 액세스할 수 없다면 Amazon FSx for Lustre 자격 증명 및 액세스 문제 해결 섹션을 참조하세요.

서비스 관리자 – 회사에서 Amazon FSx 리소스를 책임지고 있다면 Amazon FSx에 대한 완전한 액세스 권한이 있을 것입니다. 서비스 관리자는 서비스 사용자가 액세스해야 하는 Amazon FSx 기능과 리소스를 결정합니다. 그런 다음, IAM 관리자에게 요청을 제출하여 서비스 사용자의 권한을 변경해야 합니다. 이 페이지의 정보를 검토하여 IAM의 기본 개념을 이해해 두세요. 회사가 Amazon FSx에서 IAM을 사용하는 방법에 대해 자세히 알아보려면 Amazon FSx for Lustre가 IAM과 작동하는 방식 섹션을 참조하세요.

IAM 관리자 - IAM 관리자라면 Amazon FSx에 대한 액세스 관리 정책 작성 방법을 자세히 알고 싶을 수도 있습니다. IAM에서 사용할 수 있는 Amazon FSx ID 기반 정책의 예제를 확인하려면 Amazon FSx for Lustre의 자격 증명 기반 정책 예 섹션을 참조하세요.

보안 인증 정보를 통한 인증

인증은 자격 증명을 AWS 사용하여 로그인하는 방법입니다. IAM 사용자로 인증 (로그인 AWS) 하거나 IAM 역할을 맡아 인증 (로그인) 해야 합니다. AWS 계정 루트 사용자

ID 소스를 통해 제공된 자격 증명을 사용하여 페더레이션 ID로 로그인할 수 있습니다. AWS AWS IAM Identity Center (IAM ID 센터) 사용자, 회사의 싱글 사인온 인증, Google 또는 Facebook 자격 증명이 페더레이션 ID의 예입니다. 페더레이션형 ID로 로그인할 때 관리자가 이전에 IAM 역할을 사용하여 ID 페더레이션을 설정했습니다. 페더레이션을 사용하여 액세스하는 경우 AWS 간접적으로 역할을 맡게 됩니다.

사용자 유형에 따라 AWS Management Console 또는 AWS 액세스 포털에 로그인할 수 있습니다. 로그인에 대한 자세한 내용은 AWS 로그인 사용 설명서의 내 로그인 방법을 참조하십시오. AWS AWS 계정

AWS 프로그래밍 방식으로 액세스하는 경우 자격 증명을 사용하여 요청에 암호화 방식으로 서명할 수 있는 소프트웨어 개발 키트 (SDK) 와 명령줄 인터페이스 (CLI) 를 AWS 제공합니다. AWS 도구를 사용하지 않는 경우 요청에 직접 서명해야 합니다. 권장 방법을 사용하여 직접 요청에 서명하는 방법에 대한 자세한 내용은 IAM 사용 설명서의 AWS API 요청 서명을 참조하십시오.

사용하는 인증 방법에 상관없이 추가 보안 정보를 제공해야 할 수도 있습니다. 예를 들어, AWS 계정의 보안을 강화하기 위해 다단계 인증 (MFA) 을 사용할 것을 권장합니다. 자세한 내용은 AWS IAM Identity Center 사용 설명서의 다중 인증 및 IAM 사용자 설명서의 AWS에서 다중 인증(MFA) 사용을 참조합니다.

AWS 계정 루트 사용자

계정을 AWS 계정만들 때는 먼저 계정의 모든 AWS 서비스 리소스에 대한 완전한 액세스 권한을 가진 하나의 로그인 ID로 시작합니다. 이 ID를 AWS 계정 루트 사용자라고 하며, 계정을 만들 때 사용한 이메일 주소와 비밀번호로 로그인하여 액세스할 수 있습니다. 일상적인 작업에 루트 사용자를 사용하지 않을 것을 강력히 권장합니다. 루트 사용자 보안 인증 정보를 보호하고 루트 사용자만 수행할 수 있는 태스크를 수행하는 데 사용하세요. 루트 사용자로 로그인해야 하는 태스크의 전체 목록은 IAM 사용자 안내서의 루트 사용자 보안 인증이 필요한 태스크를 참조하세요.

페더레이션 자격 증명

가장 좋은 방법은 관리자 액세스가 필요한 사용자를 비롯한 수동 AWS 서비스 사용자가 ID 공급자와의 페더레이션을 사용하여 임시 자격 증명을 사용하여 액세스하도록 하는 것입니다.

페더레이션 ID는 기업 사용자 디렉토리, 웹 ID 공급자, Identity Center 디렉터리의 사용자 또는 ID 소스를 통해 제공된 자격 증명을 사용하여 액세스하는 AWS 서비스 모든 사용자를 말합니다. AWS Directory Service페더레이션 ID에 AWS 계정액세스하면 이들이 역할을 맡고 역할은 임시 자격 증명을 제공합니다.

중앙 집중식 액세스 관리를 위해 AWS IAM Identity Center을 사용하는 것이 좋습니다. IAM Identity Center에서 사용자 및 그룹을 생성하거나 자체 ID 소스의 사용자 및 그룹 집합에 연결하고 동기화하여 모든 사용자 및 애플리케이션에서 사용할 수 있습니다. AWS 계정 IAM Identity Center에 대한 자세한 내용은AWS IAM Identity Center 사용 설명서에서 IAM Identity Center란 무엇입니까?를 참조하세요.

IAM 사용자 및 그룹

IAM 사용자는 단일 사용자 또는 애플리케이션에 대한 특정 권한을 AWS 계정 가진 사용자 내 자격 증명입니다. 가능하면 암호 및 액세스 키와 같은 장기 자격 증명이 있는 IAM 사용자를 생성하는 대신 임시 자격 증명을 사용하는 것이 좋습니다. 하지만 IAM 사용자의 장기 자격 증명이 필요한 특정 사용 사례가 있는 경우 액세스 키를 교체하는 것이 좋습니다. 자세한 내용은 IAM 사용 설명서의 장기 보안 인증이 필요한 사용 사례의 경우 정기적으로 액세스 키 교체를 참조하세요.

IAM 그룹은 IAM 사용자 컬렉션을 지정하는 자격 증명입니다. 사용자는 그룹으로 로그인할 수 없습니다. 그룹을 사용하여 여러 사용자의 권한을 한 번에 지정할 수 있습니다. 그룹을 사용하면 대규모 사용자 집합의 권한을 더 쉽게 관리할 수 있습니다. 예를 들어, IAMAdmins라는 그룹이 있고 이 그룹에 IAM 리소스를 관리할 권한을 부여할 수 있습니다.

사용자는 역할과 다릅니다. 사용자는 한 사람 또는 애플리케이션과 고유하게 연결되지만, 역할은 해당 역할이 필요한 사람이라면 누구나 수임할 수 있습니다. 사용자는 영구적인 장기 자격 증명을 가지고 있

지만, 역할은 임시 보안 인증만 제공합니다. 자세한 정보는 IAM 사용 설명서의 IAM 사용자를 만들어야 하는 경우(역할이 아님)를 참조하세요.

IAM 역할

IAM 역할은 특정 권한을 가진 사용자 AWS 계정 내의 자격 증명입니다. IAM 사용자와 유사하지만, 특정 개인과 연결되지 않습니다. 역할을 AWS Management Console 전환하여 에서 일시적으로 IAM 역할을 맡을 수 있습니다. AWS CLI 또는 AWS API 작업을 호출하거나 사용자 지정 URL을 사용하여 역할을 수임할 수 있습니다. 역할 사용 방법에 대한 자세한 정보는 IAM 사용 설명서의 IAM 역할 사용을 참조하세요.

임시 보안 인증 정보가 있는 IAM 역할은 다음과 같은 상황에서 유용합니다.

- 페더레이션 사용자 액세스 - 페더레이션 아이덴티티에 권한을 부여하려면 역할을 생성하고 해당 역할의 권한을 정의합니다. 연동 자격 증명이 인증되면 역할이 연결되고 역할에 정의된 권한이 부여됩니다. 페더레이션 역할에 대한 자세한 내용은 IAM 사용 설명서의 타사 자격 증명 공급자의 역할 만들기를 참조하세요. IAM Identity Center를 사용하는 경우 권한 세트를 구성합니다. 인증 후 자격 증명이 액세스할 수 있는 항목을 제어하기 위해 IAM Identity Center는 권한 세트를 IAM의 역할과 연관짓습니다. 권한 세트에 대한 자세한 내용은AWS IAM Identity Center 사용 설명서의 권한 세트를 참조하세요.
- 임시 IAM 사용자 권한 - IAM 사용자 또는 역할은 IAM 역할을 수임하여 특정 태스크에 대한 다양한 권한을 임시로 받을 수 있습니다.
- 크로스 계정 액세스 - IAM 역할을 사용하여 다른 계정의 사용자(신뢰할 수 있는 보안 주체)가 내 계정의 리소스에 액세스하도록 허용할 수 있습니다. 역할은 계정 간 액세스를 부여하는 기본적인 방법입니다. 그러나 일부 AWS 서비스경우에는 역할을 프록시로 사용하는 대신 정책을 리소스에 직접 연결할 수 있습니다. 크로스 계정 액세스를 위한 역할과 리소스 기반 정책의 차이점을 알아보려면 IAM 사용 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.
- 서비스 간 액세스 — 일부는 다른 AWS 서비스서비스의 기능을 AWS 서비스 사용합니다. 예컨대, 어떤 서비스에서 호출을 수행하면 일반적으로 해당 서비스는 Amazon EC2에서 애플리케이션을 실행하거나 Amazon S3에 객체를 저장합니다. 서비스는 호출하는 보안 주체의 권한을 사용하거나, 서비스 역할을 사용하거나, 또는 서비스 연결 역할을 사용하여 이 작업을 수행할 수 있습니다.
  - 순방향 액세스 세션 (FAS) — IAM 사용자 또는 역할을 사용하여 작업을 수행하는 경우 보안 AWS 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는 전화를 거는 주체의 권한을 다운스트림 AWS 서비스서비스에 AWS 서비스 요청하기 위한 요청과 결합하여 사용합니다. FAS 요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업

을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 전달 액세스 세션을 참조하세요.

- 서비스 역할 - 서비스 역할은 서비스가 사용자를 대신하여 태스크를 수행하기 위해 맡는 IAM 역할입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 정보는 IAM 사용자 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 참조합니다.
- 서비스 연결 역할 — 서비스 연결 역할은 에 연결된 서비스 역할의 한 유형입니다. AWS 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM 관리자는 서비스 링크 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.

- Amazon EC2에서 실행되는 애플리케이션 — IAM 역할을 사용하여 EC2 인스턴스에서 실행되고 API 요청을 AWS CLI 하는 애플리케이션의 임시 자격 증명을 관리할 수 있습니다. AWS 이는 EC2 인스턴스 내에 액세스 키를 저장할 때 권장되는 방법입니다. EC2 인스턴스에 AWS 역할을 할당하고 모든 애플리케이션에서 사용할 수 있게 하려면 인스턴스에 연결된 인스턴스 프로필을 생성합니다. 인스턴스 프로파일에는 역할이 포함되어 있으며 EC2 인스턴스에서 실행되는 프로그램이 임시 보안 인증을 얻을 수 있습니다. 자세한 정보는 IAM 사용 설명서의 IAM 역할을 사용하여 Amazon EC2 인스턴스에서 실행되는 애플리케이션에 권한 부여를 참조하세요.

IAM 역할을 사용할지 또는 IAM 사용자를 사용할지를 알아보려면 IAM 사용자 설명서의 IAM 역할(사용자 대신)을 생성하는 경우를 참조합니다.

정책을 사용한 액세스 관리

정책을 생성하고 이를 AWS ID 또는 리소스에 AWS 연결하여 액세스를 제어할 수 있습니다. 정책은 ID 또는 리소스와 연결될 때 AWS 해당 권한을 정의하는 객체입니다. AWS 주도자 (사용자, 루트 사용자 또는 역할 세션) 가 요청할 때 이러한 정책을 평가합니다. 정책에서 권한은 요청이 허용되거나 거부되는 지를 결정합니다. 대부분의 정책은 JSON 문서로 AWS 저장됩니다. JSON 정책 문서의 구조와 콘텐츠에 대한 자세한 정보는 IAM 사용 설명서의 JSON 정책 개요를 참조하세요.

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.

기본적으로, 사용자와 역할에는 어떠한 권한도 없습니다. 사용자에게 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다. 그런 다음 관리자가 IAM 정책을 역할에 추가하고, 사용자가 역할을 수임할 수 있습니다.

IAM 정책은 작업을 수행하기 위해 사용하는 방법과 상관없이 작업에 대한 권한을 정의합니다. 예를 들어, `iam:GetRole`태스크를 허용하는 정책이 있다고 가정합니다. 해당 정책을 사용하는 사용자는 AWS Management Console, AWS CLI, 또는 AWS API에서 역할 정보를 가져올 수 있습니다.

ID 기반 정책

ID 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 자격 증명에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는 지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

자격 증명 기반 정책은 인라인 정책 또는 관리형 정책으로 한층 더 분류할 수 있습니다. 인라인 정책은 단일 사용자, 그룹 또는 역할에 직접 포함됩니다. 관리형 정책은 내 여러 사용자, 그룹 및 역할에 연결할 수 있는 독립형 정책입니다. AWS 계정관리형 정책에는 AWS 관리형 정책과 고객 관리형 정책이 포함됩니다. 관리형 정책 또는 인라인 정책을 선택하는 방법을 알아보려면 IAM 사용 설명서의 관리형 정책과 인라인 정책의 선택을 참조하세요.

리소스 기반 정책

리소스 기반 정책은 리소스에 연결하는 JSON 정책 설명서입니다. 리소스 기반 정책의 예는 IAM 역할 신뢰 정책과 Amazon S3 버킷 정책입니다. 리소스 기반 정책을 지원하는 서비스에서 서비스 관리자는 이러한 정책을 사용하여 특정 리소스에 대한 액세스를 통제할 수 있습니다. 정책이 연결된 리소스의 경우 정책은 지정된 보안 주체가 해당 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는 지를 정의합니다. 리소스 기반 정책에서 보안 주체를 지정해야 합니다. 보안 주체에는 계정, 사용자, 역할, 연동 사용자 등이 포함될 수 있습니다. AWS 서비스

리소스 기반 정책은 해당 서비스에 있는 인라인 정책입니다. IAM의 AWS 관리형 정책은 리소스 기반 정책에 사용할 수 없습니다.

액세스 제어 목록(ACLs)

액세스 제어 목록(ACLs)은 어떤 보안 주체(계정 멤버, 사용자 또는 역할)가 리소스에 액세스할 수 있는 권한을 가지고 있는 지를 제어합니다. ACLs는 JSON 정책 문서 형식을 사용하지 않지만 리소스 기반 정책과 유사합니다.

ACL을 지원하는 서비스의 예로는 아마존 S3와 아마존 VPC가 있습니다. AWS WAF ACL에 대해 자세히 알아보려면 Amazon Simple Storage Service 개발자 안내서의 액세스 제어 목록(ACL) 개요를 참조하세요.

기타 정책 타입

AWS 일반적이지 않은 추가 정책 유형을 지원합니다. 이러한 정책 타입은 더 일반적인 정책 타입에 따라 사용자에게 부여되는 최대 권한을 설정할 수 있습니다.

- 권한 경계 – 권한 경계는 보안 인증 기반 정책에 따라 IAM 엔터티(IAM 사용자 또는 역할)에 부여할 수 있는 최대 권한을 설정하는 고급 기능입니다. 개체에 대한 권한 경계를 설정할 수 있습니다. 그 결과로 얻는 권한은 엔터티의 자격 증명 기반 정책과 그 권한 경계의 교집합입니다. `Principal` 필드에서 사용자나 역할을 보안 주체로 지정하는 리소스 기반 정책은 권한 경계를 통해 제한되지 않습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 권한 경계에 대한 자세한 정보는 IAM 사용자 설명서의 IAM 엔터티에 대한 권한 경계를 참조합니다.
- 서비스 제어 정책 (SCP) - SCP는 조직 또는 조직 단위 (OU) 에 대한 최대 권한을 지정하는 JSON 정책입니다. AWS Organizations AWS Organizations 사업체가 소유한 여러 AWS 계정 개를 그룹화하고 중앙에서 관리하는 서비스입니다. 조직에서 모든 기능을 활성화할 경우 서비스 제어 정책 (SCP)을 임의의 또는 모든 계정에 적용할 수 있습니다. SCP는 구성원 계정의 엔티티 (각 엔티티 포함) 에 대한 권한을 제한합니다. AWS 계정 루트 사용자조직 및 SCP에 대한 자세한 정보는AWS Organizations 사용 설명서의 SCP 작동 방식을 참조하세요.
- 세션 정책 – 세션 정책은 역할 또는 연합된 사용자에 대해 임시 세션을 프로그래밍 방식으로 생성할 때 파라미터로 전달하는 고급 정책입니다. 결과적으로 얻는 세션의 권한은 사용자 또는 역할 자격 증명 기반 정책의 교차 및 세션 정책입니다. 또한 권한을 리소스 기반 정책에서 가져올 수도 있습니다. 이러한 정책 중 하나에 포함된 명시적 거부는 허용을 재정의합니다. 자세한 정보는 IAM 사용자 설명서의 세션 정책을 참조합니다.

여러 정책 타입

여러 정책 타입이 요청에 적용되는 경우 결과 권한은 이해하기가 더 복잡합니다. 여러 정책 유형이 관련되어 있을 때 요청을 허용할지 여부를 AWS 결정하는 방법을 알아보려면 IAM 사용 설명서의 정책 평가 로직을 참조하십시오.

Amazon FSx for Lustre가 IAM과 작동하는 방식

IAM을 사용하여 Amazon FSx에 대한 액세스를 관리하기 전에 Amazon FSx에서 사용할 수 있는 IAM 기능에 대해 알아봅니다.

Amazon FSx for Lustre에서 사용할 수 있는 IAM 기능

IAM 특성	Amazon FSx 지원
ID 기반 정책	예
리소스 기반 정책	아니요

IAM 특성	Amazon FSx 지원
정책 작업	예
정책 리소스	예
정책 조건 키	예
ACLs	아니요
ABAC(정책의 태그)	예
임시 보안 인증	예
전달 액세스 세션(FAS)	예
서비스 역할	아니요
서비스 링크 역할	예

Amazon FSx 및 AWS 기타 서비스가 대부분의 IAM 기능과 어떻게 작동하는지 자세히 알아보려면 IAM 사용 설명서의 IAM과 함께 작동하는 서비스를AWS 참조하십시오.

Amazon FSx의 자격 증명 기반 정책

ID 기반 정책 지원	예

자격 증명 기반 정책은 IAM 사용자, 사용자 그룹 또는 역할과 같은 자격 증명에 연결할 수 있는 JSON 권한 정책 문서입니다. 이러한 정책은 사용자와 역할이 어떤 리소스와 어떤 조건에서 어떤 작업을 수행할 수 있는 지를 제어합니다. 자격 증명 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

IAM 자격 증명 기반 정책을 사용하면 허용되거나 거부되는 작업와 리소스뿐 아니라 작업이 허용되거나 거부되는 조건을 지정할 수 있습니다. 자격 증명 기반 정책에서는 보안 주체가 연결된 사용자 또는 역할에 적용되므로 보안 주체를 지정할 수 없습니다. JSON 정책에서 사용하는 모든 요소에 대해 알아보려면 IAM 사용 설명서의 IAM JSON 정책 요소 참조를 참조하세요.

Amazon FSx의 자격 증명 기반 정책 예

Amazon FSx 자격 증명 기반 정책 예제를 보려면 Amazon FSx for Lustre의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon FSx 내의 리소스 기반 정책

리소스 기반 정책 지원	아니요

Amazon FSx의 정책 작업

정책 작업 지원	예

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 `Action`요소는 정책에서 액세스를 허용하거나 거부하는 데 사용할 수 있는 태스크를 설명합니다. 정책 작업은 일반적으로 관련 AWS API 작업과 이름이 같습니다. 일치하는 API 작업이 없는 권한 전용 작업 같은 몇 가지 예외도 있습니다. 정책에서 여러 작업이 필요한 몇 가지 작업도 있습니다. 이러한 추가 작업을 일컬어 종속 작업이라고 합니다.

연결된 작업을 수행할 수 있는 권한을 부여하기 위한 정책에 작업을 포함하십시오.

Amazon FSx 작업 목록을 보려면 서비스 승인 참조의 Amazon FSx for Lustre에서 정의한 작업을 참조하세요.

Amazon FSx의 정책 작업은 작업 앞에 다음 접두사를 사용합니다.

```
fsx
```

단일 문에서 여러 작업을 지정하려면 다음과 같이 쉼표로 구분합니다.

```
"Action": [
      "fsx:action1",
      "fsx:action2"
         ]
```

Amazon FSx 자격 증명 기반 정책 예제를 보려면 Amazon FSx for Lustre의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon FSx의 정책 리소스

정책 리소스 지원	예

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지 지정할 수 있습니다.

`Resource` JSON 정책 요소는 작업이 적용되는 하나 이상의 개체를 지정합니다. 문장에는 `Resource`또는 `NotResource`요소가 반드시 추가되어야 합니다. 모범 사례에 따라 Amazon 리소스 이름(ARN)을 사용하여 리소스를 지정합니다. 리소스 수준 권한이라고 하는 특정 리소스 타입을 지원하는 작업에 대해 이 작업을 수행할 수 있습니다.

작업 나열과 같이 리소스 수준 권한을 지원하지 않는 작업의 경우, 와일드카드(*)를 사용하여 해당 문이 모든 리소스에 적용됨을 나타냅니다.

```
"Resource": "*"
```

Amazon FSx 리소스 유형 및 해당 ARN의 목록을 보려면 서비스 권한 부여 참조에서 Amazon FSx for Lustre에서 정의한 리소스를 참조하세요. 각 리소스의 ARN을 지정할 수 있는 작업을 알아보려면 Amazon FSx for Lustre에서 정의한 작업을 참조하세요.

Amazon FSx 자격 증명 기반 정책 예제를 보려면 Amazon FSx for Lustre의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon FSx의 정책 조건 키

서비스별 정책 조건 키 지원	예

관리자는 AWS JSON 정책을 사용하여 누가 무엇에 액세스할 수 있는지 지정할 수 있습니다. 즉, 어떤 보안 주체가 어떤 리소스와 어떤 조건에서 작업을 수행할 수 있는지 지정할 수 있습니다.

Condition 요소(또는 Condition 블록)를 사용하면 정책이 발효되는 조건을 지정할 수 있습니다. Condition 요소는 옵션입니다. 같거나 작음과 같은 조건 연산자를 사용하여 정책의 조건을 요청의 값과 일치시키는 조건식을 생성할 수 있습니다.

한 문에서 여러 Condition요소를 지정하거나 단일 Condition요소에서 여러 키를 지정하는 경우 AWS 는 논리적 AND태스크를 사용하여 평가합니다. 단일 조건 키에 여러 값을 지정하는 경우 는 논리적 OR 연산을 사용하여 조건을 AWS 평가합니다. 명문의 권한을 부여하기 전에 모든 조건을 충족해야 합니다.

조건을 지정할 때 자리 표시자 변수를 사용할 수도 있습니다. 예를 들어, IAM 사용자에게 IAM 사용자 이름으로 태그가 지정된 경우에만 리소스에 액세스할 수 있는 권한을 부여할 수 있습니다. 자세한 정보는 IAM 사용 설명서의 IAM 정책 요소: 변수 및 태그를 참조하세요.

AWS 글로벌 조건 키 및 서비스별 조건 키를 지원합니다. 모든 AWS 글로벌 조건 키를 보려면 IAM 사용 AWS 설명서의 글로벌 조건 컨텍스트 키를 참조하십시오.

Amazon FSx 조건 키 목록을 보려면 서비스 승인 참조의 Amazon FSx for Lustre에 사용되는 조건 키를 참조하세요. 조건 키를 사용할 수 있는 작업과 리소스를 알아보려면 Amazon FSx for Lustre에서 정의한 작업을 참조하세요.

Amazon FSx 자격 증명 기반 정책 예제를 보려면 Amazon FSx for Lustre의 자격 증명 기반 정책 예 섹션을 참조하세요.

Amazon FSx의 액세스 제어 목록(ACL)

ACL 지원	아니요

Amazon FSx를 사용한 속성 기반 액세스 제어(ABAC)

ABAC 지원(정책의 태그)	예

ABAC(속성 기반 액세스 제어)는 속성을 기반으로 권한을 정의하는 권한 부여 전략입니다. AWS에서는 이러한 속성을 태그라고 합니다. IAM 엔티티 (사용자 또는 역할) 및 여러 AWS 리소스에 태그를 첨부할 수 있습니다. ABAC의 첫 번째 단계로 개체 및 리소스에 태그를 지정합니다. 그런 다음 보안 주체의 태그가 액세스하려는 리소스의 태그와 일치할 때 작업을 허용하도록 ABAC 정책을 설계합니다.

ABAC는 빠르게 성장하는 환경에서 유용하며 정책 관리가 번거로운 상황에 도움이 됩니다.

태그를 기반으로 액세스를 제어하려면 `aws:ResourceTag/key-name`, `aws:RequestTag/key-name` 또는 `aws:TagKeys` 조건 키를 사용하여 정책의 조건 요소에 태그 정보를 제공합니다.

서비스가 모든 리소스 타입에 대해 세 가지 조건 키를 모두 지원하는 경우, 값은 서비스에 대해 예입니다. 서비스가 일부 리소스 타입에 대해서만 세 가지 조건 키를 모두 지원하는 경우, 값은 부분적입니다.

ABAC에 대한 자세한 정보는 IAM 사용 설명서의 ABAC란 무엇인가요?를 참조하세요. ABAC 설정 단계가 포함된 자습서를 보려면 IAM 사용 설명서의 속성 기반 액세스 제어(ABAC) 사용을 참조하세요.

Amazon FSx 리소스 태그 지정에 대한 자세한 내용은 Amazon FSx 리소스 태그 지정 섹션을 참조하세요.

리소스의 태그를 기반으로 리소스에 대한 액세스를 제한하는 자격 증명 기반 정책의 예제는 태그를 사용하여 Amazon FSx 리소스에 대한 액세스 제어 섹션에서 확인할 수 있습니다.

Amazon FSx에서 임시 보안 인증 사용

임시 보안 인증 지원	예

임시 자격 증명을 사용하여 로그인하면 작동하지 AWS 서비스 않는 것도 있습니다. 임시 자격 증명을 사용하는 방법을 AWS 서비스 비롯한 추가 정보는 IAM 사용 설명서의 IAM과AWS 서비스 연동되는 내용을 참조하십시오.

사용자 이름과 암호를 제외한 다른 방법을 AWS Management Console 사용하여 로그인하면 임시 자격 증명을 사용하는 것입니다. 예를 들어 회사의 SSO (Single Sign-On) 링크를 AWS 사용하여 액세스하는 경우 이 프로세스에서 자동으로 임시 자격 증명을 생성합니다. 또한 콘솔에 사용자로 로그인한 다음 역할을 전환할 때 임시 보안 인증을 자동으로 생성합니다. 역할 전환에 대한 자세한 정보는 IAM 사용 설명서의 역할로 전환(콘솔)을 참조하세요.

또는 API를 사용하여 임시 자격 증명을 수동으로 생성할 수 있습니다 AWS CLI . AWS 그런 다음 해당 임시 자격 증명을 사용하여 액세스할 수 AWS있습니다. AWS 장기 액세스 키를 사용하는 대신 임시 자격 증명을 동적으로 생성할 것을 권장합니다. 자세한 정보는 IAM의 임시 보안 인증 섹션을 참조하세요.

Amazon FSx를 위한 포워드 액세스 세션

전달 액세스 세션(FAS) 지원	예

IAM 사용자 또는 역할을 사용하여 작업을 수행하는 AWS경우 사용자는 보안 주체로 간주됩니다. 일부 서비스를 사용하는 경우 다른 서비스에서 다른 작업을 시작하는 작업을 수행할 수 있습니다. FAS는 전화를 거는 주체의 권한을 다운스트림 서비스에 AWS 서비스 요청하기 위한 요청과 함께 사용합니다. AWS 서비스 FAS 요청은 다른 서비스 AWS 서비스 또는 리소스와의 상호 작용이 필요한 요청을 서비스가 수신한 경우에만 이루어집니다. 이 경우 두 작업을 모두 수행할 수 있는 권한이 있어야 합니다. FAS 요청 시 정책 세부 정보는 전달 액세스 세션을 참조하세요.

Amazon FSx의 서비스 역할

서비스 역할 지원	아니요

서비스 역할은 서비스가 사용자를 대신하여 작업을 수행하기 위해 수임하는 IAM role(IAM 역할)입니다. IAM 관리자는 IAM 내에서 서비스 역할을 생성, 수정 및 삭제할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 AWS 서비스에 대한 권한을 위임할 역할 생성을 참조하세요.

> **Warning**
>
> 서비스 역할에 대한 권한을 변경하면 Amazon FSx 기능이 중단될 수 있습니다. Amazon FSx에서 관련 지침을 제공하는 경우에만 서비스 역할을 편집합니다.

Amazon FSx의 서비스 연결 역할

서비스 링크 역할 지원	예

서비스 연결 역할은 에 연결된 서비스 역할의 한 유형입니다. AWS 서비스서비스는 사용자를 대신하여 작업을 수행하기 위해 역할을 수임할 수 있습니다. 서비스 연결 역할은 사용자에게 AWS 계정 표시되며 해당 서비스가 소유합니다. IAM 관리자는 서비스 링크 역할의 권한을 볼 수 있지만 편집은 할 수 없습니다.

Amazon FSx 서비스 연결 IAM 역할을 관리하고 생성하는 방법에 대한 자세한 w정보는 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요.

Amazon FSx for Lustre의 자격 증명 기반 정책 예

기본적으로 사용자 및 역할은 Amazon FSx 리소스를 생성하거나 수정할 수 있는 권한이 없습니다. 또한 AWS Management Console, AWS Command Line Interface (AWS CLI) 또는 AWS API를 사용하여

작업을 수행할 수 없습니다. 사용자에게 사용자가 필요한 리소스에서 작업을 수행할 권한을 부여하려면 IAM 관리자가 IAM 정책을 생성하면 됩니다. 그런 다음 관리자가 IAM 정책을 역할에 추가하고, 사용자가 역할을 맡을 수 있습니다.

이러한 예제 JSON 정책 문서를 사용하여 IAM ID 기반 정책을 생성하는 방법을 알아보려면 IAM 사용 설명서의 IAM 정책 생성을 참조하세요.

각 리소스 유형에 대한 ARN 형식을 포함하여 Amazon FSx에서 정의한 작업 및 리소스 유형에 대한 자세한 내용은 서비스 인증 참조에서 Amazon FSx for Lustre에 대한 작업, 리소스 및 조건 키를 참조하세요.

주제

- 정책 모범 사례
- Amazon FSx 콘솔 사용
- 사용자가 자신의 고유한 권한을 볼 수 있도록 허용

정책 모범 사례

ID 기반 정책에 따라 계정에서 사용자가 Amazon FSx 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 정책으로 시작하고 최소 권한 권한으로 이동 — 사용자와 워크로드에 권한을 부여하려면 여러 일반적인 사용 사례에 권한을 부여하는 AWS 관리형 정책을 사용하세요. 해당 내용은 에서 사용할 수 있습니다. AWS 계정사용 사례에 맞는 AWS 고객 관리형 정책을 정의하여 권한을 더 줄이는 것이 좋습니다. 자세한 정보는 IAM 사용 설명서의 AWS managed policies(관리형 정책) 또는 AWS managed policies for job functions(직무에 대한 관리형 정책)를 참조하세요.
- 최소 권한 적용 – IAM 정책을 사용하여 권한을 설정하는 경우 태스크를 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 Policies and permissions in IAM(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. 예를 AWS 서비스들어 특정 작업을 통해 서비스 작업을 사용하는 경우 조건을 사용하여 서비스 작업에 대한 액세스 권한을 부여할 수도 AWS CloudFormation있습니다. 자세한 정보는 IAM 사용 설명서의 IAM JSON 정책 요소: 조건을 참조하세요.

- IAM Access Analyzer를 통해 IAM 정책을 검증하여 안전하고 기능적인 권한 보장 – IAM Access Analyzer에서는 IAM 정책 언어(JSON)와 모범 사례가 정책에서 준수되도록 신규 및 기존 정책을 검증합니다. IAM Access Analyzer는 100개 이상의 정책 확인 항목과 실행 가능한 추천을 제공하여 안전하고 기능적인 정책을 작성하도록 돕습니다. 자세한 정보는 IAM 사용 설명서의 IAM Access Analyzer 정책 검증을 참조하tpdy.
- 멀티 팩터 인증 (MFA) 필요 - IAM 사용자 또는 루트 사용자가 필요한 시나리오가 있는 경우 추가 보안을 위해 AWS 계정 MFA를 활성화하십시오. API 작업을 직접 호출할 때 MFA가 필요하면 정책에 MFA 조건을 추가합니다. 자세한 정보는 IAM 사용 설명서의 Configuring MFA-protected API access(MFA 보호 API 액세스 구성)를 참조하세요.

IAM의 모범 사례에 대한 자세한 내용은 IAM 사용 설명서의 IAM의 보안 모범 사례를 참조하세요.

Amazon FSx 콘솔 사용

Amazon FSx for Lustre 콘솔에 액세스하려면 최소한의 권한 집합이 있어야 합니다. 이러한 권한을 통해 내 Amazon FSx 리소스를 나열하고 해당 리소스에 대한 세부 정보를 볼 수 있어야 합니다. AWS 계정최소 필수 권한보다 더 제한적인 자격 증명 기반 정책을 만들면 콘솔이 해당 정책에 연결된 엔터티(사용자 또는 역할)에 대해 의도대로 작동하지 않습니다.

AWS CLI 또는 API만 호출하는 사용자에게 최소 콘솔 권한을 허용할 필요는 없습니다. AWS 그 대신, 수행하려는 API 작업과 일치하는 작업에만 액세스할 수 있도록 합니다.

사용자와 역할이 Amazon FSx 콘솔을 계속 사용할 수 있도록 하려면 관리형 정책도 `AmazonFSxConsoleReadOnlyAccess` AWS 엔티티에 연결하십시오. 자세한 내용은 IAM 사용 설명서의 사용자에게 권한 추가를 참조하세요.

`AmazonFSxConsoleReadOnlyAccess` 및 기타 Amazon FSx 관리 서비스 정책은 AWS 아마존 FSx에 대한 관리형 정책 섹션에서 확인할 수 있습니다.

사용자가 자신의 고유한 권한을 볼 수 있도록 허용

이 예시는 IAM 사용자가 자신의 사용자 자격 증명에 연결된 인라인 및 관리형 정책을 볼 수 있도록 허용하는 정책을 생성하는 방법을 보여줍니다. 이 정책에는 콘솔에서 또는 API를 사용하여 프로그래밍 방식으로 이 작업을 완료할 수 있는 AWS CLI 권한이 포함됩니다. AWS

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

```
            "Effect": "Allow",
            "Action": [
                "iam:GetUserPolicy",
                "iam:ListGroupsForUser",
                "iam:ListAttachedUserPolicies",
                "iam:ListUserPolicies",
                "iam:GetUser"
            ],
            "Resource": ["arn:aws:iam::*:user/${aws:username}"]
        },
        {
            "Sid": "NavigateInConsole",
            "Effect": "Allow",
            "Action": [
                "iam:GetGroupPolicy",
                "iam:GetPolicyVersion",
                "iam:GetPolicy",
                "iam:ListAttachedGroupPolicies",
                "iam:ListGroupPolicies",
                "iam:ListPolicyVersions",
                "iam:ListPolicies",
                "iam:ListUsers"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS 아마존 FSx에 대한 관리형 정책

AWS 관리형 정책은 에서 만들고 관리하는 독립형 정책입니다. AWS AWS 관리형 정책은 많은 일반 사용 사례에 대한 권한을 제공하도록 설계되었으므로 사용자, 그룹 및 역할에 권한을 할당하기 시작할 수 있습니다.

AWS 관리형 정책은 모든 AWS 고객이 사용할 수 있으므로 특정 사용 사례에 대해 최소 권한 권한을 부여하지 않을 수도 있다는 점에 유의하세요. 사용 사례에 고유한 고객 관리형 정책을 정의하여 권한을 줄이는 것이 좋습니다.

관리형 정책에 정의된 권한은 변경할 수 없습니다. AWS AWS 관리형 정책에 정의된 권한을 업데이트하는 경우 AWS 해당 업데이트는 정책이 연결된 모든 주체 ID (사용자, 그룹, 역할) 에 영향을 미칩니다. AWS 새 API 작업이 시작되거나 기존 서비스에 새 AWS 서비스 API 작업을 사용할 수 있게 되면 AWS 관리형 정책을 업데이트할 가능성이 가장 높습니다.

자세한 내용은 IAM 사용자 설명서의 AWS 관리형 정책을 참조하세요.

AmazonF SxServiceRolePolicy

Amazon FSx가 사용자를 대신하여 리소스를 AWS 관리할 수 있도록 허용합니다. 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요.

AWS 관리형 정책: AmazonF SxDeleteServiceLinkedRoleAccess

`AmazonFSxDeleteServiceLinkedRoleAccess`를 IAM 엔터티에 연결할 수 없습니다. 이 정책은 서비스에 연결되어 있으며 해당 서비스에 대한 서비스 연결 역할에서만 사용됩니다. 이 정책은 연결, 분리, 수정 또는 삭제할 수 없습니다. 자세한 정보는 Amazon FSx에 대해 서비스 연결 역할 사용을 참조하세요.

이 정책은 Amazon FSx가 Amazon FSx for Lustre에서만 사용하는 Amazon S3 액세스에 대한 서비스 연결 역할을 삭제할 수 있도록 허용하는 관리자 권한을 부여합니다.

권한 세부 정보

이 정책에는 Amazon FSx가 Amazon S3 `iam` 액세스를 위한 FSx 서비스 연결 역할에 대한 삭제 상태를 보고, 삭제하고, 볼 수 있도록 허용하는 권한이 포함되어 있습니다.

이 정책에 대한 권한을 보려면 관리형 정책 참조 안내서의 AmazonF를 참조하십시오 SxDeleteServiceLinkedRoleAccess. AWS

AWS 관리형 정책: AmazonF SxFullAccess

Amazon F를 IAM SxFullAccess 엔티티에 연결할 수 있습니다. Amazon FSx는 사용자를 대신하여 Amazon FSx가 작업을 수행할 수 있도록 허용하는 서비스 역할에도 이 정책을 연결합니다.

Amazon FSx에 대한 전체 액세스 권한 및 관련 서비스에 대한 액세스를 제공합니다. AWS

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `fsx` - 보안 주체가 `BypassSnaplockEnterpriseRetention`을 제외한 모든 Amazon FSx 작업을 수행할 수 있습니다.
- `ds`— 주도자가 디렉터리에 대한 정보를 볼 수 있습니다. AWS Directory Service
- `ec2`

  - 주도자가 지정된 조건에서 태그를 생성할 수 있습니다.
  - VPC와 함께 사용할 수 있는 모든 보안 그룹에 대한 향상된 보안 그룹 검증을 제공합니다.
- `iam` - 보안 주체가 사용자를 대신하여 Amazon FSx 서비스 연결 역할을 생성할 수 있습니다. 이는 Amazon FSx가 사용자를 대신하여 리소스를 AWS 관리할 수 있도록 하기 위해 필요합니다.
- `logs` - 보안 주체가 로그 그룹, 로그 스트림을 생성하고, 로그 스트림에 이벤트를 기록할 수 있습니다. 이는 사용자가 감사 액세스 로그를 로그로 전송하여 FSx for Windows File Server 파일 시스템 액세스를 모니터링할 수 있도록 하기 위해 필요합니다. CloudWatch
- `firehose`— 주체가 Amazon Data Firehose에 레코드를 쓸 수 있도록 허용합니다. 이는 사용자가 Firehose에 감사 액세스 로그를 전송하여 FSx for Windows File Server 파일 시스템 액세스를 모니터링할 수 있도록 하기 위해 필요합니다.

이 정책에 대한 권한을 보려면 관리형 정책 참조 SxFullAccess 가이드의 AWS AmazonF를 참조하십시오.

AWS 관리형 정책: AmazonF SxConsoleFullAccess

`AmazonFSxConsoleFullAccess` 정책을 IAM 보안 인증에 연결할 수 있습니다.

이 정책은 Amazon FSx에 대한 전체 액세스 및 를 통한 AWS 관련 서비스 액세스를 허용하는 관리자 권한을 부여합니다. AWS Management Console

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `fsx` - 보안 주체가 Amazon FSx 관리 콘솔에서 `BypassSnaplockEnterpriseRetention`을 제외한 모든 작업을 수행할 수 있습니다.
- `cloudwatch`— 보안 주체가 Amazon FSx 관리 콘솔에서 CloudWatch 경보 및 지표를 볼 수 있습니다.
- `ds`— 보안 주체가 디렉터리에 대한 정보를 나열할 수 있습니다. AWS Directory Service
- `ec2`
  - 보안 주체가 라우팅 테이블에 태그를 생성하고, 네트워크 인터페이스, 라우팅 테이블, 보안 그룹, 서브넷 및 Amazon FSx 파일 시스템과 연결된 VPC를 나열할 수 있습니다.
  - 보안 주체가 VPC와 함께 사용할 수 있는 모든 보안 그룹에 대해 향상된 보안 그룹 검증을 제공할 수 있도록 허용합니다.
- `kms`— 보안 주체가 키의 별칭을 나열할 수 있도록 허용합니다. AWS Key Management Service

- `s3` - 보안 주체가 Amazon S3 버킷의 일부 또는 모든 객체를 나열할 수 있습니다(최대 1000개).
- `iam` - Amazon FSx가 사용자를 대신하여 작업을 수행할 수 있도록 허용하는 서비스 연결 역할을 생성할 권한을 부여합니다.

이 정책에 대한 권한을 보려면 관리형 정책 참조 안내서의 AmazonF를 SxConsoleFullAccess 참조하십시오. AWS

AWS 관리형 정책: AmazonF SxConsoleReadOnlyAccess

`AmazonFSxConsoleReadOnlyAccess` 정책을 IAM 보안 인증에 연결할 수 있습니다.

이 정책은 Amazon FSx 및 AWS 관련 서비스에 읽기 전용 권한을 부여하여 사용자가 에서 이러한 서비스에 대한 정보를 볼 수 있도록 합니다. AWS Management Console

권한 세부 정보

이 정책에는 다음 권한이 포함되어 있습니다.

- `fsx`- 보안 주체가 Amazon FSx 관리 콘솔에서 모든 태그를 비롯하여 Amazon FSx 파일 시스템에 대한 정보를 볼 수 있습니다.
- `cloudwatch`— 보안 주체가 Amazon FSx 관리 콘솔에서 CloudWatch 경보 및 지표를 볼 수 있습니다.
- `ds`— 보안 주체가 Amazon FSx 관리 콘솔의 AWS Directory Service 디렉터리에 대한 정보를 볼 수 있습니다.
- `ec2`
  - 보안 주체가 Amazon FSx 관리 콘솔에서 Amazon FSx 파일 시스템과 연결된 네트워크 인터페이스, 보안 그룹, 서브넷 및 VPC를 볼 수 있습니다.
  - VPC와 함께 사용할 수 있는 모든 보안 그룹에 대한 향상된 보안 그룹 검증을 제공합니다.
- `kms`— 보안 주체가 Amazon FSx 관리 콘솔에서 AWS Key Management Service 키의 별칭을 볼 수 있습니다.
- `log`— 보안 주체가 요청한 계정과 관련된 Amazon CloudWatch Logs 로그 그룹을 설명할 수 있습니다. 이는 보안 주체가 FSx for Windows File Server 파일 시스템에 대한 기존 파일 액세스 감사 구성을 볼 수 있도록 하기 위해 필요합니다.
- `firehose`— 주체가 요청하는 계정과 관련된 Amazon Data Firehose 전송 스트림을 설명할 수 있습니다. 이는 보안 주체가 FSx for Windows File Server 파일 시스템에 대한 기존 파일 액세스 감사 구성을 볼 수 있도록 하기 위해 필요합니다.

이 정책에 대한 권한을 보려면 관리형 정책 참조 SxConsoleReadOnlyAccess 가이드의 AWS AmazonF를 참조하십시오.

AWS 관리형 정책: AmazonF SxReadOnlyAccess

`AmazonFSxReadOnlyAccess` 정책을 IAM 보안 인증에 연결할 수 있습니다.

이 정책에는 다음 권한이 포함되어 있습니다.

- `fsx`- 보안 주체가 Amazon FSx 관리 콘솔에서 모든 태그를 비롯하여 Amazon FSx 파일 시스템에 대한 정보를 볼 수 있습니다.
- `ec2`— VPC와 함께 사용할 수 있는 모든 보안 그룹에 대한 향상된 보안 그룹 검증을 제공합니다.

이 정책에 대한 권한을 보려면 AWS 관리형 정책 참조 SxReadOnlyAccess 가이드의 AmazonF를 참조하십시오.

관리형 정책에 대한 Amazon FSx 업데이트 AWS

이 서비스가 변경 사항을 추적하기 시작한 이후 Amazon FSx의 AWS 관리형 정책 업데이트에 대한 세부 정보를 확인하십시오. 이 페이지의 변경 사항에 대한 자동 알림을 받아보려면 Amazon FSx 문서 이력 페이지에서 RSS 피드를 구독하세요.

변경 사항	설명	날짜
AmazonF SxServiceRolePolicy — 기존 정책에 대한 업데이트	Amazon FSx는 보안 주체가 VPC와 함께 사용할 수 있는 모든 보안 그룹에 대해 향상된 보안 그룹 검증을 제공할 수 있는 새로운 권한을 `ec2:GetSecurityGroupsForVpc` 추가했습니다.	2024년 1월 9일
AmazonF SxReadOnlyAccess — 기존 정책 업데이트	Amazon FSx는 보안 주체가 VPC와 함께 사용할 수 있는 모든 보안 그룹에 대해 향상된 보안 그룹 검증을 제공할 수 있는 새로운 권한을 `ec2:GetSe`	2024년 1월 9일

변경 사항	설명	날짜
	curityGroupsForVpc 추가했습니다.	
AmazonF SxConsole ReadOnlyAccess — 기존 정책 업데이트	Amazon FSx는 보안 주체가 VPC와 함께 사용할 수 있는 모든 보안 그룹에 대해 향상된 보안 그룹 검증을 제공할 수 있는 새로운 권한을 ec2:GetSecurityGroupsForVpc 추가했습니다.	2024년 1월 9일
AmazonF SxFullAccess — 기존 정책 업데이트	Amazon FSx는 보안 주체가 VPC와 함께 사용할 수 있는 모든 보안 그룹에 대해 향상된 보안 그룹 검증을 제공할 수 있는 새로운 권한을 ec2:GetSecurityGroupsForVpc 추가했습니다.	2024년 1월 9일
AmazonF SxConsole FullAccess — 기존 정책 업데이트	Amazon FSx는 보안 주체가 VPC와 함께 사용할 수 있는 모든 보안 그룹에 대해 향상된 보안 그룹 검증을 제공할 수 있는 새로운 권한을 ec2:GetSecurityGroupsForVpc 추가했습니다.	2024년 1월 9일
AmazonF SxFullAccess — 기존 정책 업데이트	Amazon FSx는 사용자가 OpenZFS용 FSX에 대해 지역 간 및 계정 간 데이터 복제를 수행할 수 있는 새로운 권한을 추가했습니다.	2023년 12월 20일

변경 사항	설명	날짜
SxConsoleFullAccesssAmazonF — 기존 정책에 대한 업데이트	Amazon FSx는 사용자가 OpenZFS용 FSX에 대해 지역 간 및 계정 간 데이터 복제를 수행할 수 있는 새로운 권한을 추가했습니다.	2023년 12월 20일
SxFullAccessAmazonF — 기존 정책에 대한 업데이트	Amazon FSx는 사용자가 OpenZFS 파일 시스템용 FSX용 볼륨의 온디맨드 복제를 수행할 수 있도록 하는 새로운 권한을 추가했습니다.	2023년 11월 26일
AmazonF — 기존 정책에 대한 업데이트 SxConsoleFullAccess	Amazon FSx는 사용자가 OpenZFS 파일 시스템용 FSX용 볼륨의 온디맨드 복제를 수행할 수 있도록 하는 새로운 권한을 추가했습니다.	2023년 11월 26일
AmazonF — 기존 정책에 대한 업데이트 SxFullAccess	Amazon FSx는 사용자가 ONTAP 다중 AZ 파일 시스템용 FSx에 대한 공유 VPC 지원을 보고, 활성화하고, 비활성화할 수 있도록 하는 새로운 권한을 추가했습니다.	2023년 11월 14일
AmazonF — 기존 정책에 대한 업데이트 SxConsoleFullAccess	Amazon FSx는 사용자가 ONTAP 다중 AZ 파일 시스템용 FSx에 대한 공유 VPC 지원을 보고, 활성화하고, 비활성화할 수 있도록 하는 새로운 권한을 추가했습니다.	2023년 11월 14일

변경 사항	설명	날짜
AmazonF — 기존 정책에 대한 업데이트 SxFullAccess	Amazon FSx가 FSx for OpenZFS 다중 AZ 파일 시스템의 네트워크 구성을 관리할 수 있도록 하는 새로운 권한을 추가했습니다.	2023년 8월 9일
AWS 관리형 정책: AmazonF SxServiceRolePolicy — 기존 정책에 대한 업데이트	Amazon FSx는 Amazon FSx가 네임스페이스에 메트릭을 게시하도록 `cloudwatch:PutMetricData` 기존 권한을 수정했습니다. CloudWatch `AWS/FSx`	2023년 7월 24일
AmazonF — 기존 정책에 대한 업데이트 SxFullAccess	Amazon FSx의 `fsx:*` 권한을 제거하고 특정 `fsx` 작업을 추가하도록 정책을 업데이트했습니다.	2023년 7월 13일
AmazonF SxConsole FullAccess — 기존 정책에 대한 업데이트	Amazon FSx의 `fsx:*` 권한을 제거하고 특정 `fsx` 작업을 추가하도록 정책을 업데이트했습니다.	2023년 7월 13일
AmazonF SxConsole ReadOnlyAccess — 기존 정책에 대한 업데이트	사용자가 Amazon FSx 콘솔에서 FSx for Windows File Server 파일 시스템에 대한 향상된 성능 지표와 권장 조치를 볼 수 있도록 하는 새로운 권한을 추가했습니다.	2022년 9월 21일
AmazonF SxConsole FullAccess — 기존 정책에 대한 업데이트	사용자가 Amazon FSx 콘솔에서 FSx for Windows File Server 파일 시스템에 대한 향상된 성능 지표와 권장 조치를 볼 수 있도록 하는 새로운 권한을 추가했습니다.	2022년 9월 21일

변경 사항	설명	날짜
AmazonF — 정책 추적 시작 SxReadOnlyAccess	이 정책은 모든 Amazon FSx 리소스 및 이와 관련된 모든 태그에 대한 읽기 전용 액세스 권한을 부여합니다.	2022년 2월 4일
AmazonF — 추적 정책 시작 SxDeleteServiceLinkedRoleAccess	이 정책은 Amazon FSx가 Amazon S3 액세스에 대한 서비스 연결 역할을 삭제할 수 있도록 허용하는 관리자 권한을 부여합니다.	2022년 1월 7일
AmazonF SxServiceRolePolicy — 기존 정책에 대한 업데이트	Amazon FSx는 Amazon FSx가 ONTAP 파일 시스템용 Amazon FSx의 네트워크 구성을 관리할 수 있도록 하는 새로운 권한을 추가했습니다. NetApp	2021년 9월 2일
AmazonF — 기존 정책에 대한 업데이트 SxFullAccess	Amazon FSx가 EC2 라우팅 테이블에서 범위를 좁힌 호출에 대한 태그를 생성할 수 있도록 하는 새로운 권한을 추가했습니다.	2021년 9월 2일
AmazonF SxConsole FullAccess — 기존 정책에 대한 업데이트	Amazon FSx는 Amazon FSx가 ONTAP 다중 AZ 파일 시스템용 Amazon FSx를 생성할 수 있도록 하는 새로운 권한을 추가했습니다. NetApp	2021년 9월 2일
SxConsoleFullAccessAmazonF — 기존 정책에 대한 업데이트	Amazon FSx가 EC2 라우팅 테이블에서 범위를 좁힌 호출에 대한 태그를 생성할 수 있도록 하는 새로운 권한을 추가했습니다.	2021년 9월 2일

변경 사항	설명	날짜
AmazonF SxServiceRolePolicy — 기존 정책에 대한 업데이트	Amazon FSx는 Amazon FSx가 로그 로그 스트림을 설명하고 이에 쓸 수 있도록 새 권한을 추가했습니다. CloudWatch 이는 사용자가 로그를 사용하여 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사 로그를 볼 수 있도록 하기 위해 필요합니다. CloudWatch	2021년 6월 8일
AmazonF SxServiceRolePolicy — 기존 정책에 대한 업데이트	Amazon FSx는 Amazon FSx가 Amazon Data Firehose 전송 스트림을 설명하고 이에 쓸 수 있도록 새 권한을 추가했습니다. 이는 사용자가 Amazon Data Firehose를 사용하여 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사 로그를 볼 수 있도록 하기 위해 필요합니다.	2021년 6월 8일

변경 사항	설명	날짜
AmazonF SxFullAccess — 기존 정책에 대한 업데이트	Amazon FSx는 보안 주체가 로그 로그 그룹, 로그 스트림을 설명하고 CloudWatch 생성하고 로그 스트림에 이벤트를 쓸 수 있도록 하는 새로운 권한을 추가했습니다. 이는 주도자가 로그를 사용하여 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사 로그를 볼 수 있도록 하기 위해 필요합니다. CloudWatch	2021년 6월 8일
AmazonF — 기존 정책에 대한 업데이트 SxFullAccess	Amazon FSx는 보안 주체가 Amazon Data Firehose에 레코드를 설명하고 기록할 수 있는 새로운 권한을 추가했습니다. 이는 사용자가 Amazon Data Firehose를 사용하여 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사 로그를 볼 수 있도록 하기 위해 필요합니다.	2021년 6월 8일

변경 사항	설명	날짜
AmazonF SxConsole FullAccess — 기존 정책에 대한 업데이트	Amazon FSx는 보안 주체가 요청한 계정과 관련된 CloudWatch Amazon Logs 로그 그룹을 설명할 수 있는 새로운 권한을 추가했습니다. 이는 주도자가 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사를 구성할 때 기존 CloudWatch 로그 로그 그룹을 선택할 수 있도록 하기 위해 필요합니다.	2021년 6월 8일
AmazonF — 기존 정책에 SxConsoleFullAccess 대한 업데이트	Amazon FSx는 보안 주체가 요청한 계정과 관련된 Amazon Data Firehose 전송 스트림을 설명할 수 있는 새로운 권한을 추가했습니다. 이는 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사를 구성할 때 보안 주체가 기존 Firehose 전송 스트림을 선택할 수 있도록 하기 위해 필요합니다.	2021년 6월 8일

변경 사항	설명	날짜
AmazonF — 기존 정책에 대한 업데이트 SxConsole ReadOnlyAccess	Amazon FSx는 보안 주체가 요청한 계정과 관련된 CloudWatch Amazon Logs 로그 그룹을 설명할 수 있는 새로운 권한을 추가했습니다. 이는 보안 주체가 FSx for Windows File Server 파일 시스템에 대한 기존 파일 액세스 감사 구성을 볼 수 있도록 하기 위해 필요합니다.	2021년 6월 8일
AmazonF — 기존 정책에 대한 업데이트 SxConsole ReadOnlyAccess	Amazon FSx는 보안 주체가 요청한 계정과 관련된 Amazon Data Firehose 전송 스트림을 설명할 수 있는 새로운 권한을 추가했습니다. 이는 보안 주체가 FSx for Windows File Server 파일 시스템에 대한 기존 파일 액세스 감사 구성을 볼 수 있도록 하기 위해 필요합니다.	2021년 6월 8일
Amazon FSx에서 변경 사항 추적 시작	Amazon FSx는 관리형 정책의 변경 사항을 추적하기 시작했습니다. AWS	2021년 6월 8일

Amazon FSx for Lustre 자격 증명 및 액세스 문제 해결

다음 정보를 사용하여 Amazon FSx 및 IAM으로 작업할 때 발생할 수 있는 일반적인 문제를 진단하고 수정할 수 있습니다.

주제

- Amazon FSx에서 작업을 수행할 권한이 없음

- 저는 IAM을 수행할 권한이 없습니다. PassRole
- 외부 사용자가 내 Amazon FSx 리소스에 액세스할 AWS 계정 수 있도록 허용하고 싶습니다.

Amazon FSx에서 작업을 수행할 권한이 없음

작업을 수행할 권한이 없다는 오류가 수신되면, 작업을 수행할 수 있도록 정책을 업데이트해야 합니다.

다음 예제 오류는 mateojacksonIAM 사용자가 콘솔을 사용하여 가상 *my-example-widget*리소스에 대한 세부 정보를 보려고 하지만 가상 fsx:*GetWidget*권한이 없을 때 발생합니다.

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
 fsx:GetWidget on resource: my-example-widget
```

이 경우 fsx:*GetWidget*작업을 사용하여 *my-example-widget*리소스에 액세스할 수 있도록 mateojackson사용자 정책을 업데이트해야 합니다.

도움이 필요한 경우 AWS 관리자에게 문의하세요. 관리자는 로그인 자격 증명을 제공한 사람입니다.

저는 IAM을 수행할 권한이 없습니다. PassRole

iam:PassRole 작업을 수행할 수 있는 권한이 없다는 오류가 수신되면 Amazon FSx에 역할을 전달할 수 있도록 정책을 업데이트해야 합니다.

새 서비스 역할 또는 서비스 연결 역할을 만드는 대신 기존 역할을 해당 서비스에 전달할 AWS 서비스 수 있는 기능도 있습니다. 이렇게 하려면 사용자가 서비스에 역할을 전달할 수 있는 권한을 가지고 있어야 합니다.

다음 예제 오류는 marymajor라는 IAM 사용자가 콘솔을 사용하여 Amazon FSx에서 작업을 수행하려고 하는 경우에 발생합니다. 하지만 작업을 수행하려면 서비스 역할이 부여한 권한이 서비스에 있어야 합니다. Mary는 서비스에 역할을 전달할 수 있는 권한을 가지고 있지 않습니다.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

이 경우 Mary가 iam:PassRole작업을 수행할 수 있도록 Mary의 정책을 업데이트해야 합니다.

도움이 필요하면 관리자에게 문의하세요. AWS 관리자는 로그인 자격 증명을 제공한 사람입니다.

외부 사용자가 내 Amazon FSx 리소스에 액세스할 AWS 계정 수 있도록 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.

자세히 알아보려면 다음을 참조하세요.

- Amazon FSx에서 이러한 기능을 지원하는지 여부를 알아보려면 Amazon FSx for Lustre가 IAM과 작동하는 방식 섹션을 참조하세요.
- 소유하고 AWS 계정 있는 모든 리소스에 대한 액세스를 제공하는 방법을 알아보려면 IAM 사용 설명서의 다른 AWS 계정 IAM 사용자에게 액세스 권한 제공을 참조하십시오.
- 제3자에게 리소스에 대한 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 타사 AWS 계정AWS 계정 소유에 대한 액세스 제공을 참조하십시오.
- ID 페더레이션을 통해 액세스 권한을 제공하는 방법을 알아보려면 IAM 사용 설명서의 외부에서 인증된 사용자에게 액세스 권한 제공(자격 증명 연동)을 참조하세요.
- 크로스 계정 액세스를 위한 역할과 리소스 기반 정책 사용의 차이점을 알아보려면 IAM 사용 설명서의 IAM 역할과 리소스 기반 정책의 차이를 참조하세요.

Amazon FSx에서 태그 사용

태그를 사용하여 Amazon FSx 리소스에 대한 액세스를 제어하고 ABAC(속성 기반 액세스 제어)를 구현할 수 있습니다. 생성 중에 Amazon FSx 리소스에 태그를 적용하려면 사용자에게 AWS Identity and Access Management(IAM) 권한이 있어야 합니다.

생성 시 리소스 태그 지정에 대한 권한 부여

일부 리소스 생성 Amazon FSx for Lustre API 작업에서는 리소스를 생성할 때 태그를 지정할 수 있습니다. 리소스 태그를 사용하여 속성 기반 액세스 제어(ABAC)를 구현할 수도 있습니다. 자세한 내용은 IAM 사용 설명서에서 AWS에 대한 속성 기반 액세스 제어란 무엇인가요?를 참조하세요.

사용자가 생성 시 리소스에 태그를 지정할 수 있으려면 `fsx:CreateFileSystem` 같은 리소스를 생성하는 작업을 사용할 권한이 있어야 합니다. 리소스 생성 작업에서 태그가 지정되면 IAM은 `fsx:TagResource` 작업에서 추가 권한 부여를 수행해 사용자에게 태그를 생성할 권한이 있는지 확인합니다. 따라서 사용자는 `fsx:TagResource` 작업을 사용할 명시적 권한도 가지고 있어야 합니다.

다음 예제의 정책은 사용자가 특정 AWS 계정에서 파일 시스템을 생성하고 생성 도중 태그를 적용하는 것을 허용합니다.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "fsx:CreateFileSystem",
         "fsx:TagResource"
      ],
      "Resource": [
         "arn:aws:fsx:region:account-id:file-system/*"
      ]
    }
  ]
}
```

마찬가지로 다음 정책은 사용자가 특정 파일 시스템에 백업을 생성하고 백업 생성 도중 백업에 임의의 태그를 적용하는 것을 허용합니다.

```
{
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
         "fsx:CreateBackup"
      ],
      "Resource": "arn:aws:fsx:region:account-id:file-system/file-system-id*"
    },
    {
      "Effect": "Allow",
      "Action": [
         "fsx:TagResource"
      ],
      "Resource": "arn:aws:fsx:region:account-id:backup/*"
    }
  ]
}
```

`fsx:TagResource` 작업은 리소스 생성 작업 도중 태그가 적용되는 경우에만 평가됩니다. 따라서 리소스를 생성할 권한이 있는 사용자(태그 지정 조건은 없다고 가정)는 요청에서 태그가 지정되지 않은

경우, `fsx:TagResource` 작업을 사용할 권한이 필요하지 않습니다. 하지만 사용자가 태그를 사용하여 리소스 생성을 시도하는 경우, 사용자에게 `fsx:TagResource` 작업을 사용할 권한이 없다면 요청은 실패합니다.

Amazon FSx 리소스 태그 지정에 대한 자세한 내용은 Amazon FSx 리소스 태그 지정 섹션을 참조하세요. 태그를 사용하여 Amazon FSx for Lustre 리소스에 대한 액세스를 제어하는 방법에 대한 자세한 내용은 태그를 사용하여 Amazon FSx 리소스에 대한 액세스 제어 섹션을 참조하세요.

태그를 사용하여 Amazon FSx 리소스에 대한 액세스 제어

Amazon FSx 리소스 및 작업에 대한 액세스를 제어하려면 태그를 기반으로 IAM 정책을 사용할 수 있습니다. 두 가지 방법으로 제어할 수 있습니다.

- Amazon FSx 리소스의 태그를 기반으로 리소스에 대한 액세스를 제어할 수 있습니다.
- IAM 요청 조건에 어떤 태그가 전달될 수 있는지를 제어할 수 있습니다.

태그를 사용하여 AWS 리소스에 대한 액세스를 제어하는 방법에 대한 자세한 내용은 IAM 사용 설명서에서 태그를 사용해 액세스 제어 섹션을 참조하세요. 생성 시 Amazon FSx 리소스 태그 지정에 대한 자세한 내용은 생성 시 리소스 태그 지정에 대한 권한 부여 섹션을 참조하세요. 리소스 태그 지정에 대한 자세한 내용은 Amazon FSx 리소스 태그 지정 섹션을 참조하세요.

리소스의 태그를 기반으로 액세스 제어

사용자나 역할이 Amazon FSx 리소스에서 어떤 작업을 수행할 수 있는지를 제어하기 위해 리소스의 태그를 사용할 수 있습니다. 예를 들어, 리소스에 있는 태그의 키-값 페어를 기반으로 파일 시스템 리소스에서 특정 API 작업을 허용하거나 거부할 수 있습니다.

Example 예제 정책 - 특정 태그를 제공할 때 파일 시스템 생성

해당 예시에서 이 정책은 사용자가 `key=Department, value=Finance` 같은 특정 태그 키 값 쌍으로 태그를 지정하는 경우에만 파일 시스템을 생성할 수 있도록 허용합니다.

```
{
    "Effect": "Allow",
    "Action": [
        "fsx:CreateFileSystem",
        "fsx:TagResource"
    ],
    "Resource": "arn:aws:fsx:region:account-id:file-system/*",
```

```
    "Condition": {
        "StringEquals": {
            "aws:RequestTag/Department": "Finance"
        }
    }
}
```

Example 예제 정책 - 특정 태그가 있는 파일 시스템에만 백업 생성

이 정책은 사용자가 키 값 쌍으로 key=Department, value=Finance 태그가 지정된 파일 시스템에만 백업을 생성할 수 있도록 허용하며, 백업은 Deparment=Finance 태그를 사용하여 생성됩니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateBackup"
            ],
            "Resource": "arn:aws:fsx:region:account-id:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:TagResource",
                "fsx:CreateBackup"
            ],
            "Resource": "arn:aws:fsx:region:account-id:backup/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        }

    ]
}
```

Example 예제 정책 - 특정 태그가 있는 백업에서 특정 태그를 사용하여 파일 시스템 생성

이 정책을 통해 사용자는 Department=Finance 태그가 지정된 백업에서만 Department=Finance 태그가 지정된 파일 시스템을 생성할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateFileSystemFromBackup",
                "fsx:TagResource"
            ],
            "Resource": "arn:aws:fsx:region:account-id:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateFileSystemFromBackup"
            ],
            "Resource": "arn:aws:fsx:region:account-id:backup/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Example 예제 정책 - 특정 태그가 있는 파일 시스템 삭제

이 정책은 사용자가 Department=Finance 태그가 지정된 파일 시스템만 삭제하도록 허용합니다. 최종 백업을 생성하는 경우 Department=Finance 태그를 지정해야 합니다. Lustre 파일 시스템의 경우 사용자에게 최종 백업을 생성할 수 있는 fsx:CreateBackup 권한이 필요합니다.

```
{
```

```
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "fsx:DeleteFileSystem"
            ],
            "Resource": "arn:aws:fsx:region:account-id:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateBackup",
                "fsx:TagResource"
            ],
            "Resource": "arn:aws:fsx:region:account-id:backup/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Example 예제 정책 - 특정 태그가 있는 파일 시스템에 데이터 리포지토리 작업 생성

이 정책을 통해 사용자는 `Department=Finance` 태그가 지정된 데이터 리포지토리 작업을 생성하고 `Department=Finance` 태그가 지정된 파일 시스템에만 생성할 수 있습니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateDataRepositoryTask"
            ],
```

```
            "Resource": "arn:aws:fsx:region:account-id:file-system/*",
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/Department": "Finance"
                }
            }
        },
        {
            "Effect": "Allow",
            "Action": [
                "fsx:CreateDataRepositoryTask",
                "fsx:TagResource"
            ],
            "Resource": "arn:aws:fsx:region:account-id:task/*",
            "Condition": {
                "StringEquals": {
                    "aws:RequestTag/Department": "Finance"
                }
            }
        }
    ]
}
```

Amazon FSx에 대해 서비스 연결 역할 사용

Amazon AWS Identity and Access Management FSx는 (IAM) 서비스 연결 역할을 사용합니다. 서비스 연결 역할은 Amazon FSx에 직접 연결된 고유한 유형의 IAM 역할입니다. 서비스 연결 역할은 Amazon FSx에서 사전 정의하며 서비스가 사용자를 대신하여 다른 서비스를 호출하는 데 필요한 모든 권한을 포함합니다. AWS

필요한 권한을 수동으로 추가할 필요가 없으므로 서비스 연결 역할로 Amazon FSx를 더 쉽게 설정할 수 있습니다. Amazon FSx에서 서비스 연결 역할의 권한을 정의하므로 다르게 정의되지 않은 한, Amazon FSx만 해당 역할을 수임할 수 있습니다. 정의된 권한에는 신뢰 정책과 권한 정책이 포함되며 이 권한 정책은 다른 IAM 엔터티에 연결할 수 없습니다.

먼저 관련 리소스를 삭제한 후에만 서비스 연결 역할을 삭제할 수 있습니다. 이렇게 하면 리소스에 대한 액세스 권한을 실수로 삭제할 수 없기 때문에 Amazon FSx 리소스가 보호됩니다.

서비스 연결 역할을 지원하는 기타 서비스에 대한 자세한 내용은 IAM으로 작업하는AWS 서비스를 참조하고 서비스 연결 역할(Service-linked roles) 열에 예(Yes)가 있는 서비스를 찾으십시오. 해당 서비스에 대한 서비스 연결 역할 설명서를 보려면 예 링크를 선택합니다.

Amazon FSx에 대한 서비스 연결 역할 권한

Amazon FSx는 `AWSServiceRoleForFSxS3Access_fs-01234567890` 이름과 계정에서 특정 작업을 수행하는 두 개의 서비스 `AWSServiceRoleForAmazonFSx` 연결 역할을 사용합니다. 이러한 작업의 예로는 VPC에서 파일 시스템을 위한 탄력적인 네트워크 인터페이스를 생성하고 Amazon S3 버킷을 통해 데이터 리포지토리에 액세스하는 것이 있습니다. `AWSServiceRoleForFSxS3Access_fs-01234567890`의 경우, 이 서비스 연결 역할은 S3 버킷에 연결된 사용자가 생성한 각 Amazon FSx for Lustre 파일 시스템에 대해 생성됩니다.

AWSServiceRoleForAmazonFSx 권한 세부 정보

의 경우`AWSServiceRoleForAmazonFSx`, 역할 권한 정책에 따라 Amazon FSx는 사용자를 대신하여 모든 해당 리소스에 대해 다음과 같은 관리 작업을 완료할 수 있습니다. AWS

이 정책에 대한 업데이트는 다음을 참조하십시오. AmazonF SxServiceRolePolicy

> **Note**
>
> AWSServiceRoleForAmazonFSx 는 모든 Amazon FSx 파일 시스템 유형에서 사용됩니다. 나열된 권한 중 일부는 FSx for Lustre에는 적용되지 않습니다.

- `ds`— Amazon FSx가 디렉터리에 있는 애플리케이션을 보고, 권한을 부여하고, 권한 부여를 취소할 수 있도록 허용합니다. AWS Directory Service
- `ec2` - Amazon FSx에서 다음 작업을 수행하도록 허용합니다.
  - Amazon FSx 파일 시스템과 연결된 네트워크 인터페이스를 확인하고, 생성하고, 연결을 해제합니다.
  - Amazon FSx 파일 시스템과 연결된 하나 이상의 탄력적 IP 주소를 확인합니다.
  - Amazon FSx 파일 시스템과 연결된 Amazon VPC, 보안 그룹 및 서브넷을 확인합니다.
  - VPC와 함께 사용할 수 있는 모든 보안 그룹에 대한 향상된 보안 그룹 검증을 제공합니다.
  - 권한이 AWS부여된 사용자가 네트워크 인터페이스에서 특정 작업을 수행할 수 있는 권한을 생성합니다.
- `cloudwatch`— Amazon FSx가 /FSx 네임스페이스 아래에 AWS메트릭 데이터 포인트를 게시할 CloudWatch 수 있도록 허용합니다.
- `route53` - Amazon FSx에서 Amazon VPC를 프라이빗 호스팅 영역과 연결할 수 있도록 허용합니다.

- `logs`— Amazon FSx가 로그 로그 스트림을 설명하고 이에 쓸 수 CloudWatch 있도록 허용합니다. 이는 사용자가 Windows File Server용 FSx 파일 시스템에 대한 파일 액세스 감사 로그를 로그 스트림으로 보낼 수 있도록 CloudWatch 하기 위한 것입니다.
- `firehose`— Amazon FSx가 Amazon Data Firehose 전송 스트림을 설명하고 이에 쓸 수 있도록 허용합니다. 이는 사용자가 Windows용 FSx 파일 시스템에 대한 파일 액세스 감사 로그를 Amazon Data Firehose 전송 스트림에 게시할 수 있도록 하기 위한 것입니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "CreateFileSystem",
            "Effect": "Allow",
            "Action": [
                "ds:AuthorizeApplication",
                "ds:GetAuthorizedApplicationDetails",
                "ds:UnauthorizeApplication",
                "ec2:CreateNetworkInterface",
                "ec2:CreateNetworkInterfacePermission",
                "ec2:DeleteNetworkInterface",
                "ec2:DescribeAddresses",
                "ec2:DescribeDhcpOptions",
                "ec2:DescribeNetworkInterfaces",
                "ec2:DescribeRouteTables",
                "ec2:DescribeSecurityGroups",
                "ec2:DescribeSubnets",
                "ec2:DescribeVPCs",
                "ec2:DisassociateAddress",
                "ec2:GetSecurityGroupsForVpc",
                "route53:AssociateVPCWithHostedZone"
            ],
            "Resource": "*"
        },
        {
            "Sid": "PutMetrics",
            "Effect": "Allow",
            "Action": [
                "cloudwatch:PutMetricData"
            ],
            "Resource": [
                "*"
```

```
            ],
            "Condition": {
                "StringEquals": {
                    "cloudwatch:namespace": "AWS/FSx"
                }
            }
        },

        {
            "Sid": "TagResourceNetworkInterface",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateTags"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*"
            ],
            "Condition": {
                "StringEquals": {
                    "ec2:CreateAction": "CreateNetworkInterface"
                },
                "ForAllValues:StringEquals": {
                    "aws:TagKeys": "AmazonFSx.FileSystemId"
                }
            }
        },
        {
            "Sid": "ManageNetworkInterface",
            "Effect": "Allow",
            "Action": [
                "ec2:AssignPrivateIpAddresses",
                "ec2:ModifyNetworkInterfaceAttribute",
                "ec2:UnassignPrivateIpAddresses"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:network-interface/*"
            ],
            "Condition": {
                "Null": {
                    "aws:ResourceTag/AmazonFSx.FileSystemId": "false"
                }
            }
        },
        {
```

```
            "Sid": "ManageRouteTable",
            "Effect": "Allow",
            "Action": [
                "ec2:CreateRoute",
                "ec2:ReplaceRoute",
                "ec2:DeleteRoute"
            ],
            "Resource": [
                "arn:aws:ec2:*:*:route-table/*"
            ],
            "Condition": {
                "StringEquals": {
                    "aws:ResourceTag/AmazonFSx": "ManagedByAmazonFSx"
                }
            }
        },
        {
            "Sid": "PutCloudWatchLogs",
            "Effect": "Allow",
            "Action": [
                "logs:DescribeLogGroups",
                "logs:DescribeLogStreams",
                "logs:PutLogEvents"
            ],
            "Resource": "arn:aws:logs:*:*:log-group:/aws/fsx/*"
        },
        {
            "Sid": "ManageAuditLogs",
            "Effect": "Allow",
            "Action": [
                "firehose:DescribeDeliveryStream",
                "firehose:PutRecord",
                "firehose:PutRecordBatch"
            ],
            "Resource": "arn:aws:firehose:*:*:deliverystream/aws-fsx-*"
        }
    ]
}
```

이 정책에 대한 모든 업데이트는 관리형 정책에 대한 Amazon FSx 업데이트 AWS에 설명되어 있습니다.

IAM 엔터티(사용자, 그룹, 역할 등)가 서비스 연결 역할을 생성하고 편집하거나 삭제할 수 있도록 권한을 구성할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 권한 섹션을 참조하세요.

AWSServiceRoleForFSxS3Access 권한 세부 정보

의 경우`AWSServiceRoleForFSxS3Access_`*`file-system-id`*, 역할 권한 정책에 따라 Amazon FSx는 Amazon FSx for Lustre 파일 시스템의 데이터 리포지토리를 호스팅하는 Amazon S3 버킷에서 다음 작업을 완료할 수 있습니다.

- `s3:AbortMultipartUpload`
- `s3:DeleteObject`
- `s3:Get*`
- `s3:List*`
- `s3:PutBucketNotification`
- `s3:PutObject`

IAM 엔터티(사용자, 그룹, 역할 등)가 서비스 링크 역할을 생성하고 편집하거나 삭제할 수 있도록 권한을 구성할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 권한을 참조하세요.

Amazon FSx에 대한 서비스 연결 역할 생성

서비스 링크 역할은 수동으로 생성할 필요가 없습니다. AWS Management Console AWS CLI, 또는 AWS API에서 파일 시스템을 생성하면 Amazon FSx가 서비스 연결 역할을 자동으로 생성합니다.

> **Important**
>
> 이러한 서비스 연결 역할은 해당 역할이 지원하는 기능을 사용하는 다른 서비스에서 작업을 완료했을 경우 계정에 나타날 수 있습니다. 자세한 내용은 내 IAM 계정에 표시되는 새 역할을 참조하세요.

이 서비스 연결 역할을 삭제했다가 다시 생성해야 하는 경우 동일한 프로세스를 사용하여 계정에서 역할을 다시 생성할 수 있습니다. 파일 시스템을 생성할 때 Amazon FSx에서는 서비스 연결 역할을 다시 생성합니다.

Amazon FSx에 대한 서비스 연결 역할 편집

Amazon FSx에서는 이러한 서비스 연결 역할을 편집할 수 없습니다. 서비스 링크 역할을 생성한 후에는 다양한 개체가 역할을 참조할 수 있기 때문에 역할 이름을 변경할 수 없습니다. 하지만 IAM을 사용하여 역할의 설명을 편집할 수 있습니다. 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 편집을 참조하세요.

Amazon FSx에 대한 서비스 연결 역할 삭제

서비스 연결 역할이 필요한 기능 또는 서비스가 더 이상 필요 없는 경우에는 해당 역할을 삭제할 것이 좋습니다. 따라서 적극적으로 모니터링하거나 유지하지 않는 미사용 엔터티가 없도록 합니다. 그러나 서비스 연결 역할을 수동으로 삭제하려면 먼저 모든 파일 시스템 및 백업을 삭제해야 합니다.

> **Note**
>
> 리소스를 삭제하려고 할 때 Amazon FSx 서비스가 역할을 사용 중이면 삭제에 실패할 수 있습니다. 이 문제가 발생하면 몇 분 기다렸다가 작업을 다시 시도하세요.

IAM을 사용하여 수동으로 서비스 연결 역할을 삭제하려면

IAM 콘솔, IAM CLI 또는 IAM API를 사용하여 서비스 연결 역할을 삭제합니다. AWSServiceRoleForAmazonFSx 자세한 내용은 IAM 사용 설명서의 서비스 연결 역할 삭제 섹션을 참조하세요.

Amazon FSx 서비스 연결 역할을 지원하는 리전

Amazon FSx는 서비스가 제공되는 모든 리전에서 서비스 연결 역할 사용을 지원합니다. 자세한 내용은 AWS 리전 및 엔드포인트 섹션을 참조하십시오.

Amazon VPC를 사용한 파일 시스템 액세스 제어

Amazon FSx 파일 시스템은 파일 시스템과 연결된 Amazon VPC 서비스에 따라 가상 프라이빗 클라우드(VPC)에 있는 탄력적 네트워크 인터페이스를 통해 액세스할 수 있습니다. 파일 시스템의 네트워크 인터페이스에 매핑되는 DNS 이름을 통해 Amazon FSx 파일 시스템에 액세스합니다. 연결된 VPC 또는 피어링된 VPC 내의 리소스만 파일 시스템의 네트워크 인터페이스에 액세스할 수 있습니다. 자세한 내용은 Amazon VPC 사용 설명서의 Amazon VPC란 무엇인가요?를 참조하세요.

> **Warning**
>
> 이 Amazon FSx 탄력적 네트워크 인터페이스는 수정하거나 삭제하면 안 됩니다. 네트워크 인터페이스를 수정하거나 삭제하면 VPC와 파일 시스템 간의 연결이 영구적으로 손실될 수 있습니다.

Amazon VPC 보안 그룹

VPC 내에서 파일 시스템의 네트워크 인터페이스를 통과하는 네트워크 트래픽을 추가로 제어하려면 보안 그룹을 사용하여 파일 시스템에 대한 액세스를 제한합니다. 보안 그룹은 가상 방화벽 역할을 하여 관련 리소스에 대한 트래픽을 제어합니다. 이 경우, 관련 리소스는 파일 시스템의 네트워크 인터페이스입니다. 또한 VPC 보안 그룹을 사용하여 Lustre 클라이언트의 네트워크 트래픽을 제어할 수 있습니다.

인바운드 및 아웃바운드 규칙을 사용한 액세스 제어

보안 그룹을 사용하여 Amazon FSx 파일 시스템 및 Lustre 클라이언트에 대한 액세스를 제어하려면 수신되는 트래픽을 제어하는 인바운드 규칙과 파일 시스템 및 Lustre 클라이언트에서 발신되는 트래픽을 제어하는 아웃바운드 규칙을 추가합니다. Amazon FSx 파일 시스템의 파일 공유를 지원되는 컴퓨팅 인스턴스의 폴더에 매핑하려면 보안 그룹에 올바른 네트워크 트래픽 규칙이 있는지 확인합니다.

보안 그룹 규칙에 대한 자세한 내용은 Amazon EC2 사용 설명서의 보안 그룹 규칙을 참조하십시오.

Amazon FSx 파일 시스템을 위한 보안 그룹을 만들려면

1. https://console.aws.amazon.com/ec2 에서 Amazon EC2 콘솔을 엽니다.
2. 탐색 창에서 보안 그룹을 선택합니다.
3. 보안 그룹 생성을 선택합니다.
4. 보안 그룹의 이름과 설명을 지정합니다.
5. VPC의에서는 Amazon FSx 파일 시스템과 연결된 VPC를 선택하여 해당 VPC 내에 보안 그룹을 생성합니다.
6. 보안 그룹을 생성하려면 생성을 선택합니다.

다음으로 방금 생성한 보안 그룹에 인바운드 규칙을 추가하여 FSx for Lustre 파일 서버 간의 Lustre 트래픽을 활성화합니다.

보안 그룹에 인바운드 규칙 추가

1. 방금 생성한 보안 그룹을 아직 선택하지 않았다면 선택합니다. 작업에서 인바운드 규칙을 선택합니다.
2. 다음 인바운드 규칙을 추가합니다.

유형	프로토콜	포트 범위	소스	설명
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 방금 생성한 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버 간의 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 Lustre 클라이언트와 연결된 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 방금 생성한 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버 간의 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 Lustre 클라이언트와 연결된 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre 트래픽을 허용합니다.

3. 저장을 선택하여 새 인바운드 규칙을 저장하고 적용합니다.

기본적으로 보안 그룹은 모든 아웃바운드 트래픽을 허용합니다(All, 0.0.0.0/0). 보안 그룹에서 모든 아웃바운드 트래픽을 허용하지 않을 경우 보안 그룹에 다음 아웃바운드 규칙을 추가합니다. 이러한 규칙은 FSx for Lustre 파일 서버와 Lustre 클라이언트 간 및 Lustre 파일 서버 간 트래픽을 허용합니다.

보안 그룹에 아웃바운드 규칙 추가

1. 인바운드 규칙을 방금 추가한 것과 동일한 보안 그룹을 선택합니다. 작업에서 아웃바운드 규칙 편집을 선택합니다.
2. 다음 아웃바운드 규칙을 추가합니다.

유형	프로토콜	포트 범위	소스	설명
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 방금 생성한 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버 간 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 Lustre 클라이언트와 연결된 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 방금 생성한 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버 간의 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 Lustre 클라이언트와 연결된 보안 그룹	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre

유형	프로토콜	포트 범위	소스	설명
			의 보안 그룹 ID를 입력합니다.	트래픽을 허용합니다.

3. 저장을 선택하여 새 아웃바운드 규칙을 저장하고 적용합니다.

Amazon FSx 파일 시스템과 보안 그룹의 연결

1. https://console.aws.amazon.com/fsx/에서 Amazon FSx 콘솔을 엽니다.
2. 콘솔 대시보드에서 세부 정보를 볼 파일 시스템을 선택합니다.
3. 네트워크 및 보안 탭에서 파일 시스템의 네트워크 인터페이스 ID(예:`ENI-01234567890123456`)를 선택합니다. 이렇게 하면 Amazon EC2 콘솔로 리디렉션됩니다.
4. 각 네트워크 인터페이스 ID를 선택합니다. 각 작업은 브라우저에서 Amazon EC2 콘솔의 새 인스턴스를 엽니다. 각 보안 그룹에 대해, 작업에서 보안 그룹 변경을 선택합니다.
5. 보안 그룹 변경 대화 상자에서 사용할 보안 그룹을 선택하고 저장을 선택합니다.

Lustre 클라이언트 VPC 보안 그룹 규칙

VPC 보안 그룹을 사용하여 수신되는 트래픽을 제어하는 인바운드 규칙과 Lustre 클라이언트에서 발신되는 트래픽을 제어하는 아웃바운드 규칙을 추가하여 Lustre 클라이언트에 대한 액세스를 제어할 수 있습니다. Lustre 클라이언트와 Amazon FSx 파일 시스템 간에 Lustre 트래픽이 흐를 수 있도록 보안 그룹에 올바른 네트워크 트래픽 규칙을 적용해야 합니다.

Lustre 클라이언트에 적용되는 보안 그룹에 다음 인바운드 규칙을 추가합니다.

유형	프로토콜	포트 범위	소스	설명
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 Lustre 클라이언트에 적용되는 보안 그룹의 보안 그룹 ID를 입력합니다.	Lustre 클라이언트 간 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 FSx for	FSx for Lustre 파일 서버와 Lustre

유형	프로토콜	포트 범위	소스	설명
			Lustre 파일 시스템과 연결된 보안 그룹의 보안 그룹 ID를 입력합니다.	클라이언트 간의 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 Lustre 클라이언트에 적용되는 보안 그룹의 보안 그룹 ID를 입력합니다.	Lustre 클라이언트 간 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 FSx for Lustre 파일 시스템과 연결된 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre 트래픽을 허용합니다.

Lustre 클라이언트에 적용된 보안 그룹에 다음 아웃바운드 규칙을 추가합니다.

유형	프로토콜	포트 범위	소스	설명
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 Lustre 클라이언트에 적용되는 보안 그룹의 보안 그룹 ID를 입력합니다.	Lustre 클라이언트 간 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	988	사용자 지정을 선택하고 FSx for Lustre 파일 시스템과 연결된 보안	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre 트래픽을 허용합니다.

유형	프로토콜	포트 범위	소스	설명
			그룹의 보안 그룹 ID를 입력합니다.	
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 Lustre 클라이언트에 적용되는 보안 그룹의 보안 그룹 ID를 입력합니다.	Lustre 클라이언트 간 Lustre 트래픽을 허용합니다.
사용자 지정 TCP 규칙	TCP	1018-1023	사용자 지정을 선택하고 FSx for Lustre 파일 시스템과 연결된 보안 그룹의 보안 그룹 ID를 입력합니다.	FSx for Lustre 파일 서버와 Lustre 클라이언트 간의 Lustre 트래픽을 허용합니다.

Amazon VPC 네트워크 ACL

VPC 내 파일 시스템에 대한 액세스를 보호하는 또 다른 방법은 네트워크 액세스 제어 목록(네트워크 ACL)을 설정하는 것입니다. 네트워크 ACL은 보안 그룹과는 별개이지만, VPC의 리소스에 추가 보안 계층을 추가하기 위한 비슷한 기능이 있습니다. 네트워크 ACL를 사용하여 액세스 컨트롤을 구현하는 것에 대한 자세한 내용은 Amazon VPC 사용 설명서의 네트워크 ACL을 사용하여 서브넷에 대한 트래픽 제어를 참조하세요.

Amazon FSx for Lustre에 대한 규정 준수 확인

특정 규정 준수 프로그램의 범위 내에 AWS 서비스 있는지 알아보려면AWS 서비스 규정 준수 프로그램의 범위 내 규정 준수AWS 서비스 프로그램별 참조하여 관심 있는 규정 준수 프로그램을 선택하십시오. 일반 정보는 AWS 규정 준수 프로그램AWS 보증 프로그램 규정AWS 참조하십시오.

를 사용하여 AWS Artifact타사 감사 보고서를 다운로드할 수 있습니다. 자세한 내용은 의 보고서 https://docs.aws.amazon.com/artifact/latest/ug/downloading-documents.html 참조하십시오 AWS Artifact.

사용 시 규정 준수 AWS 서비스 책임은 데이터의 민감도, 회사의 규정 준수 목표, 관련 법률 및 규정에 따라 결정됩니다. AWS 규정 준수에 도움이 되는 다음 리소스를 제공합니다.

- 보안 및 규정 준수 퀵 스타트 가이드 - 이 배포 가이드에서는 아키텍처 고려 사항을 설명하고 보안 및 규정 준수에 AWS 중점을 둔 기본 환경을 배포하기 위한 단계를 제공합니다.
- Amazon Web Services의 HIPAA 보안 및 규정 준수를 위한 설계 — 이 백서에서는 기업이 HIPAA 적격 애플리케이션을 만드는 AWS 데 사용할 수 있는 방법을 설명합니다.

  > **Note**
  >
  > 모든 AWS 서비스 사람이 HIPAA 자격을 갖춘 것은 아닙니다. 자세한 내용은 HIPAA 적격 서비스 참조를 참조하십시오.

- AWS 규정 준수 리소스AWS — 이 워크북 및 가이드 모음은 해당 산업 및 지역에 적용될 수 있습니다.
- AWS 고객 규정 준수 가이드 — 규정 준수의 관점에서 공동 책임 모델을 이해하십시오. 이 가이드에서는 보안을 유지하기 위한 모범 사례를 AWS 서비스 요약하고 여러 프레임워크 (미국 표준 기술 연구소 (NIST), 결제 카드 산업 보안 표준 위원회 (PCI), 국제 표준화기구 (ISO) 등) 에서 보안 제어에 대한 지침을 매핑합니다.
- AWS Config 개발자 안내서의 규칙을 사용하여 리소스 평가 — 이 AWS Config 서비스는 리소스 구성이 내부 관행, 업계 지침 및 규정을 얼마나 잘 준수하는지 평가합니다.
- AWS Security Hub— 이를 AWS 서비스 통해 내부 AWS보안 상태를 포괄적으로 파악할 수 있습니다. Security Hub는 보안 제어를 사용하여 AWS 리소스를 평가하고 보안 업계 표준 및 모범 사례에 대한 규정 준수를 확인합니다. 지원되는 서비스 및 제어 목록은 Security Hub 제어 참조를 참조하십시오.
- Amazon GuardDuty — 환경에 의심스럽고 악의적인 활동이 있는지 AWS 계정모니터링하여 워크로드, 컨테이너 및 데이터에 대한 잠재적 위협을 AWS 서비스 탐지합니다. GuardDuty 특정 규정 준수 프레임워크에서 요구하는 침입 탐지 요구 사항을 충족하여 PCI DSS와 같은 다양한 규정 준수 요구 사항을 해결하는 데 도움이 될 수 있습니다.
- AWS Audit Manager— 이를 AWS 서비스 통해 AWS 사용량을 지속적으로 감사하여 위험을 관리하고 규정 및 업계 표준을 준수하는 방법을 단순화할 수 있습니다.

Amazon FSx for Lustre 및 인터페이스 VPC 엔드포인트(AWS PrivateLink)

인터페이스 VPC 엔드포인트를 사용하도록 Amazon FSx를 구성하여 VPC의 보안 상태를 향상시킬 수 있습니다. 인터페이스 VPC 엔드포인트는 인터넷 게이트웨이, NAT 디바이스 AWS PrivateLink, VPN 연결 또는 연결 없이 Amazon FSx API에 비공개로 액세스할 수 있는 기술인 에 의해 구동됩니다. AWS Direct Connect VPC의 인스턴스는 Amazon FSx API와 통신하는 데 퍼블릭 IP 주소를 필요로 하지 않습니다. VPC와 Amazon FSx 간의 트래픽은 네트워크를 벗어나지 않습니다. AWS

각 인터페이스 VPC 엔드포인트는 서브넷에서 하나 이상의 탄력적 네트워크 인터페이스로 표현됩니다. 네트워크 인터페이스는 트래픽에 대한 진입점 역할을 하는 프라이빗 IP 주소를 Amazon FSx API에 제공합니다.

Amazon FSx 인터페이스 VPC 엔드포인트에 대한 고려 사항

Amazon FSx에 대한 인터페이스 VPC 엔드포인트를 설정하기 전에 Amazon VPC 사용 설명서에서 인터페이스 VPC 엔드포인트 속성 및 제한 사항을 검토해야 합니다.

VPC에서 모든 Amazon FSx API 작업을 호출할 수 있습니다. 예를 들어, VPC 내에서 API를 CreateFileSystem 호출하여 FSx for Lustre 파일 시스템을 생성할 수 있습니다. Amazon FSx API의 전체 목록은 Amazon FSx API 참조의 작업을 참조하세요.

VPC 피어링 고려 사항

VPC 피어링을 사용하여 인터페이스 VPC 엔드포인트가 있는 VPC에 다른 VPC를 연결할 수 있습니다. VPC 피어링은 두 VPC 간의 네트워킹 연결입니다. 사용자의 자체 두 VPC 간에 또는 다른 AWS 계정의 VPC와 VPC 피어링 연결을 설정할 수 있습니다. VPC는 서로 다를 수도 있습니다. AWS 리전

피어링된 VPC 간의 트래픽은 AWS 네트워크에 머무르며 공용 인터넷을 통과하지 않습니다. VPC가 피어링되면 두 VPC의 Amazon Elastic Compute Cloud(Amazon EC2) 인스턴스와 같은 리소스는 VPC 중 하나에서 생성된 인터페이스 VPC 엔드포인트를 통해 Amazon FSx API에 액세스할 수 있습니다.

Amazon FSx API에 대한 인터페이스 VPC 엔드포인트 생성

Amazon VPC 콘솔 또는 () 를 사용하여 Amazon FSx API에 대한 VPC 엔드포인트를 생성할 수 있습니다. AWS Command Line Interface AWS CLI자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 VPC 엔드포인트 생성을 참조하세요.

Amazon FSx 엔드포인트의 전체 목록은 Amazon Web Services 일반 참조에서 Amazon FSx 엔드포인트 및 할당량을 참조하세요.

Amazon FSx에 대한 인터페이스 VPC 엔드포인트를 생성하려면 다음 중 하나를 사용합니다.

- **com.amazonaws.*region*.fsx** - Amazon FSx API 작업을 위한 엔드포인트를 생성합니다.
- **com.amazonaws.*region*.fsx-fips** - Federal Information Processing Standard(FIPS) 140-2를 준수하는 Amazon FSx API에 대한 엔드포인트를 생성합니다.

프라이빗 DNS 옵션을 사용하려면 VPC의 `enableDnsHostnames` 및 `enableDnsSupport` 속성을 설정해야 합니다. 자세한 내용은 Amazon VPC 사용 설명서의 VPC에 대한 DNS 지원 보기 및 업데이트를 참조하세요.

중국을 제외하고 AWS 리전 , 엔드포인트에 대해 프라이빗 DNS를 활성화하면, 예를 들어 VPC 엔드포인트의 기본 DNS 이름을 사용하여 Amazon FSx에 API 요청을 할 수 있습니다. AWS 리전`fsx.us-east-1.amazonaws.com` 중국 (베이징) 과 중국 (닝샤) AWS 리전의 경우 각각 및 를 사용하여 `fsx-api.cn-north-1.amazonaws.com.cn` VPC 엔드포인트로 API 요청을 할 수 있습니다. `fsx-api.cn-northwest-1.amazonaws.com.cn`

자세한 내용은 Amazon VPC 사용 설명서의 인터페이스 VPC 엔드포인트를 통해 서비스 액세스를 참조하세요.

Amazon FSx에 대한 VPC 엔드포인트 정책 생성

Amazon FSx API에 대한 액세스를 추가로 제어하려면 선택적으로 (IAM) 정책을 VPC AWS Identity and Access Management 엔드포인트에 연결할 수 있습니다. 이 정책은 다음을 지정합니다.

- 작업을 수행할 수 있는 보안 주체.
- 수행할 수 있는 작업.
- 작업을 수행할 수 있는 리소스.

자세한 내용은 Amazon VPC 사용 설명서의 VPC 엔드포인트를 통해 서비스에 대한 액세스 제어를 참조하세요.

할당량

다음에서는 Amazon FSx for Lustre 작업 시 할당량에 대해 알아봅니다.

주제

- 늘릴 수 있는 할당량
- 각 파일 시스템의 리소스 할당량
- 추가 고려 사항

늘릴 수 있는 할당량

계정별, 지역별 Amazon FSx for AWS Lustre의 할당량은 다음과 같습니다. 할당량을 늘릴 수 있습니다. AWS

Resource	기본값	설명
Lustre 영구_1 파일 시스템	100	이 계정에서 생성할 수 있는 Amazon FSx for Lustre 영구_1 파일 시스템의 최대 수입니다.
Lustre 영구_2 파일 시스템	100	이 계정에서 생성할 수 있는 Amazon FSx for Lustre 영구_2 파일 시스템의 최대 수입니다.
Lustre 영구 HDD 스토리지 용량(파일 시스템당)	102000	Amazon FSx for Lustre 영구 파일 시스템에 대해 구성할 수 있는 최대 HDD 스토리지 용량(GiB 단위).
Lustre 영구_1 파일 스토리지 용량	100800	이 계정의 모든 Amazon FSx for Lustre 영구_1 파일 시스템에 대해 구성할 수 있는 최대 스토리지 용량(GiB 단위)입니다.
Lustre 영구_2 파일 스토리지 용량	100800	이 계정의 모든 Amazon FSx for Lustre 영구_2 파일 시스템

Resource	기본값	설명
		에 대해 구성할 수 있는 최대 스토리지 용량(GiB 단위)입니다.
Lustre 스크래치 파일 시스템	100	이 계정에서 생성할 수 있는 Amazon FSx for Lustre 스크래치 파일 시스템의 최대 수입니다.
Lustre 스크래치 스토리지 용량	100800	이 계정에서 모든 Amazon FSx for Lustre 스크래치 파일 시스템에 대해 구성할 수 있는 최대 스토리지 용량(GiB 단위)입니다.
Lustre 백업	500	이 계정의 모든 Amazon FSx for Lustre 파일 시스템에 대해 보유할 수 있는 최대 사용자 시작 백업 수입니다.

할당량 증가 요청

1. Service Quotas 콘솔을 엽니다.
2. 탐색 창에서 AWS 서비스를 선택합니다.
3. Amazon FSx를 선택합니다.
4. 할당량을 선택합니다.
5. 할당량 증가 요청을 선택한 다음, 지침에 따라 할당량 증가를 요청합니다.
6. 할당량 요청 상태를 보려면 콘솔 탐색 창에서 할당량 요청 기록을 선택합니다.

자세한 내용은Service Quotas 사용 설명서의 할당량 증가 요청을 참조하세요.

각 파일 시스템의 리소스 할당량

다음은 AWS 지역 내 각 파일 시스템에 대한 Amazon FSx for Lustre 리소스 제한입니다.

Resource	파일 시스템당 한도
최대 태그 수	50
자동 백업의 최대 보존 기간	90일
계정당 단일 대상 리전으로 진행 중인 최대 백업 복사 요청 수입니다.	5
파일 시스템당 연결된 S3 버킷의 파일 업데이트 수	1,000만 건/월
최소 스토리지 용량, SSD 파일 시스템	1.2TiB
최소 스토리지 용량, HDD 파일 시스템	6TiB
스토리지 단위당 최소 처리량, SSD	50MBps
스토리지 단위당 최대 처리량, SSD	1,000MBps
스토리지 단위당 최소 처리량, SSD	12MBps
스토리지 단위당 최대 처리량, SSD	40MBps

추가 고려 사항

또한 다음 사항에 유의하세요.

- 최대 125개의 Amazon FSx for Lustre 파일 시스템에서 각 AWS Key Management Service (AWS KMS) 키를 사용할 수 있습니다.
- 파일 시스템을 생성할 수 있는 AWS 지역 목록은 의 Amazon FSx 엔드포인트 및 할당량을 참조하십시오. AWS 일반 참조

문제 해결

다음 정보를 사용하여 Amazon FSx for Lustre 파일 시스템에서 작업할 때 발생할 수 있는 문제를 해결할 수 있습니다.

아래에 나열되지 않은 문제가 발생하는 경우 Amazon FSx for Lustre 포럼에 질문해 보세요.

주제

- FSx for Lustre 파일 시스템 생성 실패
- 파일 시스템 마운트 문제 해결
- 파일 시스템 액세스 불가
- 데이터 리포지토리 연결을 생성할 때 S3 버킷에 대한 액세스를 검증할 수 없습니다.
- 디렉터리 이름을 바꾸는 데 시간이 오래 걸립니다.
- 잘못 구성된 연결된 S3 버킷 문제 해결
- 스토리지 문제 해결
- FSx for Lustre CSI 드라이버 문제 해결

FSx for Lustre 파일 시스템 생성 실패

다음 토픽에 설명된 것처럼 파일 시스템 생성 요청이 실패하는 잠재적 원인은 여러 가지가 있습니다.

잘못 구성된 보안 그룹 때문에 파일 시스템을 생성할 수 없습니다.

FSx for Lustre 파일 시스템 생성이 실패하면 다음 오류 메시지가 표시됩니다.

```
The file system cannot be created because the default security group in the subnet
 provided
or the provided security groups do not permit Lustre LNET network traffic on port 988
```

취할 조치

생성 작업에 사용하는 VPC 보안 그룹이 Amazon VPC를 사용한 파일 시스템 액세스 제어에 설명된 대로 구성되어 있는지 확인합니다. 보안 그룹 자체 또는 전체 서브넷 CIDR에서 포트 988 및 1018~1023의 인바운드 트래픽을 허용하도록 보안 그룹을 설정해야 합니다. 이 트래픽은 파일 시스템 호스트가 서로 통신할 수 있도록 하는 데 필요합니다.

S3 버킷에 연결된 파일 시스템을 생성할 수 없습니다.

S3 버킷에 연결된 새 파일 시스템 생성이 실패하는 경우 다음과 비슷한 오류 메시지가 표시됩니다.

```
User: arn:aws:iam::012345678901:user/username is not authorized to perform:
 iam:PutRolePolicy on resource: resource ARN
```

필요한 IAM 권한 없이 Amazon S3 버킷에 연결된 파일 시스템을 생성하려고 하면 이 오류가 발생할 수 있습니다. 필요한 IAM 권한은 사용자를 대신하여 지정된 Amazon S3 버킷에 액세스하는 데 사용되는 Amazon FSx for Lustre 서비스 연결 역할을 지원합니다.

취할 조치

IAM 엔터티(사용자, 그룹 또는 역할)에 파일 시스템을 생성할 수 있는 적절한 권한이 있는지 확인합니다. 여기에는 Amazon FSx for Lustre 서비스 연결 역할을 지원하는 권한 정책을 추가하는 작업이 포함됩니다. 자세한 내용은 Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가 섹션을 참조하세요.

서비스 연결 역할에 대한 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용 섹션을 참조하세요.

파일 시스템 마운트 문제 해결

다음 항목에 설명된 것처럼 파일 시스템 마운트 명령이 실패하는 원인에는 여러 가지가 있습니다.

파일 시스템 마운트 즉시 실패

파일 시스템 마운트 명령이 바로 실패 다음 코드에 예가 나와 있습니다.

```
mount.lustre: mount fs-0123456789abcdef0.fsx.us-east-1.aws@tcp:/fsx at /lustre
failed: No such file or directory

Is the MGS specification correct?
Is the filesystem name correct?
```

mount 명령을 사용하여 영구 또는 스크래치 2 파일 시스템을 마운트할 때 올바른 `mountname` 값을 사용하지 않으면 이 오류가 발생할 수 있습니다. describe-file-systems AWS CLI 명령 또는 DescribeFileSystemsAPI 작업의 응답에서 `mountname` 값을 가져올 수 있습니다.

파일 시스템 마운트가 중단된 후 실패하고 제한 시간 초과 오류가 표시됨

파일 시스템 탑재 명령이 1~2분 동안 중단된 후 실패하고 제한 시간 초과 오류가 표시됩니다.

다음 코드에 예가 나와 있습니다.

```
sudo mount -t lustre file_system_dns_name@tcp:/mountname /mnt/fsx

[2+ minute wait here]
Connection timed out
```

이 오류는 Amazon EC2 인스턴스 또는 파일 시스템의 보안 그룹이 제대로 구성되지 않았기 때문에 발생할 수 있습니다.

취할 조치

파일 시스템의 보안 그룹에 Amazon VPC 보안 그룹 지정된 인바운드 규칙이 있는지 확인합니다.

자동 마운트 실패 및 인스턴스 무응답

경우에 따라 파일 시스템의 자동 마운트가 실패하여 Amazon EC2 인스턴스가 응답을 중지할 수 있습니다.

이 문제는 _netdev 옵션이 선언되지 않은 경우 발생할 수 있습니다. _netdev이 빠진 경우 Amazon EC2 인스턴스가 응답을 중지합니다. 컴퓨팅 인스턴스가 네트워킹을 시작한 후 네트워크 파일 시스템의 초기화를 완료해야 하기 때문입니다.

취할 조치

이 문제가 발생하는 경우 문의하세요 AWS Support.

시스템 부팅 중에 파일 시스템 마운트 실패

시스템 부팅 중에 파일 시스템 마운트가 실패합니다. /etc/fstab를 사용하여 마운트를 자동화합니다. 파일 시스템이 마운트되지 않은 경우 인스턴스 부팅 기간 동안 syslog에 다음과 같은 오류가 표시됩니다.

```
LNetError: 3135:0:(lib-socket.c:583:lnet_sock_listen()) Can't create socket: port 988
 already in use
```

```
LNetError: 122-1: Can't start acceptor on port 988: port already in use
```

이 오류는 포트 988을 사용할 수 없을 때 발생할 수 있습니다. 인스턴스가 NFS 파일 시스템을 마운트하도록 구성된 경우 NFS 마운트가 클라이언트 포트를 포트 988에 바인딩할 수 있습니다.

취할 조치

가능한 경우 NFS 클라이언트 `noresvport` 및 `noauto` 마운트 옵션을 조정하여 이 문제를 해결할 수 있습니다.

DNS 이름을 사용한 파일 시스템 마운트에 실패

잘못 구성된 도메인 이름 서비스(DNS) 이름은 다음 시나리오와 같이 파일 시스템 마운트 실패를 일으킬 수 있습니다.

시나리오 1: 도메인 이름 서비스(DNS) 이름을 사용하는 파일 시스템 마운트가 실패합니다. 다음 코드에 예가 나와 있습니다.

```
sudo mount -t lustre file_system_dns_name@tcp:/mountname /mnt/fsx
mount.lustre: Can't parse NID
'file_system_dns_name@tcp:/mountname'
```

취할 조치

Virtual Private Cloud(VPC) 구성을 확인합니다. 사용자 지정 VPC를 사용하는 경우, DNS 설정이 활성화되어 있는지 확인합니다. 자세한 내용은 Amazon VPC 사용 설명서의 VPC에서 DNS 사용 을 참조하세요.

`mount` 명령에서 DNS 이름을 지정하려면 다음 작업을 수행해야 합니다.

- Amazon EC2 인스턴스가 Amazon FSx for Lustre용 파일 시스템과 동일한 VPC에 있는지 확인합니다.
- Amazon에서 제공하는 DNS 서버를 사용하도록 구성된 VPC 내에서 Amazon EC2 인스턴스를 연결합니다. 자세한 내용은 Amazon VPC 사용 설명서의 DHCP 옵션 세트를 참조하세요.
- 연결 중인 Amazon EC2 인스턴스의 Amazon VPC에 DNS 호스트 이름이 활성화되어 있는지 확인합니다. 자세한 내용은 Amazon VPC 사용 설명서의 VPC에 대한 DNS 지원 업데이트를 참조하세요.

시나리오 2: 도메인 이름 서비스(DNS) 이름을 사용하는 파일 시스템 마운트가 실패합니다. 다음 코드에 예가 나와 있습니다.

```
mount -t lustre file_system_dns_name@tcp:/mountname /mnt/fsx
mount.lustre: mount file_system_dns_name@tcp:/mountname at /mnt/fsx failed: Input/
output error Is the MGS running?
```

취할 조치

클라이언트의 VPC 보안 그룹에 올바른 아웃바운드 트래픽 규칙이 적용되었는지 확인합니다. 이 권장 사항은 특히 기본 보안 그룹을 사용하지 않거나 기본 보안 그룹을 수정한 경우 유효합니다. 자세한 내용은 Amazon VPC 보안 그룹 섹션을 참조하세요.

파일 시스템 액세스 불가

파일 시스템에 액세스할 수 없는 잠재적 원인은 여러 가지가 있으며, 원인마다 해결 방법이 다릅니다.

파일 시스템 탄력적 네트워크 인터페이스에 연결된 탄력적 IP 주소 삭제됨

Amazon FSx는 퍼블릭 인터넷에서 파일 시스템에 액세스하는 것을 지원하지 않습니다. Amazon FSx는 인터넷에서 연결할 수 있는 퍼블릭 IP 주소인 탄력적 IP 주소를 자동으로 분리합니다. 이 주소는 파일 시스템의 탄력적 네트워크 인터페이스에 연결됩니다.

수정 또는 삭제된 파일 시스템 탄력적 네트워크 인터페이스

파일 시스템의 탄력적 네트워크 인터페이스를 수정하거나 삭제해서는 안 됩니다. 네트워크 인터페이스를 수정하거나 삭제하면 VPC와 파일 시스템 간의 연결이 영구적으로 손실될 수 있습니다. 새 파일 시스템을 생성하고, FSx 탄력적 네트워크 인터페이스를 수정하거나 삭제하지 마세요. 자세한 내용은 Amazon VPC를 사용한 파일 시스템 액세스 제어 섹션을 참조하세요.

데이터 리포지토리 연결을 생성할 때 S3 버킷에 대한 액세스를 검증할 수 없습니다.

Amazon FSx 콘솔에서 데이터 리포지토리 연결 (DRA) 을 생성하거나 `create-data-repository-association` CLI 명령 CreateDataRepositoryAssociation(동일한 API 작업) 을 사용하면 작업이 실패하고 다음 오류 메시지가 표시됩니다.

```
Amazon FSx is unable to validate access to the S3 bucket. Ensure the IAM role or user
```

```
you are using has s3:Get*, s3:List* and s3:PutObject permissions to the S3 bucket
 prefix.
```

> **Note**
>
> Amazon FSx 콘솔 또는 `create-file-system` CLI 명령 CreateFileSystem(동일한 API 작업) 을 사용하여 데이터 리포지토리 (S3 버킷 또는 접두사) 에 연결된 스크래치 1, 스크래치 2 또는 영구 1 파일 시스템을 생성할 때도 위 오류가 발생할 수 있습니다.

취할 조치

FSx for Lustre 파일 시스템이 S3 버킷과 동일한 계정에 있는 경우 이 오류는 생성 요청에 사용한 IAM 역할에 S3 버킷에 액세스하는 데 필요한 권한이 없음을 의미합니다. IAM 역할에 오류 메시지에 나열된 권한이 있는지 확인합니다. 이러한 권한은 사용자를 대신하여 지정된 Amazon S3 버킷에 액세스하는 데 사용되는 Amazon FSx for Lustre 서비스 연결 역할을 지원합니다.

FSx for Lustre 파일 시스템이 S3 버킷과 다른 계정에 있는 경우(계정 간 사례), 사용한 IAM 역할에 필요한 권한이 있는지 확인하는 것 외에도 FSx for Lustre가 생성된 계정에서 액세스를 허용하도록 S3 버킷 정책을 구성해야 합니다. 다음은 샘플 정책입니다.

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
            "Principal": {
                "AWS": "*"
            },
            "Action": [
                "s3:AbortMultipartUpload",
                "s3:DeleteObject",
                "s3:PutObject",
                "s3:GetObject",
                "s3:GetBucketAcl",
                "s3:GetBucketNotification",
                "s3:ListBucket",
                "s3:PutBucketNotification"
            ],
            "Resource": [
                "arn:aws:s3:::bucket_name",
```

```
                "arn:aws:s3:::bucket_name/*"
            ],
            "Condition": {
                "StringLike": {
                    "aws:PrincipalArn": [
                        "arn:aws:iam::file_system_account_ID:role/aws-service-role/
s3.data-source.lustre.fsx.amazonaws.com/AWSServiceRoleForFSxS3Access_fs-*"
                }
            }
        }
    ]
}
```

Amazon S3의 크로스 계정 버킷 권한 구성에 대한 자세한 내용은 Amazon Simple Storage Service 사용 설명서의 예제 2 : 계정주의 크로스 계정 버킷 권한 부여를 참조하세요.

디렉터리 이름을 바꾸는 데 시간이 오래 걸립니다.

질문

Amazon S3 버킷에 연결된 파일 시스템의 디렉터리 이름을 변경하고 자동 내보내기를 활성화했습니다. S3 버킷에서 이 디렉터리 내의 파일 이름을 바꾸는 데 시간이 오래 걸리는 이유는 무엇인가요?

답변

파일 시스템에서 디렉터리 이름을 바꾸면 FSx for Lustre는 이름이 바뀐 디렉터리 내의 모든 파일 및 디렉터리에 대해 새 S3 객체를 생성합니다. 디렉터리 이름 변경을 S3로 전파하는 데 걸리는 시간은 이름을 바꾸는 디렉터리의 하위 항목인 파일 및 디렉터리의 양과 직접적인 상관관계가 있습니다.

잘못 구성된 연결된 S3 버킷 문제 해결

경우에 따라 FSx for Lustre 파일 시스템의 연결된 S3 버킷의 데이터 리포지토리 수명 주기 상태가 잘못 구성되어 있을 수 있습니다.

가능한 원인

Amazon FSx에 연결된 데이터 리포지토리에 액세스하는 AWS Identity and Access Management 데 필요한 (IAM) 권한이 없는 경우 이 오류가 발생할 수 있습니다. 필요한 IAM 권한은 사용자를 대신하여 지정된 Amazon S3 버킷에 액세스하는 데 사용되는 Amazon FSx for Lustre 서비스 연결 역할을 지원합니다.

취할 조치

1. IAM 엔터티(사용자, 그룹 또는 역할)에 파일 시스템을 생성할 수 있는 적절한 권한이 있는지 확인합니다. 여기에는 Amazon FSx for Lustre 서비스 연결 역할을 지원하는 권한 정책을 추가하는 작업이 포함됩니다. 자세한 정보는 Amazon S3의 데이터 리포지토리를 사용하기 위한 권한 추가을 참조하세요.
2. Amazon FSx CLI 또는 API를 사용하여 다음과 같이 CLI UpdateFileSystem명령 (동일한 API 작업)으로 `update-file-system` 파일 시스템을 `AutoImportPolicy` 새로 고칩니다.

```
aws fsx update-file-system \
--file-system-id fs-0123456789abcdef0 \
--lustre-configuration AutoImportPolicy=the_existing_AutoImportPolicy
```

서비스 연결 역할에 대한 자세한 내용은 Amazon FSx에 대해 서비스 연결 역할 사용를 참조하세요.

가능한 원인

이 오류는 연결된 Amazon S3 데이터 리포지토리에 Amazon FSx 이벤트 알림 구성(`s3:ObjectCreated:*`, `s3:ObjectRemoved:*`)과 중복되는 이벤트 유형을 가진 기존 이벤트 알림 구성이 있는 경우 발생할 수 있습니다.

이는 연결된 S3 버킷의 Amazon FSx 이벤트 알림 구성이 삭제되거나 수정된 경우에도 발생할 수 있습니다.

취할 조치

1. 연결된 S3 버킷에서 FSx 이벤트 구성에서 사용하는 이벤트 유형인 `s3:ObjectCreated:*`, `s3:ObjectRemoved:*` 중 하나 또는 둘 다를 사용하는 기존 이벤트 알림을 제거합니다.
2. 연결된 S3 버킷에 이름 `FSx`, 이벤트 유형 `s3:ObjectCreated:*` 및 `s3:ObjectRemoved:*`를 포함하는 S3 이벤트 알림 구성이 있는지 확인하고 ARN:*topic_arn_returned_in_API_response*으로 SNS 토픽으로 보냅니다.
3. Amazon FSx CLI 또는 API를 사용하여 S3 버킷에 FSx 이벤트 알림 구성을 다시 적용하여 파일 시스템의 `AutoImportPolicy`를 새로고침합니다. 다음과 같이 `update-file-system` CLI 명령 (동일한 API 작업) 을 사용하여 UpdateFileSystem이 작업을 수행합니다.

```
aws fsx update-file-system \
--file-system-id fs-0123456789abcdef0 \
--lustre-configuration AutoImportPolicy=the_existing_AutoImportPolicy
```

스토리지 문제 해결

경우에 따라 파일 시스템에 스토리지 문제가 발생할 수 있습니다. `lfs migrate` 명령과 같은 `lfs` 명령을 사용하여 이러한 문제를 해결할 수 있습니다.

스토리지 대상에 공간이 없어서 쓰기 오류가 발생했습니다.

파일 시스템 스토리지 레이아웃에 설명된 대로 `lfs df -h` 명령을 사용하여 파일 시스템의 스토리지 사용량을 확인할 수 있습니다. 이 `filesystem_summary` 필드에는 총 파일 시스템 스토리지 사용량이 보고됩니다.

파일 시스템 디스크 사용량이 100% 인 경우 파일 시스템의 스토리지 용량을 늘리는 것을 고려해 보세요. 자세한 내용은 스토리지 용량 관리 섹션을 참조하세요.

파일 시스템 스토리지 사용량이 100% 가 아닌데도 쓰기 오류가 계속 발생하는 경우, 쓰기 중인 파일이 꽉 찬 OST에 스트라이핑될 수 있습니다.

취할 조치

- 많은 OST가 꽉 찬 경우 파일 시스템의 스토리지 용량을 늘리세요. OST의 스토리지 불균형 섹션의 작업에 따라 OST의 스토리지 용량이 불균형하지 않은지 확인합니다.
- OST가 가득 차 있지 않은 경우 모든 클라이언트 인스턴스에 다음 조정을 적용하여 클라이언트 더티 페이지 버퍼 크기를 조정하십시오.

```
sudo lctl set_param osc.*.max_dirty_mb=64
```

OST의 스토리지 불균형

Amazon FSx for Lustre는 새 파일 스트라이프를 OST 전체에 균등하게 배포합니다. 하지만 I/O 패턴 또는 파일 스토리지 레이아웃으로 인해 파일 시스템이 여전히 불균형해질 수 있습니다. 따라서 일부 스토리지 타겟은 꽉 찬 반면 다른 타겟은 상대적으로 비어 있을 수 있습니다.

`lfs migrate` 명령을 사용하여 파일 또는 디렉터리를 가득 찬 OST에서 덜 꽉 찬 OST로 이동할 수 있습니다. 이 `lfs migrate` 명령은 블록 또는 비블록 모드에서 사용할 수 있습니다.

- 블록 모드는 `lfs migrate` 명령의 기본 모드입니다. 차단 모드에서 실행하면 데이터 마이그레이션 전에 `lfs migrate` 코드가 먼저 파일 및 디렉터리에 대한 그룹 잠금을 획득하여 파일이 수정되지 않도록 한 다음 마이그레이션이 완료되면 잠금을 해제합니다. 차단 모드는 다른 프로세스가 파일을 수정하지 못하도록 함으로써 이러한 프로세스가 마이그레이션을 방해하지 않도록 합니다. 단점은

응용 프로그램에서 파일을 수정하지 못하게 하면 응용 프로그램이 지연되거나 오류가 발생할 수 있다는 것입니다.

- `-n` 옵션이 있는 `lfs migrate` 명령에 대해 비차단 모드가 활성화됩니다. 비블록 모드에서 `lfs migrate` 명령이 실행 중인 경우에도 다른 프로세스가 마이그레이션 중인 파일을 수정할 수 있습니다. `lfs migrate` 명령이 마이그레이션을 완료하기 전에 프로세스에서 파일을 수정하면 `lfs migrate` 명령이 해당 파일을 마이그레이션하는 데 실패하고 파일은 원래 스트라이프 레이아웃으로 남게 됩니다.

비차단 모드는 응용 프로그램에 방해가 될 가능성이 적으므로 사용하는 것이 좋습니다.

취할 조치

1. 비교적 큰 클라이언트 인스턴스(예: Amazon EC2 인스턴스`c5n.4xlarge` 유형)를 시작하여 파일 시스템에 마운트합니다.
2. 비차단 모드 스크립트 또는 차단 모드 스크립트를 실행하기 전에 먼저 각 클라이언트 인스턴스에서 다음 명령을 실행하여 프로세스 속도를 높이세요.

   ```
   sudo lctl set_param 'mdc.*.max_rpcs_in_flight=60'
   sudo lctl set_param 'mdc.*.max_mod_rpcs_in_flight=59'
   ```

3. 스크린 세션을 시작하고 비차단 모드 스크립트 또는 차단 모드 스크립트를 실행합니다. 스크립트에서 적절한 변수를 변경합니다.

   - 비차단 모드 스크립트:

     ```
     #!/bin/bash

     # UNCOMMENT THE FOLLOWING LINES:
     #
     # TRY_COUNT=0
     # MAX_MIGRATE_ATTEMPTS=100
     # OSTS="fsname-OST0000_UUID"
     # DIR_OR_FILE_MIGRATED="/mnt/subdir/"
     # BATCH_SIZE=10
     # PARALLEL_JOBS=16 # up to max-procs processes, set to 16 if client is
      c5n.4xlarge with 16 vcpu
     # LUSTRE_STRIPING_CONFIG="-E 100M -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32" #
      should be consistent with the existing striping setup
     #
     ```

```
if [ -z "$TRY_COUNT" -o -z "$MAX_MIGRATE_ATTEMPTS" -o -z "$OSTS" -o -z
 "$DIR_OR_FILE_MIGRATED" -o -z "$BATCH_SIZE" -o -z "$PARALLEL_JOBS" -o -z
 "$LUSTRE_STRIPING_CONFIG" ]; then
    echo "Some variables are not set."
    exit 1
fi

echo "lfs migrate starts"
while true; do
    output=$(sudo lfs find ! -L released --ost $OSTS --print0
 $DIR_OR_FILE_MIGRATED | shuf -z | /bin/xargs -0 -P $PARALLEL_JOBS -n $BATCH_SIZE
 sudo lfs migrate -n $LUSTRE_STRIPING_CONFIG 2>&1)
    if [[ $? -eq 0 ]]; then
        echo "lfs migrate succeeds for $DIR_OR_FILE_MIGRATED at the $TRY_COUNT
 attempt, exiting."
        exit 0
    elif [[ $? -eq 123 ]]; then
        echo "WARN: Target data objects are not located on these OSTs. Skipping
 lfs migrate"
        exit 1
    else
        echo "lfs migrate fails for $DIR_OR_FILE_MIGRATED at the $TRY_COUNT
 attempt, retrying..."
        if (( ++TRY_COUNT >= MAX_MIGRATE_ATTEMPTS )); then
            echo "WARN: Exceeds max retry attempt. Skipping lfs migrate for
 $DIR_OR_FILE_MIGRATED. Failed with the following error"
            echo $output
            exit 1
        fi
    fi
done
```

- 차단 모드 스크립트:
  - OSTS의 값을 OST 값으로 바꿉니다.
  - 병렬로 실행할 max-procs 프로세스 수를 설정하려면 nproc에 정수 값을 제공합니다. 예를 들어 Amazon EC2 인스턴스c5n.4xlarge 유형에는 16개의 vCPU가 있으므로 nproc에 16(또는 16보다 작은 값)을 사용할 수 있습니다.
  - mnt_dir_path에 마운트 디렉터리 경로를 입력합니다.

```
# find all OSTs with usage above a certain threshold; for example, greater than
 or equal to 85% full
```

```
for OST in $(lfs df -h |egrep '( 8[5-9]| 9[0-9]|100)%'|cut -d' ' -f1); do echo
 ${OST};done|tr '\012' ','

# customer can also just pass OST values directly to OSTS variable
OSTS='dzfevbmv-OST0000_UUID,dzfevbmv-OST0002_UUID,dzfevbmv-OST0004_UUID,dzfevbmv-
OST0005_UUID,dzfevbmv-OST0006_UUID,dzfevbmv-OST0008_UUID'

nproc=<Run up to max-procs processes if client is c5n.4xlarge with 16 vcpu, this
 value can be set to 16>

mnt_dir_path=<mount dir, e.g. '/my_mnt'>

lfs find ${mnt_dir_path} --ost ${OSTS}| xargs -P ${nproc} -n2 lfs migrate -E 100M
 -c 1 -E 10G -c 8 -E 100G -c 16 -E -1 -c 32
```

참고

- 파일 시스템 읽기 성능에 영향이 있는 경우 `ctrl-c` 또는 `ill -9`를 사용하여 언제든지 마이그레이션을 중지하고 스레드 수(nproc 값)를 더 낮은 수(예: 8)로 줄인 다음 파일 마이그레이션을 재개할 수 있습니다.
- 클라이언트 워크로드에서도 열려 있는 파일에서는 `lfs migrate` 명령이 실패합니다. 오류가 발생하고 다음 파일로 이동합니다. 따라서 액세스하는 파일이 많으면 스크립트에서 파일을 마이그레이션하지 못할 수 있으며, 마이그레이션 진행 속도가 매우 느리기 때문에 반영될 수 있습니다.
- 다음 방법 중 하나를 사용하여 OST 사용량을 모니터링할 수 있습니다.
  - 클라이언트 마운트에서 다음 명령을 실행하여 OST 사용량을 모니터링하고 사용량이 85%를 초과하는 OST를 찾습니다.

    ```
    lfs df -h |egrep '( 8[5-9]| 9[1-9]|100)%'
    ```

  - Amazon CloudWatch 메트릭을 확인해 보세요`Minimum`. 확인해 보세요. OST `FreeDataStorageCapacity` 스크립트에서 85%가 넘는 OST가 검색되면 지표가 15%에 가까워지면 `ctrl-c` 또는 `kill -9`를 사용하여 마이그레이션을 중단합니다.
- 새 파일이 여러 스토리지 대상에 스트라이핑되도록 파일 시스템 또는 디렉터리의 스트라이프 구성을 변경하는 것도 고려할 수 있습니다. 자세한 내용은 파일 시스템의 스트라이핑 데이터 섹션을 참조하세요.

FSx for Lustre CSI 드라이버 문제 해결

Amazon EKS에서 실행되는 컨테이너용 FSx for Lustre CSI 드라이버에서 문제가 발생하는 경우 에서 사용할 수 있는 CSI 드라이버 문제 해결 (일반 문제) 을 참조하십시오. GitHub

추가 정보

이 섹션에서는 지원되지만 사용 중단된 Amazon FSx 기능에 대한 참조를 제공합니다.

주제

- 사용자 지정 백업 일정 설정

사용자 지정 백업 일정 설정

파일 시스템에 대한 사용자 지정 백업 일정을 설정하는 데 AWS Backup을 사용하는 것이 좋습니다. 여기에 제공된 정보는 AWS Backup 사용 시 더 자주 백업을 예약해야 하는 경우에 참조할 수 있습니다.

활성화된 경우 Amazon FSx for Windows File Server는 일일 백업 기간 동안 하루에 한 번 파일 시스템을 자동으로 백업합니다. Amazon FSx는 이러한 자동 백업에 대해 사용자가 지정한 보존 기간을 적용합니다. 또한 사용자 시작 백업을 지원하므로 언제든지 백업할 수 있습니다.

다음에서 사용자 지정 백업 예약을 배포하기 위한 리소스 및 구성을 찾을 수 있습니다. 사용자 지정 백업 예약은 사용자가 정의한 사용자 지정 일정에 따라 Amazon FSx for Lustre 파일 시스템에서 사용자가 시작한 백업을 수행합니다. 6시간에 한 번, 일주일에 한 번 등을 예로 들 수 있습니다. 또한 이 스크립트는 지정된 보존 기간보다 오래된 백업을 삭제하도록 구성합니다.

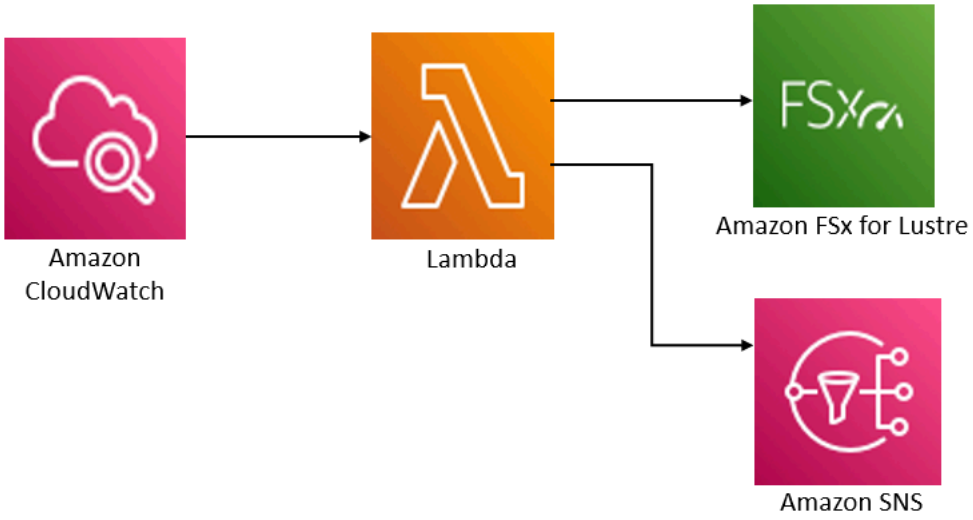
솔루션은 필요한 모든 구성 요소를 자동으로 배포하고 다음 파라미터를 사용합니다.

- 파일 시스템.
- 백업 수행을 위한 CRON 일정 패턴
- 백업 보존 기간(일 단위)
- 백업 이름 태그

CRON 스케줄 패턴에 대한 자세한 내용은 Amazon CloudWatch 사용 설명서의 규칙에 대한 스케줄 표현식을 참조하십시오.

아키텍처 개요

이 솔루션을 배포하면 AWS 클라우드에 다음과 같은 리소스가 빌드됩니다.



이 솔루션은 다음 작업을 수행합니다.

1. AWS CloudFormation템플릿은 CloudWatch 이벤트, Lambda 함수, Amazon SNS 대기열 및 IAM 역할을 배포합니다. IAM 역할은 Lambda 함수에 Amazon FSx for Lustre API 작업을 호출할 수 있는 권한을 부여합니다.
2. CloudWatch 이벤트는 초기 배포 시 사용자가 CRON 패턴으로 정의한 일정에 따라 실행됩니다. 이 이벤트는 Amazon FSx for Lustre `CreateBackup` API 작업을 호출하여 백업을 시작하는 솔루션의 백업 관리자 Lambda 함수를 호출합니다.
3. 백업 관리자는 `DescribeBackups`를 사용하여 지정된 파일 시스템에 대해 사용자가 시작한 기존 백업 목록을 검색합니다. 그런 다음 초기 배포 시 지정한 보존 기간보다 오래된 백업을 삭제합니다.
4. 초기 배포 중에 알림을 받는 옵션을 선택하면 백업 관리자가 백업 성공 시 Amazon SNS 대기열에 알림 메시지를 보냅니다. 장애 발생 시 항상 알림이 전송됩니다.

AWS CloudFormation 템플릿

이 솔루션은 Amazon FSx for Lustre 사용자 지정 백업 예약 솔루션의 배포를 자동화하는 데 AWS CloudFormation을 사용합니다. 이 솔루션을 fsx-scheduled-backup사용하려면.template 템플릿을 AWS CloudFormation 다운로드하십시오.

배포 자동화

다음 절차는 이 사용자 지정 백업 예약 솔루션을 구성하고 배포합니다. 배포에는 약 5분이 소요됩니다. 시작하기 전에 AWS 계정에 Amazon Virtual Private Cloud(VPC)에서 실행되는 Amazon FSx for Lustre 파일 시스템이 있어야 합니다. 리소스 생성에 대한 자세한 내용은 Amazon FSx for Lustre 시작하기 섹션을 참조하세요.

> **Note**
>
> 이 솔루션을 구현하면 연결된 AWS 서비스에 대한 요금이 청구됩니다. 자세한 내용은 해당 서비스에 대한 요금 세부 정보 페이지를 참조하세요.

사용자 지정 백업 솔루션 스택 시작

1. fsx-scheduled-backup.template 템플릿을 다운로드합니다. AWS CloudFormation AWS CloudFormation 스택 생성에 대한 자세한 내용은 AWS CloudFormation 사용 설명서의 AWS CloudFormation 콘솔에서 스택 생성을 참조하세요.

   > **Note**
   >
   > 기본적으로 이 템플릿은 미국 동부(버지니아 북부) AWS 리전에서 실행됩니다. Amazon FSx for Lustre는 현재 특정 AWS 리전 리전에서만 사용할 수 있습니다. Amazon FSx for Lustre를 사용할 수 있는 AWS 리전에서 이 솔루션을 시작해야 합니다. 자세한 내용은 AWS 일반 참조의 AWS 리전 및 엔드포인트의 Amazon FSx 섹션을 참조하세요.

2. 파라미터의 경우 템플릿의 파라미터를 검토하고 파일 시스템의 필요에 맞게 수정합니다. 이 솔루션은 다음과 같은 기본값을 사용합니다.

파라미터	기본값	설명
Amazon FSx for Lustre 파일 시스템 ID	기본값 없음	백업하려는 파일 시스템의 파일 시스템 ID
백업을 위한 CRON 일정 패턴.	0 0/4 * *? *	CloudWatch 이벤트를 실행하여 새 백업을 트리거하고 보존 기간이 지난 오래된 백업을 삭제하는 일정
백업 보존 기간(일)	7	사용자 시작 백업을 보존할 일수입니다. Lambda 함수는 이 일수보다 오래된 사용자 시작 백업을 삭제합니다.

파라미터	기본값	설명
백업 이름	사용자 예약 백업	이러한 백업의 이름은 Amazon FSx for Lustre 관리 콘솔의 백업 이름 열에 표시됩니다.
백업 알림	예	백업이 시작되었을 때 알림을 받을지 여부를 선택합니다. 오류가 있는 경우 항상 알림이 전송됩니다.
이메일 주소	기본값 없음	SNS 알림을 구독하기 위한 이메일 주소

3. 다음을 선택합니다.
4. 옵션에서 다음을 선택합니다.
5. 검토에서 설정을 검토하고 확인합니다. 템플릿이 IAM 리소스를 생성한다는 것을 확인하는 확인란을 선택해야 합니다.
6. 생성을 선택하여 스택을 배포합니다.

AWS CloudFormation 콘솔의 상태 열에서 스택의 상태를 볼 수 있습니다. 약 5분 후에 생성_완료 상태를 확인할 수 있습니다.

추가 옵션

이 솔루션으로 생성된 Lambda 함수를 사용하여 둘 이상의 Amazon FSx for Lustre 파일 시스템에 대한 사용자 지정 예약 백업을 수행할 수 있습니다. 파일 시스템 ID는 이벤트에 대한 입력 JSON의 Amazon FSx for Lustre 함수로 전달됩니다. CloudWatch Lambda 함수에 전달되는 기본 JSON은 다음과 같습니다. 여기서`FileSystemId` 및 `SuccessNotification` 값은 AWS CloudFormation 스택을 시작할 때 지정된 파라미터에서 전달됩니다.

```
{
 "start-backup": "true",
 "purge-backups": "true",
 "filesystem-id": "${FileSystemId}",
 "notify_on_success": "${SuccessNotification}"
}
```

추가 Amazon FSx for Lustre 파일 시스템의 백업을 예약하려면 다른 이벤트 규칙을 생성하십시오. CloudWatch 이렇게 하려면 이 솔루션에서 생성한 Lambda 함수를 대상으로 하는 일정 이벤트 소스를 사용합니다. 입력 구성에서 상수(JSON 텍스트)를 선택합니다. JSON 입력의 경우 `${FileSystemId}` 대신 백업할 Amazon FSx for Lustre 파일 시스템의 파일 시스템 ID로 대체하면 됩니다. 또한 위의 JSON에서 `${SuccessNotification}` 대신 `Yes` 또는 `No`를 사용할 수 있습니다.

수동으로 생성하는 추가 CloudWatch 이벤트 규칙은 Amazon FSx for Lustre 사용자 지정 예약 백업 솔루션 스택에 포함되지 않습니다. AWS CloudFormation 따라서 스택을 삭제해도 해당 스택은 제거되지 않습니다.

문서 이력

- API 버전: 2018년 3월 1일
- 최신 설명서 업데이트: 2024년 6월 6일

아래 표에 Amazon FSx for Lustre 사용자 가이드의 주요 변경 사항이 설명되어 있습니다. 설명서 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하시면 됩니다.

변경 사항	설명	날짜
메타데이터 성능 향상을 위한 지원 추가	이제 메타데이터 성능을 높이는 기능을 제공하는 메타데이터 구성을 사용하여 FSx for Lustre Persistent_2 파일 시스템을 생성할 수 있습니다. 자세한 내용은 파일 시스템 메타데이터 성능 및 메타데이터 성능 관리를 참조하십시오.	2024년 6월 6일
AWS 리전 퍼시스턴트_2 배포 유형에 대한 추가 지원이 추가되었습니다.	Lustre 파일 시스템용 Persistent_2 SSD FSx를 이제 미국 동부 (애틀랜타) 로컬 존에서 사용할 수 있습니다. 자세한 내용은 사용 가능한 지역을 참조하십시오.	2024년 5월 29일
CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 9.4에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS, 록키 리눅스 및 레드햇 엔터프라이즈 리눅스 (RHEL) 9.4를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2024년 5월 16일

AWS 리전 Persistent_2 배포 유형에 대한 추가 지원이 추가되었습니다.	이제 캐나다 서부 (캘거리)에서 Lustre 파일 시스템용 Persistent_2 SSD FSx를 사용할 수 있습니다. AWS 리전자세한 내용은 사용 가능한 지역을 참조하십시오.	2024년 5월 3일
아마존 리눅스 2023에 대한 Lustre 클라이언트 지원 추가	FSx for Lustre 클라이언트는 이제 아마존 리눅스 2023을 실행하는 아마존 EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2024년 3월 25일
CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 8.9에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS, 록키 리눅스 및 레드햇 엔터프라이즈 리눅스 (RHEL) 8.9를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2024년 1월 9일
Amazon FSx는 AmazonF 액세스, AmazonF, SxFull AmazonF, AmazonF 액세스 및 SxConsole FullAccess AmazonF 관리형 정책을 업데이트했습니다. SxRead OnlyAccess SxConsole ReadOnly SxService RolePolicy AWS	Amazon FSx는 AmazonF 액세스, AmazonF, SxFull AmazonF, SxConsole FullAccess AmazonF 액세스 및 SxRead OnlyAccess AmazonF 정책을 업데이트하여 권한을 추가했습니다 SxConsoleReadOnly. SxServiceRolePolicy `ec2:GetSecurityGroupsForVpc` 자세한 내용은 관리형 정책에 대한 Amazon FSx 업데이트를 AWS 참조하십시오.	2024년 1월 9일

CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 9.0 및 9.3에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 9.0 및 9.3을 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2023년 12월 20일
Amazon FSx for Lustre는 AmazonF 및 AmazonF 관리형 정책을 업데이트했습니다. SxFullAccess SxConsole FullAccess AWS	Amazon FSx는 SxFullAccess AmazonF 및 SxConsole FullAccess AmazonF 정책을 업데이트하여 작업을 추가했습니다. `ManageCrossAccountDataReplication` 자세한 내용은 관리형 정책에 대한 Amazon FSx 업데이트를 AWS 참조하십시오.	2023년 12월 20일
Amazon FSx는 AmazonF 및 SxFullAccess AmazonF 관리형 정책을 업데이트했습니다. SxConsole FullAccess AWS	Amazon FSx는 SxFullAccess AmazonF 및 SxConsole FullAccess AmazonF 정책을 업데이트하여 권한을 추가했습니다. `fsx:CopySnapshotAndUpdateVolume` 자세한 내용은 관리형 정책에 대한 Amazon FSx 업데이트를 AWS 참조하십시오.	2023년 11월 26일
처리량 용량 확장에 대한 지원 추가	이제 처리량 요구 사항의 변화에 따라 기존 FSx for Lustre 영구 SSD 기반 파일 시스템의 처리량 용량을 수정할 수 있습니다. 자세한 내용은 처리량 용량 관리를 참조하세요.	2023년 11월 16일

Amazon FSx는 AmazonF 및 SxFullAccess AmazonF 관리형 정책을 업데이트했습니다. SxConsole FullAccess AWS	Amazon FSx는 AmazonF 및 SxFullAccess AmazonF 정책을 업데이트하여 및 권한을 SxConsoleFullAccess 추가했습니다. `fsx:DescribeSharedVPCConfiguration` `fsx:UpdateSharedVPCConfiguration` 자세한 내용은 관리형 정책에 대한 Amazon FSx 업데이트를 AWS 참조하십시오.	2023년 11월 14일
프로젝트 할당량 지원 추가	이제 프로젝트에 대한 스토리지 할당량을 생성할 수 있습니다. 프로젝트 할당량은 프로젝트와 관련된 모든 파일 또는 디렉터리에 적용됩니다. 자세한 내용은 스토리지 할당량을 참조하세요.	2023년 8월 29일
Lustre 버전 2.15에 대한 지원이 추가되었습니다.	이제 모든 FSx for Lustre 파일 시스템은 Amazon FSx 콘솔을 사용하여 생성할 때 Lustre 버전 2.15를 기반으로 구축됩니다. 자세한 내용은 1단계: Amazon FSx for Lustre 파일 시스템 생성을 참조하세요.	2023년 8월 29일
AWS 리전 Persistent_2 배포 유형에 대한 추가 지원이 추가되었습니다.	이제 이스라엘 (텔아비브) 에서 Lustre 파일 시스템용 Persistent_2 FSx를 사용할 수 있습니다. AWS 리전자세한 내용은 FSx for Lustre 파일 시스템 배포 옵션을 참조하세요.	2023년 8월 24일

릴리스 데이터 리포지토리 작업에 대한 지원 추가	FSx for Lustre는 이제 S3 데이터 리포지토리에 연결된 파일 시스템에서 아카이브된 파일을 릴리스하는 릴리스 데이터 리포지토리 작업을 제공합니다. 파일을 릴리스하면 파일 목록과 메타데이터는 유지되지만 해당 파일 콘텐츠의 로컬 사본은 제거됩니다. 자세한 내용은 데이터 리포지토리 작업을 사용하여 파일 릴리스를 참조하세요.	2023년 8월 9일
Amazon FSx는 AmazonF 관리형 정책을 업데이트했습니다. SxService RolePolicy AWS	Amazon FSx가 `cloudwatch:PutMetricData` AmazonF의 권한을 업데이트했습니다. SxService RolePolicy 자세한 내용은 관리형 정책에 대한 Amazon FSx 업데이트를 AWS 참조하십시오.	2023년 7월 24일
Amazon FSx는 AmazonF Access 관리형 정책을 업데이트했습니다. SxFull AWS	Amazon FSx는 SxFullAccess AmazonF 정책을 업데이트하여 권한을 제거하고 특정 작업을 추가했습니다`fsx:*`. `fsx` 자세한 내용은 AmazonF 액세스 정책을 참조하십시오. SxFull	2023년 7월 13일
Amazon FSx는 AmazonF 관리형 정책을 업데이트했습니다. SxConsole FullAccess AWS	Amazon FSx는 SxConsole FullAccess AmazonF 정책을 업데이트하여 권한을 제거하고 특정 작업을 추가했습니다`fsx:*`. `fsx` 자세한 내용은 AmazonF 정책을 참조하십시오. SxConsole FullAccess	2023년 7월 13일

CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 8.8에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS, 록키 리눅스 및 레드햇 엔터프라이즈 리눅스 (RHEL) 8.8을 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2023년 5월 25일
AutoImport 및 AutoExport 지표에 대한 지원 추가	FSx for Lustre는 이제 데이터 리포지토리에 연결된 파일 시스템의 자동 가져오기 및 자동 내보내기 업데이트를 모니터링하는 CloudWatch Amazon 지표를 제공합니다. 자세한 내용은 Amazon을 통한 모니터링을 참조하십시오 CloudWatch.	2023년 3월 31일
영구_1 및 Scratch_2 배포 유형에 대한 DRA 지원이 추가되었습니다.	이제 데이터 리포지토리 연결을 생성하고 영구_1 또는 Scratch_2 배포 유형을 사용하여 데이터 리포지토리를 Lustre 2.12 파일 시스템에 연결할 수 있습니다. 자세한 내용은 Amazon FSx for Lustre에서 데이터 리포지토리 사용을 참조하세요.	2023년 3월 29일
CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 8.7에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS, 록키 리눅스 및 레드햇 엔터프라이즈 리눅스 (RHEL) 8.7을 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2022년 12월 5일

AWS 리전 Persistent_2 배포 유형에 대한 추가 지원이 추가되었습니다.	이제 유럽 (스톡홀름), 아시아 태평양 (홍콩), 아시아 태평양 (뭄바이) 및 아시아 태평양 (서울) 에서 FsX for Lustre 파일 시스템을 위한 차세대 Persistent_2 SSD FSx를 사용할 수 있습니다. AWS 리전자세한 내용은 FSx for Lustre 파일 시스템 배포 옵션을 참조하세요.	2022년 11월 10일
CentOS, 록키 리눅스, 레드햇 엔터프라이즈 리눅스 (RHEL) 8.6에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS, 록키 리눅스 및 레드햇 엔터프라이즈 리눅스 (RHEL) 8.6을 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2022년 9월 8일
Ubuntu 22에 대한 Lustre 클라이언트 지원이 추가되었습니다.	이제 FSx for Lustre 클라이언트는 Ubuntu 22.04를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2022년 7월 28일
Rocky Linux에 대한 Lustre 클라이언트 지원이 추가되었습니다.	이제 FSx for Lustre 클라이언트는 Rocky Linux를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2022년 7월 8일

Lustre 루트 스쿼시에 대한 지원이 추가되었습니다	이제 Lustre 루트 스쿼시 기능을 사용하여 FSx for Lustre 파일 시스템에 루트로 액세스하려는 클라이언트의 루트 수준 액세스를 제한할 수 있습니다. 자세한 내용은 Lustre 루트 스쿼시를 참조하세요.	2022년 5월 25일
AWS 리전 Persistent_2 배포 유형에 대한 추가 지원이 추가되었습니다.	이제 FsX 파일 시스템용 차세대 Persistent_2 SSD FSx를 유럽 (런던), 아시아 태평양 (싱가포르) 및 아시아 태평양 (시드니) 에서 사용할 수 있습니다. AWS 리전자세한 내용은 FSx for Lustre 파일 시스템 배포 옵션을 참조하세요.	2022년 4월 19일
Amazon FSx for AWS DataSync Lustre 파일 시스템으로 파일을 마이그레이션하는 데 사용할 수 있는 지원이 추가되었습니다.	이제 를 AWS DataSync 사용하여 기존 파일 시스템에서 FSx for Lustre 파일 시스템으로 파일을 마이그레이션할 수 있습니다. 자세한 내용은 AWS DataSync를 사용하여 기존 파일을 FSx for Lustre로 마이그레이션하는 방법을 참조하세요.	2022년 4월 5일
AWS PrivateLink 인터페이스 VPC 엔드포인트에 대한 지원 추가	이제 인터넷을 통해 트래픽을 보내지 않고 인터페이스 VPC 엔드포인트를 사용하여 VPC에서 Amazon FSx API에 액세스할 수 있습니다. 자세한 내용은 Amazon FSx 및 인터페이스 VPC 엔드포인트를 참조하세요.	2022년 4월 5일

Lustre DRA 대기열에 대한 지원 추가됨	이제 FSx for Lustre 파일 시스템을 생성할 때 데이터 리포지토리 연결(DRA)을 생성할 수 있습니다. 요청은 대기열에 추가되며 파일 시스템을 사용할 수 있게 되면 DRA가 생성됩니다. 자세한 내용은 S3 버킷에 파일 시스템 연결을 참조하세요.	2022년 2월 28일
CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.5에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.5를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2021년 12월 20일
FSx for Lustre의 변경 내용을 연결된 데이터 리포지토리로 내보내기 지원	이제 파일 시스템에서 새 파일, 변경 및 삭제된 파일을 연결된 Amazon S3 데이터 리포지토리로 자동으로 내보내도록 FSx for Lustre를 구성할 수 있습니다. 데이터 리포지토리 작업을 사용하여 데이터 및 메타데이터 변경 사항을 데이터 리포지토리로 내보낼 수 있습니다. 여러 데이터 리포지토리에 대한 링크를 구성할 수도 있습니다. 자세한 내용은 데이터 리포지토리로 변경 내용 내보내기를 참조하세요.	2021년 11월 30일

Lustre 로깅에 대한 지원 추가	이제 파일 시스템과 연결된 데이터 리포지토리의 오류 및 경고 이벤트를 Amazon Logs에 기록하도록 FSx for Lustre를 구성할 수 있습니다. CloudWatch 자세한 내용은 Amazon CloudWatch Logs를 사용한 로깅을 참조하십시오.	2021년 11월 30일
영구 SSD 파일 시스템은 더 높은 처리량과 더 작은 스토리지 용량을 지원합니다.	차세대 영구 SSD FSx for Lustre 파일 시스템은 처리량 옵션이 더 높고 최소 스토리지 용량은 더 낮습니다. 자세한 내용은 FSx for Lustre 파일 시스템 배포 옵션을 참조하세요.	2021년 11월 30일
Lustre 버전 2.12에 대한 지원이 추가되었습니다.	이제 Lustre용 FSx를 생성할 때 Lustre 버전 2.12를 선택할 수 있습니다. 자세한 내용은 1단계: Amazon FSx for Lustre 파일 시스템 생성을 참조하세요.	2021년 10월 5일
CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.4에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.4를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2021년 6월 9일

데이터 압축에 대한 지원 추가	이제 FSx for Lustre 파일 시스템을 만들 때 데이터 압축을 활성화할 수 있습니다. 기존 FSx for Lustre 파일 시스템에서 데이터 압축을 활성화하거나 비활성화할 수도 있습니다. 자세한 내용은 Lustre 데이터 압축을 참조하세요.	2021년 5월 27일
백업 복사에 대한 지원 추가	이제 Amazon FSx를 사용하여 동일한 백업에서 AWS 리전 다른 백업으로 (지역 간 사본) 또는 AWS 계정 동일한 백업 (리전 내 사본) 에 AWS 리전 복사할 수 있습니다. 자세한 내용은 백업 복사를 참조하세요.	2021년 4월 12일
Lustre 파일 세트에 대한 Lustre 클라이언트 지원	이제 FSx for Lustre 클라이언트는 파일 세트를 사용하여 파일 시스템 네임스페이스의 하위 집합만 마운트할 수 있습니다. 자세한 내용은 특정 파일 세트 마운트를 참조하세요.	2021년 3월 18일
프라이빗이 아닌 IP 주소를 사용한 클라이언트 액세스에 대한 지원 추가	프라이빗이 아닌 IP 주소를 사용하여 온프레미스 클라이언트에서 FSx for Lustre 파일 시스템에 액세스할 수 있습니다. 자세한 내용은 온프레미스 또는 피어링된 Amazon VPC에서 Amazon FSx 파일 시스템 마운트를 참조하세요.	2020년 12월 17일

ARM 기반 CentOS 7.9에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 ARM 기반 CentOS 7.9를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2020년 12월 17일
CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.3에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.3을 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2020년 12월 16일
스토리지 및 처리량 용량 확장에 대한 지원 추가	이제 스토리지 및 처리량 요구 사항이 발전함에 따라 기존 FSx for Lustre 파일 시스템의 스토리지 및 처리량 용량을 늘릴 수 있습니다. 자세한 내용은 스토리지 및 처리량 용량 관리 섹션을 참조하세요.	2020년 11월 24일
스토리지 할당량 지원 추가	이제 사용자와 그룹에 대한 스토리지 할당량을 생성할 수 있습니다. 스토리지 할당량은 사용자 또는 그룹이 FSx for Lustre 파일 시스템에서 사용할 수 있는 디스크 공간 및 파일 수를 제한합니다. 자세한 내용은 스토리지 할당량을 참조하세요.	2020년 11월 9일

Amazon FSx는 이제 다음과 통합되었습니다. AWS Backup	이제 기본 Amazon FSx 백업을 사용하는 AWS Backup 것 외에도 FSx 파일 시스템을 백업 및 복원하는 데 사용할 수 있습니다. 자세한 내용은 Amazon AWS Backup FSx와 함께 사용을 참조하십시오.	2020년 11월 9일
하드 디스크 드라이브(HDD) 스토리지 옵션에 대한 지원 추가	이제 FSx for Lustre는 솔리드 스테이트 드라이브(SSD) 스토리지 옵션 외에도 하드 디스크 드라이브(HDD) 스토리지 옵션을 지원합니다. 일반적으로 대용량의 순차적인 파일 작업이 필요한 처리량 집약적인 워크로드에 HDD를 사용하도록 파일 시스템을 구성할 수 있습니다. 자세한 내용은 다중 스토리지 옵션을 참조하세요.	2020년 8월 12일
연결된 데이터 리포지토리 변경 내용을 FSx for Lustre로 가져오기 지원	이제 파일 시스템 생성 후 연결된 데이터 리포지토리에서 추가된 새 파일과 변경된 파일을 자동으로 가져오도록 FSx for Lustre 파일 시스템을 구성할 수 있습니다. 자세한 내용은 데이터 리포지토리에서 업데이트 자동 가져오기를 참조하세요.	2020년 7월 23일
SUSE Linux SP4 및 SP5에 대한 Lustre 클라이언트 지원이 추가되었습니다.	이제 FSx for Lustre 클라이언트는 SUSE Linux SP4 및 SP5를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2020년 7월 20일

CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.2에 대한 Lustre 클라이언트 지원이 추가되었습니다.	FSx for Lustre 클라이언트는 이제 CentOS와 레드햇 엔터프라이즈 리눅스 (RHEL) 8.2를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2020년 7월 20일
자동 및 수동 파일 시스템 백업 지원 추가	이제 Amazon S3의 내구성 데이터 리포지토리에 연결되지 않은 파일 시스템의 자동 일일 백업 및 수동 백업을 수행할 수 있습니다. 자세한 내용은 백업 작업을 참조하세요.	2020년 6월 23일
두 가지 새로운 파일 시스템 배포 유형이 출시되었습니다.	스크래치 파일 시스템은 데이터의 임시 저장 및 단기 처리를 위해 설계되었습니다. 영구 파일 시스템은 장기 스토리지 및 워크로드를 위해 설계되었습니다. 자세한 내용은 FSx for Lustre 배포 옵션을 참조하세요.	2020년 2월 12일
POSIX 메타데이터 지원 추가	FSx for Lustre는 Amazon S3의 연결된 영구 데이터 리포지토리로 파일을 가져오고 내보낼 때 관련 POSIX 메타데이터를 유지합니다. 자세한 내용은 데이터 리포지토리에 대한 POSIX 메타데이터 지원을 참조하세요.	2019년 12월 23일

새 데이터 리포지토리 작업 기능 출시	이제 데이터 리포지토리 작업을 사용하여 Amazon S3의 연결된 내구성 데이터 리포지토리로 변경된 데이터 및 관련 POSIX 메타데이터를 내보낼 수 있습니다. 자세한 내용은 데이터 리포지토리 작업을 사용한 데이터 및 메타데이터 전송을 참조하세요.	2019년 12월 23일
AWS 리전 추가 지원이 추가되었습니다.	이제 유럽(런던) 리전 AWS 리전에서 FSx for Lustre를 사용할 수 있습니다. FSx for Lustre 지역별 한도는 한도를 참조하세요.	2019년 7월 9일
추가 AWS 리전 지원 추가	이제 아시아 태평양 (싱가포르) 에서 FSx for Lustre를 사용할 수 있습니다. AWS 리전 FSx for Lustre 지역별 한도는 한도를 참조하세요.	2019년 6월 26일
Amazon Linux 및 Amazon Linux 2에 대한 Lustre 클라이언트 지원 추가	이제 FSx for Lustre 클라이언트는 Amazon Linux 및 Amazon Linux 2를 실행하는 Amazon EC2 인스턴스를 지원합니다. 자세한 내용은 Lustre 클라이언트 설치를 참조하세요.	2019년 3월 11일

사용자 정의 데이터 내보내기 경로 지원이 추가되었습니다.	이제 사용자는 Amazon S3 버킷의 원본 객체를 덮어쓰거나 새 파일 또는 변경된 파일을 지정한 접두사에 쓸 수 있습니다. 이 옵션을 사용하면 FSx for Lustre를 데이터 처리 워크플로우에 통합할 수 있는 유연성이 더욱 높아집니다. 자세한 내용은 Amazon S3 버킷에 데이터 내보내기를 참조하세요.	2019년 2월 6일
총 스토리지 기본 한도가 증가했습니다.	모든 FSx for Lustre 파일 시스템의 기본 총 스토리지는 100,800GiB로 증가했습니다. 자세한 내용은 제한을 참조하세요.	2019년 1월 11일
이제 Amazon FSx for Lustre는 정식 출시되었습니다	Amazon FSx for Lustre는 고성능 컴퓨팅, 기계 학습, 미디어 처리 워크플로와 같은 컴퓨팅 집약적 워크로드에 최적화된 완전 관리형 파일 시스템입니다.	2018년 11월 28일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.