DELL Technologies

기술 백서

ECS 개요 및 아키텍처

백서 소개

이 문서에서는 Dell EMC™ ECS™ 소프트웨어 정의 클라우드 스케일 오브젝트 스토리지 플랫폼의 기술 개요 및 설계에 대해 설명합니다.

2021 년 2 월

개정 내역

날짜	설명
2015 년 12 월	최초 릴리스
2016 년 5 월	2.2.1 업데이트
2016 년 9 월	3.0 업데이트
2017 년 8 월	3.1 업데이트
2018 년 3 월	3.2 업데이트
2018 년 9 월	Gen3 하드웨어 업데이트
2019 년 2 월	3.3 업데이트
2019 년 9 월	3.4 업데이트
2020 년 2 월	업데이트된 ECSDOC-628 변경 사항
2020 년 5 월	3.5 업데이트
2020 년 11 월	3.6 업데이트
2021 년 2 월	3.6.1 업데이트

감사의 말

이 백서는 다음에 의해 작성되었습니다.

작성자: Zhu, Jarvis

목차

개정 내역 2
감사의 말 2
목차 3
핵심 요약 6
1 소개 7
1.1 대상 7
1.2 범위 7
2 ECS 의 가치 8
3 아키텍처 11
3.1 개요 11
3.2 ECS 포털 및 프로비저닝 서비스 12
3.3 데이터 서비스 14
3.3.1 객체 15
3.3.2 HDFS 16
3.3.3 NFS 19
3.3.4 커넥터 및 게이트웨이 20
3.4 스토리지 엔진 20
3.4.1 스토리지 서비스 20
3.4.2 데이터 21
3.4.3 데이터 관리 22
3.4.4 데이터 흐름 24
3.4.5 파일 크기를 위한 쓰기 최적화 26
3.4.6 공간 재확보 26
3.4.7 SSD 메타데이터 캐싱 27
3.4.8 클라우드 DVR 28
3.5 Fabric 29
3.5.1 노드 에이전트 29
3.5.2 수명주기 관리자 30

3.5.3 레지스트리 30
3.5.4 이벤트 라이브러리 30
3.5.5 하드웨어 관리자 30
3.6 인프라스트럭처 30
3.6.1 Docker 31
4 어플라이언스 하드웨어 모델 33
4.1 EX-Series 33
4.2 어플라이언스 네트워킹 35
4.2.1 S5148F - 프런트엔드 퍼블릭 스위치 35
4.2.2 S5148F - 백엔드 프라이빗 스위치 36
4.2.3 S5248F - 프런트엔드 퍼블릭 스위치 37
4.2.4 S5248F - 백엔드 프라이빗 스위치 37
4.2.5 S5232 - 통합 스위치 38
5 네트워크 분리 39
6 보안 41
6.1 인증 41
6.2 데이터 서비스 인증 42
6.3 D@RE(Data-At-Rest Encryption) 42
6.3.1 키 순환 43
6.4 ECS IAM 44
6.5 Object tagging 45
6.5.1 오브젝트 태그 지정에 대한 추가 정보 46
7 데이터 무결성 및 보호 47
7.1 규정 준수 48
8 배포 49
8.1 단일 사이트 구축 51
8.2 멀티 사이트 구축 52
8.2.1 데이터 정합성 53
8.2.2 액티브 복제 그룹 53
8.2.3 패시브 복제 그룹 54
8.2.4 원격 데이터 원거리 캐싱 57

8.2.5 사이트 운영 중단 중 동작 57
8.3 내결함성 59
8.4 디스크 교체 자동화 62
8.5 Tech Refresh 63
9 스토리지 보호 오버헤드 64
10 결론 67
A 기술 지원 및 리소스 68

핵심 요약

조직에는 프라이빗 클라우드 인프라스트럭처의 신뢰성과 제어 기능을 갖춘 퍼블릭 클라우드 서비스를 사용할 수 있는 옵션이 필요합니다. Dell EMC ECS 는 S3, Atmos, CAS, Swift, NFSv3 및 HDFS 스토리지 서비스를 단일 모던 플랫폼에서 제공하는 소프트웨어 정의, IPv6 지원, 클라우드 규모, 오브젝트 스토리지 플랫폼입니다.

ECS 를 사용하는 관리자는 장소에 구애받지 않고 콘텐츠에 액세스할 수 있는 단일 글로벌 네임스페이스를 이용해 전사적으로 분산된 인프라스트럭처를 쉽게 관리할 수 있습니다. ECS 핵심 구성 요소는 유연성과 복원력을 위해 계층화되어 있습니다. 각 계층은 추상화되어 있으며 고가용성 덕분에 개별적인 확장이 가능합니다.

개발자들은 스토리지 서비스에 대해 간단한 RESTful API 액세스를 채택하고 있습니다. GET 및 PUT 같은 HTTP 의미 체계를 사용하면 기존의 익숙한 경로 기반 파일 작업과 비교했을 때 필요한 애플리케이션 로직이 단순해집니다. 또한 ECS 의 기본 스토리지 시스템은 강력한 정합성을 유지하므로 신뢰할 수 있는 응답을 보장할 수 있습니다. 신뢰할 수 있는 데이터 전달을 보장해야 하는 애플리케이션은 복잡한 코드 로직 없이 ECS 를 사용하여 이를 수행할 수 있습니다.

1 소개

이 문서는 Dell EMC ECS 오브젝트 스토리지 플랫폼에 대해 간략하게 설명합니다. 또한 ECS 설계 아키텍처를 비롯해 스토리지 서비스 및 데이터 보호 메커니즘 같은 핵심 구성 요소에 대해 자세히 설명합니다.

1.1 대상

이 백서는 ECS 의 가치와 아키텍처에 관심이 있는 모든 사용자를 대상으로 합니다. 추가 정보에 대한 링크와 함께 관련된 설명을 제공하는 것을 목표로 합니다.

1.2 범위

이 문서에서는 주로 ECS 아키텍처를 다룹니다. ECS 소프트웨어 또는 하드웨어의 설치, 관리, 업그레이드 절차는 다루지 않습니다. ECS API 를 통해 애플리케이션을 사용하고 제작하는 구체적인 내용도 다루지 않습니다.

이 문서는 일반적으로 주요 릴리스 또는 새로운 기능이 발표되는 시점에 주기적으로 업데이트됩니다.

2 ECS 의 가치

ECS 는 빠른 데이터 증가를 지원하도록 설계된 플랫폼을 찾는 기업 및 서비스 공급업체에 상당한 가치를 제공합니다. 기업이 대규모 분산 콘텐츠를 전사적으로 관리 및 저장할 수 있도록 돕는 ECS 의 주요 이점과 특징은 다음과 같습니다.

- **클라우드 스케일** - ECS 는 기존 및 차세대 워크로드 모두를 위한 오브젝트 스토리지 플랫폼입니다. ECS 의 소프트웨어 정의 계층형 아키텍처는 무제한의 확장성을 지원합니다. 주요 특징은 다음과 같습니다.
  - 전사적으로 분산된 오브젝트 인프라스트럭처
  - 스토리지 풀, 클러스터 또는 페더레이션된 환경 용량에 제한이 없는 엑사바이트 이상의 확장성
  - 시스템, 네임스페이스 또는 버킷에서 오브젝트 수에 제한이 없음
  - 오브젝트 크기에 제한 없이 크고 작은 파일 워크로드 모두에서 뛰어난 효율성 제공
- **유연한 구축** - ECS 는 다음과 같은 기능으로 탁월한 유연성을 제공합니다.
  - 어플라이언스 구축
  - 인증 또는 맞춤형 업계 표준 하드웨어를 지원하는 소프트웨어 전용 구축
  - 멀티 프로토콜 지원: 오브젝트(S3, Swift, Atmos, CAS) 및 파일(HDFS, NFSv3)
  - 다양한 워크로드: 최신 애플리케이션 및 장기간 아카이브
  - CloudPools 를 사용하는 Data Domain Cloud Tier 및 Isilon 을 위한 보조 스토리지
  - 현재 세대 ECS 모델로의 운영 중단 없는 업그레이드 경로
- **엔터프라이즈급** - ECS 는 다음과 같은 기능을 통해 사용자가 안전한 호환 시스템 내에서 엔터프라이즈급 스토리지를 사용하여 데이터 자산을 더 효과적으로 제어할 수 있도록 합니다.
  - 키 순환 및 외부 키 관리를 사용하는 D@RE(Data-at-Rest Encryption)
  - 암호화된 사이트 간 통신
  - 기본적으로 포트 9101/9206 을 비활성화하여 조직이 규정 준수 정책을 충족하도록 지원
  - 법적 증거 자료 보존과 최소-최대 거버넌스 등의 고급 보존 관리를 포함하여 SEC 규칙 17a-4(f) 규정 준수를 위한 보고, 정책 및 이벤트 기반 레코드 보존 및 플랫폼 강화
  - DISA(Defense Information Systems Agency) STIG(Security Technical Implementation Guide) 보안 강화 지침 준수
  - Active Directory 및 LDAP 를 사용하는 인증, 권한 부여, 액세스 제어
  - 모니터링 및 알림 인프라스트럭처(SNMP 트랩 및 SYSLOG)와의 통합

  - 향상된 엔터프라이즈 기능(멀티 테넌시, 용량 모니터링, 알림)

- **TCO 절감** - ECS 는 기존 스토리지와 퍼블릭 클라우드 스토리지에 비해 TCO(Total Cost of Ownership)를 크게 줄일 수 있습니다. 장기간 보존의 경우 TCO 가 테이프보다도 더 낮습니다. 다음과 같은 기능이 포함되어 있습니다.

  - 글로벌 네임스페이스
  - 소용량 및 대용량 파일 성능
  - 원활한 Centera 마이그레이션
  - Atmos REST 와의 완벽한 호환
  - 관리 부담 감소
  - 데이터 센터 상면 감소
  - 스토리지 활용도 향상

ECS 설계는 다음과 같은 기본 활용 사례에 최적화되어 있습니다.

- **최신 애플리케이션** - ECS 는 차세대 웹, 모바일 및 클라우드 애플리케이션과 같은 최신 개발에 맞도록 설계되었습니다. 강력한 정합성이 보장되는 스토리지 덕분에 애플리케이션 개발이 간단해집니다. 다중 사이트, 동시 다중 사용자 읽기/쓰기 액세스 덕분에 ECS 용량이 변경되고 증가하더라도 개발자는 애플리케이션을 다시 코딩할 필요가 없습니다.
- **보조 스토리지** - ECS 는 자주 액세스하지 않는 데이터를 운영 스토리지에서 비우면서도 필요할 때 액세스 가능하도록 유지하기 위한 보조 스토리지로 사용됩니다. Data Domain Cloud Tier 및 Isilon CloudPools 와 같은 정책 기반 계층화 제품을 예로 들 수 있습니다. Windows 기반 애플리케이션인 GeoDrive 는 Windows 시스템이 ECS 에 직접 액세스하여 데이터를 저장할 수 있도록 지원합니다.
- **원거리 보호가 지원되는 아카이브** - ECS 는 아카이브 및 장기간 보존 목적을 위한 안전하고 경제적인 온프레미스 클라우드 역할을 합니다. ECS 를 아카이브 계층으로 사용하면 운영 스토리지 용량을 대폭 절감할 수 있습니다. 콜드 아카이브 활용 사례에서 스토리지 효율성을 높이기 위해 기본값인 12+4 외에 10+2 EC(Erasure Coding) 스키마를 사용할 수도 있습니다.
- **글로벌 콘텐츠 저장소** - 이미지 및 비디오 같은 데이터를 포함하는 비정형 콘텐츠 저장소가 고비용 스토리지 시스템에 보관되어 기업이 대규모 데이터 증가를 비용 효율적으로 관리할 수 없는 경우가 종종 있습니다. ECS 를 사용하면 전사적 액세스가 가능한 효율적인 단일 콘텐츠 저장소로 여러 스토리지 시스템을 통합할 수 있습니다.
- **IoT 용 스토리지** - IoT(Internet of Things)는 고객 데이터에서 가치를 이끌어내는 기업에 새로운 수익 창출 기회를 제공합니다. ECS 는 대규모의 비정형 데이터 수집을 위한 효율적인 IoT 아키텍처를 제공합니다. 오브젝트 수, 오브젝트 크기, 맞춤형 메타데이터에 제한이 없는 ECS 는 IoT 데이터를 저장하기에 이상적인 플랫폼입니다. 시간이 많이 걸리는 ETL(Extract, Transform and Load) 프로세스를 거칠 필요 없이 ECS 플랫폼에서 바로 데이터를 분석할 수 있으므로 일부 분석 워크플로도 간소화됩니다. Hadoop 클러스터는 S3 또는 NFS 와 같은 다른 프로토콜 API 가 ECS 에 저장한 데이터를 사용하여 쿼리를 실행할 수 있습니다.
- **영상 관제 증거 저장소** - IoT 데이터와 달리, 영상 관제 데이터는 오브젝트 저장 횟수가 훨씬 적지만 파일당 차지하는 용량 상면은 훨씬 더 큽니다. 데이터 신뢰성은 중요하지만 데이터 보존은 그렇게 중요하지 않습니다. ECS 는 이러한 데이터를 위한 경제적인 보관 영역 또는 보조 스토리지 공간이 될 수 있습니다. 비디오 관리 소프트웨어는 카메라 위치, 보존 요구 사항, 데이터 보호 요구 사항 등 중요한 세부 정보 태그를 파일에 지정하는 풍부한 맞춤형 메타데이터 기능을 활용할 수 있습니다. 또한 메타데이터를 사용하여 파일을 읽기 전용 상태로 설정함으로써 파일의 증거 보존 체계를 구현할 수 있습니다.
- **데이터 레이크 및 분석** - 데이터와 분석은 경쟁력 차별화 요소이면서 조직에서 가치를 창출하는 주 원천이 되었습니다. 그러나 데이터를 가치 있는 기업 자산으로 변환하는 작업은 대개 수십 개의 기술과 툴, 환경의 사용이 수반되기 때문에 매우 복잡한 사안이라 할 수 있습니다. ECS 는 고객이 규모와 관계없이 데이터를 수집, 저장, 관리 및 분석할 수 있도록 지원하는 일련의 서비스를 제공합니다.

3 아키텍처

ECS 는 강력한 정합성이 보장되는 글로벌 네임스페이스, 스케일 아웃 기능, 안전한 멀티 테넌시, 오브젝트 크기를 가리지 않는 뛰어난 성능 등 몇 가지 핵심 설계 원칙을 바탕으로 설계되었습니다. ECS 는 클라우드 애플리케이션의 원칙에 따라 완전히 분산된 시스템으로 구축되었고 시스템의 모든 기능은 독립적인 계층으로 구축되어 있습니다. 이러한 설계에서 각 계층은 시스템의 모든 노드에서 수평적으로 확장될 수 있습니다. 리소스는 가용성을 높이고 로드를 공유하기 위해 모든 노드에 걸쳐 분산됩니다.

이 섹션에서는 소프트웨어와 하드웨어의 ECS 아키텍처 및 설계에 대해 자세히 다룰 것입니다.

3.1 개요

ECS 는 검증된 업계 표준 하드웨어 세트에 구축되거나 턴키 스토리지 어플라이언스로 구축됩니다. ECS 의 주요 구성 요소는 다음과 같습니다.

- **ECS 포털 및 프로비저닝 서비스** - ECS 노드의 셀프 서비스, 자동화, 보고 및 관리를 위한 API 기반 WebUI 및 CLI 입니다. 이 계층은 라이선싱, 인증, 멀티 테넌시 그리고 네임스페이스 생성 같은 프로비저닝 서비스도 처리합니다.
- **데이터 서비스** - 시스템에 대한 오브젝트 및 파일 액세스를 지원하는 여러 서비스, 툴, API 입니다.
- **스토리지 엔진** - 데이터 저장 및 검색, 트랜잭션 관리, 로컬 및 사이트 간 데이터 보호 및 복제를 담당하는 핵심 서비스입니다.
- **패브릭** - 상태, 구성, 업그레이드 관리 및 알림을 위한 클러스터링 서비스입니다.
- **인프라스트럭처** - 턴키 어플라이언스의 기본 운영 체제로 SUSE Linux Enterprise Server 12 를 사용하거나, 업계 표준 하드웨어 구성으로 검증된 Linux 운영 체제를 사용합니다.
- **하드웨어** - 턴키 어플라이언스 또는 검증된 업계 표준 하드웨어입니다.

그림 1 은 이후 섹션에서 자세히 설명하고 있는 계층을 그래픽으로 보여줍니다.

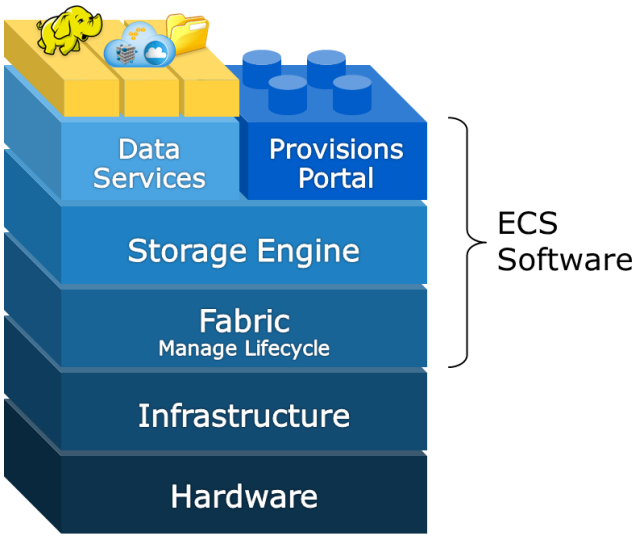


그림 1 ECS 아키텍처 계층

3.2 ECS 포털 및 프로비저닝 서비스

스토리지 관리자는 ECS 포털 및 프로비저닝 서비스를 사용하여 ECS 를 관리합니다. ECS 는 ECS 노드를 관리, 라이선싱, 프로비저닝하기 위한 웹 기반 GUI(WebUI)를 제공합니다. 이 포털은 다음과 같은 포괄적인 보고 기능을 제공합니다.

- 각 사이트, 스토리지 풀, 노드, 디스크별 용량 활용도
- 레이턴시, 처리량, 복제 진행 상황에 대한 성능 모니터링
- 노드 및 디스크 복구 상태와 같은 진단 정보

ECS 대시보드는 전반적인 시스템 레벨 상태 및 성능 정보를 제공합니다. 이 통합된 뷰는 전반적인 시스템 가시성을 향상시킵니다. 알림은 용량 제한, 할당량 제한, 디스크 또는 노드 장애, 소프트웨어 장애와 같은 중요한 이벤트를 사용자에게 알립니다. ECS 는 ECS 를 설치, 업그레이드, 모니터링하기 위한 명령줄 인터페이스도 제공합니다. 명령줄을 사용하기 위한 노드는 SSH 를 통해 액세스합니다. ECS 대시보드의 스크린샷이 아래 그림 2 에 나와 있습니다.

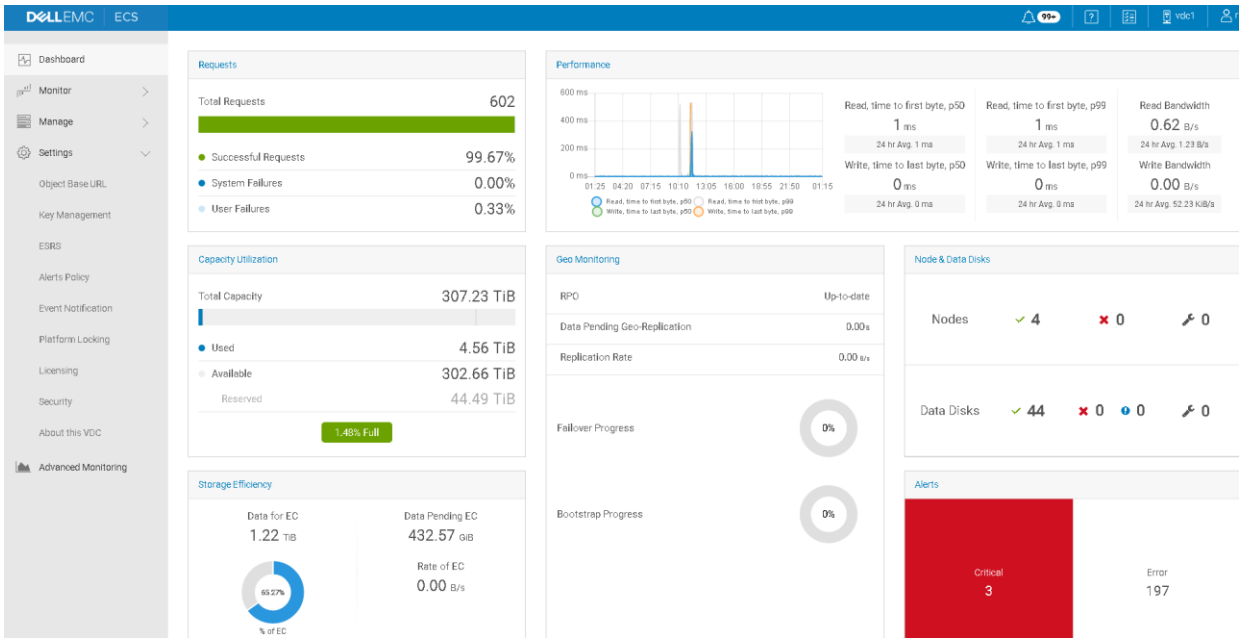


그림 2 ECS 웹 UI 대시보드

상세한 성능 보고서는 고급 모니터링 폴더 아래의 UI 에서 확인할 수 있습니다. 해당 보고서는 Grafana 대시보드에 표시됩니다. 필터를 사용하여 지정된 네임스페이스, 프로토콜 또는 노드를 자세히 살펴볼 수 있습니다. S3 프로토콜 성능 보고서의 예는 아래 그림 3 에 나와 있습니다.

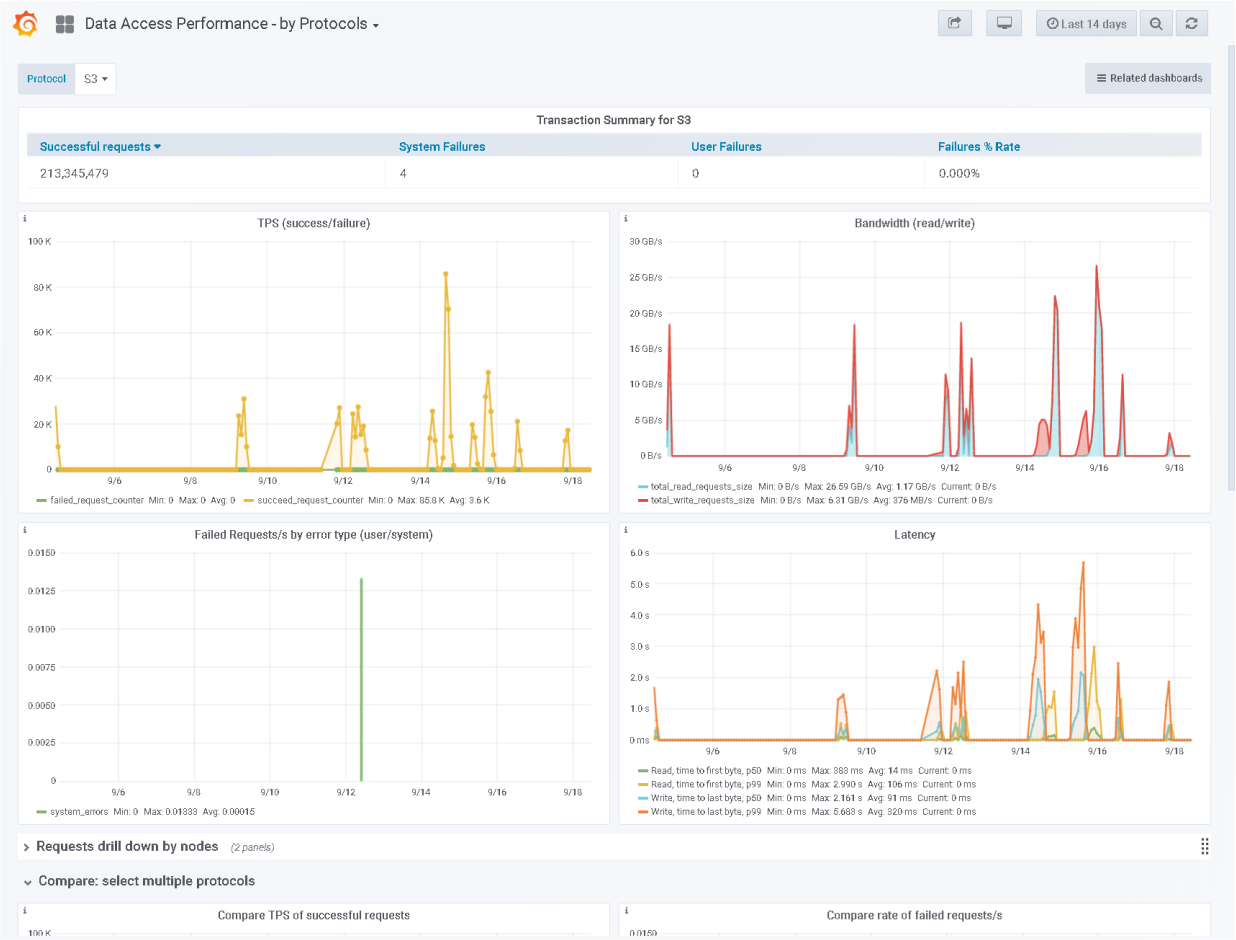


그림 3　Grafana 를 사용하는 고급 모니터링 시각화

RESTful API 를 사용하여 ECS 를 관리할 수도 있습니다. 사용자는 관리 API 를 통해 자신의 툴, 스크립트, 신규 또는 기존 애플리케이션 내에서 ECS 를 관리할 수 있습니다. ECS 웹 UI 및 명령줄 툴은 ECS REST 관리 API 를 사용하여 설계됩니다.

ECS 는 웹 UI, API 또는 CLI 를 사용하여 설정할 수 있는 다음과 같은 이벤트 알림 서버를 지원합니다.

- SNMP(Simple Network Management Protocol) 서버
- Syslog 서버

알림 서비스를 구성하는 방법에 대한 자세한 내용은 *ECS 관리자 가이드*를 참조하십시오.

3.3 데이터 서비스

ECS 스토리지 서비스에 액세스하려면 표준 오브젝트 및 파일 방식을 사용합니다. S3, Atmos 및 Swift 의 경우 HTTP 를 통한 RESTful API 를 사용하여 액세스합니다. CAS(Content Addressable Storage)의 경우 전용 액세스 방식/SDK 가 사용됩니다. ECS 는 LINK 를 제외한 모든 NFSv3 프로시저를 기본적으로 지원합니다. 이제 S3a 에서 ECS 버킷에 액세스할 수 있습니다.

ECS 는 한 프로토콜을 통해 수집한 데이터를 다른 프로토콜을 통해 액세스할 수 있는 멀티 프로토콜 액세스를 제공합니다. 즉, 데이터를 S3 를 통해 수집하고 NFSv3 또는 Swift 를 통해 수정할 수 있고 그 반대로도 가능합니다. 프로토콜 의미 체계와 프로토콜 설계 표현으로 인해 멀티 프로토콜 액세스에는 몇 가지 예외가 있습니다. 표 1 은 액세스 방식 및 상호 운용되는 프로토콜을 요약해 보여줍니다.

표 1 ECS 지원 데이터 서비스 및 프로토콜 상호 운용성

Protocols		**지원됨**	**상호 운용성**
객체	S3	바이트 범위 업데이트, 리치 ACL 등의 추가 기능	HDFS, NFS, Swift
	Atmos	버전 2.0	NFS(경로 기반 오브젝트만 해당, 오브젝트 ID 스타일 기반은 아님)
	Swift	V2 API, Swift 및 Keystone v3 인증	HDFS, NFS, S3
	CAS	SDK v3.1.544 이상	해당 없음
File	HDFS	Hadoop 2.7 호환성	S3, NFS, Swift
	NFS	NFSv3	S3, Swift, HDFS, Atmos(경로 기반 오브젝트만 해당, 오브젝트 ID 스타일 기반은 아님)

헤드 서비스라고도 하는 데이터 서비스는 클라이언트 요청을 받고 필요한 정보를 추출하여 추가 처리를 위해 스토리지 엔진에 전달하는 역할을 합니다. 모든 헤드 서비스는 인프라스트럭처 계층 내에서 실행되는 *dataheadsvc* 라는 단일 프로세스에 결합됩니다. 이 프로세스는 *object-main* 이라는 Docker 컨테이너 내에 추가로 캡슐화되며, 이 컨테이너는 ECS 의 모든 노드에서 실행됩니다. 이 문서의 *인프라스트럭처* 섹션에서 Docker 를 자세히 다룹니다. S3 통신을 위한 포트 9020 과 같은 ECS 프로토콜 서비스 포트 요구 사항은 최신 *ECS 보안 구성 가이드*에서 확인할 수 있습니다.

3.3.1 객체

ECS 는 오브젝트 액세스를 위해 S3, Atmos, Swift, CAS API 를 지원합니다. CAS 를 제외하고 오브젝트나 데이터는 GET, POST, PUT, DELETE, HEAD 의 HTTP 또는 HTTPS 호출을 통해 작성, 검색, 업데이트 및 삭제됩니다. CAS 의 경우 표준 TCP 통신과 특정 액세스 방법 및 호출이 사용됩니다.

ECS 는 풍부한 쿼리 언어를 사용하여 오브젝트에 대한 메타데이터 검색 기능을 제공합니다. 이 기능은 S3 오브젝트 클라이언트가 시스템 및 맞춤형 메타데이터를 사용하여 버킷 내의 오브젝트를 검색할 수 있도록 하는 ECS 의 강력한 기능입니다. 모든 메타데이터를 통해 검색이 가능하지만, 그 중에서도 버킷에서 인덱싱되도록 특별히 구성된 메타데이터를 검색할 경우 ECS 에서 특히 수십억 개의 오브젝트가 있는 버킷에 대해 쿼리를 더 빠르게 반환할 수 있습니다.

버킷당 최대 30 개의 사용자 정의 메타데이터 필드를 인덱싱할 수 있습니다. 메타데이터는 버킷 생성 시 지정됩니다. 메타데이터 검색 기능은 서버 측 암호화를 사용하도록 설정된 버킷에서 사용할 수 있습니다. 그러나 검색 키로 활용되는 인덱싱된 사용자 메타데이터 특성은 암호화되지 않습니다.

참고: 메타데이터를 인덱싱하도록 구성된 버킷에 데이터를 쓰면 성능에 영향을 미칩니다. 인덱싱된 필드 수가 증가함에 따라 작업에 미치는 영향이 커집니다. 이처럼 성능에 영향을 미치므로 버킷에서 메타데이터를 인덱싱할 것인지, 그렇다면 몇 개의 인덱스를 유지할 것인지를 신중하게 선택해야 합니다.

CAS 오브젝트의 경우, CAS 쿼리 API 는 CAS 오브젝트에 대해 유지되며 명시적으로 사용 설정할 필요가 없는 메타데이터를 기준으로 오브젝트를 검색하는 유사한 기능을 제공합니다.

ECS API 및 메타데이터 검색용 API 에 대한 자세한 내용은 최신 *ECS 데이터 액세스 가이드*를 참조하십시오. Atmos 및 S3 SDK 에 대해서는 GitHub 사이트 Dell EMC 데이터 서비스 SDK 또는 Dell EMC ECS 를 참조하십시오. CAS 에 대해서는 Centera 커뮤니티 사이트를 참조하십시오. 개발자를 위한 여러 가지 예제, 리소스, 지원은 ECS 커뮤니티에서 이용하실 수 있습니다.

S3 Browser 및 Cyberduck 같은 클라이언트 애플리케이션은 ECS 에 저장된 데이터를 빠르게 테스트하거나 해당 데이터에 액세스할 수 있는 방법을 제공합니다. 테스트 및 개발 목적으로 공개 ECS 시스템에 액세스할 수 있는 ECS 테스트 드라이브는 Dell EMC 에서 무료로 제공합니다. ECS 테스트 드라이브를 등록하면 REST 엔드포인트가 각 오브젝트 프로토콜에 대한 사용자 자격 증명과 함께 제공됩니다. 누구든지 ECS 테스트 드라이브를 사용하여 자신의 S3 API 애플리케이션을 테스트할 수 있습니다.

참고: ECS 에서는 버킷당 인덱싱할 수 있는 메타데이터 수만 30 개로 제한됩니다. 오브젝트당 저장되는 맞춤형 메타데이터의 총 개수에는 제한이 없으며, 빠른 조회를 위해 인덱싱되는 개수만 제한됩니다.

3.3.2 HDFS

ECS 는 Hadoop 파일 시스템 데이터를 저장할 수 있습니다. 따라서 Hadoop 분석이 사용하고 처리할 수 있는 빅데이터 저장소를 ECS 에 Hadoop 호환 파일 시스템으로 생성할 수 있습니다. HDFS 데이터 서비스는 Apache Hadoop 2.7 과 호환되며 세분화된 ACL 과 확장 파일 시스템 속성을 지원합니다.

ECS 는 Hortonworks(HDP 2.7)를 사용한 검증 및 테스트를 마쳤습니다. ECS 는 YARN, MapReduce, Pig, Hive/Hiveserver2, HBase, Zookeeper, Flume, Spark 및 Sqoop 와 같은 서비스도 지원합니다.

3.3.2.1 Hadoop S3A 지원

ECS 는 Hadoop 데이터를 저장하기 위한 Hadoop S3A 클라이언트를 지원합니다. S3A 는 공식 AWS(Amazon Web Services) SDK 를 기반으로 하는 Hadoop 의 오픈 소스 커넥터이며 수많은 Hadoop 관리자들이 HDFS 로 인해 겪고 있던 스토리지 확장 및 비용 문제를 해결할 목적으로 제작되었습니다. Hadoop S3A 는 퍼블릭, 하이브리드 또는 온프레미스 클라우드에 있는 모든 S3 호환 가능 오브젝트 저장소에 Hadoop 클러스터를 연결합니다.

참고: S3A 지원은 Hadoop 2.7 이상 버전에서 사용할 수 있습니다.

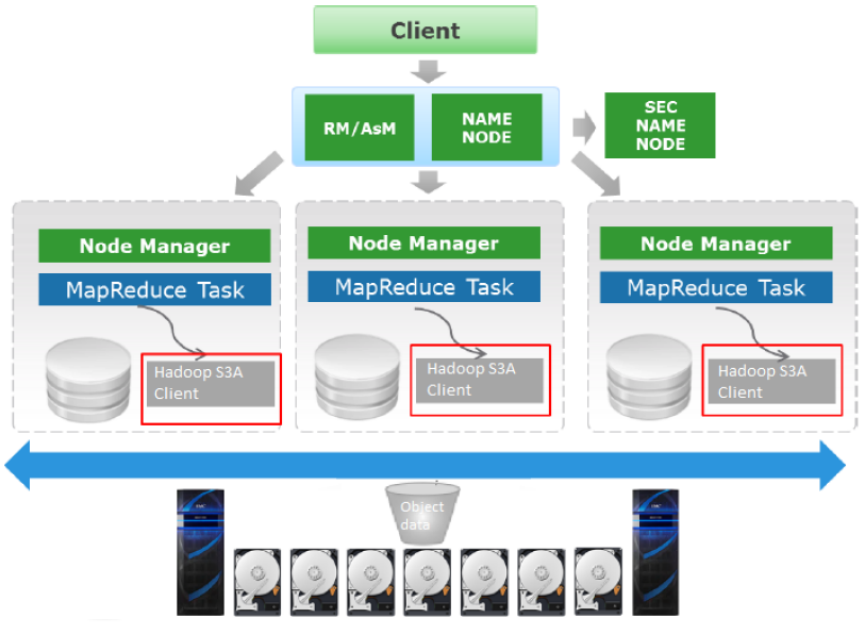


그림 4 Hadoop 및 ECS 아키텍처

그림 4 와 같이 고객이 기존 HDFS 에 Hadoop 클러스터를 설정할 때 S3A 구성은 모든 HDFS 작업을 수행하기 위해 ECS 오브젝트 데이터를 가리킵니다. 각 Hadoop HDFS 노드에서 기존의 모든 Hadoop 구성 요소는 Hadoop 의 S3A 클라이언트를 사용하여 HDFS 작업을 수행합니다.

ECS 서비스 콘솔을 사용하는 Hadoop 구성 분석

ECS SC(Service Console)는 S3A 의 ECS 에 대한 연결을 고려하여 Hadoop 구성 매개변수를 읽고 해석할 수 있습니다. 또한 SC 는 Hadoop 클러스터 구성을 읽고 S3A 설정에서 오타, 오류 및 값을 확인하는 함수인 *Get_Hadoop_Config* 를 제공합니다. ECS SC 설치에 대한 지원을 받으려면 ECS 지원 팀에 문의하십시오.

Hadoop S3A 를 사용한 Privacera 구현

Privacera 는 Hadoop 클라이언트 측 에이전트를 구현하고 Ambari for S3(AWS 및 ECS)의 세분화된 보안과 통합한 타사 공급업체입니다. Privacera 는 CDH(Cloudera Distribution of Hadoop)를 지원하지만 다른 타사 공급업체인 Cloudera 는 CDH 에서 Privacera 를 지원하지 않습니다.

참고: CDH 사용자는 ECS IAM 보안 서비스를 사용해야 합니다. ECS IAM 을 사용하지 않고 S3A 에 안전하게 액세스하려면 지원 팀에 문의하십시오.

S3A 지원에 대한 자세한 내용은 최신 *ECS 데이터 액세스 가이드*를 참조하십시오.

Hadoop S3A 보안

ECS IAM 을 사용하면 Hadoop 관리자가 액세스 정책을 설정하여 S3A Hadoop 데이터에 대한 액세스를 제어할 수 있습니다. 액세스 정책이 정의되면 Hadoop 관리자는 다음 두 가지 사용자 액세스 옵션을 구성해야 합니다.

- IAM 사용자/그룹
  - 정책에 연결되는 IAM 그룹 생성
  - IAM 그룹의 구성원인 IAM 사용자 생성
- SAML 어설션(페더레이션 사용자)
  - 정책에 연결되는 IAM 역할 생성
  - AD 그룹을 IAM 역할에 매핑하는 ID 공급자(AD FS)와 ECS 간의 CrossTrustRelationship 구성

ECS 관리자와 Hadoop 관리자는 함께 협력하여 적절한 정책을 미리 정의해야 합니다. 이어지는 가상의 예시에서는 정책 생성의 대상이 되는 세 가지 유형의 Hadoop 사용자를 간략하게 설명합니다. 다음과 같습니다.

- **Hadoop 관리자 -** 버킷 생성 및 버킷 삭제를 제외한 모든 작업을 수행합니다.
- **Hadoop 고급 사용자 -** 버킷 생성, 버킷 삭제 및 오브젝트 삭제를 제외한 모든 작업을 수행합니다.
- **Hadoop 읽기 전용 사용자 -** 오브젝트를 나열하고 읽을 수만 있습니다.

ECS IAM 에 대한 자세한 내용은 44 페이지에서 ESC IAM 을 참조하십시오.

3.3.2.2 ECS HDFS 클라이언트 지원

ECS 는 Ambari 와 통합되었기 때문에 ECS HDFS 클라이언트 jar 파일을 손쉽게 배포하고 ECS HDFS 를 Hadoop 클러스터의 기본 파일 시스템으로 지정할 수 있습니다. jar 파일은 참여 중인 Hadoop 클러스터 내의 각 노드에 설치됩니다. ECS 는 Hadoop 구축에서 이름 및 데이터 노드가 수행하는 것과 동일한 파일 시스템 및 스토리지 기능을 제공합니다. ECS 는 로컬 Hadoop DAS 로 데이터를 마이그레이션해야 할 필요를 없애거나 최소 세 개의 복제본을 생성함으로써 Hadoop 워크플로를 간소화합니다. 아래의 그림 5 는 각 Hadoop 컴퓨팅 노드에 설치된 ECS HDFS 클라이언트 jar 파일과 일반적인 통신 흐름을 보여줍니다.

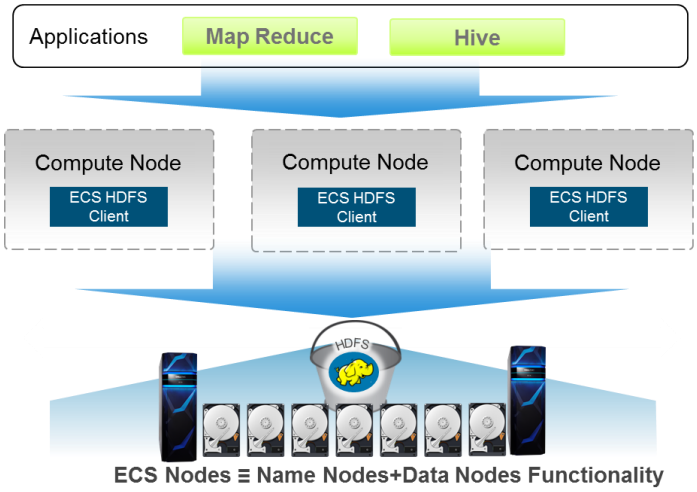


그림 5 Hadoop 클러스터의 이름 및 데이터 노드 역할을 하는 ECS

HDFS 용 ECS 에 추가된 다른 향상된 기능은 다음과 같습니다.

- **프록시 사용자 인증** - Hive, HBase, Oozie 를 가장합니다.
- **보안** - 서버 측 ACL 을 적용하고 Hadoop 슈퍼 유저 및 슈퍼 유저 그룹은 물론 기본 그룹을 버킷에 추가합니다.

3.3.3 NFS

ECS 는 NFSv3 를 사용한 기본 파일을 지원합니다. NFSv3 파일 데이터 서비스의 주요 특징은 다음과 같습니다.

- **글로벌 네임스페이스** - 모든 사이트의 모든 노드에서 파일에 액세스합니다.
- **글로벌 잠금** - NFSv3 에서 잠금은 **권장 사항일 뿐**입니다. ECS 는 공유 및 전용, 범위 기반 및 필수 잠금을 허용하는 호환 클라이언트 구현을 지원합니다.
- **멀티 프로토콜 액세스** - 서로 다른 프로토콜 방식을 사용하여 데이터에 액세스합니다.

WebUI 또는 API 를 사용하여 NFS 내보내기, 사용 권한 및 사용자 그룹 매핑을 생성할 수 있습니다. NFSv3 호환 클라이언트는 네임스페이스 및 버킷 이름을 사용하여 내보내기를 마운트합니다. 다음은 버킷을 마운트하는 샘플 명령입니다.

```
mount -t nfs -o vers=3 s3.dell.com:/namespace/bucket
```

노드 장애 중에 클라이언트 투명성을 확보하려면 이 워크플로에 로드 밸런서를 사용하는 것이 좋습니다.

ECS 는 *lockmgr*, *statd*, *nfsd*, *mountd* 등과 같은 다른 NFS 서버 구현을 강력하게 통합하기 때문에 이 서비스를 인프라스트럭처 계층(호스트 운영 체제)에 구애받지 않고 관리할 수 있습니다. NFSv3 지원에는 다음과 같은 특징이 있습니다.

- 파일 또는 디렉토리 수에 설계상의 제한이 없습니다.
- 파일 쓰기 크기는 최대 16TB 입니다.
- 단일 글로벌 네임스페이스/내보내기를 통해 최대 8 개 사이트에 걸쳐 확장할 수 있습니다.
- Kerberos 및 AUTH_SYS 인증을 지원합니다.

NFS 파일 서비스는 클라이언트에서 오는 NFS 요청을 처리합니다. 그러나 데이터는 ECS 내에 오브젝트로 저장됩니다. NFS 파일 핸들이 오브젝트 ID 에 매핑됩니다. 파일은 기본적으로 오브젝트에 매핑되므로 NFS 는 다음을 포함하여 오브젝트 데이터 서비스와 같은 기능을 갖추고 있습니다.

- 버킷 레벨에서의 할당량 관리
- 오브젝트 레벨에서의 암호화
- 버킷 레벨에서의 WORM(Write-Once-Read-Many)
  - WORM 은 새 버킷 생성 중에 자동 커밋 기간을 사용하여 구현됩니다.
  - WORM 은 비호환 버킷에만 적용됩니다.

3.3.4 커넥터 및 게이트웨이

여러 타사 소프트웨어 제품에서 ECS 오브젝트 스토리지에 액세스할 수 있습니다. Panzura, Ctera, Syncplicity 등의 ISV(Independent Software Vendor)는 SMB/CIFS, NFS, iSCSI 와 같은 기존 프로토콜을 통해 클라이언트가 ECS 오브젝트 스토리지에 액세스할 수 있도록 하는 서비스 계층을 생성합니다. 다음 Dell EMC 제품을 사용하여 데이터에 액세스하거나 ECS 스토리지에 데이터를 업로드할 수도 있습니다.

- **Isilon CloudPools** - Isilon 에서 ECS 로 데이터의 정책 기반 계층화를 지원합니다.
- **Data Domain Cloud Tier** - 장기간 보존을 위해 중복 제거된 데이터를 Data Domain 에서 ECS 로 자동 기본 계층화합니다. Data Domain Cloud Tier 는 스토리지 상면 및 네트워크 대역폭을 줄이면서 클라우드의 데이터를 암호화하는 안전하고 비용 효율적인 솔루션을 제공합니다.
- **GeoDrive** - Microsoft®Windows® 데스크탑 및 서버를 위한 ECS 스텝 기반 스토리지 서비스입니다.

3.4 스토리지 엔진

ECS 의 핵심은 스토리지 엔진입니다. 스토리지 엔진 계층에는 요청 처리는 물론 데이터 저장, 검색, 보호 및 복제를 책임지는 주요 구성 요소가 포함되어 있습니다.

이 섹션에서는 설계 원리와 데이터를 내부적으로 표현하고 처리하는 방법에 대해 설명합니다.

3.4.1 스토리지 서비스

ECS 스토리지 엔진은 그림 6 에 나온 것처럼 다음과 같은 서비스를 포함합니다.

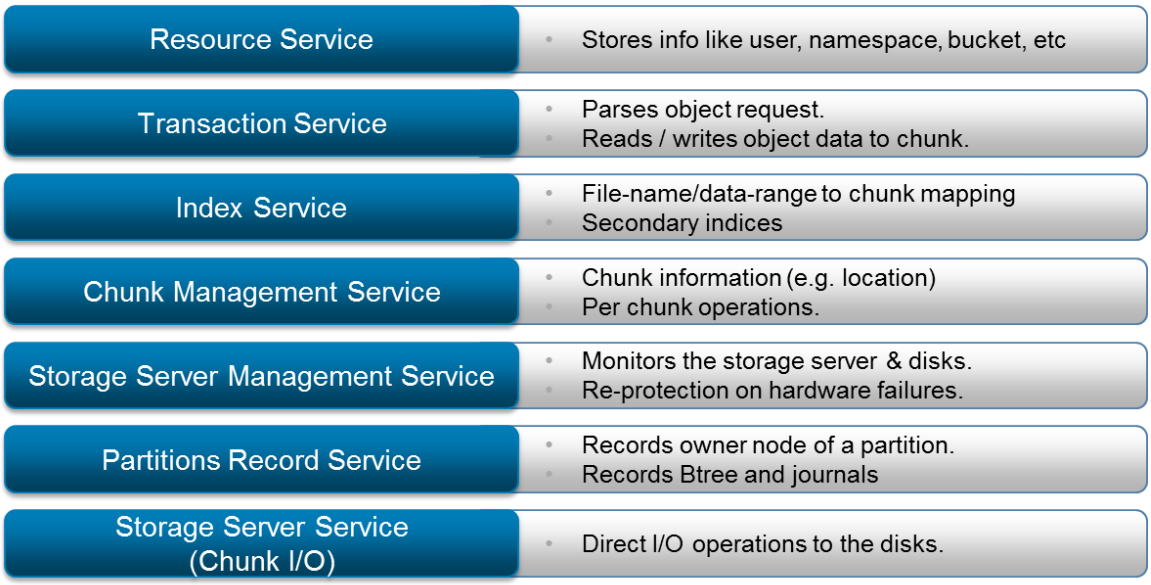


그림 6 스토리지 엔진 서비스

스토리지 엔진의 서비스는 분산 및 공유 서비스를 제공하기 위해 모든 ECS 노드에서 실행되는 Docker 컨테이너 내에 캡슐화됩니다.

3.4.2 데이터

ECS 에 저장되는 데이터의 기본 유형을 다음과 같이 요약할 수 있습니다.

- **데이터** - 저장되는 애플리케이션 또는 사용자 레벨 콘텐츠(예: 이미지)입니다. 데이터는 오브젝트, 파일 또는 콘텐츠와 동의어로 사용됩니다. 애플리케이션은 각 오브젝트와 함께 무제한의 맞춤형 메타데이터를 저장할 수 있습니다. 스토리지 엔진은 데이터 그리고 애플리케이션에서 제공하는 관련 맞춤형 메타데이터를 함께 논리 저장소에 씁니다. 맞춤형 메타데이터는 저장 중인 데이터에 대한 추가 정보 또는 분류를 제공하는 모던 스토리지 시스템의 강력한 기능입니다. 맞춤형 메타데이터는 키-값 쌍 형식이며 쓰기 요청과 함께 제공됩니다.
- **시스템 메타데이터** - 사용자 데이터 및 시스템 리소스와 관련된 시스템 정보 및 특성입니다. 시스템 메타데이터는 크게 다음과 같이 분류할 수 있습니다.
  - **식별자 및 설명자** - 오브젝트 및 해당 버전을 식별하기 위해 내부적으로 사용되는 특성 집합입니다. 식별자는 ECS 소프트웨어 컨텍스트 밖에서는 쓸모가 없는 숫자 ID 또는 해시 값입니다. 설명자는 인코딩 유형과 같은 정보를 정의합니다.
  - **암호화된 형식의 암호화 키** - 데이터 암호화 키는 시스템 메타데이터로 간주되며 코어 디렉토리 테이블 구조 내에서 암호화된 형식으로 저장됩니다.
  - **내부 플래그** - 바이트 범위 업데이트 또는 암호화가 사용 설정되어 있는지를 추적하고 캐싱 및 삭제를 조정하는 데 사용되는 지표 집합입니다.
  - **위치 정보** - 바이트 오프셋과 같이 인덱스 및 데이터 위치가 있는 특성 집합입니다.
  - **타임스탬프** - 오브젝트 생성 또는 업데이트 등을 위해 시간을 추적하는 특성 집합입니다.
  - **구성/테넌시 정보** - 네임스페이스 및 오브젝트 액세스 제어입니다.

데이터 및 시스템 메타데이터는 ECS 에서 *청크*로 기록됩니다. ECS 청크는 연속 공간의 128MB 논리적 컨테이너입니다. 각 청크는 그림 7 과 같이 서로 다른 오브젝트의 데이터를 포함할 수 있습니다. ECS 는 인덱싱을 사용하여 서로 다른 청크 및 노드에 분산될 수 있는 오브젝트의 모든 부분을 추적합니다.

청크는 추가 전용 패턴으로 기록됩니다. 추가 전용 동작이란 애플리케이션에서 기존 오브젝트의 수정 또는 업데이트를 요청할 때 청크 내에서 이전에 기록된 데이터를 수정하거나 삭제하는 것이 아니라 새로운 청크에 새로운 수정 사항 및 업데이트 사항을 기록한다는 것을 의미합니다. 따라서 I/O 를 잠글 필요가 없으며 캐시 무효화가 필요하지 않습니다. 추가 전용 설계는 데이터 버전 관리도 간소화합니다. 이전 버전의 데이터는 이전 청크에서 유지됩니다. S3 버전 관리가 활성화되어 있고 이전 버전의 데이터가 필요한 경우 S3 REST API 를 사용하여 가져오거나 이전 버전으로 복원할 수 있습니다.

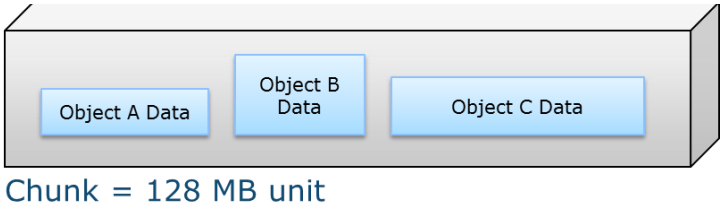


그림 7 오브젝트 세 개의 데이터를 저장하는 128MB 청크

아래의 *데이터 무결성 및 보호* 섹션에서는 청크 레벨에서 데이터를 보호하는 방법을 설명합니다.

3.4.3 데이터 관리

ECS 는 논리 테이블 집합을 사용하여 오브젝트와 관련된 정보를 저장합니다. 키-값 쌍은 데이터 위치를 빠르게 인덱싱하기 위해 결국 B+ 트리의 디스크에 저장됩니다. 키-값 쌍을 B+ 트리와 같이 균형 잡힌 검색된 트리에 저장하면 데이터 및 메타데이터의 위치에 빠르게 액세스할 수 있습니다. ECS 는 2 개의 트리식 구조가 있는 2 단계 로그 구조 병합 트리를 구현합니다. 작은 트리는 메모리(메모리 테이블)에 있고 기본 B+ 트리는 디스크에 있습니다. 키-값 쌍은 먼저 메모리에서 조회되고 이후 필요한 경우 디스크의 기본 B+ 트리에서 조회됩니다. 이러한 논리 테이블의 항목은 먼저 저널 로그에 기록되고 이 로그는 삼중 미러링된 청크의 디스크에 기록됩니다. 저널은 아직 B+ 트리에 커밋되지 않은 트랜잭션을 추적하는 데 사용됩니다. 각 트랜잭션이 저널에 로깅된 후에 메모리 내 테이블이 업데이트됩니다. 메모리의 테이블이 가득 차거나 일정 기간이 지나면 병합이 정렬되거나 디스크의 B+ 트리에 덤프됩니다. B+ 트리 청크와 비교했을 때 시스템에서 사용하는 저널 청크 수는 중요하지 않습니다. 그림 8 에 이 프로세스가 나와 있습니다.

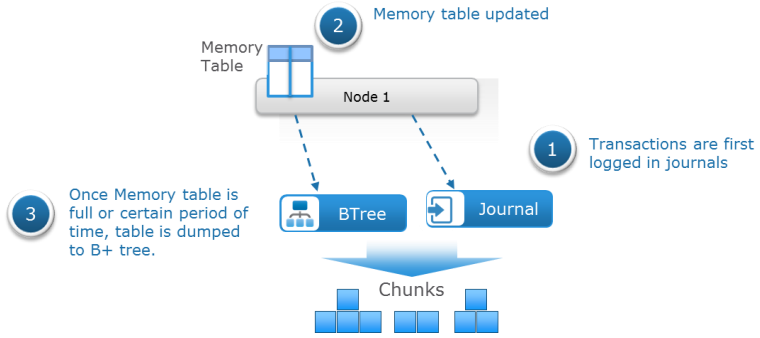


그림 8 B+ 트리에 덤프된 메모리 테이블

OB(Object Table)에 저장된 정보가 표 2 에 나와 있습니다. OB 테이블에는 오브젝트 이름 그리고 해당 청크 내에서 특정 오프셋 및 길이에 있는 청크 위치가 포함됩니다. 이 테이블에서 오브젝트 이름은 인덱스의 키이며 값은 청크 위치입니다. 스토리지 엔진 내의 인덱스 계층은 오브젝트 이름-청크 매핑을 담당합니다.

표 2 오브젝트 테이블 항목

Object Name	청크 위치
ImgA	• C1:offset:length
FileB	• C2:offset:length • C3:offset:length

CT(Chunk Table)는 표 3 에 자세히 설명된 것처럼 각 청크에 대한 위치를 기록합니다.

표 3 청크 테이블 항목

청크 ID	위치
C1	• Node1:Disk1:File1:Offset1:Length • Node2:Disk2:File1:Offset2:Length • Node3:Disk2:File6:Offset:Length

ECS 는 데이터 저장 및 액세스가 모든 노드에 고르게 분산되는 분산형 시스템으로 설계되었습니다. 오브젝트 데이터 및 메타데이터를 관리하는 데 사용되는 테이블은 스토리지가 사용되고 증가함에 따라 점차 커집니다. 테이블은 파티션으로 분할되고 서로 다른 노드에 할당됩니다. 여기에서 각 노드는 각 테이블에 대해 호스팅하는 파티션의 소유자가 됩니다. 예를 들어, 청크 위치를 가져오려면 청크 위치를 알고 있는 소유자 노드에 대해 PR(Partition Records) 테이블을 쿼리합니다. 아래 표 4 에 기본 PR 테이블이 나와 있습니다.

표 4 파티션 레코드 테이블 항목

파티션 ID	Owner
P1	노드 1
P2	노드 2
P3	노드 3

노드가 다운되면 다른 노드가 파티션을 소유하게 됩니다. 파티션은 B+ 트리 루트를 읽고 디스크에 저장된 저널을 재생하여 다시 생성됩니다. 그림 9 는 파티션 소유권의 페일오버(failover)를 보여줍니다.

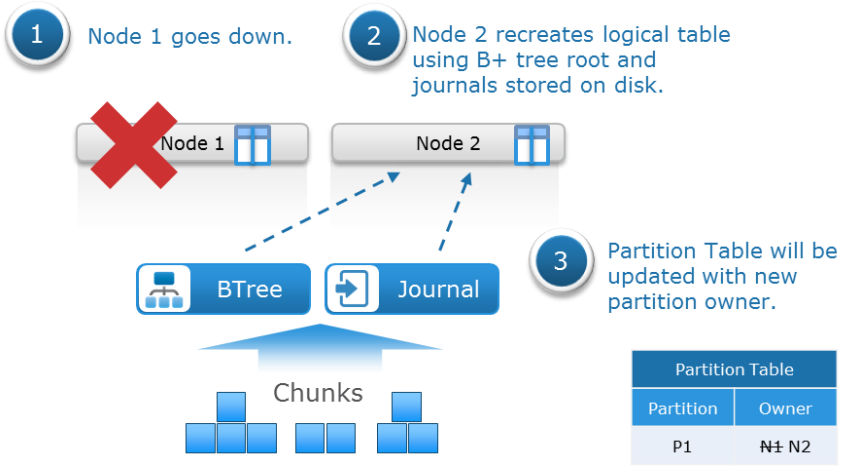


그림 9 파티션 소유권의 페일오버

3.4.4 데이터 흐름

스토리지 서비스는 모든 노드에서 사용할 수 있습니다. 데이터는 드라이브, 노드, 랙 전반에 분산된 EC 세그먼트에 의해 보호됩니다. ECS 는 체크섬 함수를 실행하고 그 결과를 각각의 쓰기와 함께 저장합니다. 데이터의 처음 몇 바이트가 압축 가능한 경우 ECS 는 데이터를 압축합니다. 읽기와 동시에 데이터가 압축 해제되고 저장된 체크섬의 유효성이 검사됩니다. 다음 예는 쓰기의 데이터 흐름을 다섯 단계로 보여 주고 있습니다.

1. 클라이언트가 오브젝트 생성 요청을 노드에 보냅니다.
2. 해당 요청을 처리하는 노드가 새 오브젝트의 데이터를 저장소 청크에 씁니다.
3. 디스크 쓰기에 성공하면 이름과 청크 위치를 입력하는 PR 트랜잭션이 발생합니다.
4. 파티션 소유자가 저널 로그에 트랜잭션을 기록합니다.
5. 트랜잭션이 저널 로그에 기록되면 클라이언트에 확인이 전송됩니다.

아래 그림 10 은 Gen2 와 EX300, EX500 및 EX3000 과 같은 하드 디스크 드라이브 아키텍처의 읽기 데이터 흐름 예시를 보여줍니다.

1. 클라이언트가 오브젝트 읽기 요청을 노드 1 에 전송합니다.
2. 노드 1 은 오브젝트 이름을 사용하는 해시 함수를 활용하여 어떤 노드가 이 오브젝트 정보가 포함된 논리 테이블의 파티션 소유자인지를 확인합니다. 이 예제에서는 노드 2 가 소유자이므로 노드 2 가 논리 테이블을 조회하여 청크 위치를 가져옵니다. 경우에 따라 서로 다른 두 노드에서 조회가 이루어질 수도 있습니다. 예를 들어 위치가 노드 2 의 논리 테이블에 캐시되지 않은 경우가 그렇습니다.
3. 이전 단계에서 청크 위치가 노드 1 에 제공되고, 노드 1 은 데이터를 보유한 노드(이 예에서는 노드 3)에 바이트 오프셋 읽기 요청을 보내고, 노드 3 은 데이터를 노드 1 에 보냅니다.
4. 노드 1 은 데이터를 요청 클라이언트에 보냅니다.

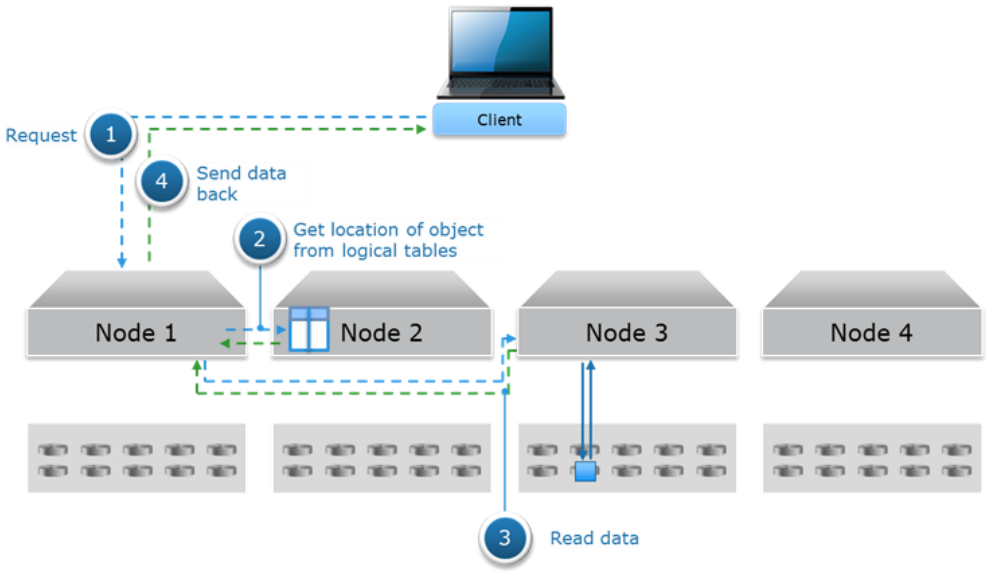


그림 10 하드 디스크 드라이브 아키텍처의 읽기 데이터 흐름

아래 그림 11 은 EXF900 과 같은 올플래시 아키텍처의 읽기 데이터 흐름 예시를 보여줍니다.

1. 클라이언트가 오브젝트 읽기 요청을 노드 1 에 전송합니다.
2. 노드 1 은 오브젝트 이름을 사용하는 해시 함수를 활용하여 어떤 노드가 이 오브젝트 정보가 포함된 논리 테이블의 파티션 소유자인지를 확인합니다. 이 예제에서는 노드 2 가 소유자이므로 노드 2 가 논리 테이블을 조회하여 청크 위치를 가져옵니다. 경우에 따라 서로 다른 두 노드에서 조회가 이루어질 수도 있습니다. 예를 들어 위치가 노드 2 의 논리 테이블에 캐시되지 않은 경우가 그렇습니다.
3. 이전 단계에서 청크의 위치가 노드 1 에 제공되면 노드 1 에서 직접 노드 3 의 데이터를 읽습니다.
4. 노드 1 은 데이터를 요청 클라이언트에 보냅니다.

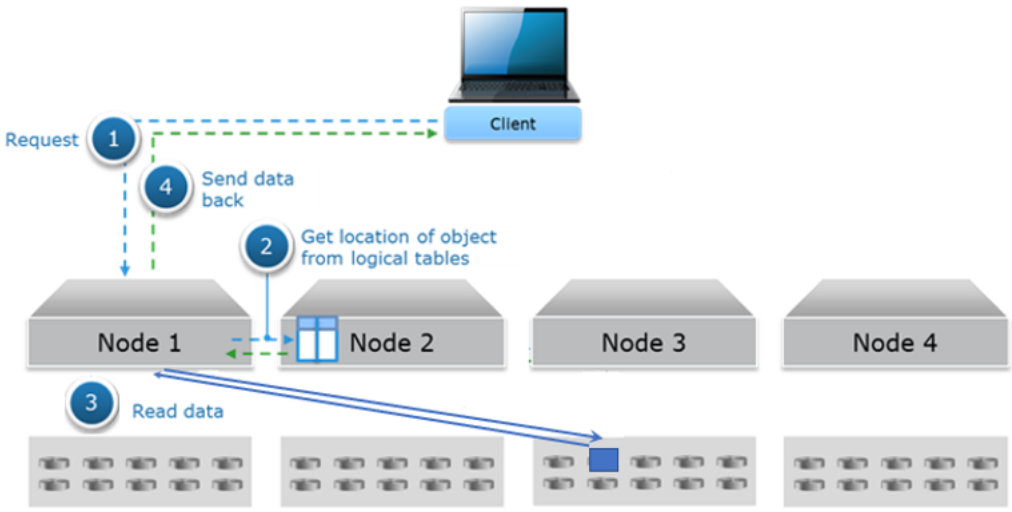


그림 11 올플래시 아키텍처의 읽기 데이터 흐름

참고: EXF900 과 같은 올플래시 아키텍처에서 각 노드는 각 노드가 자체 데이터 저장소만 읽을 수 있는 하드 디스크 드라이브 아키텍처와 달리 다른 노드에서 직접 데이터를 읽을 수 있습니다.

3.4.5 파일 크기를 위한 쓰기 최적화

비교적 소용량인 스토리지 쓰기의 경우 ECS 는 *박스 카팅(Box-Carting)*이라는 방법을 사용하여 성능에 미치는 영향을 최소화합니다. 박스 카팅은 메모리에서 2MB 이하의 여러 작은 쓰기를 취합하여 단일 디스크 작업으로 씁니다. 박스 카팅은 라운드 트립 수를 디스크가 필요한 프로세스의 개별 쓰기 횟수로 제한합니다.

비교적 대용량인 오브젝트 쓰기의 경우 ECS 내의 노드는 동일한 오브젝트에 대한 쓰기 요청을 동시에 처리하고 ECS 클러스터의 여러 스핀들에서 동시 쓰기를 활용할 수 있습니다. 따라서 ECS 는 크고 작은 오브젝트를 효율적으로 수집하고 저장할 수 있습니다.

3.4.6 공간 재확보

추가 전용 방식으로 청크를 쓴다는 것은 우선 원래의 기록된 데이터를 제자리에 유지하고, 그 다음에 원래 오브젝트의 청크 컨테이너에 포함되었거나 포함되지 않았을 수 있는 완전히 새로운 청크 세그먼트를 생성하여 데이터가 추가되거나 업데이트된다는 의미입니다. 추가 전용 데이터 수정의 이점은 기존 파일 시스템의 파일 잠금 문제로 인해 방해받지 않는 활성/활성 데이터 액세스 모델을 이용한다는 것입니다. 이는 오브젝트가 업데이트되거나 삭제될 때 청크의 데이터가 더 이상 참조되거나 필요하지 않게 되기 때문입니다. ECS 가 삭제된 전체 청크로부터 또는 더 이상 참조되지 않는 삭제된 오브젝트 조각과 삭제되지 않은 오브젝트 조각이 혼합된 청크로부터 공간을 재확보하는 데 사용하는 두 가지 가비지 컬렉션 방법은 다음과 같습니다.

- **일반 가비지 컬렉션** - 전체 청크가 가비지인 경우 공간을 재확보합니다.
- **병합에 의한 부분 가비지 컬렉션** - 청크가 2/3 가비지인 경우, 유효한 부분을 다른 부분적으로 채워진 청크와 함께 새로운 청크로 병합하여 청크를 재확보하고 공간을 재확보합니다.

가비지 컬렉션은 고립된 Blob 을 정리하기 위해 ECS CAS 데이터 서비스 액세스 API 에도 적용되었습니다. 고립된 Blob 이란 ECS 에 저장된 CAS 데이터에서 식별된 참조되지 않은 Blob 로서, 일반 가비지 컬렉션 방법을 통해 공간을 재확보할 수 있습니다.

3.4.7 SSD 메타데이터 캐싱

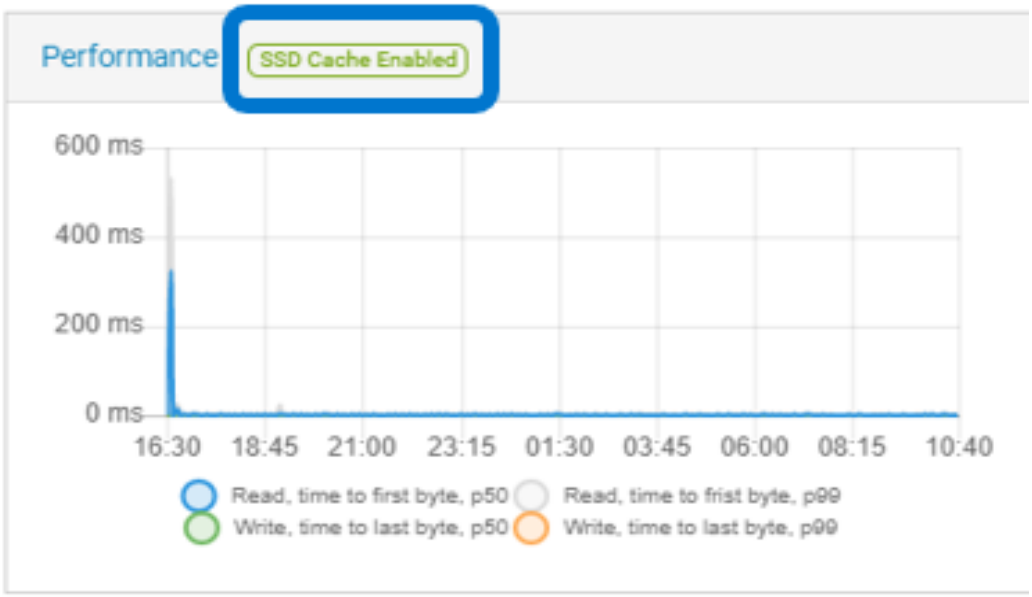
ECS 메타데이터는 B 트리에 저장됩니다. 각 B 트리의 메모리, 저널 트랜잭션 및 디스크에 항목이 있을 수 있습니다. 시스템에서 특정 B 트리의 전체 그림을 확보하기 위해 세 위치 모두 쿼리되며, 이 작업에는 종종 디스크에 대한 여러 번의 조회가 포함됩니다.

메타데이터 조회의 레이턴시를 최소화하기 위해 ECS 3.5 에 선택적 SSD 기반 캐시 메커니즘이 구현되었습니다. 캐시는 최근에 액세스한 B 트리 페이지를 보유합니다. 즉, 최신 B 트리의 읽기 작업은 항상 SSD 기반 캐시에 도달하며 회전식 디스크로의 이동이 방지됩니다.

다음은 새로운 SSD 메타데이터 캐싱 기능의 몇 가지 주요 특징입니다.

- 소규모 파일의 전체 시스템 읽기 레이턴시 및 TPS(Transactions Per Second) 개선
- 노드당 1 개의 960GB 플래시 드라이브
- 제조 단계의 새로운 노드에 SSD 드라이브가 옵션으로 포함됨
- 업그레이드 키트와 셀프 서비스 설치를 통해 기존 현장 노드(Gen3 및 Gen2) 업그레이드 가능
- ECS 가 온라인 상태일 때 SSD 드라이브 추가 가능
- 대규모 데이터 세트의 빠른 읽기가 필요한 소규모 파일 분석 워크로드 개선
- 이 기능을 활성화하려면 VDC 의 모든 노드에 SSD 가 있어야 합니다.

SSD 키트가 설치되어 있는 경우 ECS 패브릭에서 이를 감지합니다. 이러한 방식으로 시스템이 자동으로



초기화되어 새 드라이브 사용을 시작합니다. 그림 12 는 SSD 캐시가 활성화되었음을 보여줍니다.
그림 12 SSD 캐시 활성화됨

SSD 메타데이터 캐싱은 소규모 읽기 및 버킷 나열을 개선합니다. 연구실에서 테스트한 결과, 10MB 오브젝트에서 나열 성능이 50% 향상되었습니다. 읽기 성능은 각각 10KB 오브젝트에서 35%, 100KB 오브젝트에서 70% 향상되었습니다.

3.4.8 클라우드 DVR

ECS 는 케이블 및 위성 회사에 대한 법적 저작권 요구 사항을 해결하는 클라우드 DVR(Digital Video Recording) 기능을 지원합니다. 요구 사항은 ECS 에서 오브젝트에 매핑된 모든 레코딩 단위를 미리 정해진 횟수만큼 복제해야 한다는 것입니다. 미리 정해진 복제 횟수를 팬아웃이라고 합니다. 미리 정해진 복제 횟수(팬아웃)가 이중화 또는 성능 향상을 위한 요구 사항은 아니지만, 그보다는 케이블 및 위성 회사에 대한 법적 저작권 요구 사항에 해당합니다. ECS 는 다음을 지원합니다.

- ECS 에서 생성된 오브젝트 복제본의 팬 아웃 수 생성
- 특정 복제본의 읽기 허용
- 특정 복제본의 삭제 허용
- 모든 복제본의 삭제 허용
- 특정 복제본의 복제 허용
- 복제본 나열 허용

- 팬 아웃 오브젝트의 버킷 나열 허용

클라우드 DVR 기능은 Service Console 을 통해 활성화할 수 있습니다. 처음 사용할 때는 Service Console 을 사용하여 클라우드 DVR 기능을 활성화해야 합니다. 클라우드 DVR 을 활성화하면 기본적으로 모든 새 노드에 대해 Cloud DVR 이 활성화됩니다.

Service Console 에서 아래 명령을 실행하여 클라우드 DVR 기능을 활성화합니다.

```
service-console run Enable_CloudDVR
```

클라우드 DVR 기능은 API 를 지원합니다. 자세한 내용은 *ECS 데이터 액세스 가이드*를 참조하십시오.

3.5 Fabric

패브릭 계층은 클러스터링, 시스템 상태, 소프트웨어 관리, 구성 관리, 업그레이드 기능 및 알림을 제공합니다. 서비스 실행을 유지하고 디스크, 컨테이너, 네트워크와 같은 리소스를 관리합니다. 장애 감지와 같이 환경 변화를 추적하고 대응하며 시스템 상태와 관련된 알림을 제공합니다. 패브릭 계층에는 다음과 같은 구성 요소가 있습니다.

- **노드 에이전트** - 호스트 리소스(디스크, 네트워크, 컨테이너 등) 및 시스템 프로세스를 관리합니다.
- **수명주기 관리자** - 서비스 시작, 복구, 알림, 오류 감지와 관련된 애플리케이션 수명주기 관리를 제공합니다.
- **지속성 관리자** - ECS 분산 환경을 조정하고 동기화합니다.
- **레지스트리** - ECS 소프트웨어용 Docker 이미지 저장소입니다.
- **이벤트 라이브러리** - 시스템에서 발생하는 이벤트 집합을 보관합니다.
- **하드웨어 관리자** - 상태 및 이벤트 정보 그리고 하드웨어 계층의 프로비저닝을 더 높은 수준의 서비스에 제공합니다. 이들 서비스는 상용 하드웨어를 지원하기 위해 통합되었습니다.

3.5.1 노드 에이전트

노드 에이전트는 모든 ECS 노드에서 기본적으로 실행되며 Java 로 작성된 경량 에이전트입니다. 주요 기능은 호스트 리소스(Docker 컨테이너, 디스크, 방화벽, 네트워크)를 관리 및 제어하고 시스템 프로세스를 모니터링하는 것입니다. 디스크 포맷 및 마운팅, 필수 포트 열기, 모든 프로세스의 실행 확인, 퍼블릭 및 프라이빗 네트워크 인터페이스 결정 등의 관리 업무를 수행합니다. 이벤트 스트림으로 수명주기 관리자에게 순서가 지정된 이벤트를 제공함으로써 시스템에서 발생하는 이벤트를 나타냅니다. 패브릭 CLI 는 문제를 진단하고 전체 시스템 상태를 살펴보는 데 유용합니다.

3.5.2 수명주기 관리자

수명주기 관리자는 3 개 또는 5 개 노드의 하위 집합에서 실행되며 노드에서 실행되는 애플리케이션의 수명주기를 관리합니다. 각 수명주기 관리자가 여러 개의 노드를 추적합니다. 기본 목적은 장애 감지, 복구, 알림, 마이그레이션 등 부팅에서 배포까지 ECS 애플리케이션의 전체 수명주기를 관리하는 것입니다. 이를 위해 노드 에이전트 스트림을 살펴보고 에이전트로 하여금 상황을 처리하도록 합니다. 노드가 다운되면 시스템을 알려진 정상 상태로 복원함으로써 노드 상태의 장애 또는 불일치에 대처합니다. 수명주기 관리자 인스턴스가 다운된 경우에는 다른 인스턴스가 그 자리를 대신합니다.

3.5.3 레지스트리

레지스트리에는 설치, 업그레이드 및 노드 교체 중에 사용되는 ECS Docker 이미지가 포함되어 있습니다. *fabric-registry* 라는 Docker 컨테이너는 ECS 랙 내의 한 노드에서 실행되며, 설치와 업그레이드에 필요한 ECS Docker 이미지 및 정보의 저장소를 포함합니다. 레지스트리는 한 번에 하나의 노드에서만 사용할 수 있지만 모든 Docker 이미지는 모든 노드에서 로컬로 캐시되므로 모든 노드가 레지스트리를 처리할 수 있습니다.

3.5.4 이벤트 라이브러리

이벤트 라이브러리는 패브릭 계층 내에서 수명주기 및 노드 에이전트 이벤트 스트림을 노출하는 데 사용됩니다. 시스템에서 생성된 이벤트는 공유 메모리 및 디스크에서 유지되면서 ECS 시스템의 상태에 대한 기록 정보를 제공합니다. 순서가 지정된 이러한 이벤트 스트림을 활용하면 시스템을 특정 상태로 복원할 수 있습니다. 즉, 저장되어 있는 순서가 지정된 이벤트를 재생하면 됩니다. 이벤트의 몇 가지 예로는 시작됨, 중지됨, 성능 저하됨 등의 노드 이벤트가 있습니다.

3.5.5 하드웨어 관리자

하드웨어 관리자는 패브릭 에이전트에 통합되어 업계 표준 하드웨어를 지원합니다. 기본 목적은 ECS 내에서 더 높은 수준의 서비스에 하드웨어별 상태 및 이벤트 정보 그리고 하드웨어 계층 프로비저닝을 제공하는 것입니다.

3.6 인프라스트럭처

현재 ECS 어플라이언스 노드는 인프라스트럭처용으로 SUSE Linux Enterprise Server 12 를 실행합니다. 맞춤형 업계 표준 하드웨어에 배포된 ECS 소프트웨어의 경우 운영 체제는 RedHat Enterprise Linux 또는 CoreOS 일 수도 있습니다. 맞춤형 구축은 공식적인 요청 및 검증 프로세스를 통해 수행됩니다. Docker 는 캡슐화된 ECS 계층을 구축하기 위해 인프라스트럭처에 설치됩니다. ECS 소프트웨어는 Java 로 작성되었기 때문에 JVM(Java Virtual Machine)이 인프라스트럭처의 일부로 설치됩니다.

3.6.1 Docker

ECS 는 운영 체제 위에서 Java 애플리케이션으로 실행되며 여러 Docker 컨테이너 내에 캡슐화됩니다. 컨테이너들은 서로 분리되어 있지만 기본 운영 체제 리소스와 하드웨어를 공유합니다. ECS 소프트웨어의 일부는 모든 노드에서 실행되고 다른 일부는 단일 노드나 여러 노드에서 실행됩니다. Docker 컨테이너 내에서 실행되는 구성 요소는 다음과 같습니다.

- **object-main** - 데이터 서비스, 스토리지 엔진, 포털 및 프로비저닝 서비스와 관련된 리소스와 프로세스가 포함되어 있습니다. ECS 의 모든 노드에서 실행됩니다.
- **fabric-lifecycle** - 시스템 레벨 모니터링, 구성 관리 및 상태 관리에 필요한 프로세스, 정보, 리소스가 포함되어 있습니다. 항상 홀수의 fabric-lifecycle 인스턴스가 실행됩니다. 예를 들어 4 노드 시스템에서는 3 개의 인스턴스가 실행되고, 8 노드 시스템에서는 5 개의 인스턴스가 실행됩니다.
- **fabric-zookeeper** - 분산 프로세스, 구성 정보, 그룹 및 명명 서비스를 조정하고 동기화하기 위한 중앙 집중식 서비스입니다. 지속성 관리자라고도 하며 홀수의 인스턴스에서 실행됩니다(예: 8 노드 시스템에서 5 개).
- **fabric-registry** - ECS Docker 이미지의 레지스트리입니다. ECS 랙당 하나의 인스턴스만 실행됩니다.

Fabric 노드 에이전트와 하드웨어 추상화 계층 툴처럼 Docker 컨테이너 외부에서 실행되는 다른 프로세스 및 툴도 있습니다. 아래 그림 13 에서는 8 노드 구축에서 ECS 컨테이너를 실행하는 방법의 예를 보여줍니다.

그림 13 8 노드 구축에서의 Docker 컨테이너 및 에이전트 예시

그림 14 에는 노드에 대한 *docker ps* 명령의 명령줄 출력이 나와 있습니다. 이 출력은 Docker 안에서 ECS 가 사용하는 4 개의 컨테이너를 보여줍니다. 시스템에서 사용할 수 있는 모든 오브젝트 관련 서비스와 함께 목록이 표시됩니다.

```
admin@hop-u300-11-pub-01:~> sudo docker ps
CONTAINER ID        IMAGE                                                COMMAND                  CREATED             STATUS
      PORTS               NAMES
7ba30ce42be2        ecs-monitoring/telegraf:3.5.0.0-825.b6b07cf9         "/entrypoint.sh "        5 weeks ago         Up 5 weeks
                    object-telegraf
e22513635cab        ecs-monitoring/grafana:3.5.0.0-825.b6b07cf9          "/entrypoint.sh "        5 weeks ago         Up 5 weeks
                    object-grafana
ee9db1ea40bc        emcvipr/object:3.5.0.0-120417.6a358e139f1            "/opt/vipr/boot/boot…"   5 weeks ago         Up 5 weeks
                    object-main
d11a7acd55e5        ecs-monitoring/throttler:3.5.0.0-825.b6b07cf9        "/entrypoint.sh "        5 weeks ago         Up 5 weeks
                    object-throttler
f94026797bb3        ecs-monitoring/fluxd:3.5.0.0-825.b6b07cf9            "/entrypoint.sh "        5 weeks ago         Up 5 weeks
                    object-fluxd
c7b8530a8bb9        caspian/fabric:3.5.0.0-4076.7d40a97                  "./boot.sh lifecycle"    5 weeks ago         Up 5 weeks
                    fabric-lifecycle
bffdf8836853        caspian/fabric-zookeeper:3.5.6.0-99.0354df7          "./boot.sh 1 1=169.2…"   5 weeks ago         Up 5 weeks
                    fabric-zookeeper
f4420f7f7d51        caspian/fabric-registry:2.3.1.0-68.10d1aca           "/opt/docker-registr…"   5 weeks ago         Up 5 weeks
                    fabric-registry
admin@hop-u300-11-pub-01:~> sudo dockobj
hop-u300-11-pub-01:/ # cd /opt/storageos/
hop-u300-11-pub-01:/opt/storageos # ls bin/*svc
bin/blobsvc     bin/coordinatorsvc  bin/eventsvc  bin/objcontrolsvc  bin/storagemanagementsvc
bin/cassvc      bin/dataheadsvc     bin/filesvc   bin/objheadsvc     bin/syssvc
bin/controlsvc  bin/ecsportalsvc    bin/hdfssvc   bin/resourcesvc    bin/transformsvc
```

그림 14 object-main 컨테이너의 프로세스, 리소스, 툴 및 바이너리

4 어플라이언스 하드웨어 모델

ECS 는 유연한 초기 구축을 통해 페타바이트 및 엑사바이트급 데이터로 빠르게 확장할 수 있습니다. ECS 솔루션은 업무에 미치는 영향을 최소화하면서 노드 및 디스크를 추가하여 용량과 성능 두 측면 모두에서 선형적으로 확장할 수 있습니다.

ECS 어플라이언스 하드웨어 모델은 하드웨어 세대로 구분됩니다. Gen3 또는 EX-Series 로 알려진 3 세대 어플라이언스 시리즈에는 세 가지 하드웨어 모델이 포함됩니다. 이 섹션에서는 EX-Series 를 개략적으로 살펴봅니다. 자세한 내용은 *ECS EX-Series 하드웨어 가이드*를 참조하십시오.

1 세대 및 2 세대 ECS 어플라이언스 하드웨어에 대한 정보는 *Dell EMC ECS D-and U-Series 하드웨어 가이드*에서 확인할 수 있습니다.

4.1 EX-Series

EX-Series 어플라이언스 모델은 표준 Dell 서버 및 스위치를 기반으로 합니다. 이 시리즈의 제품은 다음과 같습니다.

- **EX300** - EX300 은 최소 물리적 용량이 60TB 이며 클라우드 네이티브 애플리케이션과 고객 디지털 혁신 이니셔티브를 위한 완벽한 스토리지 플랫폼입니다. EX300 은 Centera 구축을 현대화하는 데 적합합니다. 무엇보다도, EX300 은 비용 효율적으로 더 큰 용량으로 확장할 수 있습니다. 노드당 12 개의 드라이브와 1TB, 2TB, 4TB, 8TB, 16TB 디스크 옵션(노드에서 모두 동일)을 제공합니다.
- **EX500** - EX500 은 경제성과 집적도를 동시에 구현하는 것을 목표로 하는 최신 어플라이언스입니다. 12 개 또는 24 개의 드라이브 옵션과 8TB, 12TB, 16TB 디스크 옵션(노드에서 모두 동일)을 제공합니다. 클러스터 규모는 랙당 480TB 에서 6.1PB 까지입니다. 이 시리즈는 최신 애플리케이션 및/또는 장기 아카이브 활용 사례를 지원하고자 하는 중간 규모 기업에 다양한 옵션을 제공합니다.
- **EX3000** - EX3000 은 랙당 최대 11.5PB 의 물리적 스토리지 용량을 갖추고 있으며, 노드당 드라이브 30~90 개와 12TB 또는 16TB 디스크 옵션을 제공합니다. 여러 사이트에 걸쳐 엑사바이트 단위로 확장되어 심층적이고 확장 가능한 데이터 센터 솔루션을 제공하므로 많은 데이터 상면이 필요한 워크로드에 이상적입니다. 이러한 노드는 EX3000S 및 EX3000D 라고 하는 두 가지 구성으로 사용할 수 있습니다. EX3000S 는 단일 노드이고 EX3000D 는 이중 노드 섀시입니다. 이러한 고집적도 노드는 디스크 핫 스왑이 가능합니다. 노드당 최소 30 개 디스크부터 시작합니다. ECS 노드당 드라이브가 30 개 정도가 되면 드라이브를 추가함으로써 얻을 수 있는 성능 향상 효과가 줄어들기 시작합니다. 각 노드에 최소 30 개 이상의 드라이브가 있으면 드라이브 개수에 관계없이 모든 EX3000 노드에서 비슷한 성능을 기대할 수 있습니다.

- **EXF900** – EXF900 은 레이턴시가 낮고 IOPS 가 높은 ECS 를 구축하기 위한 하이퍼 컨버지드 노드의 올플래시 오브젝트 스토리지 솔루션입니다. 12 개 또는 24 개의 드라이브 옵션과 3.84TB NVMe SSD 드라이브 옵션(7.68TB NVMe SSD 드라이브는 하드웨어가 제공되면 지원될 예정)이 제공됩니다. 이 플랫폼은 230TB 물리적 용량의 최소 구성에서 시작하며 랙당 1.4PB 물리적 용량까지 확장됩니다. 그림 15 는 EXF900 의 노드를 보여줍니다.

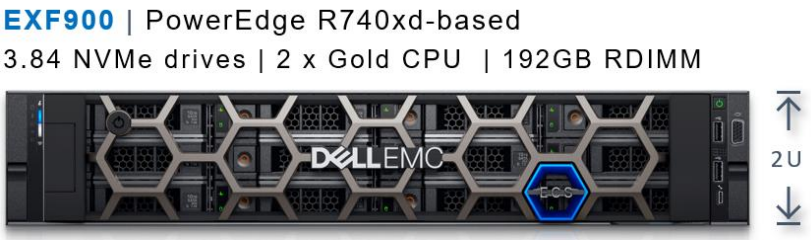


그림 15 EXF900 노드

참고: SSD 읽기 캐시 기능은 EXF900 에 적용되지 않습니다. Cloud DVR 은 EXF900 에서 지원되지 않습니다. TECH Refresh 는 EXF900 에서 지원되지 않습니다. EXF900 은 VDC 에 다른 비 EXF900 하드웨어와 공존할 수 없습니다. EXF900 은 GEO 에 다른 비 EXF900 하드웨어와 공존할 수 없습니다(모든 사이트는 EXF900 이어야 함).

EX-Series 시작 용량 옵션을 통해 고객은 필요한 용량만으로 ECS 구축을 시작하고 나중에 상황에 맞춰 쉽게 확장할 수 있습니다. EX-Series 어플라이언스에 대한 자세한 내용은 *ECS Appliance 스펙 시트*를 참조하십시오. 이전 Gen2 U-Series 및 D-Series 어플라이언스도 자세히 설명되어 있습니다.

EX-Series 노드의 구축 후 업데이트는 지원되지 않습니다. 다음과 같은 섹션이 있습니다.

- CPU 변경
- 메모리 용량 조정
- 하드 드라이브 크기 업그레이드

4.2 어플라이언스 네트워킹

EX-Series 어플라이언스의 출시와 함께 전용 백엔드 관리 스위치의 이중화 쌍이 사용됩니다. 새로운 어플라이언스 스위치 기어로 전환함으로써 ECS 는 이제 프런트엔드 및 백엔드 스위칭 구성 모드를 이용할 수 있습니다.

EX300, EX500 및 EX3000 어플라이언스는 전부 프런트엔드 스위치 쌍과 백엔드 스위치 쌍에 모두 Dell EMC S5148F 를 사용합니다. EXF900 어플라이언스는 프런트엔드 스위치 쌍과 백엔드 스위치 쌍에 Dell EMC S5248F 를, 통합 백엔드 스위치에 S5232F 를 각각 사용합니다. 참고로 고객은 Dell EMC 스위치 대신 자체 프런트엔드 스위치를 사용할 수 있습니다.

4.2.1 S5148F - 프런트엔드 퍼블릭 스위치

2 개의 옵션 Dell EMC S5148F 25GbE 1U 이더넷 스위치를 네트워크 연결용으로 구입할 수 있으며, 고객은 프런트엔드 연결용으로 자체 10GbE 또는 25GbE HA 쌍을 제공할 수도 있습니다. 퍼블릭 스위치를 종종 *산토끼*와 *집토끼* 또는 단순히 프런트엔드라고 합니다.

주의: ECS 어플라이언스의 고가용성 아키텍처를 유지하기 위해서는 고객 네트워크에서 프런트엔드 스위치(집토끼 및 산토끼) 모두에 연결해야 합니다. 고객이 필수 HA 방식으로 네트워크에 연결하지 않기로 선택하는 경우에는 이 제품 사용으로 얻을 수 있는 높은 데이터 가용성을 보장할 수 없습니다.

이 스위치는 25GbE SFP28 포트 48 개와 100GbE QSFP28 포트 6 개를 제공합니다. 다음은 이 두 포트 유형에 대한 자세한 정보입니다.

- SFP28 은 SFP+의 향상된 버전
  - SFP+는 최대 16Gb/s 를 지원하고 SFP28 은 최대 28Gb/s 를 지원
  - 동일한 폼 팩터
  - SFP+ 모듈 이전 버전과의 호환성
- QSFP28 은 QSFP+의 향상된 버전
  - QSFP+는 최대 4 개의 16Gb/s 레인을 지원하고 QSFP28 은 최대 4 개의 28Gb/s 레인을 지원
    - QSFP+ 집계된 레인으로 40Gb/s 이더넷 확보
    - QSFP28 집계된 레인으로 100Gb/s 이더넷 확보
  - 동일한 폼 팩터

- QSFP+ 모듈 이전 버전과의 호환성
- 4 개의 개별 SFP28 레인으로 나눌 수 있음

참고: 두 개의 100GbE LAG 케이블이 Dell EMC S5148F 25GbE 퍼블릭 스위치와 함께 제공됩니다. 자체 퍼블릭 스위치를 제공하는 조직은 필요한 LAG, SF 또는 외부 연결 케이블을 제공해야 합니다.

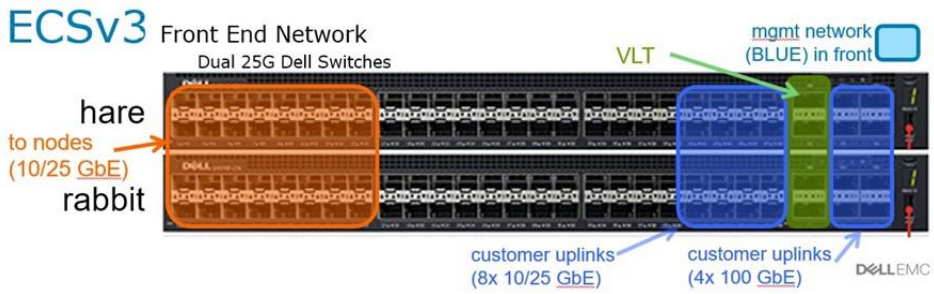


그림 16 프런트엔드 네트워크 스위치 포트 지정 및 사용

위의 그림 16 은 포트를 사용하여 ECS 노드 트래픽과 고객 업링크 포트를 활성화하는 방법을 시각적으로 보여줍니다. 이 방식은 모든 구현에서 표준입니다.

4.2.2 S5148F - 백엔드 프라이빗 스위치

48 개의 25GbE SFP 포트와 6 개의 100GbE 업링크 포트가 있는 두 가지 필수 Dell EMC S5148F 25GbE 1U 이더넷 스위치 모두가 각 ECS 랙에 포함되어 있습니다. 종종 *여우*와 *사냥개* 또는 백엔드 스위치라고 하는 이 스위치는 관리 네트워크를 담당합니다. 향후 ECS 릴리스에서 백엔드 스위치는 복제 트래픽을 위한 네트워크 분리도 제공할 것입니다. 프라이빗 네트워크의 기본 목적은 원격 관리 및 콘솔에 사용하고, 설치 관리자를 위해 PXE 부팅에 사용하고, 랙 및 클러스터 전반의 관리와 프로비저닝을 지원하는 것입니다. 그림 17 은 두 Dell 25GbE 스위치의 전면을 보여줍니다.

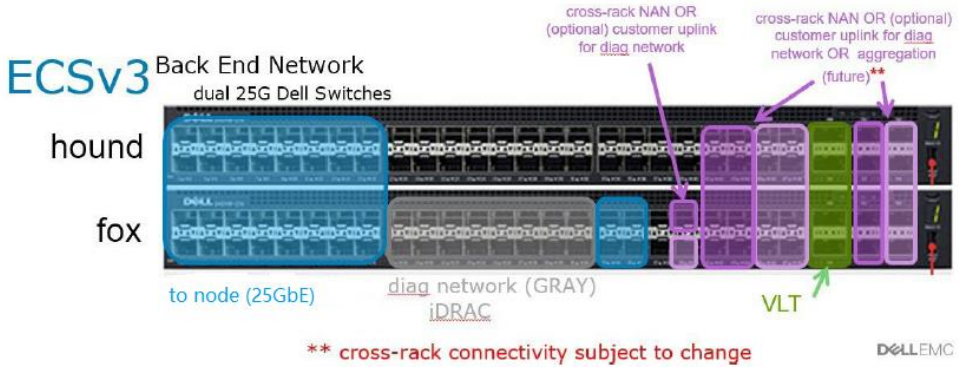


그림 17 백엔드 네트워크 스위치 포트 지정 및 사용

위의 다이어그램은 ECS 관리 트래픽 및 진단 포트를 활성화하기 위한 포트의 용도를 시각적으로 보여줍니다. 이러한 포트 할당 방식은 모든 구현에서 표준입니다. 향후 사용 가능한 포트는 보라색으로 표시되어 있지만 앞으로 변경될 수도 있습니다.

4.2.3 S5248F - 프런트엔드 퍼블릭 스위치

Dell EMC 에서는 고객이 랙에 대한 네트워크 연결을 구성할 수 있도록 단일 HA 쌍의 프런트엔드 25GbE S5248F 스위치(선택 사항)를 제공합니다. HA 쌍당 2 개의 200GbE(QSFP28-DD) VLT(Virtual Link Trunking) 케이블이 제공됩니다. 이러한 스위치를 산토끼 및 집토끼 스위치라고 합니다. 위의 그림 18 은 포트를 사용하여 ECS 노드 트래픽과 고객 업링크 포트를 활성화하는 방법을 시각적으로 보여줍니다.

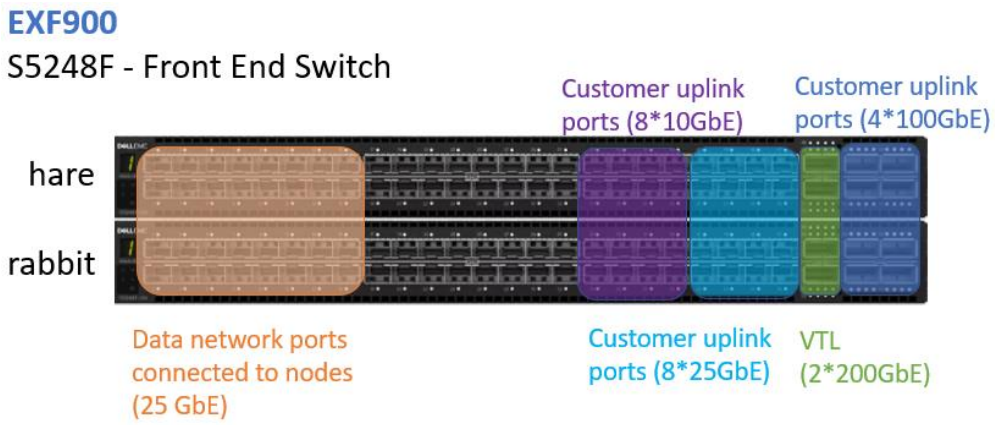


그림 18 프런트엔드 네트워크 스위치 포트 지정 및 사용

4.2.4 S5248F - 백엔드 프라이빗 스위치

Dell EMC 는 25GbE S5248F 백엔드 스위치 2 개와 200GbE(QSFP28-DD) VLT 케이블 2 개를 제공합니다. 이러한 스위치를 사냥개 및 여우 스위치라고 합니다. 노드의 모든 iDRAC 케이블과 모든 프런트엔드 스위치 관리 케이블 연결은 여우 스위치까지 이어집니다. 위의 그림 19 는 ECS 관리 트래픽 및 진단 포트를 활성화하기 위한 포트의 용도를 시각적으로 보여줍니다. 이러한 포트 할당 방식은 모든 구현에서 표준입니다.

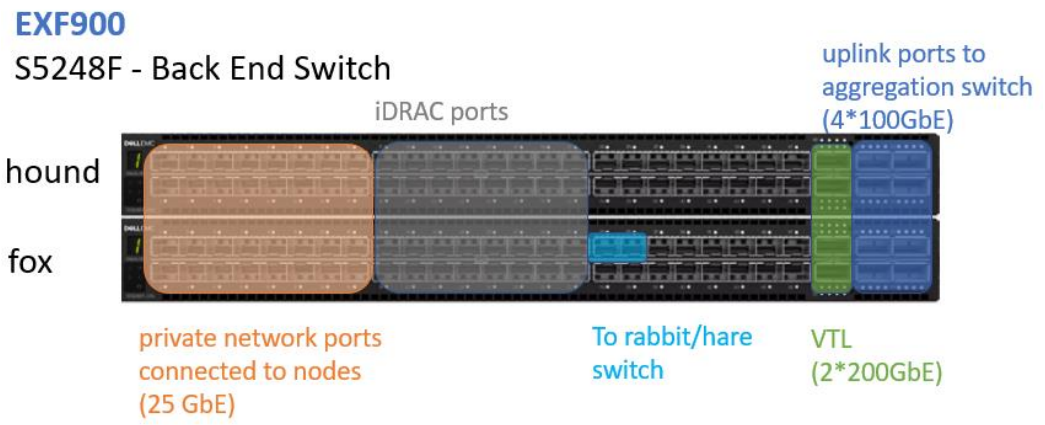


그림 19 백엔드 네트워크 스위치 포트 지정 및 사용

4.2.5 S5232 - 통합 스위치

Dell EMC 는 100GbE S5232F 백엔드 통합 스위치 2 개(AGG1 및 AGG2)와 100GbE VLT 케이블 4 개를 제공합니다. 이러한 스위치를 매 및 독수리 스위치라고 합니다. 다음 그림 20 에서 레이블이 지정된 모든 포트는 포트 명칭을 나타냅니다. 이 구성에서는 EXF900 노드의 랙을 7 개까지 연결할 수 있습니다.

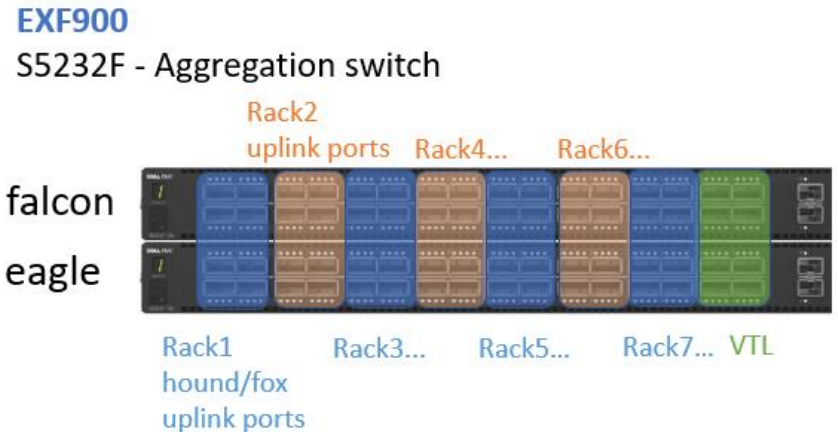


그림 20 통합 스위치 포트 지정 및 사용

네트워킹 및 케이블 연결에 대한 자세한 내용은 *ECS EX Series 하드웨어 가이드*를 참조하십시오.

5 네트워크 분리

ECS 는 보안 및 성능 격리를 위해 상이한 유형의 네트워크 트래픽에 대한 분리를 지원합니다. 분리할 수 있는 트래픽 유형은 다음과 같습니다.

- 관리
- 복제
- 데이터

*네트워크 분리 모드*라는 운영 모드가 있습니다. 이 모드에서는 각각의 트래픽 유형에 대해 운영 체제 레벨에서 각 노드를 최대 3 개의 IP 주소 또는 논리 네트워크로 구성할 수 있습니다. 이 기능은 관리, 복제, 데이터를 위한 세 개의 개별 논리 네트워크를 생성하거나 이들 네트워크를 결합하여 두 개의 논리 네트워크(예: 한 논리 네트워크는 관리 및 복제 트래픽에 사용하고 다른 논리 네트워크는 데이터 트래픽에 사용)를 생성하는 방식 중에서 선택할 수 있도록 설계되었습니다. CAS 전용 트래픽을 위한 두 번째 논리 데이터 네트워크를 구성하여 CAS 트래픽을 S3 와 같은 다른 유형의 데이터 트래픽과 분리할 수 있습니다.

ECS 에서 네트워크를 분리하려면 각 논리 네트워크 트래픽을 서비스 및 포트와 연결해야 합니다. 예를 들어 ECS 포털 서비스는 포트 80 또는 443 을 통해 통신하므로 이 포트와 서비스는 관리 논리 네트워크에 연결됩니다. 두 번째 데이터 네트워크를 구성할 수 있지만 이 네트워크는 CAS 트래픽 전용이 됩니다. 아래 표 5 는 논리 네트워크 유형에 고정된 서비스를 요약해 보여줍니다. 포트와 관련된 서비스의 전체 목록은 최신 *ECS 보안 구성 가이드*를 참조하십시오.

표 5 서비스-논리 네트워크 매핑

서비스	논리 네트워크	Identifier
WebUI 및 API, SSH, DNS, NTP, AD, SMTP	관리	public.mgmt
클라이언트 데이터	데이터	public.data
	CAS 전용 데이터	public.data2
복제 데이터	복제	public.repl
Dell EMC SRS(Secure Remote Services)	SRS 게이트웨이가 연결된 네트워크를 기준으로 함	public.data 또는 public.mgmt

참고: ECS 3.6 은 데이터(기본값) 및 data2 네트워크 모두에서 S3 데이터 액세스를 허용하지만, data2 에서는 기본적으로 S3 가 활성화되어 있지 않습니다. data2 네트워크에서 S3 데이터 액세스를 활성화하려면 public.data 가 필요하며 ECS 원격 지원 담당자에게 문의하십시오.

서로 다른 IP 주소를 사용하여 논리적으로, 서로 다른 VLAN 을 사용하여 가상으로, 또는 서로 다른 케이블을 사용하여 물리적으로 네트워크를 분리할 수 있습니다. *setrackinfo* 명령은 IP 주소와 VLAN 을 구성하는 데 사용됩니다. 스위치 레벨 또는 클라이언트 측 VLAN 구성은 고객이 담당합니다. 물리적 네트워크 분리를 원하는 고객은 Dell EMC Global Business Service 에 연락하여 RPQ(Request for Product Qualification)를 제출해야 합니다. 네트워크 분리에 대한 자세한 내용은 네트워크 분리를 개략적으로 살펴보는 *ECS 네트워킹 및 모범 사례(ECS Networking and Best Practices)* 백서를 참조하십시오.

6 보안

ECS 보안은 관리, 전송 및 데이터 레벨에서 구현됩니다. 사용자 및 관리자 인증은 Active Directory, LDAP 메서드, Keystone 을 통해 또는 ECS 포털 내에서 직접 수행됩니다. 데이터 레벨 보안은 이동 중인 데이터는 HTTPS 를 통해, 저장된 데이터는 서버 측 암호화를 통해 이루어집니다.

6.1 인증

ECS 는 ECS 를 관리하고 구성할 수 있는 액세스 권한을 제공하기 위해 Active Directory, LDAP 및 Keystone 인증 방법을 지원합니다. 그러나 표 6 에 표기된 것과 같은 제한이 있습니다. 보안에 대한 자세한 내용은 최신 *ECS 보안 구성 가이드*를 참조하십시오.

표 6 지원되는 인증 방법

인증 방법	지원됨
Active Directory	• 관리 사용자를 위한 AD 그룹 지원 • API 를 통한 셀프 서비스 키를 사용하는 오브젝트 사용자 셀프 프로비저닝 방식에 대해 AD 그룹 지원 • 다중 도메인 지원
LDAP	• 관리 사용자가 LDAP 을 통해 개별적으로 인증할 수 있음 • 관리 사용자에게는 LDAP 그룹이 지원되지 않음 • 오브젝트 사용자에게는 LDAP 이 지원됨(API 를 통한 셀프 서비스 키) • 다중 도메인 지원
Keystone	• RBAC 정책은 아직 지원되지 않음 • 범위가 없는 토큰은 지원되지 않음 • 한 ECS 시스템에서 여러 Keystone 서버가 지원되지 않음
IAM	• SAML 2.0 표준을 통해 ID 페더레이션 및 SSO(Single Sign-On) 제공 • S3 프로토콜을 통해서만 사용 가능

6.2 데이터 서비스 인증

RESTful API 를 사용하는 오브젝트 액세스는 HTTPS(TLS v1.2)를 통해 보호됩니다. 들어오는 요청은 HBAC(Hash-based Message Authentication Code), Kerberos, 토큰 인증 등의 정의된 방식을 사용하여 인증됩니다. 아래 표 7 은 각 프로토콜에 사용되는 다양한 방법을 소개합니다.

표 7 데이터 서비스 인증

Protocols		**인증 방법**
객체	S3	V2(HMAC-SHA1), V4(HMAC-SHA256)
	Swift	토큰 - Keystone v2 및 v3(범위, UUID, PKI 토큰), SWAuth v1
	Atmos	HMAC-SHA1
	CAS	비밀 키 PEA 파일
File	HDFS	Kerberos
	NFS	Kerberos, AUTH_SYS

6.3 D@RE(Data-At-Rest Encryption)

규정 준수 요건에 따라 디스크에 기록된 데이터를 암호화를 통해 보호하는 것이 필수적인 경우가 많습니다. ECS 에서는 네임스페이스 및 버킷 레벨에서 암호화를 활성화할 수 있습니다. ECS D@RE 의 주요 기능은 다음과 같습니다.

- 기본 제공되는 간편한 저장된 데이터 암호화 - 쉽게 활성화됨, 간단한 구성
- CIPHER(AES-256 CTR) 사용
- 2,048 비트 길이로 RSA 공개 키 암호화
- EKM(External Key Management) 클러스터 레벨 지원:
  - Gemalto SafeNet
  - IBM Security Key Lifecycle Manager
- 키 순환
- *x-amz-server-side-encryption* 과 같은 HTTP 헤더를 사용하는 S3 암호화 의미 체계 지원
- 미국 정부 암호화 보안 표준을 준수하는 FIPS 140-2

참고: FIPS 140-2 모드는 D@RE 내에서 승인된 전용 알고리듬의 사용을 시행합니다. FIPS 140-2 규정 준수는 전체 ECS 제품이 아니라 D@RE 모듈에만 적용됩니다.

ECS 는 키 계층 구조를 사용하여 데이터를 암호화하고 해독합니다. 기본 키 매니저는 기본 키를 해독하기 위해 모든 노드에 공통적인 개인 키를 저장합니다. EKM 구성에서는 EKM 이 기본 키를 제공합니다. EKM 에서 제공하는 키는 ECS 에서 메모리에만 상주하며 절대로 ECS 내의 영구 스토리지에 저장되지 않습니다.

원거리 복제 환경에서 새로운 ECS 시스템이 기존 페더레이션에 조인되면 기존 시스템의 공개-개인 키를 사용하여 기본 키가 추출되고 페더레이션 구성에 조인된 새로운 시스템에서 생성된 새로운 공개-개인 키를 사용하여 마스터 키가 암호화됩니다. 이때부터는 기본 키가 전역 키가 되어 페더레이션 구성에 포함된 두 시스템이 모두 마스터 키를 인식합니다. EKM 을 사용할 때는 모든 페더레이션된 시스템이 키 관리 시스템에서 기본 키를 검색합니다.

6.3.1 키 순환

ECS 에서는 암호화 키 변경을 지원합니다. 특정 KEK(Key Encryption Key) 집합에 의해 보호되는 데이터 양을 제한하거나 잠재적인 유출 또는 침해에 대응하기 위해 주기적으로 암호화 키를 변경할 수 있습니다. 순환 KEK 레코드는 다른 상위 키와 함께 DEK(Data Encryption Key) 및 네임스페이스 KEK 를 보호하기 위한 가상 래핑 키를 생성하는 데 사용됩니다.

순환 키는 기본적으로 생성되거나 제공되며 EKM 에서 관리합니다. ECS 는 현재 순환 키를 사용하여 키 관리가 기본적으로 이루어지는지 혹은 외부에서 이루어지는지에 관계없이 DEK 또는 KEK 를 보호하기 위한 가상 래핑 키를 생성합니다.

쓰기 중에 ECS 는 버킷 및 활성 순환 키를 통해 생성된 가상 래핑 키를 사용하여 임의로 생성된 DEK 를 래핑합니다.

키 순환의 일환으로 ECS 는 모든 네임스페이스 KEK 레코드를 새 순환 키로부터 생성된 새 가상 기본 KEK, 관련 비밀 컨텍스트, 활성 기본 키와 함께 다시 래핑합니다. 이는 이전 순환 키에 의해 보호되는 데이터에 대한 액세스를 보호하기 위한 것입니다.

EKM 을 사용하면 암호화된 오브젝트의 읽기/쓰기 경로에 영향을 줍니다. 키 순환은 DEK 및 네임스페이스 KEK 에 대한 가상 래핑 키를 사용함으로써 데이터를 추가적으로 보호할 수 있습니다. 가상 래핑 키는 보존되지 않으며, 보존되는 키의 두 독립적인 계층 구조에서 파생됩니다. EKM 을 사용하면 순환 키가 ECS 에 저장되지 않으며 데이터 보안이 강화됩니다. 관리자는 주로 새로운 KEK 레코드를 추가하고 활성 ID 를 업데이트하지만 아무것도 삭제하지 않습니다.

ECS 의 키 순환과 관련하여 다음 사항을 추가로 고려해야 합니다.

- 키를 순환하는 프로세스는 현재 순환 키만 변경합니다. 키 순환 프로세스 중에 기존 기본, 네임스페이스 및 버킷 키는 변경되지 않습니다.
- 네임스페이스 또는 버킷 레벨 키 순환은 지원되지 않습니다. 그러나 회전 범위가 클러스터 레벨이므로 시스템에서 암호화하는 모든 신규 오브젝트가 영향을 받습니다.
- 기존 데이터는 순환 키로 인해 다시 암호화되지 않습니다.
- ECS 는 운영 중단 중에 키 순환을 지원하지 않습니다.
  - 순환 중 TSO: 키 순환 작업은 시스템이 TSO 에서 빠져나올 때까지 일시 중단됩니다.
  - PSO 진행 중: ECS 는 키 순환이 활성화되기 전에 PSO 에서 빠져나와야 합니다. 순환 중에 PSO 가 발생하면 순환이 즉시 실패합니다.
- S3 를 통해 오브젝트를 암호화할 때 버킷 암호화가 필요하지 않습니다.
- 검색 키로 활용되는 인덱싱된 클라이언트 오브젝트 메타데이터는 암호화되지 않습니다.

D@RE, EKM, 키 순환에 대한 자세한 내용은 최신 *ECS 보안 구성 가이드*를 참조하십시오.

6.4 ECS IAM

ECS IAM(Identity and Access Management)을 사용하면 ECS S3 리소스를 제어하고 해당 리소스에 안전하게 액세스할 수 있습니다. 이 기능은 ECS 리소스에 대한 각 액세스 요청이 식별되고 인증 및 사용 권한 부여되도록 합니다. ECS IAM 을 사용하면 관리자가 사용자, 역할 및 그룹을 추가할 수 있습니다. 또한 관리자는 ECS IAM 엔터티에 정책을 추가하여 액세스 권한을 제한할 수 있습니다.

참고: ECS IAM 은 S3 에서만 사용할 수 있습니다. CAS 또는 파일 시스템 지원 버킷은 지원하지 않습니다.

ECS IAM 의 구성 요소는 다음과 같습니다.

- **어카운트 관리** - 각 네임스페이스 내에서 사용자, 그룹 및 역할과 같은 IAM ID 를 관리할 수 있습니다.
- **액세스 관리** - 정책을 생성하고 IAM ID 또는 리소스에 연결하여 액세스를 관리합니다.
- **ID 페더레이션** - SAML(Security Assertion Markup Language)이 ID 를 설정하고 인증합니다. ID 가 설정되면 보안 토큰 서비스를 사용하여 리소스에 액세스하는 데 사용할 임시 자격 증명을 얻습니다.
- **보안 토큰 서비스** - 리소스에 대한 교차 계정 액세스, 그리고 엔터프라이즈 ID 공급자 또는 디렉토리 서비스의 SAML 인증을 사용하여 인증을 받은 사용자를 위해 임시 자격 증명을 요청할 수 있습니다.

IAM 을 사용하면 다음 항목을 생성하고 관리할 수 있으며, 이를 통해 인증을 받고 사용 권한이 부여된 후 ECS 리소스를 사용할 수 있는 사람을 제어할 수 있습니다.

- **사용자** - IAM 사용자는 네임스페이스에서 ECS 리소스와 상호 작용할 수 있는 사람 또는 애플리케이션을 나타냅니다.
- **그룹** - IAM 그룹은 IAM 사용자의 집합입니다. 그룹을 사용하면 IAM 사용자 집합에 대한 권한을 지정할 수 있습니다.
- **역할** - IAM 역할은 역할이 필요한 모든 사용자가 맡을 수 있는 ID 입니다. 역할은 ID 가 무엇을 할 수 있고 무엇을 할 수 없는지를 결정하는 사용 권한 정책이 있는 ID 라는 점에서 사용자와 유사합니다.
- **정책** - IAM 정책은 역할에 대한 사용 권한을 정의하는 JSON 형식의 문서입니다. IAM 사용자, IAM 그룹 및 IAM 역할에 정책을 할당하고 연결합니다.
- **SAML 공급자** - SAML 은 ID 공급업체와 서비스 공급업체 간에 인증 및 사용 권한 부여 데이터를 교환하는 개방형 표준입니다. ECS 의 SAML 공급자는 SAML 호환 IdP(Identity Provider)와 ECS 간에 신뢰를 구축하는 데 사용됩니다.

각 ECS 시스템은 ECS IAM 계정에 할당됩니다. 이 계정은 여러 네임스페이스를 지원하며 해당 네임스페이스에 정의된 관련 IAM 엔터티를 보유하고 있습니다.

- 개별 네임스페이스는 사용자, 역할 및 그룹과 같은 ECS IAM 엔터티를 사용하여 계정을 관리할 수 있도록 지원합니다.
- 정책, 권한, ECS IAM 엔터티와 연결된 ACL(Access Control List), ECS S3 리소스는 ECS IAM 기능에 대한 액세스 관리를 지원합니다.
- ECS IAM 은 SAML(Security Assertion Markup Language) 및 역할을 사용하여 계정 간 액세스를 지원합니다.
- ECS IAM 은 ECS 에서 IAM 및 S3 에 액세스할 수 있도록 AWS(Amazon Web Services) 액세스 키를 지원합니다.

ECS IAM 에 대한 자세한 내용은 최신 *ECS 보안 가이드*를 참조하십시오.

6.5 Object tagging

오브젝트 태그 지정을 사용하면 개별 오브젝트에 태그를 할당하여 오브젝트를 분류할 수 있습니다. 단일 오브젝트에는 여러 태그가 연결되어 다차원 분류가 가능합니다.

태그는 건강 기록과 같은 기밀 정보를 설명할 수 있으며, 기밀 정보로 분류될 수 있는 특정 제품에 대한 오브젝트에 태그를 지정할 수 있습니다. 태그 지정은 오브젝트 작업에 통합된 수명주기가 있는 오브젝트의 하위 리소스입니다. 새 오브젝트를 업로드할 때 태그를 추가하거나 기존 오브젝트에 태그를 추가할 수 있습니다. 태그를 사용하여 PII(Personally Identifiable Information) 또는 PHI(Protected Health Information)와 같은 기밀 데이터가 포함된 오브젝트에 레이블을 지정할 수 있습니다. 오브젝트에 대한 실질적인 읽기 사용 권한 없이도 태그를 볼 수 있으므로 태그에 기밀 정보가 포함되어서는 안 됩니다.

6.5.1 오브젝트 태그 지정에 대한 추가 정보

이 섹션에서는 IAM 의 오브젝트 태그 지정, 버킷 정책을 사용하는 오브젝트 태그 지정, TSO/PSO 중 오브젝트 태그 지정 처리, 그리고 오브젝트 수명주기 관리 중 오브젝트 태그 지정에 대한 정보를 제공합니다. 다음은 몇 가지 추가 참고 사항입니다.

- IAM 의 오브젝트 태그 지정
  - 분류 시스템으로서 오브젝트 태그 지정의 주요 기능은 IAM 의 정책과 통합될 때 제공됩니다. 이를 활용하여 관리자는 특정 사용자 권한을 구성할 수 있습니다. 예를 들어 관리자는 모든 사용자가 지정된 태그를 사용하여 오브젝트에 액세스할 수 있도록 하는 정책을 추가하거나 특정 오브젝트의 태그를 관리할 수 있는 사용자를 위한 권한을 구성하고 이를 부여할 수 있습니다. 오브젝트 태그 지정과 관련된 다른 주요 측면은 태그를 유지하는 방법과 위치입니다. 이는 시스템의 다양한 측면에 직접적인 영향을 끼치기 때문에 중요합니다.
- 버킷 정책을 사용하여 오브젝트 태그 지정
  - 오브젝트 태그 지정을 사용하면 오브젝트를 분류할 수 있으며 이외에도 다양한 정책과 통합하여 태그를 지정할 수 있습니다. 수명주기 관리 정책을 사용하여 버킷 수준에서 구성할 수 있습니다. 이전 버전의 ECS 는 만료, 불완전한 업로드 중단, 만료된 오브젝트 태그 지정 삭제 마커의 삭제를 지원합니다. 필터에는 태그 기반 조건을 포함한 여러 조건이 포함될 수 있습니다. 필터 조건의 각 태그가 키와 값과 일치해야 합니다.
- TSO/PSO 중 오브젝트 태그 지정
  - 오브젝트 태그 지정은 시스템 메타데이터의 또 다른 항목 세트로서 TSO/PSO 중에 특별한 처리가 필요하지 않습니다. 각 오브젝트와 연결될 수 있는 태그 수에는 제한 값이 설정되어 있으며, 오브젝트 태그 지정을 사용하는 시스템 메타데이터의 크기는 메모리 제한을 초과하지 않습니다.
- 오브젝트 수명주기 관리 중 오브젝트 태그 지정
  - 오브젝트 태그 지정은 시스템 메타데이터의 일부로서 수명주기 관리 도중에 시스템 메타데이터 처리와 동시에 진행됩니다. 만료 논리 및 수명주기 삭제 스캐너는 태그 기반 정책을 이해하는 데 필요합니다. 오브젝트 태그를 사용하면 수명주기 규칙에서 키 이름 접두사와 함께 태그 기반 필터를 지정할 수 있는 상세한 오브젝트 수명주기 관리가 가능합니다.

ECS 오브젝트 태그 지정에 대한 자세한 내용은 최신 *ECS 보안 구성 가이드*를 참조하십시오.

7 데이터 무결성 및 보호

ECS 는 데이터 무결성을 위해 체크섬을 활용합니다. 체크섬은 쓰기 작업 중에 생성되며 데이터와 함께 저장됩니다. 데이터를 읽을 때 체크섬이 계산되어 저장된 버전과 비교됩니다. 백그라운드 작업 검사는 체크섬 정보를 사전에 확인합니다.

ECS 는 데이터 보호를 위해 저널 청크에 트리플 미러링을 활용하고 *repo*(사용자 저장소 데이터) 및 *btree*(B+ 트리) 청크에는 별도의 EC 스키마를 활용합니다.

삭제 코딩은 기존의 보호 스키마에 비해 스토리지 효율적인 방식으로 디스크, 노드 및 랙 장애로부터 데이터를 더 안전하게 보호합니다. ECS 스토리지 엔진은 다음의 두 가지 스키마를 사용하여 Reed Solomon 오류 수정을 구현합니다.

- 12+4(기본값) - 청크가 12 개의 데이터 세그먼트로 분할됩니다. 4 개의 코딩(패리티) 세그먼트가 생성됩니다.
- 10+2(콜드 아카이브) - 청크가 10 개의 데이터 세그먼트로 분할됩니다. 2 개의 코딩 세그먼트가 생성됩니다.

기본값인 12+4 를 사용하면 최종적으로 16 개의 세그먼트가 로컬 사이트의 노드 사이에 분산됩니다. 각 청크의 데이터 및 코딩 세그먼트는 클러스터의 노드 간에 균등하게 분산됩니다. 예를 들어 노드가 8 개인 경우에는 각 노드에 2 개의 세그먼트가 있습니다(총 16 개). 스토리지 엔진은 16 개의 세그먼트 중 12 개에서 청크를 재구성할 수 있습니다.

ECS 에서 콜드 아카이브 옵션에는 최소 6 개의 노드가 필요하며 이 경우 12+4 대신 10+2 스키마가 사용됩니다. 노드 수가 EC 스키마에 필요한 최소값 아래로 떨어지면 EC 가 중지됩니다.

청크가 가득 찰 때 또는 설정된 기간 이후에 청크가 봉인되고, 패리티가 계산되고, 코딩 세그먼트가 장애 도메인 전체의 디스크에 기록됩니다. 청크 데이터는 클러스터 전체에 분산된 16 개의 세그먼트(12 개 데이터, 4 개 코드)로 구성된 단일 복제본으로 유지됩니다. ECS 는 장애가 발생하면 청크 재구성을 위해 코드 세그먼트만 사용합니다.

VDC 의 기본 인프라스트럭처가 노드 또는 랙 레벨에서 변경될 때 패브릭 계층은 변경 사항을 감지하며 재조정 스캐너를 백그라운드 작업으로 트리거합니다. 스캐너는 새 토폴로지를 사용하여 각 청크에 대해 장애 도메인 전반에서 가장 적합한 EC 세그먼트 레이아웃을 계산합니다. 새 레이아웃이 기존 레이아웃보다 더 효과적으로 보호하는 경우 ECS 는 EC 세그먼트를 백그라운드 작업으로 다시 배포합니다. 이 작업은 시스템 성능에 미치는 영향을 최소화합니다. 그러나 재조정 중에 노드 간 트래픽이 증가합니다. 새 노드로의 논리 테이블 파티션의 밸런싱도 이루어지며 새로 생성된 저널과 B+ 트리 청크는 앞으로 이전 노드와 새 노드에 균등하게 할당됩니다. 재배포는 인프라스트럭처 내의 모든 리소스를 활용하여 로컬 보호를 강화합니다.

참고: 드라이브나 노드를 스토리지 플랫폼이 완전히 가득 찰 때까지 기다렸다가 추가하는 것은 좋지 않은 방법입니다. 일일 수집 속도와 추가할 드라이브/노드의 예상 주문, 배송, 통합 시간을 고려했을 때 합리적인 스토리지 사용률 임계값은 70%입니다.

7.1 규정 준수

ECS 는 데이터 스토리지에 대한 기업과 업계의 규정 준수 요건(SEC Rule 17a-4(f))을 충족하기 위해 다음 사항을 구현했습니다.

- **플랫폼 강화** - 노드 또는 클러스터에 대한 액세스를 비활성화하는 플랫폼 잠금, *ftpd, sshd* 같은 필수적이지 않은 모든 포트 닫기, sudo 명령에 대한 전체 감사 로깅, 노드에 대한 원격 액세스를 종료하도록 하는 Dell EMC SRS(Secure Remote Services) 지원 등의 보안 강화로 ECS 의 보안 취약성을 해결합니다.
- **규정 준수 보고** - 시스템 에이전트가 시스템의 규정 준수 상태를 규정 준수 시 *Good*, 미준수 시 *Bad* 와 같이 보고합니다.
- **정책 기반 레코드 보존 및 규칙** - 정책, 기간, 규칙을 사용하여 보존 중인 레코드 또는 데이터의 변경을 제한할 수 있습니다.
- **ARM(Advanced Retention Management)** - Centera 규정 준수 요건을 충족하기 위해 CAS 전용의 보존 규칙 집합이 정의되었습니다.
  - **이벤트 기반 보존** - 지정된 이벤트가 발생할 때 시작되는 보존 기간을 활성화합니다.
  - **법적 증거 자료 보존** - 법적 조치를 받을 수 있는 데이터의 임시 삭제 방지를 활성화합니다.
  - **최소/최대 관리자** - 최소 및 최대 기본 보존 기간에 대한 버킷별 설정입니다.

규정 준수는 네임스페이스 레벨에서 활성화됩니다. 보존 기간은 버킷 레벨에서 구성됩니다. 규정 준수 요건은 플랫폼을 인증합니다. 따라서 규정 준수 기능은 어플라이언스 하드웨어에서 실행되는 ECS 에서만 사용할 수 있습니다. ECS 의 규정 준수 활성화 및 구성에 대한 자세한 내용은 최신 *ECS 데이터 액세스 가이드* 및 최신 *ECS 관리자 가이드*를 참조하십시오.

8 배포

ECS 는 단일 또는 다중 사이트 인스턴스로 구축될 수 있습니다. ECS 구축의 구성 요소는 다음과 같습니다.

- **VDC(Virtual Data Center)** - 단일 패브릭 인스턴스에 의해 관리되는 ECS 인프라스트럭처 집합으로 구성된 클러스터로 일반적으로 사이트 또는 지리적으로 구분된 리전이라고도 합니다.
- **SP(Storage Pool)** - SP 는 노드의 하위 집합과 VDC 에 속하는 관련 스토리지라고 생각하면 됩니다. 한 노드는 하나의 SP 에만 속할 수 있습니다. EC 는 12+4 또는 10+2 스키마로 SP 레벨에서 설정됩니다. SP 는 ECS 의 저장소에 액세스하는 클라이언트 또는 클라이언트 그룹 간에 데이터를 물리적으로 분리하는 툴로 사용될 수 있습니다.
- **RG(Replication Group)** - RG, 즉 복제 그룹은 SP 콘텐츠가 보호되는 위치와 데이터에 액세스할 수 있는 위치를 정의합니다. 멤버 사이트가 하나만 있는 RG 를 로컬 RG 라고도 합니다. 데이터는 항상 디스크, 노드 및 랙 장애에 대비하여 기록되는 곳에서 로컬로 보호됩니다. 두 개 이상의 사이트가 있는 RG 를 흔히 글로벌 RG 라고 합니다. 글로벌 RG 는 최대 8 개의 VDC 에 걸쳐 확장되며 디스크, 노드, 랙 및 사이트 장애로부터 보호합니다. 한 VDC 는 여러 RG 에 속할 수 있습니다.
- **네임스페이스** - 네임스페이스는 개념적으로 ECS 의 테넌트와 동일합니다. 네임스페이스의 주요 특성은 한 네임스페이스의 사용자가 다른 네임스페이스에 있는 오브젝트에 액세스할 수 없다는 점입니다.
- **버킷** - 버킷은 네임스페이스에서 생성된 오브젝트를 위한 컨테이너이며 때로는 하위 테넌트를 위한 논리 컨테이너로 간주되기도 합니다. S3 에서는 컨테이너를 버킷이라고 하며 이 용어가 ECS 에서 채택되었습니다. Atmos 에서는 서브테넌트, Swift 에서는 컨테이너, CAS 에서는 CAS 풀이 각각 버킷에 해당하는 용어입니다. 버킷은 ECS 의 글로벌 리소스입니다. 각 버킷은 네임스페이스에 생성되고 각 네임스페이스는 RG 에서 생성됩니다.

ECS 는 다음과 같은 인프라스트럭처 시스템을 활용합니다.

- **DNS** - (필수) 각 ECS 노드에 필요한 순방향 및 역방향 조회입니다.
- **NTP** - (필수) Network Time Protocol 서버입니다.
- **SMTP** - (선택 사항) 보고하고 알림을 보내기 위한 Simple Mail Transfer Protocol 서버입니다.
- **DHCP** - (선택 사항) DHCP 를 통해 IP 주소를 할당하는 경우 필수입니다.
- **인증 공급자** - (선택 사항) ECS 관리자는 Active Directory 및 LDAP 그룹을 사용하여 인증할 수 있습니다. 오브젝트 사용자는 Keystone 을 사용하여 인증할 수 있습니다. ECS 에는 인증 공급자가 필요하지 않습니다. ECS 에는 로컬 사용자 관리 기능이 내장되어 있지만 로컬에서 생성된 사용자는 VDC 간에 복제되지 않습니다.

- **로드 밸런서** - (워크플로에서 지시하는 경우 필수, 그 외에는 선택 사항) 시스템에서 사용할 수 있는 모든 리소스를 효과적으로 활용하려면 클라이언트 로드를 노드 간에 분산해야 합니다. ECS 노드에서 로드를 관리하기 위해 전용 로드 밸런서 어플라이언스 또는 서비스가 필요한 경우에는 필수로 간주해야 합니다. ECS S3 SDK 를 사용하여 애플리케이션을 제작하는 개발자는 기본 제공되는 로드 밸런서 기능을 활용할 수 있습니다. 정교한 로드 밸런서는 서버의 보고된 로드, 응답 시간, 증가/감소 상태, 활성 연결 수, 지리적 위치 등의 추가적인 요소를 고려할 수 있습니다. 고객은 클라이언트 트래픽을 관리하고 액세스 요구 사항을 결정할 책임이 있습니다. 다양한 방식 중에서도 수동 IP 할당, DNS 라운드 로빈(Round Robin), 클라이언트 측 로드 밸런싱, 로드 밸런서 어플라이언스, 지리적 로드 밸런서 등 일반적으로 고려되는 몇 가지 기본 옵션이 있습니다. 다음은 각각의 방식에 대한 간략한 설명입니다.

  - **수동 IP 할당** - IP 주소를 애플리케이션에 수동으로 분배합니다. 로드를 분산하지 못하거나 내결함성을 구현하지 못할 수 있으므로 일반적으로 권장되지 않습니다.
  - **DNS 라운드 로빈** - 모든 노드 IP 주소를 포함하는 DNS 항목이 생성됩니다. 클라이언트는 ECS 서비스의 FQDN(Fully-Qualified Domain Name)을 확인하기 위해 DNS 를 쿼리하고 임의 노드의 IP 주소로 응답을 받습니다. 이로써 일종의 유사 로드 밸런싱을 제공할 수 있습니다. DNS 에서 실패한 노드의 IP 주소를 수동으로 제거하는 경우가 종종 있기 때문에 이 방법은 내결함성을 제공하지 못할 수 있습니다. 이 방법에서 TTL(Time To Live) 문제가 발생할 수 있습니다. 일부 DNS 서버 구현은 가까운 시간 내에 연결하는 클라이언트가 동일한 IP 주소에 바인딩되도록 일정 기간 동안 DNS 조회를 캐시함으로써 데이터 노드에 분산되는 로드 양을 줄일 수 있습니다. DNS 를 사용하여 트래픽을 라운드 로빈 방식으로 분산하는 것은 권장되지 않습니다.
  - **로드 밸런싱** - 로드 밸런서는 클라이언트 로드를 분산하는 가장 일반적인 방법입니다. 클라이언트는 트래픽을 로드 밸런서로 보낼 수 있으며 로드 밸런서는 이를 받아서 정상 ECS 노드에 전달합니다. 각 노드의 서비스 요청 가용성을 확인하기 위해 사전 예방적 상태 점검 또는 연결 상태를 사용합니다. 사용할 수 없는 노드는 상태 점검을 통과할 때까지 사용에서 배제됩니다. CPU 를 많이 사용하는 SSL 처리를 오프로드하면 ECS 에서 이러한 리소스를 확보할 수 있습니다.
  - **지리적 로드 밸런싱** - 지리적 로드 밸런싱은 DNS 를 활용하여 조회를 어플라이언스에 라우팅합니다. 이러한 어플라이언스의 예로 지리적 IP 또는 다른 메커니즘을 사용하여 클라이언트를 라우팅할 최적 사이트를 결정하는 Riverbed SteelApp 이 있습니다.

8.1 단일 사이트 구축

단일 사이트 또는 단일 클러스터 초기 구축 중에는 노드가 먼저 SP 에 추가됩니다. SP 는 물리적 노드의 논리 컨테이너입니다. SP 를 구성하려면 필요한 최소의 사용 가능한 노드 수를 선택하고 기본 12+4 또는 콜드 아카이브 10+2 EC 스키마를 선택해야 합니다. SP 를 구성할 때 처음에 그리고 나중에 중요 알림 수준을 설정할 수 있습니다. 하지만 SP 초기화 후에는 EC 스키마를 변경할 수 없습니다. 생성된 첫 번째 SP 가 시스템 SP 로 지정되며 시스템 메타데이터를 저장하는 데 사용됩니다. 시스템 SP 는 삭제할 수 없습니다.

클러스터는 일반적으로 그림 21 과 같이 각 EC 스키마에 하나씩, 1~2 개의 SP 를 포함합니다. 하지만 데이터를 물리적으로 분리해야 하는 조직에서는 경계를 구현하기 위해 추가적인 SP 가 사용됩니다.

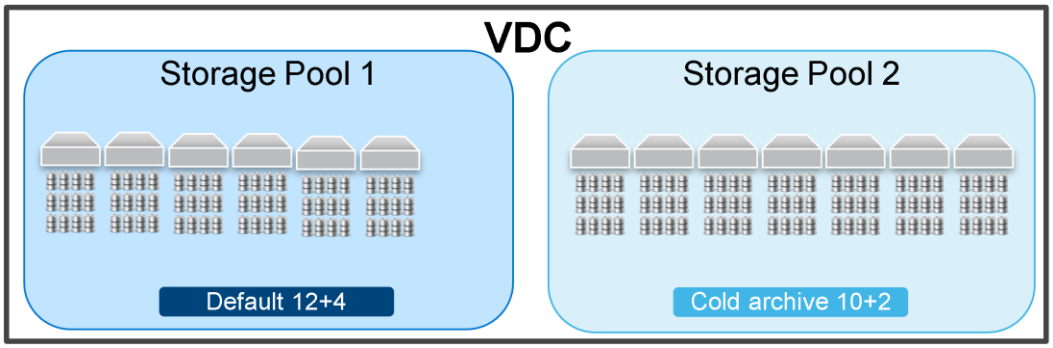


그림 21 각각 서로 다른 EC 스키마로 구성된 두 개의 스토리지 풀이 있는 VDC

첫 번째 SP 를 초기화한 후에 VDC 를 생성할 수 있습니다. VDC 를 구성하려면 복제 및 관리 엔드포인트를 지정해야 합니다. VDC 를 생성하기 전에 시스템 SP 초기화가 필요하지만 VDC 구성은 SP 를 할당하지 않고 노드의 IP 주소를 할당합니다.

VDC 가 생성된 후에 RG 가 구성됩니다. RG 는 단일 또는 초기 사이트 설정에서 적어도 하나의 VDC 자체를 VDC 의 SP 중 하나와 함께 지정하여 구성해야 하는 글로벌 리소스입니다. 단일 VDC 멤버가 있는 RG 는 디스크, 노드 및 랙 레벨에서 로컬로 데이터를 보호합니다. 다음 섹션에서는 멀티 사이트 구축을 포함하도록 RG 를 확장합니다.

네임스페이스는 생성된 후에 RG 에 할당되는 글로벌 리소스입니다. 네임스페이스 레벨 보존 정책에서는 할당량, 규정 준수 및 네임스페이스 관리자가 정의됩니다. ADO(Access During Outage)는 다음 섹션에서 다루는 네임스페이스 레벨에서 구성할 수 있습니다. 일반적으로 네임스페이스 레벨에서 테넌트가 구성됩니다. 테넌트는 애플리케이션 인스턴스이거나 팀, 사용자, 비즈니스 그룹이거나 조직에 적합한 다른 그룹일 수 있습니다.

버킷은 여러 사이트에 걸쳐 확장될 수 있는 글로벌 리소스입니다. 버킷을 생성하려면 네임스페이스 및 RG 에 할당하는 작업이 필요합니다. 버킷 레벨은 소유권과 파일 또는 CAD 액세스가 활성화되는 곳입니다. 아래 그림 22 는 두 개의 버킷이 포함된 네임스페이스를 사용하는 VDC 의 SP 하나를 보여줍니다.

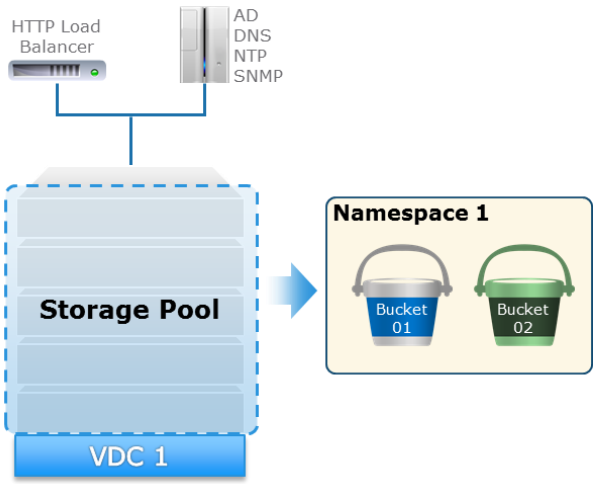


그림 22 단일 사이트 구축 예시

8.2 멀티 사이트 구축

페더레이션 환경 또는 페더레이션된 ECS 라고도 하는 멀티 사이트 구축은 최대 8 개의 VDC 에 걸쳐 있을 수 있습니다. ECS 에서 데이터는 청크 레벨에서 복제됩니다. RG 에 참여하는 노드는 로컬 데이터를 다른 사이트 하나 또는 다른 모든 사이트에 비동기식으로 보냅니다. 데이터는 WAN 을 거쳐 HTTP 를 통해 전송되기 전에 AES256 을 사용하여 암호화됩니다. 여러 VDC 를 페더레이션할 때의 알려진 주요 이점은 다음과 같습니다.

- 단일 논리 리소스에서 여러 VDC 를 통합하여 관리할 수 있음
- 노드, 디스크 및 랙 레벨의 로컬 보호 외에 사이트 레벨 보호도 가능함

- 어디서나 일정한 방법으로 지리적으로 분산된 스토리지에 액세스할 수 있음

멀티 사이트 구축을 다루는 이 섹션에서는 다음과 같은 페더레이션된 ECS 의 고유 기능에 대해 설명합니다.

- **데이터 정합성** - 기본적으로 ECS 는 강력한 정합성이 보장되는 스토리지 서비스를 제공합니다.
- **복제 그룹** - 보호 및 액세스 경계를 지정하는 데 사용되는 글로벌 컨테이너입니다.
- **지리적 캐싱** - 멀티 사이트 구축에서 원격 사이트 액세스 워크플로를 최적화합니다.
- **ADO** - TSO(Temporary Site Outage) 중의 클라이언트 액세스 동작입니다.

8.2.1 데이터 정합성

ECS 는 소유권을 사용하여 각 네임스페이스, 버킷, 오브젝트의 신뢰할 수 있는 버전을 관리하는 강력한 정합성이 보장되는 시스템입니다. 소유권은 네임스페이스, 버킷, 오브젝트가 생성되는 VDC 에 할당됩니다. 예를 들어 네임스페이스 NS1 이 VDC1 에서 생성된 경우, VDC1 은 NS1 을 소유하며 NS1 내에서 신뢰할 수 있는 버전의 버킷을 관리해야 할 책임이 있습니다. 버킷 B1 이 NS1 내부의 VDC2 에서 생성된 경우, VDC2 는 B1 을 소유하며 버킷 콘텐츠의 신뢰할 수 있는 버전은 물론 각 오브젝트의 소유자 VDC 도 관리해야 할 책임이 있습니다. 마찬가지로, 오브젝트 O1 이 VDC3 의 B1 내부에서 생성된 경우 VDC3 는 O1 을 소유하며 O1 및 관련 메타데이터의 신뢰할 수 있는 버전을 관리해야 할 책임이 있습니다.

멀티 사이트 데이터 보호의 복원력이 향상될 경우 그 대신 스토리지 보호 오버헤드와 WAN 대역폭 소비가 증가합니다. 인덱스 쿼리는 오브젝트를 소유하지 않는 사이트에서 오브젝트를 액세스하거나 업데이트할 때 필요합니다. 마찬가지로, 네임스페이스에 있는 신뢰할 수 있는 버킷 목록이나 원격 사이트가 소유한 버킷에 있는 오브젝트 등의 정보를 검색할 때는 WAN 전체에서 인덱스를 조회해야 합니다.

관리자와 애플리케이션 소유자는 ECS 가 소유권을 활용하여 네임스페이스, 버킷 및 오브젝트 레벨에서 데이터를 신뢰할 수 있는 방식으로 추적하는 방법을 이해함으로써 액세스 환경을 더 효과적으로 구성할 수 있습니다.

8.2.2 액티브 복제 그룹

RG 를 생성하는 동안 *Replicate to All Sites* 설정을 꺼진 상태(기본값)로 사용하거나 토글하여 활성화할 수 있습니다. 모든 사이트에 데이터를 복제한다는 것은 각 VDC 에 개별적으로 기록된 데이터가 다른 모든 RG 멤버 VDC 에 복제된다는 의미입니다. 예를 들어, 모든 사이트에 데이터를 복제하도록 구성된 활성 RG 가 있는 페더레이션된 X-number-of-sites ECS 인스턴스는 X 배의 보호 오버헤드 또는 X*1.33(콜드 아카이브 EC 에서는 1.2) 총 데이터 보호 오버헤드를 초래합니다. 특히 로컬 액세스가 중요한 소규모 데이터 세트의 경우 모든 사이트에 복제하는 것이 적합할 수 있습니다. 이 설정을 해제하면 각 VDC 에 기록된 모든 데이터가 다른 VDC 에 복제됩니다. 오브젝트가 생성된 주 사이트 그리고 복제본을 저장하는 사이트 각자가 로컬 SP 에 할당된 EC 스키마를 사용하여 데이터를 로컬에서 보호합니다. 즉, 원래 데이터만 WAN 을 거쳐 복제되고 연관된 EC 코딩 세그먼트는 복제되지 않습니다.

클라이언트는 사용 가능한 RG 멤버 VDC 를 통해 액티브 RG 에 저장된 데이터에 액세스할 수 있습니다. 아래 그림 23 은 VDC1, VDC2 및 VDC3 을 사용하여 구축된 페더레이션 ECS 의 예를 보여줍니다. 두 개의 RG 가 보이며 RG1 에는 VDC1 멤버 하나만 있고 RG2 에는 세 개의 VDC 모두가 멤버입니다. 그리고 B1, B2, B3 의 세 가지 버킷이 있습니다.

이 예에서 클라이언트의 특성은 다음과 같습니다.

- VDC1 에 액세스하는 클라이언트는 모든 버킷에 액세스할 수 있습니다.
- VDC2 및 VDC3 에 액세스하는 클라이언트는 버킷 B2 및 B3 에만 액세스할 수 있습니다.

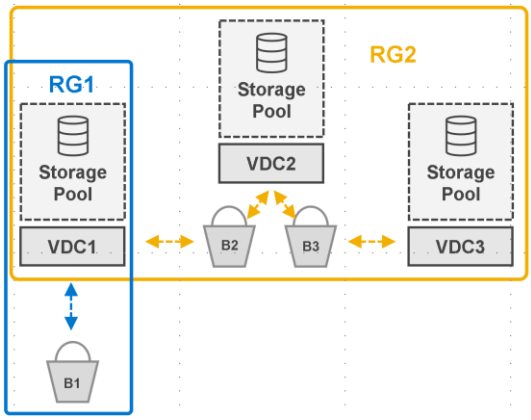


그림 23 단일 및 멀티 사이트 복제 그룹이 있는 사이트별 버킷 레벨 액세스

8.2.3 패시브 복제 그룹

패시브 RG 에는 세 개의 멤버 VDC 가 있습니다. 두 개의 VDC 는 액티브로 지정되어 있으며 클라이언트가 액세스할 수 있습니다. 세 번째 VDC 는 패시브로 지정되어 있으며 복제 타겟으로만 사용됩니다. 패시브 사이트는 복구 목적으로만 사용되며 직접 클라이언트 액세스를 허용하지 않습니다. 원거리 패시브 복제의 이점은 다음과 같습니다.

- XOR 연산의 가능성을 높여 스토리지 보호 오버헤드를 줄임
- 복제 전용 스토리지에 사용되는 위치를 관리자 레벨에서 제어

그림 24 는 VDC1 및 VDC2 가 주(소스) 사이트이고 둘 다 복제 타겟인 VDC3 에 데이터(청크)를 복제하는 원거리 패시브 구성의 예를 보여줍니다.

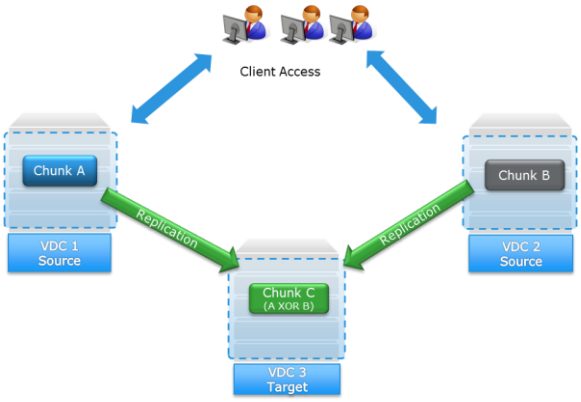


그림 24 원거리 패시브 복제 그룹에 대한 클라이언트 액세스 및 복제 경로

RG 멤버 사이트 전반에서 네임스페이스, 버킷 및 오브젝트 소유권을 사용하여 강력한 정합성이 보장되는 데이터에 대한 멀티 사이트 액세스가 가능합니다. 필요한 논리 구조를 소유하지 않은 VDC 에서 API 액세스가 발생할 때는 WAN 을 거쳐 사이트 간에 인덱스를 쿼리해야 합니다. WAN 조회는 신뢰할 수 있는 버전의 데이터를 확인하는 데 사용됩니다. 따라서 사이트 1 에서 생성된 오브젝트를 사이트 2 에서 읽는 경우에는 사이트 2 에 복제된 오브젝트의 데이터가 최신 버전의 데이터인지 확인하기 위해 오브젝트의 소유자 VDC 인 사이트 1 을 쿼리하는 WAN 조회가 필요합니다. 사이트 2 에 최신 버전이 없는 경우 사이트 1 에서 필요한 데이터를 가져옵니다. 그렇지 않으면 이전에 복제된 데이터를 사용합니다. 이는 아래 그림 25 에 잘 설명되어 있습니다.

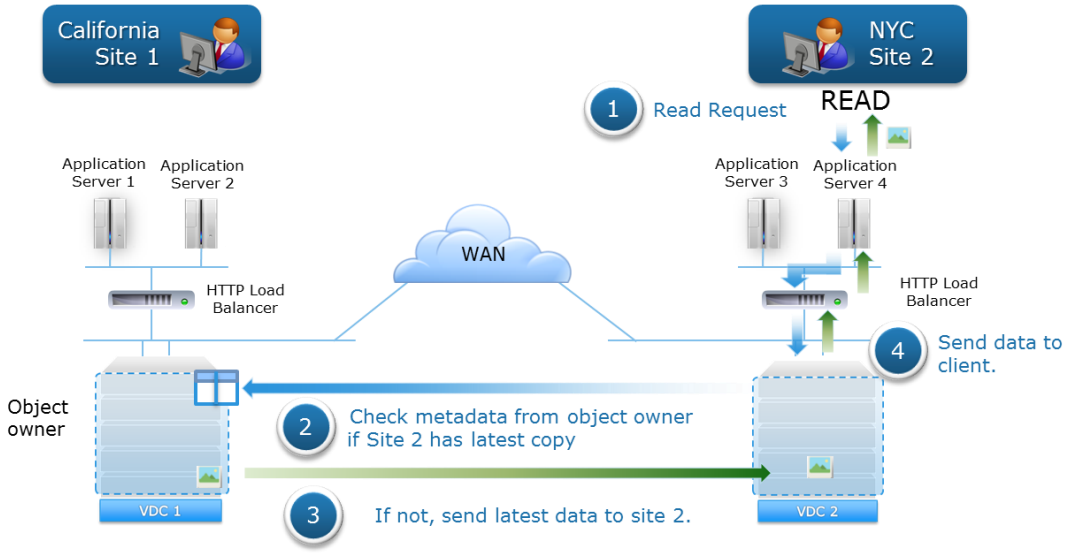


그림 25 비소유자 VDC 에 대한 읽기 요청이 오브젝트 소유자 VDC 에 대한 WAN 조회를 트리거함

그림 26 에는 두 사이트가 동일한 오브젝트를 업데이트하는 원거리 복제 환경에서 수행되는 쓰기의 데이터 흐름이 나와 있습니다. 이 예에서는 먼저 사이트 1 이 생성되고 오브젝트를 소유합니다. 오브젝트는 삭제 코딩되었으며 관련 저널 트랜잭션이 사이트 1 의 디스크에 기록됩니다. 사이트 2 에 수신된 오브젝트를 업데이트하는 데이터 흐름은 다음과 같습니다.

1. 사이트 2 에서 먼저 데이터를 로컬로 씁니다.
2. 사이트 2 에서 오브젝트 소유자인 사이트 1 에 메타데이터(저널 쓰기)를 동기식으로 업데이트하고 사이트 1 의 메타데이터 업데이트 확인을 기다립니다.
3. 사이트 1 에서 메타데이터 쓰기 확인을 사이트 2 에 보냅니다.
4. 사이트 2 에서 쓰기 확인을 클라이언트에 보냅니다.

참고: 사이트 2 는 평소와 같이 오브젝트 소유자 사이트인 사이트 1 에 데이터를 비동기식으로 복제합니다. 사이트 1 이 사이트 2 에서 데이터를 복제하기 전에 데이터를 제공해야 하는 경우, 사이트 1 은 사이트 2 에서 직접 데이터를 검색합니다.

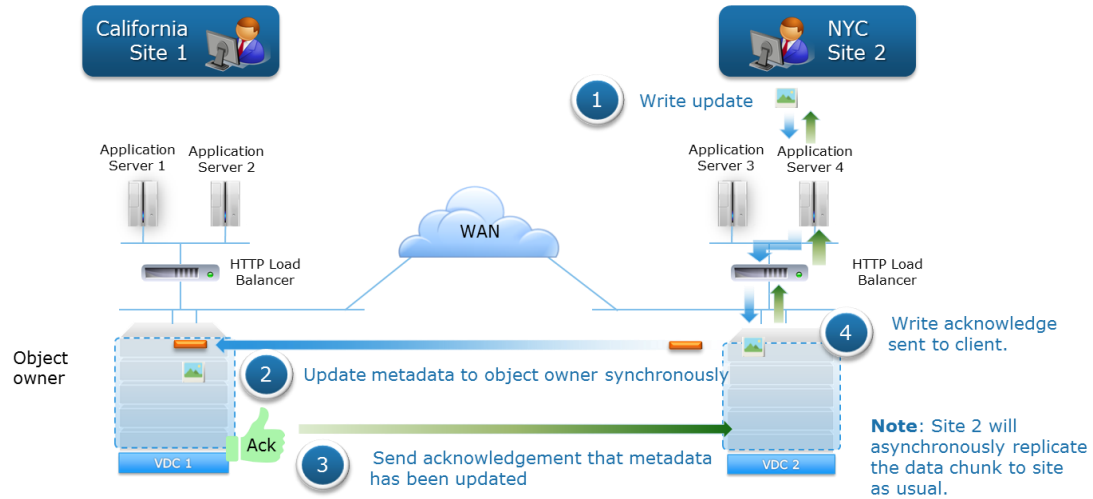


그림 26 원거리 복제 환경에서 동일한 오브젝트 업데이트의 데이터 흐름

원거리 복제 환경의 읽기 및 쓰기 시나리오에서 메타데이터를 읽고 업데이트하고 오브젝트 소유자 사이트에서 데이터를 검색할 경우 레이턴시가 발생합니다.

참고: ECS 3.4 버전부터 VDC 또는 VDC 에 연결된 다른 RG(Replication Group)에 영향을 미치지 않고 멀티 VDC 페더레이션에서 VDC 를 제거할 수 있습니다. RG 에서 VDC 를 제거해도 더 이상 PSO 가 시작되지 않습니다(영구 사이트 운영 중단). RG 에서 VDC 를 제거하면 복구가 시작됩니다.

복제 그룹에 대한 자세한 내용은 최신 *ECS 관리자 가이드*를 참조하십시오.

8.2.4 원격 데이터 원거리 캐싱

ECS 는 WAN 을 통해 읽은 오브젝트를 로컬에 캐싱함으로써 원격 사이트에 저장된 데이터에 액세스하는 응답 시간을 최적화합니다. 이는 원격 사이트 또는 비소유자 사이트에서 종종 데이터를 가져오는 멀티 사이트 액세스 패턴에 유용할 수 있습니다. 오브젝트가 VDC1 에 기록되어 있고 오브젝트의 복제본이 VDC2 에 저장되어 있는 VDC1, VDC2, VDC3 의 세 사이트를 포함하는 원거리 복제 환경을 생각해 보십시오. 이 시나리오에서는 VDC1 에서 생성되고 VDC2 에 복제된 오브젝트에 대해 VDC3 에서 받은 읽기 요청을 처리하려면 VDC1 또는 VDC2 에서 VDC3 로 오브젝트 데이터를 보내야 합니다. 자주 액세스하는 원격 데이터를 원거리 캐싱하면 응답 시간을 줄일 수 있습니다. 캐싱에는 Least Recently Used 알고리듬이 사용됩니다. 디스크, 노드, 랙 등의 하드웨어 인프라스트럭처가 원거리 복제 SP 에 추가될 경우 원거리 캐시 크기가 조정됩니다.

8.2.5 사이트 운영 중단 중 동작

TSO(Temporary Site Outage)는 일반적으로 WAN 연결 장애 또는 자연 재해 때 발생하는 것과 같은 전체 사이트 장애를 의미합니다. ECS 는 하트비트 메커니즘을 사용하여 일시적인 사이트 장애를 감지하고 처리합니다. TSO 중에 네임스페이스, 버킷, 오브젝트 레벨의 클라이언트 액세스 및 API 작업 가용성은 네임스페이스 및 버킷 레벨에서 설정된 다음과 같은 ADO 옵션에 따라 관리됩니다.

- **Off(기본값)** - 일시적인 운영 중단 중에 강력한 정합성이 보장됩니다.
- **On** - 일시적인 사이트 운영 중단 중에 최종 정합성이 보장되는 액세스가 허용됩니다.

TSO 중의 데이터 정합성은 버킷 레벨에서 구현됩니다. 구성은 네임스페이스 레벨에서 설정되며, 이 작업은 새 버킷 생성이 진행되는 도중에 ADO 에 대한 기본 ADO 설정을 알맞게 구성하며 새 버킷을 생성할 때 재정의할 수 있습니다. 즉, TSO 를 일부 버킷에 대해 구성할 수 있지만 다른 버킷에는 구성할 수 없습니다.

8.2.5.1 ADO(Access During Outage) 비활성화

기본적으로 ADO 는 활성화되지 않으며 강력한 정합성이 유지됩니다. 신뢰할 수 있는 네임스페이스, 버킷 또는 오브젝트 데이터가 필요하지만 일시적으로 사용할 수 없는 경우 모든 클라이언트 API 요청은 실패합니다. 읽기, 생성, 업데이트 같은 오브젝트 작업뿐만 아니라 온라인 사이트에서 소유하지 않은 버킷의 나열도 실패합니다. 또한 버킷, 사용자, 네임스페이스 등의 생성 및 편집 작업도 실패합니다.

앞서 설명한 것처럼 버킷, 네임스페이스 및 오브젝트의 초기 사이트 소유자는 리소스가 처음 생성된 사이트입니다. TSO 중에 리소스의 사이트 소유자에 액세스할 수 없는 경우 특정 작업은 실패할 수 있습니다. 임시 사이트 가동 중단 중에 허용되거나 허용되지 않는 대표적인 작업은 다음과 같습니다.

- 버킷, 네임스페이스, 오브젝트 사용자, 인증 공급자, RG 및 NFS 사용자, 그룹 매핑의 생성, 삭제, 업데이트는 모든 사이트에서 허용되지 않습니다.
- 네임스페이스 소유자 사이트를 사용할 수 없는 경우 네임스페이스 내의 버킷을 나열하는 작업은 허용됩니다.

HDFS/NFS 를 사용하면 액세스할 수 없는 사이트에서 소유하는 버킷을 읽기 전용으로 볼 수 있습니다.

8.2.5.2 ADO 활성화

ADO 가 활성화된 버킷에서 TSO 중에 스토리지 서비스는 최종 정합성이 보장되는 응답을 제공합니다. 이 시나리오에서는 보조(비소유자) 사이트에서 읽기 및 쓰기(선택 사항)를 사용할 수 있습니다. 또한 TSO 중에 보조 사이트에 데이터를 쓰면 보조 사이트가 해당 오브젝트의 소유권을 갖게 됩니다. 이를 통해 각 VDC 가 공유 네임스페이스에 있는 버킷의 오브젝트를 계속 읽고 쓸 수 있습니다. 마지막으로, 다른 애플리케이션이 소유자 VDC 에서 오브젝트를 업데이트하는 경우에도 오브젝트의 새 버전은 TSO 이후 조정 중에 신뢰할 수 있는 오브젝트 버전이 됩니다.

네트워크 가동 중단 중에도 많은 오브젝트 작업이 계속되지만 새로운 버킷, 네임스페이스 또는 사용자의 생성 같은 특정 작업은 허용되지 않습니다. 두 VDC 간에 네트워크 연결이 복구되면 하트비트 메커니즘이 자동으로 연결을 감지하여 서비스를 복구하고 두 VDC 의 오브젝트를 조정합니다. VDC A 와 VDC B 모두에서 동일한 오브젝트가 업데이트되는 경우 비소유자 VDC 의 복제본이 신뢰할 수 있는 복제본입니다. 따라서 동기화 중에 VDC B 가 소유한 오브젝트가 VDC A 와 VDC B 모두에서 업데이트되는 경우, VDC A 의 복제본은 신뢰할 수 있는 복제본이 되어 유지되는 반면 다른 복제본은 참조되지 않고 공간 재확보 대상이 됩니다.

3 개 이상의 VDC 가 RG 에 포함된 경우 한 VDC 와 다른 두 VDC 간에 네트워크 연결이 중단되면 쓰기/업데이트/소유권 작업은 두 VDC 가 있는 경우와 마찬가지로 계속되지만, 아래에 설명된 것처럼 읽기 요청에 응답하기 위한 프로세스가 더 복잡해집니다.

애플리케이션이 연결할 수 없는 VDC 가 소유한 오브젝트를 요청하는 경우 ECS 는 오브젝트의 보조 복제본으로 해당 VDC 에 요청을 전송합니다. 하지만 보조 사이트 복제본은 서로 다른 두 데이터 세트 간 XOR 연산을 통해 새 데이터 세트가 생성되는 데이터 축소 작업을 거쳤을 수 있습니다. 따라서 보조 사이트 VDC 는 먼저 원래 XOR 연산에 포함된 오브젝트의 청크를 검색하고 복구 복제본을 사용하여 해당 청크를 XOR 처리해야 합니다. 이 연산은 장애가 발생한 VDC 에 원래 저장된 청크 콘텐츠를 반환합니다. 그런 다음 복구된 오브젝트의 청크가 재구성되어 반환될 수 있습니다. 청크가 재구성되는 경우 VDC 가 후속 요청에 더 신속하게 응답할 수 있도록 청크가 캐싱됩니다. 재구성에는 시간이 많이 소요됩니다. RG 에 VDC 가 많을수록 다른 VDC 에서 검색해야 하는 청크가 많아져서 오브젝트 재구성에 더 오랜 시간이 걸립니다.

재해가 발생하면 전체 VDC 복구가 불가능할 수 있습니다. ECS 는 복구 불가능한 VDC 를 일시적 사이트 장애로 취급합니다. 장애가 영구적인 경우 System Administrator 가 페더레이션에서 VDC 를 영구적으로 페일오버하여 페일오버 처리를 시작해야 합니다. 그러면 장애가 발생한 VDC 에 저장된 오브젝트의 재동기화와 재보호가 시작됩니다. 복구 작업은 백그라운드 프로세스로 실행됩니다. ECS 포털에서 복구 진행 상황을 검토할 수 있습니다.

RO(Read-Only) ADO 를 위해 추가 버킷 옵션을 사용할 수 있으며, 이 RO ADO 는 오브젝트 소유권이 변경되지 않도록 하고 일시적인 사이트 가동 중단 중에 장애 사이트와 온라인 사이트 모두에서 오브젝트가 업데이트됨으로써 발생할 수 있는 충돌 가능성을 제거합니다. RO ADO 의 단점은 일시적인 사이트 가동 중단 중에 새 오브젝트를 생성할 수 없으며 모든 사이트가 다시 온라인 상태가 될 때까지 버킷에서 기존 오브젝트를 업데이트할 수 없다는 것입니다. RO ADO 옵션은 버킷 생성 중에만 사용할 수 있으며 나중에 수정할 수 없습니다. 이 옵션은 기본적으로 비활성화되어 있습니다.

표 8 멀티 사이트 내결함성

장애 모델	허용 한도
원거리 복제 환경	최대 하나의 사이트 장애

8.3 내결함성

ECS 는 여러 개의 장애 도메인을 사용하여 다양한 장비 장애 상황을 견딜 수 있도록 설계되었습니다. 장애 조건의 범위는 다음과 같이 다양합니다.

- 단일 노드에서의 단일 하드 드라이브 장애
- 단일 노드에서의 다중 하드 드라이브 장애
- 단일 하드 드라이브 장애가 있는 다중 노드
- 다중 하드 드라이브 장애가 있는 다중 노드
- 단일 노드 장애
- 다중 노드 장애
- 복제된 한 VDC 에서의 통신 손실
- 복제된 전체 VDC 의 손실

단일 사이트, 이중 사이트 또는 원거리 복제 구성에서 장애의 영향은 영향을 받는 구성 요소의 수량과 유형에 따라 달라집니다. 하지만 ECS 는 각 수준에서 구성 요소 장애의 영향을 방어하는 메커니즘을 제공합니다.

이러한 메커니즘 중 다수는 이미 이 문서에서 설명했지만 솔루션에 어떻게 적용되는지를 보여주기 위해 이 페이지와 그림 27 에서 다시 살펴봅니다. 다음과 같은 섹션이 있습니다.

- Disk failure
  - 동일한 청크의 EC 세그먼트 또는 복제본은 동일한 디스크에 저장되지 않음
  - 쓰기 및 읽기 작업에 대한 체크섬 계산
  - 체크섬을 다시 확인하는 백그라운드 정합성 검사기
- Node failure
  - 청크의 세그먼트 또는 복제본을 VDC 의 노드 간에 균등하게 분산
  - ECS Fabric 이 서비스를 계속 실행하고 디스크 및 네트워크 같은 리소스를 관리함
  - 노드 간의 파티션 소유권 페일오버에 의해 보호되는 레코드와 테이블을 분할함
- VDC 내 랙 장애
  - 청크의 세그먼트 또는 복제본을 VDC 의 랙 간에 균등하게 분산
  - 하나의 패브릭 레지스트리 인스턴스가 각 랙에서 실행되며 노드가 실패할 경우 동일한 랙의 다른 노드에서 다시 시작될 수 있음

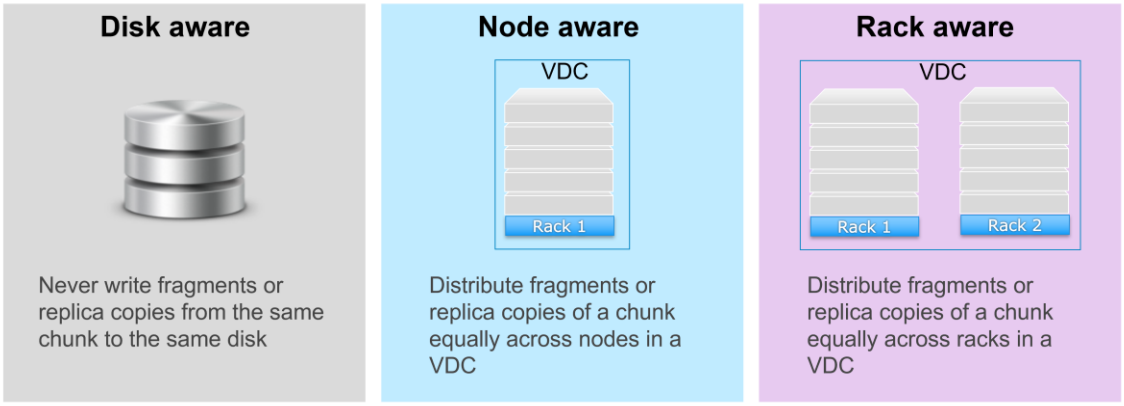


그림 27 디스크, 노드, 랙 레벨의 보호 메커니즘

다음 차트에서는 각 EC 스키마가 기본 랙 구성당 보호하는 구성 요소 장애의 유형과 개수를 정의합니다. 표 9 는 각 EC 스키마에 필요한 노드 수 측면에서 보호 장애 도메인이 전체 데이터 및 서비스 가용성에 미치는 영향의 중요성을 요약해 보여줍니다.

표 9 장애 도메인 전반에서의 삭제 코드 보호

EC 스키마	VDC 의 노드 수	노드당 청크 조각 수	보호되는 EC 데이터
12+4 기본값	5 이하	4	• 최대 4 개 디스크 손실 • 1 개 노드 손실
	6 또는 7	3	• 최대 4 개 디스크 손실 • 두 번째 노드에서 1 개 노드 및 1 개 디스크 손실
	8 이상	2	• 최대 4 개 디스크 손실 • 2 개 노드 손실 • 2 개 노드 및 2 개 디스크 손실
	16 이상	1	• 4 개 노드 손실 • 1 개의 추가 노드에서 3 개 노드 및 디스크 손실 • 최대 2 개의 서로 다른 노드에서 2 개 노드 및 디스크 손실 • 최대 3 개의 서로 다른 노드에서 1 개 노드 및 디스크 손실 • 4 개의 서로 다른 노드에서 4 개 디스크 손실
10+2 콜드 스토리지	11 이하	2	• 최대 2 개 디스크 손실 • 1 개 노드 손실
	12 이상	1	• 2 개의 서로 다른 노드에서 디스크 손실 • 2 개 노드 손실

8.4 디스크 교체 자동화

ECS 3.5 버전부터 고객은 직관적인 ECS 포털(웹 UI) 워크플로를 사용하여 장애가 발생한 디스크를 Dell EMC 서비스로 교체할 수 있습니다. 이 기능은 다음을 제공합니다.

- 드라이브 장애 문제 DIY(Do-It-Yourself) 해결
- 문제 해결 시간 단축
- 운영 유연성 및 TCO 절감

ECS 포털의 유지 보수 페이지에는 각 노드의 모든 디스크에 대한 관리자 가시성이 제공됩니다. 드라이브에 장애가 발생하면 시스템이 자동으로 복구를 시작합니다. 드라이브 내 모든 유형의 리소스가 복구되며 노드에서 드라이브를 제거할 준비가 되면 그림 28 에 나온 대로 ECS 포털에 교체 버튼이 표시됩니다.

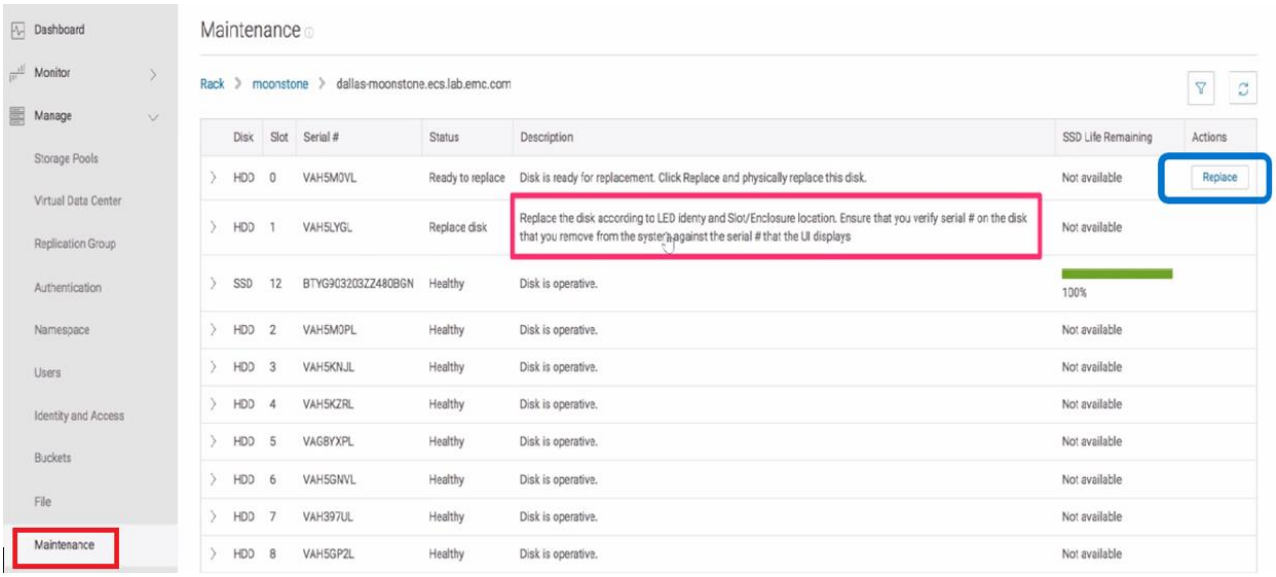


그림 28 디스크 교체 자동화

참고: 한 번에 하나의 드라이브만 교체할 수 있습니다. 이는 드라이브를 잘못 교체하는 상황을 방지하기 위한 것입니다.

8.5 Tech Refresh

Tech Refresh 는 ECS 3.5 버전부터 제공되는 Dell EMC Professional Services 주도형 참여 서비스로, 내장형 소프트웨어 기능을 사용하여 운영 중단 없이 ECS 클러스터에서 오래된 하드웨어 노드를 제거합니다. 임계치를 정밀하게 조정하는 이 작업은 효율적일 뿐 아니라 리소스도 적게 사용합니다. 또한 이 기능은 이전의 ECS 하드웨어 폐기와 관련된 오버헤드를 절감합니다.

Tech Refresh 에는 다음 세 가지 요소가 포함됩니다.

- **노드 확장**: 기존 클러스터에 Gen3 노드 추가
- **리소스 마이그레이션**: 모든 리소스를 기존 노드에서 Gen3 노드로 이동
- **노드 제거**: 이전 노드를 정리하고 클러스터에서 제거

Tech Refresh 유지 보수가 진행될 때 Professional Services 가 참여해야 합니다. Tech Refresh 에 대한 자세한 내용은 최신 *ECS Tech Refresh 가이드*를 참조하십시오.

9 스토리지 보호 오버헤드

RG 의 각 VDC 멤버가 로컬 레벨에서 자체 EC 데이터 보호를 담당합니다. 즉, 데이터는 복제되지만 관련 코딩 세그먼트는 복제되지 않습니다. EC 는 전체 복제 드라이브 미러링과 같은 다른 형태의 보호보다 스토리지 효율성이 높지만 로컬 수준에서 필연적인 스토리지 비용 부담이 발생합니다. 하지만, 보조 복제본을 오프사이트에 복제해야 하고 단일 사이트를 사용할 수 없을 때에도 모든 사이트가 데이터에 액세스할 수 있도록 해야 할 경우에는 기존 사이트 간 데이터 복제 보호 방법을 사용할 때보다 스토리지 비용이 더 늘어납니다. 특히 고유한 데이터를 세 개 이상의 사이트에 분산해야 하는 경우 더욱 그렇습니다.

ECS 는 3 개 이상의 사이트가 페더레이션될 때 스토리지 보호 오버헤드 효율성을 높일 수 있는 메커니즘을 제공합니다. ECS 는 2-VDC 복제 환경에서 기본 또는 소유자 VDC 에서 원격 사이트로 청크를 복제하여 높은 가용성과 복원력을 제공합니다. 2 사이트 페더레이션 ECS 구축에서 데이터 전체 복제의 100% 보호 오버헤드를 피할 수 있는 방법은 없습니다.

이제 멀티 사이트 환경에 세 개의 VDC 가 있다고 가정해 보십시오. VDC1, VDC2, VDC3 가 있고 각 VDC 의 고유 데이터가 다른 VDC 에 서로 복제됩니다. VDC2 및 VDC3 는 보호를 위해 해당 데이터의 복제본을 VDC1 으로 보낼 수 있습니다. 따라서 VDC1 은 자체 원본 데이터에 더하여 VDC2 및 VDC3 의 복제 데이터까지 포함합니다. 즉, VDC1 은 자체 사이트에 기록된 데이터의 3 배에 해당하는 양을 저장하게 됩니다.

이러한 상황에서 ECS 는 VDC1 에 로컬로 저장된 VDC2 및 VDC3 데이터의 XOR 연산을 수행할 수 있습니다. 이 수학 연산은 동일한 수량의 고유 데이터인 청크를 비교합니다. 그런 다음, 원래의 두 세트 중 어느 한쪽을 복원할 수 있도록 두 원래 데이터 청크의 충분한 특성을 포함하는 결과를 새로운 청크에 렌더링합니다. 따라서, 이전에는 세 개의 고유한 데이터 청크 집합이 VDC1 에 저장되어 가용 용량의 3 배를 차지했지만 이제는 원래의 로컬 데이터 세트 그리고 XOR 연산을 통해 줄인 보호 복제본, 이렇게 두 종류의 데이터만 남습니다.

같은 시나리오에서 VDC3 를 사용할 수 없게 될 경우 ECS 는 VDC2 에서 회수된 청크 복제본과 VDC1 에서 로컬로 저장된 VDC3 의 (C1 ⊕ C2) 데이터를 사용하여 VDC3 데이터 청크를 재구성할 수 있습니다. 이 원리는 RG 에 참여하는 세 개 사이트 모두에 적용됩니다. 단, 세 개의 VDC 는 각각 고유한 데이터 세트를 가지고 있어야 합니다. 그림 29 는 두 사이트가 세 번째 사이트에 복제되는 XOR 연산을 보여줍니다.

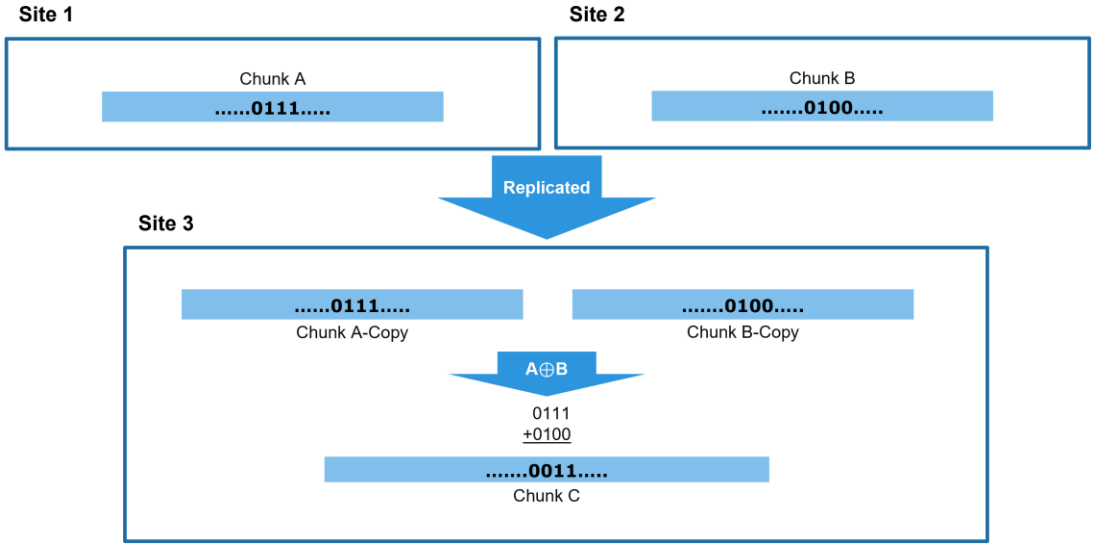


그림 29 XOR 데이터 보호 효율성

전체 사이트 장애가 발생하더라도 최적의 읽기 액세스 속도를 제공할 것을 요구하는 비즈니스 서비스 수준 계약의 경우, 모든 사이트에 대한 복제 설정은 ECS 가 복제된 데이터의 전체 복제본을 모든 사이트에 저장하도록 만듭니다. 당연히 이는 RG 에 참여하는 VDC 수에 비례하여 스토리지 비용을 증가시킵니다. 따라서 3 사이트 구성은 3 배의 스토리지 보호 오버헤드로 되돌아갑니다. RG 생성 중에 Replicate to All Sites 설정을 사용할 수 있으며 전환할 수 없습니다.

페더레이션된 사이트 수가 증가할수록 XOR 최적화는 복제로 인한 스토리지 보호 오버헤드를 더 효율적으로 줄입니다. 표 10 은 일반 EC 12+4 및 콜드 아카이브 EC 10+2 의 사이트 수에 따른 스토리지 보호 오버헤드 정보를 제공함으로써 더 많은 사이트가 연결될수록 ECS 의 스토리지 효율성이 더 높아지는 것을 보여줍니다.

참고: 3 개 및 최대 8 개 사이트에서 복제된 데이터 오버헤드를 낮추려면 각 사이트에서 고유 데이터를 비교적 균등하게 기록해야 합니다. 사이트 간에 동일한 양의 데이터를 기록하면 각 사이트의 복제본 청크 수가 비슷해집니다. 각 사이트마다 복제본 청크 수가 비슷하면 각 사이트에서 비슷한 횟수의 XOR 연산이 수행됩니다. XOR 연산을 사용하여 저장된 최대 복제본 청크 수를 줄임으로써 멀티 사이트 스토리지 효율성을 극대화할 수 있습니다.

표 10 스토리지 보호 오버헤드

RG 의 사이트 수	12+4 EC	10+2 EC
1	1.33	1.2
2	2.67	2.4
3	2.00	1,8
4	1.77	1.6
5	1.67	1.5
6	1.60	1.44
7	1.55	1.40
8(RG 의 최대 사이트 수)	1.52	1.37

10 결론

많은 조직은 특히 퍼블릭 클라우드 공간에서 데이터 양과 스토리지 비용이 점점 늘어나는 어려움을 겪고 있습니다. ECS 의 스케일 아웃 및 지리적 분산 아키텍처는 퍼블릭 클라우드 스토리지보다 훨씬 적은 *총 소유 비용*으로 엑사바이트 단위 데이터까지 확장하는 온프레미스 클라우드 플랫폼을 제공합니다. ECS 는 다기능성, 매우 뛰어난 확장성, 강력한 기능, 상용 하드웨어를 갖춘 훌륭한 솔루션입니다.

A 기술 지원 및 리소스

Dell.com/support 는 검증된 서비스와 지원으로 고객의 요구 사항에 부응하기 위해 최선을 다하고 있습니다.

스토리지 기술 문서 및 비디오에서는 고객이 Dell EMC 스토리지 플랫폼을 성공적으로 활용하는 데 필요한 전문 지식을 제공합니다.