



# **StorageGRID를 사용합니다**

StorageGRID software

NetApp  
February 12, 2026

# 목차

StorageGRID 테넌트 및 클라이언트 사용	1
테넌트 계정을 사용합니다	1
테넌트 계정을 사용합니다	1
로그인 및 로그아웃 방법	2
테넌트 관리자 대시보드 이해	4
테넌트 관리 API	7
그리드 페더레이션 연결을 사용합니다	12
그룹 및 사용자를 관리합니다	25
S3 액세스 키를 관리합니다	42
S3 버킷을 관리합니다	48
S3 플랫폼 서비스 관리	75
S3 REST API 사용	106
S3 REST API 지원 버전 및 업데이트	106
빠른 참조: 지원되는 S3 API 요청	109
S3 REST API 구성을 테스트합니다	128
StorageGRID에서 S3 REST API를 구현하는 방법	129
Amazon S3 REST API 지원	144
StorageGRID 사용자 정의 작업	192
액세스 정책 관리	211
감사 로그에서 S3 작업을 추적했습니다	237
Swift REST API 사용(지원 종료)	238
Swift REST API를 사용합니다	238

# StorageGRID 테넌트 및 클라이언트 사용

## 테넌트 계정을 사용합니다

### 테넌트 계정을 사용합니다

테넌트 계정을 사용하면 S3(Simple Storage Service) REST API를 사용하여 StorageGRID 시스템에 오브젝트를 저장하고 검색할 수 있습니다.

#### 테넌트 계정이란 무엇입니까?

각 테넌트 계정에는 고유한 페더레이션 또는 로컬 그룹, 사용자, S3 버킷 및 오브젝트가 있습니다.

테넌트 계정은 저장된 객체를 다른 엔터티로 분리하는 데 사용할 수 있습니다. 예를 들어, 다음과 같은 사용 사례에서 여러 테넌트 계정을 사용할 수 있습니다.

- \* 기업 활용 사례: \* 기업 내에서 StorageGRID 시스템을 사용하는 경우 그리드의 객체 스토리지를 조직의 여러 부서에서 분리할 수 있습니다. 예를 들어 마케팅 부서, 고객 지원 부서, 인사 부서 등의 테넌트 계정이 있을 수 있습니다.



S3 클라이언트 프로토콜을 사용하면 S3 버킷과 버킷 정책을 사용하여 기업 내 부서 간에 객체를 분리할 수도 있습니다. 별도의 테넌트 계정을 만들 필요가 없습니다. 구현에 대한 지침을 참조하세요. ["S3 버킷 및 버킷 정책"](#) 자세한 내용은.

- \* 서비스 공급자 사용 사례: \* 서비스 공급자가 StorageGRID 시스템을 사용 중인 경우, 스토리지를 임대하는 다른 엔터티로 그리드의 개체 스토리지를 분리할 수 있습니다. 예를 들어 회사 A, 회사 B, 회사 C 등에 대한 테넌트 계정이 있을 수 있습니다.

### 테넌트 계정을 생성하는 방법

테넌트 계정은 예 의해 ["그리드 관리자를 사용하는 StorageGRID 그리드 관리자"](#) 생성됩니다. 테넌트 계정을 생성할 때 그리드 관리자는 다음을 지정합니다.

- 테넌트 이름, 클라이언트 유형(S3) 및 선택적 스토리지 할당량을 포함한 기본 정보입니다.
- 테넌트 계정에서 S3 플랫폼 서비스를 사용할 수 있는지 여부, 해당 ID 소스를 구성할 수 있는지 여부, S3 Select를 사용할 것인지, 그리드 페더레이션 연결을 사용할 수 있는지 여부 등의 테넌트 계정에 대한 사용 권한
- StorageGRID 시스템에서 로컬 그룹 및 사용자, ID 페더레이션 또는 SSO(Single Sign-On)를 사용하는지 여부에 따라 테넌트의 초기 루트 액세스입니다.

또한, S3 테넌트 계정이 규정 요구 사항을 준수해야 하는 경우 그리드 관리자는 StorageGRID 시스템에 대해 S3 오브젝트 잠금 설정을 활성화할 수 있습니다. S3 오브젝트 잠금이 활성화된 경우 모든 S3 테넌트 계정에서 호환 버킷을 생성하고 관리할 수 있습니다.

#### S3 테넌트를 구성합니다

그런 다음 ["S3 테넌트 계정이 생성됩니다"](#) 테넌트 관리자에 액세스하여 다음과 같은 작업을 수행할 수 있습니다.

- ID 페더레이션 설정(ID 소스가 그리드와 공유되지 않는 경우)

- 그룹 및 사용자를 관리합니다
- 계정 클론 및 교차 그리드 복제에 그리드 페더레이션을 사용합니다
- S3 액세스 키를 관리합니다
- S3 버킷을 생성하고 관리합니다
- S3 플랫폼 서비스 사용
- S3 Select를 사용합니다
- 스토리지 사용량을 모니터링합니다



테넌트 관리자로 S3 버킷을 생성하고 관리할 수 있지만, 또는 ["S3 콘솔"](#)을 사용하여 ["S3 클라이언트"](#) "오브젝트를 수집 및 관리해야 합니다.

## 로그인 및 로그아웃 방법

테넌트 관리자에 로그인합니다

의 주소 표시줄에 테넌트의 URL을 입력하여 테넌트 관리자에 ["지원되는 웹 브라우저"](#) ["액세스합니다."](#)

시작하기 전에

- 로그인 자격 증명이 있습니다.
- 그리드 관리자가 제공한 테넌트 관리자 액세스 URL이 있습니다. URL은 다음 예 중 하나로 표시됩니다.

`https://FQDN_or_Admin_Node_IP/`

`https://FQDN_or_Admin_Node_IP:port/`

`https://FQDN_or_Admin_Node_IP/?accountId=20-digit-account-id`

`https://FQDN_or_Admin_Node_IP:port/?accountId=20-digit-account-id`

URL에는 항상 FQDN(정규화된 도메인 이름), 관리 노드의 IP 주소 또는 관리 노드의 HA 그룹의 가상 IP 주소가 포함됩니다. 포트 번호, 20자리 테넌트 계정 ID 또는 둘 다 포함될 수도 있습니다.

- URL에 테넌트의 20자리 계정 ID가 포함되어 있지 않은 경우 이 계정 ID가 있는 것입니다.
- 을 사용하고 ["지원되는 웹 브라우저"](#)입니다.
- 쿠키는 웹 브라우저에서 활성화됩니다.
- 가 있는 사용자 그룹에 속해 ["특정 액세스 권한"](#)입니다.

단계

1. 를 실행합니다. ["지원되는 웹 브라우저"](#)
2. 브라우저의 주소 표시줄에 Tenant Manager에 액세스하기 위한 URL을 입력합니다.
3. 보안 경고 메시지가 나타나면 브라우저의 설치 마법사를 사용하여 인증서를 설치합니다.
4. 테넌트 관리자에 로그인합니다.

표시되는 로그인 화면은 입력한 URL과 StorageGRID에 대해 SSO(Single Sign-On)가 구성되었는지 여부에 따라 달라집니다.

#### **SSO를 사용하지 않습니다**

StorageGRID에서 SSO를 사용하지 않는 경우 다음 화면 중 하나가 나타납니다.

- Grid Manager 로그인 페이지 Tenant Sign-In \* 링크를 선택합니다.
- 테넌트 관리자 로그인 페이지. 계정 필드는 이미 작성되었을 수 있습니다.
  - i. 테넌트의 20자리 계정 ID가 표시되지 않으면 최근 계정 목록에 테넌트 계정 이름이 나타날 경우 해당 계정 이름을 선택하거나 계정 ID를 입력합니다.
  - ii. 사용자 이름과 암호를 입력합니다.
  - iii. 로그인 \* 을 선택합니다.

Tenant Manager 대시보드가 나타납니다.

- iv. 다른 사람으로부터 초기 암호를 받은 경우 \*username \* > \* 암호 변경 \* 을 선택하여 계정을 보호하십시오.

#### **SSO 사용**

StorageGRID에서 SSO를 사용하는 경우 다음 화면 중 하나가 나타납니다.

- 귀하의 조직의 SSO 페이지입니다.  
표준 SSO 자격 증명을 입력하고 \* 로그인 \* 을 선택합니다.
- Tenant Manager SSO 로그인 페이지.
  - i. 테넌트의 20자리 계정 ID가 표시되지 않으면 최근 계정 목록에 테넌트 계정 이름이 나타날 경우 해당 계정 이름을 선택하거나 계정 ID를 입력합니다.
  - ii. 로그인 \* 을 선택합니다.
  - iii. 조직의 SSO 로그인 페이지에서 표준 SSO 자격 증명을 사용하여 로그인합니다.

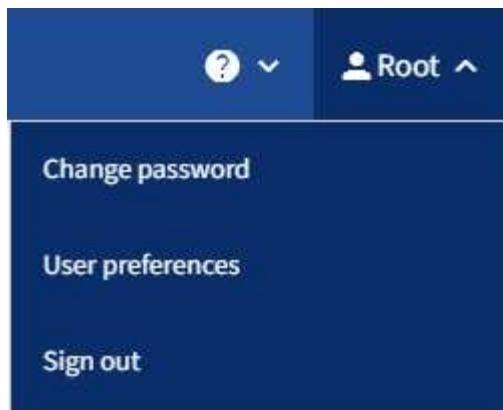
Tenant Manager 대시보드가 나타납니다.

#### 테넌트 관리자에서 로그아웃합니다

테넌트 관리자 작업을 마치면 로그아웃하여 권한이 없는 사용자가 StorageGRID 시스템에 액세스할 수 없도록 해야 합니다. 브라우저를 닫아도 브라우저 쿠키 설정에 따라 시스템에서 로그아웃되지 않을 수 있습니다.

#### 단계

1. 사용자 인터페이스의 오른쪽 위 모서리에서 사용자 이름 드롭다운을 찾습니다.



2. 사용자 이름을 선택한 다음 \* 로그아웃 \* 을 선택합니다.

◦ SSO가 사용되지 않는 경우:

관리자 노드에서 로그아웃되었습니다. Tenant Manager 로그인 페이지가 표시됩니다.



두 개 이상의 관리 노드에 로그인한 경우 각 노드에서 로그아웃해야 합니다.

◦ SSO가 활성화된 경우:

액세스 중인 모든 관리 노드에서 로그아웃되었습니다. StorageGRID 로그인 페이지가 표시됩니다. 방금 액세스한 테넌트 계정의 이름이 \* 최근 계정 \* 드롭다운에 기본값으로 나열되고 테넌트의 \* 계정 ID \* 가 표시됩니다.



SSO가 활성화되어 있고 Grid Manager에도 로그인한 경우, Grid Manager에서 로그아웃하여 SSO를 로그아웃해야 합니다.

## 테넌트 관리자 대시보드 이해

테넌트 관리자 대시보드는 테넌트 계정 구성과 테넌트의 S3 버킷에 있는 개체가 사용하는 공간의 양에 대한 개요를 제공합니다. 세입자에게 할당량이 있는 경우 대시보드에는 할당량 중 얼마나 사용되었고 얼마나 남았는지가 표시됩니다. 테넌트 계정과 관련된 오류가 있는 경우 해당 오류가 대시보드에 표시됩니다.



이 테넌트에 속한 모든 객체의 논리적 크기에는 완료되지 않은 멀티파트 업로드와 진행 중인 멀티파트 업로드가 포함됩니다. 이 크기에는 ILM 정책에 사용되는 추가적인 물리적 공간은 포함되지 않습니다. 사용된 공간 값은 추정치입니다. 이러한 추정치는 수집 타이밍, 네트워크 연결, 노드 상태의 영향을 받습니다.

객체가 업로드되면 대시보드는 다음 예와 같이 표시됩니다.

# Dashboard

**16** Buckets  
View buckets

**2** Platform services endpoints  
View endpoints

**0** Groups  
View groups

**1** User  
View users

## Storage usage ?

6.5 TB of 7.2 TB used

0.7 TB (10.1%) remaining



Bucket name	Space used	Number of objects
Bucket-15	969.2 GB	913,425
Bucket-04	937.2 GB	576,806
Bucket-13	815.2 GB	957,389
Bucket-06	812.5 GB	193,843
Bucket-10	473.9 GB	583,245
Bucket-03	403.2 GB	981,226
Bucket-07	362.5 GB	420,726
Bucket-05	294.4 GB	785,190
8 other buckets	1.4 TB	3,007,036

## Top buckets by capacity limit usage ?

Bucket name	Usage
Bucket-10	82%
Bucket-03	57%
Bucket-15	20%

## Tenant details ?

Name:	Tenant02
ID:	3341 1240 0546 8283 2208
<input checked="" type="checkbox"/> Platform services enabled	
<input checked="" type="checkbox"/> Can use own identity source	
<input checked="" type="checkbox"/> S3 Select enabled	

## 테넌트 계정 정보입니다

대시보드 상단에는 구성된 버킷 또는 컨테이너, 그룹 및 사용자의 수가 표시됩니다. 또한 구성된 플랫폼 서비스 끝점의 수도 표시됩니다. 세부 정보를 보려면 링크를 선택하십시오.

사용 중인 과 구성한 옵션에 따라 ["테넌트 관리 권한"](#) 대시보드의 나머지 부분에는 지침, 스토리지 사용, 개체 정보 및 테넌트 세부 정보의 다양한 조합이 표시됩니다.

## 스토리지 및 할당량 사용

Storage usage(저장소 사용) 패널에는 다음과 같은 정보가 포함되어 있습니다.

- 테넌트에 대한 객체 데이터의 양입니다.

이 값은 업로드된 총 오브젝트 데이터 양을 나타내며 해당 오브젝트 및 해당 메타데이터의 복사본을 저장하는 데 사용되는 공간을 나타내지 않습니다.

- 할당량이 설정된 경우 개체 데이터에 사용할 수 있는 총 공간과 남은 공간의 양과 백분율이 표시됩니다. 할당량은 섭취 가능한 오브젝트 데이터의 양을 제한합니다.



할당량 사용은 내부 추정치에 기초하며 경우에 따라 초과될 수 있습니다. 예를 들어, 테넌트가 객체를 업로드하기 시작할 때 StorageGRID는 할당량을 확인하고 테넌트가 할당량을 초과할 경우 새 베스트(ingest)를 거부합니다. 그러나 StorageGRID에서는 할당량이 초과되었는지 확인할 때 현재 업로드 크기를 고려하지 않습니다. 개체를 삭제하면 할당량 사용이 다시 계산될 때까지 테넌트가 일시적으로 새 개체를 업로드하지 못할 수 있습니다. 할당량 사용량 계산에는 10분 이상이 소요될 수 있습니다.

- 가장 큰 버킷 또는 컨테이너의 상대적 크기를 나타내는 막대 차트.

차트 세그먼트 위에 커서를 놓으면 해당 버킷이나 컨테이너에서 소비한 전체 공간을 볼 수 있습니다.



- 막대 도표에 대응하려면 총 오브젝트 데이터 양과 각 버킷 또는 컨테이너의 오브젝트 수를 포함하여 가장 큰 버킷 또는 컨테이너의 목록입니다.

Bucket name	Space used	Number of objects
Bucket-02	944.7 GB	7,575
Bucket-09	899.6 GB	589,677
Bucket-15	889.6 GB	623,542
Bucket-06	846.4 GB	648,619
Bucket-07	730.8 GB	808,655
Bucket-04	700.8 GB	420,493
Bucket-11	663.5 GB	993,729
Bucket-03	656.9 GB	379,329
9 other buckets	2.3 TB	5,171,588

테넌트에 9개 이상의 버킷이나 컨테이너가 있는 경우 다른 모든 버킷이나 컨테이너는 목록 하단의 단일 항목으로 결합됩니다.



테넌트 관리자에 표시된 스토리지 값의 단위를 변경하려면 테넌트 관리자 오른쪽 상단의 사용자 드롭다운을 선택한 다음 \* User preferences \* 를 선택합니다.

## 할당량 사용 알림을 표시합니다

Grid Manager에서 할당량 사용 경고가 활성화된 경우 할당량이 낮거나 초과되면 다음과 같이 테넌트 관리자에 이러한 경고가 표시됩니다.

- 테넌트 할당량의 90% 이상이 사용된 경우 \* Tenant quota usage high \* 경고가 트리거됩니다.

그리드 관리자에게 할당량을 늘리도록 요청하십시오.

- 할당량을 초과하면 새 개체를 업로드할 수 없다는 알림이 표시됩니다.

### 용량 제한 사용량

버킷에 대한 용량 제한을 설정한 경우 Tenant Manager 대시보드는 용량 제한 사용량별로 상위 버킷 목록을 표시합니다.

버킷에 제한이 설정되어 있지 않으면 해당 용량은 무제한입니다. 그러나 테넌트 계정에 총 저장소 할당량이 있고 이 할당량에 도달한 경우 버킷의 남은 용량 제한에 관계없이 더 많은 오브젝트를 수집할 수 없습니다.

### 끝점 오류

Grid Manager를 사용하여 플랫폼 서비스에 사용할 하나 이상의 엔드포인트를 구성한 경우, 지난 7일 내에 끝점 오류가 발생한 경우 Tenant Manager 대시보드에 경고가 표시됩니다.

 One or more endpoints have experienced an error and might not be functioning properly. Go to the [Endpoints](#) page to view the error details. The last error occurred 2 hours ago.

에 대한 세부 정보를 "[플랫폼 서비스 끝점 오류입니다](#)"보려면 \* 끝점 \* 을 선택하여 끝점 페이지를 표시합니다.

## 테넌트 관리 API

### 테넌트 관리 API 이해

테넌트 관리자 사용자 인터페이스 대신 테넌트 관리 REST API를 사용하여 시스템 관리 작업을 수행할 수 있습니다. 예를 들어, API를 사용하여 작업을 자동화하거나 사용자와 같은 여러 엔터티를 더 빠르게 생성할 수 있습니다.

테넌트 관리 API:

- Swagger 오픈 소스 API 플랫폼을 사용합니다. Swagger는 개발자와 개발자가 아닌 사용자가 API와 상호 작용할 수 있는 직관적인 사용자 인터페이스를 제공합니다. Swagger 사용자 인터페이스는 각 API 작동에 대한 전체 세부 정보와 문서를 제공합니다.
- 사용 "[무중단 업그레이드를 지원하는 버전 관리](#)".

테넌트 관리 API에 대한 Swagger 문서에 액세스하려면 다음을 수행합니다.

1. 테넌트 관리자에 로그인합니다.
2. 테넌트 관리자 상단에서 도움말 아이콘을 선택하고 \* API documentation \* 을 선택합니다.

### API 운영

테넌트 관리 API는 사용 가능한 API 작업을 다음 섹션으로 구성합니다.

- \* 계정 \*: 스토리지 사용 정보를 가져오는 것을 포함하여 현재 테넌트 계정의 작업.
- \* auth \*: 사용자 세션 인증을 수행하기 위한 작업.

Tenant Management API는 Bearer Token Authentication Scheme을 지원합니다. 테넌트 로그인의 경우 인증 요청의 JSON 본문에 사용자 이름, 암호 및 accountId를 입력합니다(즉, POST /api/v3/authorize). 사용자가 성공적으로 인증되면 보안 토큰이 반환됩니다. 이 토큰은 후속 API 요청 헤더("Authorization: Bearer token")에 제공되어야 합니다.

인증 보안 강화에 대한 자세한 내용은 ["사이트 간 요청 위조 방지"](#) 참조하십시오.



StorageGRID 시스템에서 SSO(Single Sign-On)가 활성화된 경우 인증을 위해 다른 단계를 수행해야 합니다. ["Grid Management API 사용 지침"](#) 참조하십시오.

- \* config \*: 제품 릴리스 및 테넌트 관리 API 버전과 관련된 작업. 제품 릴리스 버전과 해당 릴리스에서 지원하는 API의 주요 버전을 나열할 수 있습니다.
- 컨테이너: S3 버킷에 대한 작업입니다.
- \* 비활성화됨 - 기능 \*: 비활성화된 기능을 보기 위한 작업.
- \* 엔드포인트 \*: 엔드포인트를 관리하는 운영 엔드포인트는 S3 버킷이 StorageGRID CloudMirror 복제, 알림 또는 검색 통합에 외부 서비스를 사용할 수 있도록 합니다.
- \* 그리드 페더레이션 - 연결 \*: 그리드 페더레이션 연결 및 교차 그리드 복제에서의 작업.
- \* 그룹 \*: 로컬 테넌트 그룹을 관리하고 외부 ID 소스에서 통합 테넌트 그룹을 검색하는 작업입니다.
- \* identity-source \*: 외부 ID 소스를 구성하고 통합 그룹 및 사용자 정보를 수동으로 동기화하는 작업
- \* ILM \*: 정보 수명 주기 관리(ILM) 설정에 대한 작업.
- \* 지역 \*: StorageGRID 시스템에 대해 구성된 지역을 결정하는 작업.
- \* S3 \*: 테넌트 사용자를 위한 S3 액세스 키를 관리하는 운영
- \* S3 오브젝트 잠금 \*: 글로벌 S3 오브젝트 잠금 설정에서 운영, 규정 준수 지원에 사용됩니다.
- \* 사용자 \*: 테넌트 사용자를 보고 관리하는 작업.

#### 작업 세부 정보

각 API 작업을 확장하면 HTTP 동작, 끝점 URL, 필수 또는 선택적 매개 변수 목록, 요청 본문(필요한 경우) 예제 및 가능한 응답을 볼 수 있습니다.

## groups Operations on groups

GET

/org/groups Lists Tenant User Groups

Try it out

### Parameters

Name	Description
type	filter by group type
string (query)	
limit	maximum number of results
integer (query)	
marker	marker-style pagination offset (value is Group's URN)
string (query)	
includeMarker	if set, the marker element is also returned
boolean (query)	
order	pagination order (desc requires marker)
string (query)	

### Responses

Response content type

application/json

### Code Description

200

Example Value Model

```
[{"responseTime": "2018-02-01T16:22:31.066Z", "status": "success", "apiVersion": "2.0"}]
```

## API 요청을 발행합니다



API 문서 웹 페이지를 사용하여 수행하는 모든 API 작업은 라이브 작업입니다. 실수로 구성 데이터나 기타 데이터를 작성, 업데이트 또는 삭제하지 않도록 주의하십시오.

## 단계

1. 요청 세부 정보를 보려면 HTTP 작업을 선택합니다.
2. 요청에 그룹 또는 사용자 ID와 같은 추가 매개 변수가 필요한지 확인합니다. 그런 다음 이 값을 구합니다. 필요한 정보를 얻기 위해 먼저 다른 API 요청을 발급해야 할 수도 있습니다.
3. 예제 요청 본문을 수정해야 하는지 확인합니다. 이 경우 \* Model \* 을 선택하여 각 필드의 요구 사항을 확인할 수 있습니다.

4. 체험하기 \* 를 선택합니다.
5. 필요한 매개 변수를 제공하거나 요청 본문을 필요에 따라 수정합니다.
6. Execute \* 를 선택합니다.
7. 응답 코드를 검토하여 요청이 성공했는지 확인합니다.

#### 테넌트 관리 API 버전 관리

테넌트 관리 API는 버전 관리를 사용하여 무중단 업그레이드를 지원합니다.

예를 들어, 이 요청 URL은 API 버전 4를 지정합니다.

```
https://hostname_or_ip_address/api/v4/authorize
```

이전 버전과 호환되지 않는 변경 사항이 발생하면 API의 주 버전이 범핑됩니다. API의 부 버전은 이전 버전과 \_ 호환 \_ 을(를) 변경할 때 범핑됩니다. 호환 가능한 변경 사항에는 새 끝점 또는 새 속성 추가가 포함됩니다.

다음 예제에서는 변경 유형에 따라 API 버전을 충돌하는 방법을 보여 줍니다.

API 변경 유형입니다	이전 버전	새 버전
이전 버전과 호환 가능합니다	2.1	2.2
이전 버전과 호환되지 않습니다	2.1	3.0

StorageGRID 소프트웨어를 처음 설치하면 최신 버전의 API만 활성화됩니다. 그러나 StorageGRID의 새 기능 릴리즈로 업그레이드하면 하나 이상의 StorageGRID 기능 릴리즈에 대한 이전 API 버전에 계속 액세스할 수 있습니다.



지원되는 버전을 구성할 수 있습니다. 자세한 내용은 Swagger API 설명서의 \* config \* 섹션을 참조하십시오 "[Grid Management API를 참조하십시오](#)". 최신 버전을 사용하려면 모든 API 클라이언트를 업데이트한 후 이전 버전에 대한 지원을 비활성화해야 합니다.

오래된 요청은 다음과 같은 방법으로 더 이상 사용되지 않는 것으로 표시됩니다.

- 응답 헤더가 "DEPRECATED:TRUE"입니다.
- JSON 응답 본문에는 "DEPRECATED"가 포함됩니다. TRUE
- 더 이상 사용되지 않는 경고가 NMS.log에 추가됩니다. 예를 들면 다음과 같습니다.

```
Received call to deprecated v2 API at POST "/api/v2/authorize"
```

현재 릴리즈에서 지원되는 API 버전을 확인합니다

API 요청을 사용하여 GET /versions 지원되는 API 주요 버전 목록을 반환합니다. 이 요청은 Swagger API 설명서의 \* config \* 섹션에 있습니다.

```
GET https://{{IP-Address}}/api/versions
{
  "responseTime": "2023-06-27T22:13:50.750Z",
  "status": "success",
  "apiVersion": "4.0",
  "data": [
    2,
    3,
    4
  ]
}
```

요청에 대한 **API** 버전을 지정합니다

경로 매개 변수(/api/v4) 또는 헤더를 사용하여 API 버전을 지정할 수 ('Api-Version: 4' 있습니다. 두 값을 모두 제공하면 헤더 값이 경로 값을 재정의합니다.

```
curl https://[IP-Address]/api/v4/grid/accounts
curl -H "Api-Version: 4" https://[IP-Address]/api/grid/accounts
```

사이트 간 요청 위조(**CSRF**)로부터 보호

CSRF 토큰을 사용하여 쿠키를 사용하는 인증을 강화하면 StorageGRID에 대한 CSRF(사이트 간 요청 위조) 공격으로부터 보호할 수 있습니다. Grid Manager 및 Tenant Manager는 이 보안 기능을 자동으로 활성화합니다. 다른 API 클라이언트는 로그인할 때 활성화 여부를 선택할 수 있습니다.

HTTP 양식 POST와 같이 다른 사이트에 대한 요청을 트리거할 수 있는 공격자는 로그인한 사용자의 쿠키를 사용하여 특정 요청을 만들 수 있습니다.

StorageGRID는 CSRF 토큰을 사용하여 CSRF 공격으로부터 보호합니다. 활성화된 경우 특정 쿠키의 내용은 특정 헤더 또는 특정 POST 본문 매개 변수의 내용과 일치해야 합니다.

이 기능을 활성화하려면 csrfToken 인증 중에 매개 변수를 로 true 설정합니다. 기본값은입니다 false.

```
curl -X POST --header "Content-Type: application/json" --header "Accept: application/json" -d "{
  \"username\": \"MyUserName\",
  \"password\": \"MyPassword\",
  \"cookie\": true,
  \"csrfToken\": true
}" "https://example.com/api/v3/authorize"
```

true 인 경우, GridCsrfToken 쿠키는 Grid Manager 로그인에 대한 임의 값으로 설정되고 AccountCsrfToken 쿠키는 Tenant Manager에 로그인하기 위한 임의 값으로 설정됩니다.

쿠키가 있는 경우 시스템 상태(POST, PUT, 패치, 삭제)를 수정할 수 있는 모든 요청에 다음 중 하나가 포함되어야 합니다.

- 'X-Csrf-Token' 헤더 값이 CSRF 토큰 쿠키의 값으로 설정된 헤더입니다.
- 폼으로 인코딩된 본문을 수락하는 끝점의 경우: csrfToken 폼으로 인코딩된 요청 본문 매개 변수입니다.

CSRF 보호를 구성하려면 또는 를 "[Grid Management API를 참조하십시오](#)" "[테넌트 관리 API](#)" 사용합니다.



CSRF 토큰 쿠키 세트가 있는 요청은 CSRF 공격에 대한 추가 보호로서 JSON 요청 본문을 기대하는 모든 요청에 대해 "Content-Type: application/json" 헤더를 적용합니다.

## 그리드 페더레이션 연결을 사용합니다

클론 테넌트 그룹 및 사용자

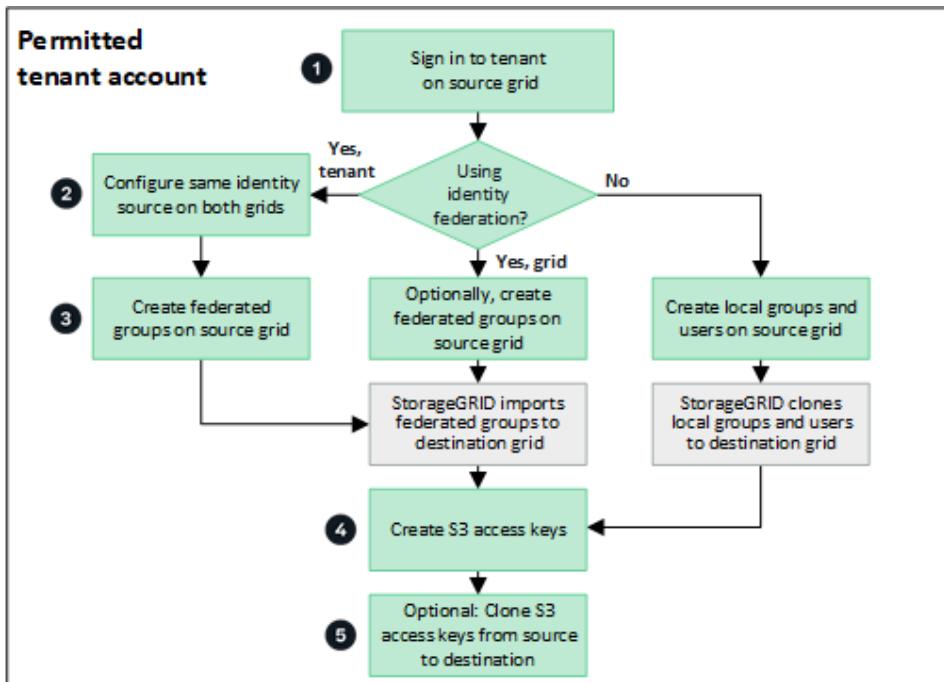
그리드 페더레이션 연결을 사용하도록 테넌트를 만들거나 편집한 경우 해당 테넌트는 한 StorageGRID 시스템(소스 테넌트)에서 다른 StorageGRID 시스템(복제본 테넌트)으로 복제됩니다. 테넌트가 복제되면 소스 테넌트에 추가된 모든 그룹 및 사용자가 복제본 테넌트에 클론됩니다.

테넌트가 처음 생성된 StorageGRID 시스템은 테넌트의 \_source GRID\_입니다. 테넌트가 복제되는 StorageGRID 시스템은 테넌트의 \_destination GRID\_입니다. 두 테넌트 계정 모두 동일한 계정 ID, 이름, 설명, 스토리지 할당량 및 할당된 권한이 있습니다. 그러나 대상 테넌트는 처음에 루트 사용자 암호를 가지고 있지 않습니다. 자세한 내용은 및 을 "[계정 클론이란 무엇입니까](#)" "[허용된 테넌트 관리](#)" 참조하십시오.

버킷 객체의 경우 테넌트 계정 정보의 복제가 "[교차 그리드 복제](#)" 필요합니다. 두 그리드에 동일한 테넌트 그룹 및 사용자가 있으면 양쪽 그리드의 해당 버킷 및 오브젝트에 액세스할 수 있습니다.

계정 클론에 대한 테넌트 워크플로우

테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 워크플로 디아그램을 검토하여 그룹, 사용자 및 S3 액세스 키를 복제하는 단계를 확인합니다.



워크플로의 주요 단계는 다음과 같습니다.

1

테넌트에 로그인합니다

소스 그리드(테넌트가 처음 생성된 그리드)에서 테넌트 계정에 로그인합니다.

2

필요한 경우 ID 페더레이션을 구성합니다

테넌트 계정에 통합 그룹 및 사용자를 사용할 수 있는 \* 고유 ID 소스 사용 \* 권한이 있는 경우 소스 및 대상 테넌트 계정 모두에 대해 동일한 ID 소스(동일한 설정 사용)를 구성합니다. 두 그리드에서 동일한 ID 소스를 사용하고 있지 않으면 통합 그룹과 사용자를 복제할 수 없습니다. 자세한 내용은 ["ID 페더레이션을 사용합니다"](#) 참조하십시오.

3

그룹 및 사용자를 생성합니다

그룹 및 사용자를 생성할 때는 항상 테넌트의 소스 그리드에서 시작합니다. 새 그룹을 추가하면 StorageGRID에서 자동으로 대상 그리드에 클론을 생성합니다.

- 전체 StorageGRID 시스템 또는 테넌트 계정에 대해 ID 페더레이션이 구성된 경우 ID 소스에서 페더레이션 ["테넌트 그룹을 생성합니다"](#) 그룹을 가져옵니다.
- ID 페더레이션을 사용하지 않는 경우 ["새 로컬 그룹을 생성합니다"](#) 그런 다음 ["로컬 사용자를 생성합니다"](#).

4

S3 액세스 키를 생성합니다

소스 그리드 또는 대상 그리드에서 또는 로 ["다른 사용자의 액세스 키를 생성합니다"](#) 이동하여 해당 그리드의 버킷을 액세스할 수 ["사용자 고유의 액세스 키를 생성합니다"](#) 있습니다.

## 5

### 필요한 경우 S3 액세스 키를 클론 복제합니다

두 그리드에 동일한 액세스 키가 있는 버킷에 액세스해야 하는 경우 소스 그리드에 액세스 키를 생성한 다음 테넌트 관리자 API를 사용하여 수동으로 대상 그리드에 클론을 생성합니다. 자세한 내용은 ["API를 사용하여 S3 액세스 키의 클론을 생성합니다"](#) 참조하십시오.

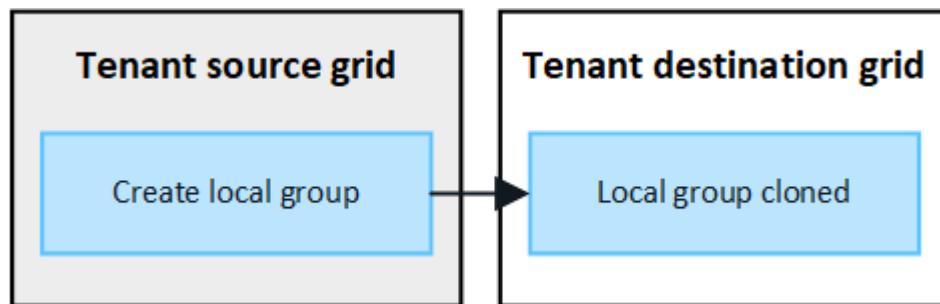
그룹, 사용자 및 S3 액세스 키의 클론을 생성하는 방법은 무엇입니까?

이 섹션을 검토하여 테넌트 소스 그리드와 테넌트 대상 그리드 간에 그룹, 사용자 및 S3 액세스 키의 클론 생성 방법을 이해합니다.

소스 그리드에 생성된 로컬 그룹이 복제됩니다

테넌트 계정이 생성되어 대상 그리드에 복제된 후 StorageGRID는 테넌트의 소스 그리드에 추가한 모든 로컬 그룹을 테넌트의 대상 그리드에 자동으로 복제합니다.

원래 그룹과 해당 클론 모두에 동일한 액세스 모드, 그룹 권한 및 S3 그룹 정책이 있습니다. 자세한 내용은 ["S3 테넌트용 그룹을 생성합니다"](#) 참조하십시오.

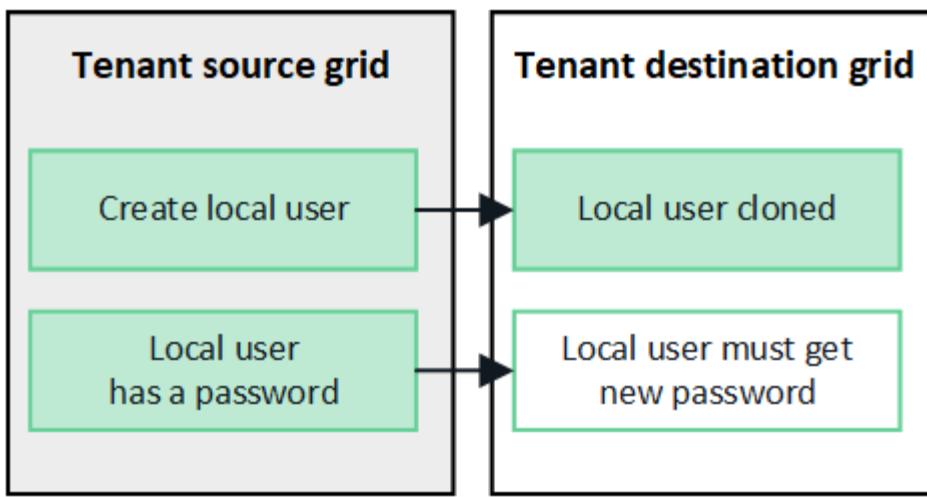


소스 그리드에 로컬 그룹을 생성할 때 선택한 사용자는 그룹이 대상 그리드에 클론될 때 포함되지 않습니다. 따라서 그룹을 만들 때 사용자를 선택하지 마십시오. 대신 사용자를 생성할 때 그룹을 선택합니다.

소스 그리드에 생성된 로컬 사용자의 클론이 생성됩니다

소스 그리드에 새로운 로컬 사용자를 생성하면 StorageGRID 자동으로 해당 사용자를 대상 그리드에 복제합니다. 원래 사용자와 복제본 모두 동일한 전체 이름, 사용자 이름 및 액세스 거부 설정을 갖습니다. 두 사용자는 모두 같은 그룹에 속합니다. 자침은 다음을 참조하세요. ["사용자 관리"](#).

보안상의 이유로 로컬 사용자 비밀번호는 대상 그리드에 복제되지 않습니다. 로컬 사용자가 대상 그리드에서 테넌트 관리자에 액세스해야 하는 경우 테넌트 계정의 루트 사용자는 대상 그리드에서 해당 사용자에 대한 비밀번호를 추가해야 합니다. 자침은 다음을 참조하세요. ["사용자 관리"](#).

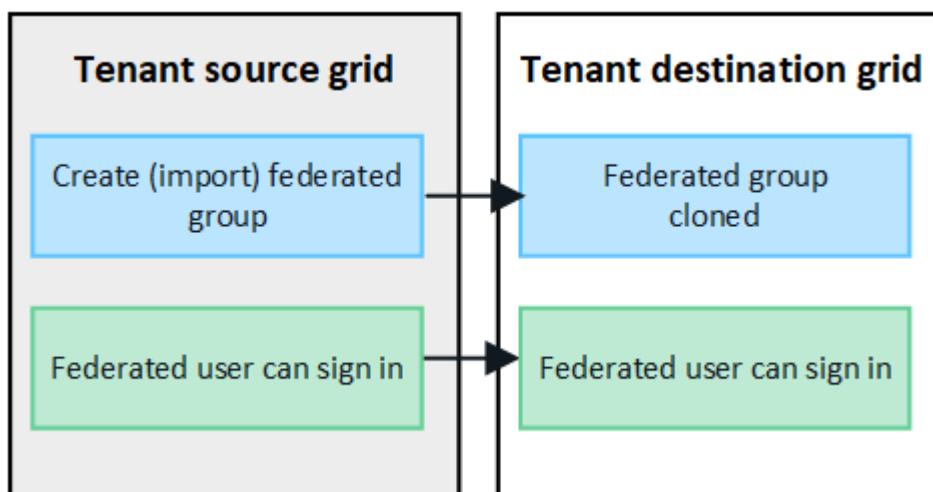


소스 그리드에 생성된 연합 그룹이 클론됩니다

에서 계정 클론을 사용하기 위한 요구 사항이 충족되었다고 가정할 때 "SSO(Single Sign-On)" "ID 제휴"소스 그리드에서 테넌트에 대해 생성(가져오기)한 페더레이션 그룹이 대상 그리드의 테넌트에 자동으로 복제됩니다.

두 그룹 모두 동일한 액세스 모드, 그룹 권한 및 S3 그룹 정책을 가집니다.

소스 테넌트의 통합 그룹이 생성되고 대상 테넌트에 클론이 생성되면 통합 사용자는 두 그리드 중 하나에서 테넌트에 로그인할 수 있습니다.

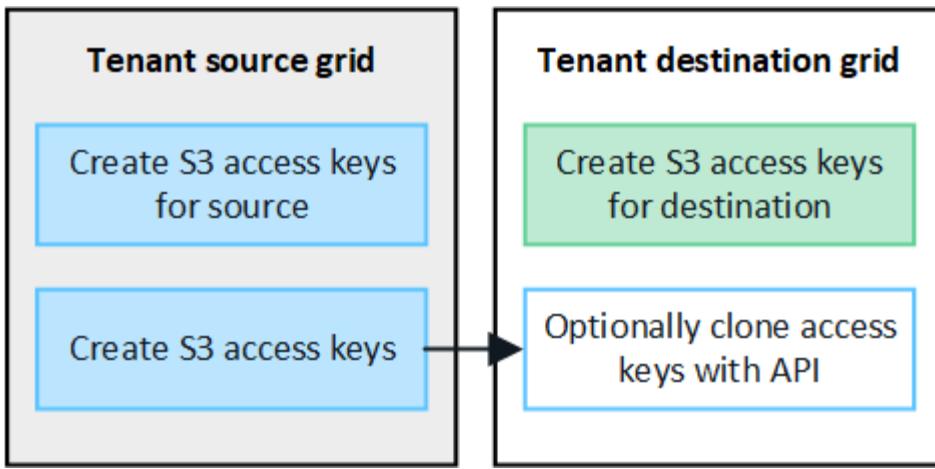


S3 액세스 키를 수동으로 복제할 수 있습니다

StorageGRID는 각 그리드에 서로 다른 키를 추가하여 보안을 강화하기 때문에 S3 액세스 키를 자동으로 복제하지 않습니다.

두 그리드에서 액세스 키를 관리하려면 다음 중 하나를 수행합니다.

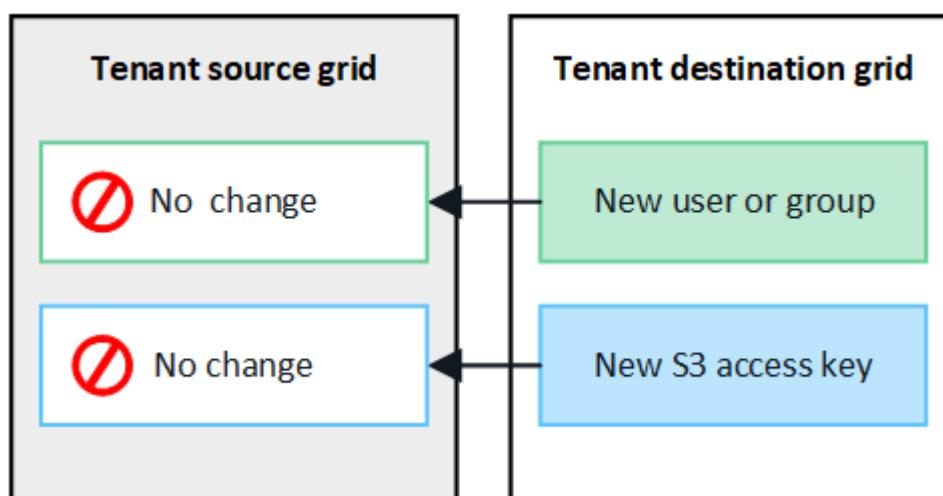
- 각 그리드에 대해 동일한 키를 사용할 필요가 없는 경우 또는 "다른 사용자의 액세스 키를 생성합니다"각 그리드에서 사용할 수 "사용자 고유의 액세스 키를 생성합니다"있습니다.
- 두 그리드 모두에서 동일한 키를 사용해야 하는 경우 소스 그리드에 키를 생성한 다음 Tenant Manager API를 사용하여 대상 그리드로 수동으로 이동할 수 "키를 복제합니다"있습니다.



통합 사용자의 S3 액세스 키를 클론하면 사용자 및 S3 액세스 키가 모두 대상 테넌트에 복제됩니다.

대상 그리드에 추가된 그룹 및 사용자는 클론이 생성되지 않습니다

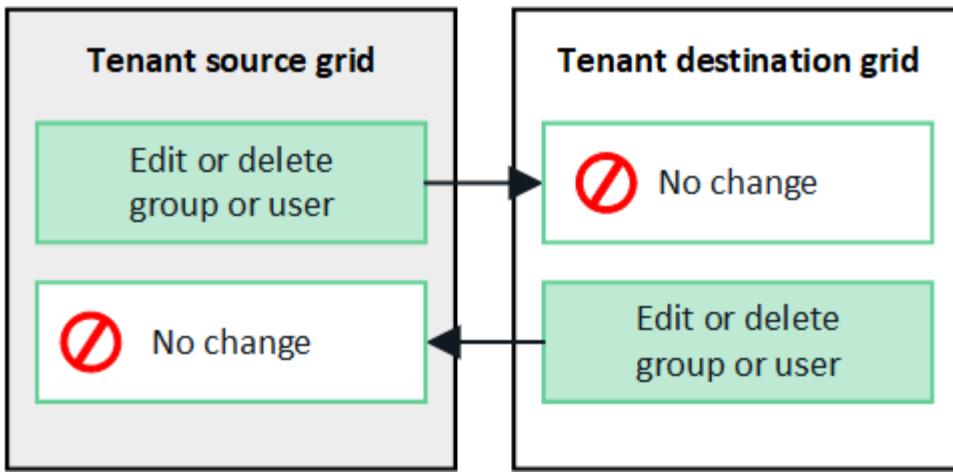
클론 생성은 테넌트의 소스 그리드에서 테넌트의 대상 그리드로만 이루어집니다. 테넌트의 대상 그리드에 그룹 및 사용자를 생성하거나 가져오는 경우 StorageGRID는 이러한 항목을 테넌트의 소스 그리드에 다시 복제하지 않습니다.



편집 또는 삭제된 그룹, 사용자 및 액세스 키는 복제되지 않습니다

클론 생성은 새 그룹 및 사용자를 생성할 때만 발생합니다.

두 눈금 중 하나에서 그룹, 사용자 또는 액세스 키를 편집하거나 삭제하면 변경 내용이 다른 눈금에 복제되지 않습니다.



API를 사용하여 S3 액세스 키의 클론을 생성합니다

테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 테넌트 관리 API를 사용하여 소스 그리드의 테넌트에서 대상 그리드의 테넌트로 S3 액세스 키를 수동으로 복제할 수 있습니다.

시작하기 전에

- 테넌트 계정에는 \* 그리드 페더레이션 연결 사용 \* 권한이 있습니다.
- 그리드 페더레이션 연결에는 \* 연결 상태 \* 가 \* 연결됨 \* 으로 표시됩니다.
- 를 사용하여 테넌트의 소스 그리드에서 테넌트 관리자에 로그인되어 "지원되는 웹 브라우저"입니다.
- 이 있는 사용자 그룹에 속해 "자신의 S3 자격 증명 또는 루트 액세스 권한을 관리합니다"입니다.
- 로컬 사용자에 대한 액세스 키를 클론하는 경우 사용자는 이미 두 그리드에 존재합니다.



통합 사용자의 S3 액세스 키를 클론하면 사용자 및 S3 액세스 키가 모두 대상 테넌트에 추가됩니다.

사용자 고유의 액세스 키를 복제합니다

두 그리드의 동일한 버킷에 액세스해야 하는 경우 고유한 액세스 키를 복제할 수 있습니다.

단계

1. 소스 그리드에서 Tenant Manager를 사용하여 "사용자 고유의 액세스 키를 생성합니다" `.csv` 파일을 다운로드합니다.
2. 테넌트 관리자 상단에서 도움말 아이콘을 선택하고 \* API documentation \* 을 선택합니다.
3. S3 \* 섹션에서 다음 끝점을 선택합니다.

POST /org/users/current-user/replicate-s3-access-key

POST

/org/users/current-user/replicate-s3-access-key Clone the current user's S3 key to the other grids.



4. 체험하기 \* 를 선택합니다.
5. body \* 텍스트 상자에서 \* AccessKey \* 및 \* secretAccessKey \* 의 예제 항목을 다운로드한 \*.csv \* 파일의 값으로 바꿉니다.

각 문자열 주위에 큰따옴표를 붙여야 합니다.



```
body * required
Edit Value | Model
(body)

{
  "accessKey": "AKIAIOSFODNN7EXAMPLE",
  "secretAccessKey": "wJalrXUtnFEMI/K7MDENG/bPxRfiCYEXAMPLEKEY",
  "expires": "2028-09-04T00:00:00.000Z"
}
```

6. 키가 만료되면 \* Expires \* 에 대한 예제 항목을 ISO 8601 데이터-시간 형식(예:)의 문자열로 2024-02-28T22:46:33-08:00 대체합니다. 키가 만료되지 않으면 \* expires \* 항목의 값으로 \* null \* 을 입력하거나 \* expires \* 행 및 앞의 쉼표를 제거합니다.
7. Execute \* 를 선택합니다.
8. 서버 응답 코드가 \* 204 \* 인지 확인합니다. 이는 키가 대상 그리드에 성공적으로 복제되었음을 나타냅니다.

다른 사용자의 액세스 키를 복제합니다

두 그리드의 동일한 버킷에 액세스해야 하는 경우 다른 사용자의 액세스 키를 복제할 수 있습니다.

단계

1. 소스 그리드에서 Tenant Manager를 사용하여 ["다른 사용자의 S3 액세스 키를 생성합니다"](#) `.csv` 파일을 다운로드합니다.
2. 테넌트 관리자 상단에서 도움말 아이콘을 선택하고 \* API documentation \* 을 선택합니다.
3. 사용자 ID를 얻습니다. 다른 사용자의 액세스 키를 복제하려면 이 값이 필요합니다.
  - a. 사용자 \* 섹션에서 다음 끝점을 선택합니다.

GET /org/users

- b. 체험하기 \* 를 선택합니다.
- c. 사용자를 찾을 때 사용할 매개 변수를 지정합니다.
- d. Execute \* 를 선택합니다.
- e. 복제할 키가 있는 사용자를 찾고 \* id \* 필드에 번호를 복사합니다.

4. S3 \* 섹션에서 다음 끝점을 선택합니다.

POST /org/users/{userId}/replicate-s3-access-key

POST

/org/users/{userId}/replicate-s3-access-key Clone an S3 key to the other grids.



5. 체험하기 \* 를 선택합니다.
6. userId \* 텍스트 상자에 복사한 사용자 ID를 붙여 넣습니다.
7. body \* 텍스트 상자에서 \* 예제 액세스 키 \* 및 \* 비밀 액세스 키 \* 와 같은 항목을 해당 사용자의 \*.csv \* 파일 값으로 바꿉니다.

문자열 주위에 큰따옴표를 붙여야 합니다.

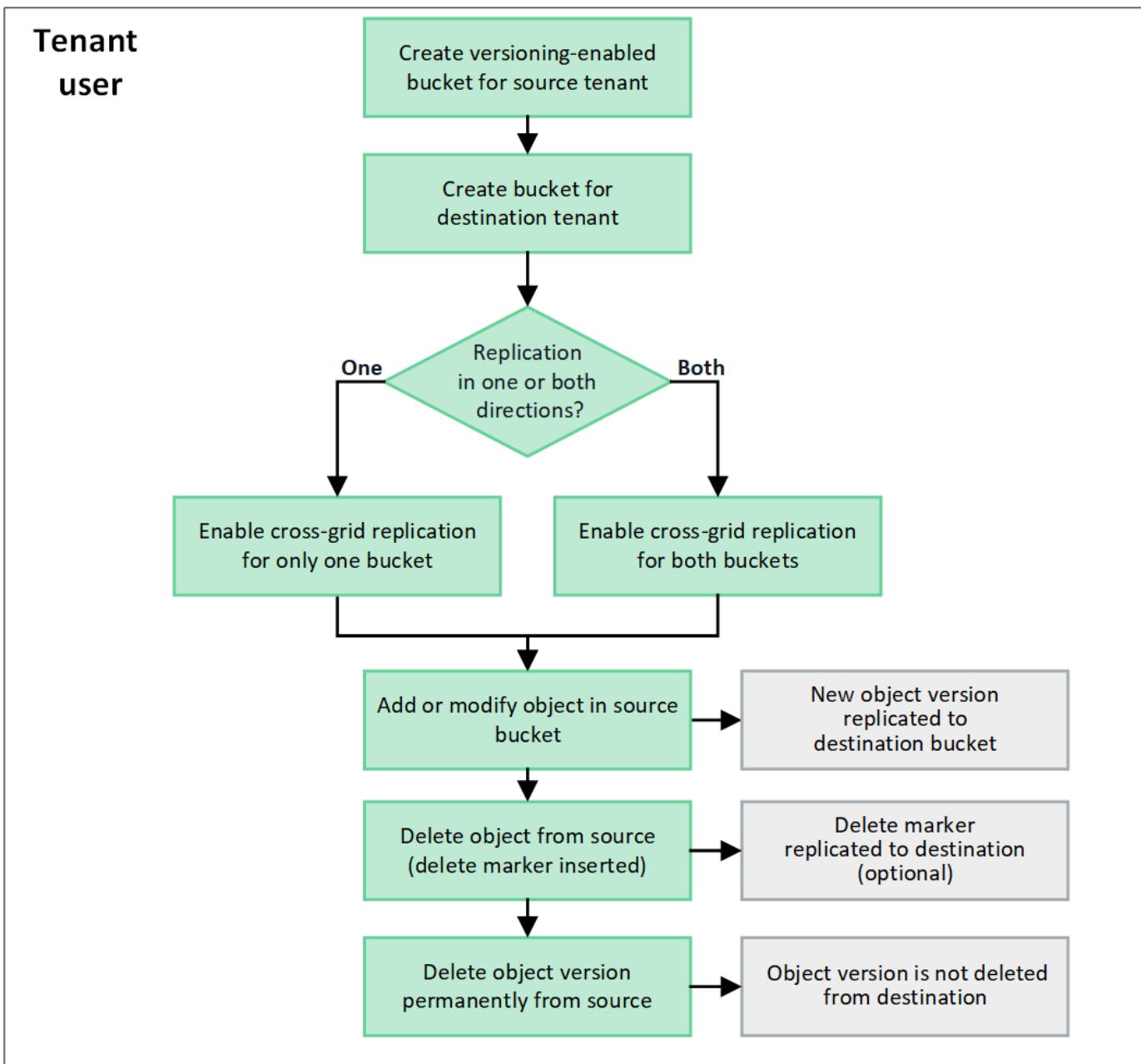
8. 키가 만료되면 \* Expires \* 에 대한 예제 항목을 ISO 8601 데이터-시간 형식(예:)의 문자열로 2023-02-28T22:46:33-08:00 대체합니다. 키가 만료되지 않으면 \* expires \* 항목의 값으로 \* null \* 을 입력하거나 \* expires \* 행 및 앞의 쉼표를 제거합니다.
9. Execute \* 를 선택합니다.
10. 서버 응답 코드가 \* 204 \* 인지 확인합니다. 이는 키가 대상 그리드에 성공적으로 복제되었음을 나타냅니다.

#### 교차 그리드 복제 관리

테넌트 계정이 생성되었을 때 \* 그리드 페더레이션 연결 사용 \* 권한이 할당된 경우 교차 그리드 복제를 사용하여 테넌트의 소스 그리드에 있는 버킷과 테넌트 대상 그리드에 있는 버킷 간에 객체를 자동으로 복제할 수 있습니다. 크로스 그리드 복제는 한 방향 또는 두 방향으로 수행될 수 있습니다.

#### 그리드 간 복제를 위한 워크플로우

워크플로우 다이어그램은 두 그리드의 버킷 간에 크로스 그리드 복제를 구성하기 위해 수행하는 단계를 요약한 것입니다. 이러한 단계는 아래 다이어그램에서 더 자세히 설명됩니다.



교차 그리드 복제를 구성합니다

크로스 그리드 복제를 사용하려면 먼저 각 그리드의 해당 테넌트 계정에 로그인하고 두 개의 버킷을 만들어야 합니다. 그런 다음 두 버킷 중 하나 또는 둘 다에서 크로스 그리드 복제를 활성화할 수 있습니다.

시작하기 전에

- 크로스 그리드 복제에 대한 요구 사항을 검토했습니다. ["크로스 그리드 복제란"](#).
- 당신은 사용 중입니다 ["지원되는 웹 브라우저"](#).
- 테넌트 계정에는 그리드 페더레이션 연결 사용 권한이 있으며, 두 그리드에 동일한 테넌트 계정이 있습니다. ["그리드 페더레이션 연결에 대해 허용된 테넌트를 관리합니다"](#).
- 로그인하려는 테넌트 사용자는 두 그리드에 모두 이미 존재하며 다음 권한이 있는 사용자 그룹에 속합니다. ["루트 액세스 권한"](#).
- 로컬 사용자로 테넌트의 대상 그리드에 로그인하는 경우 테넌트 계정의 루트 사용자가 해당 그리드에서 사용자

계정에 대한 비밀번호를 설정했습니다.

두 개의 버킷을 만듭니다

첫 번째 단계로, 각 그리드의 해당 테넌트 계정에 로그인하고 각 그리드에 버킷을 만듭니다.

단계

1. 그리드 페더레이션 연결의 두 그리드 중 하나에서 시작하여 새 버킷을 만듭니다.

a. 두 그리드에 있는 테넌트 사용자의 자격 증명을 사용하여 테넌트 계정에 로그인합니다.

로컬 사용자로 테넌트의 대상 그리드에 로그인할 수 없는 경우 테넌트 계정의 루트 사용자가 사용자 계정에 대한 비밀번호를 설정했는지 확인하세요.

b. 의 지침을 "[S3 버킷을 생성합니다](#)" 따릅니다.



버킷 이름과 지역은 각 그리드마다 다를 수 있습니다.

c. 개체 설정 관리 \* 탭에서 \* 개체 버전 관리 사용 \* 을 선택합니다.

d. StorageGRID 시스템에 S3 개체 잠금이 활성화된 경우 다음을 참조하세요. "[S3 객체 잠금을 통한 크로스 그리드 복제](#)".

e. Create bucket \* 을 선택합니다.

f. 마침 \* 을 선택합니다.

2. 그리드 페더레이션 연결의 다른 그리드에서 동일한 테넌트 계정에 대한 버킷을 만들려면 이 단계를 반복합니다.



필요에 따라 각 버킷은 다른 지역을 사용할 수 있습니다.

교차 그리드 복제를 설정합니다

두 버킷 중 하나에 물체를 추가하기 전에 이 단계를 수행해야 합니다.

단계

1. 복제할 객체가 있는 그리드에서 시작하여 "[한 방향으로 크로스 그리드 복제](#)" 다음을 활성화합니다.

a. 버켓의 테넌트 계정에 로그인합니다.

b. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

c. 버킷 세부 정보 페이지에 액세스하려면 테이블에서 버킷 이름을 선택합니다.

d. Cross-grid replication \* 탭을 선택합니다.

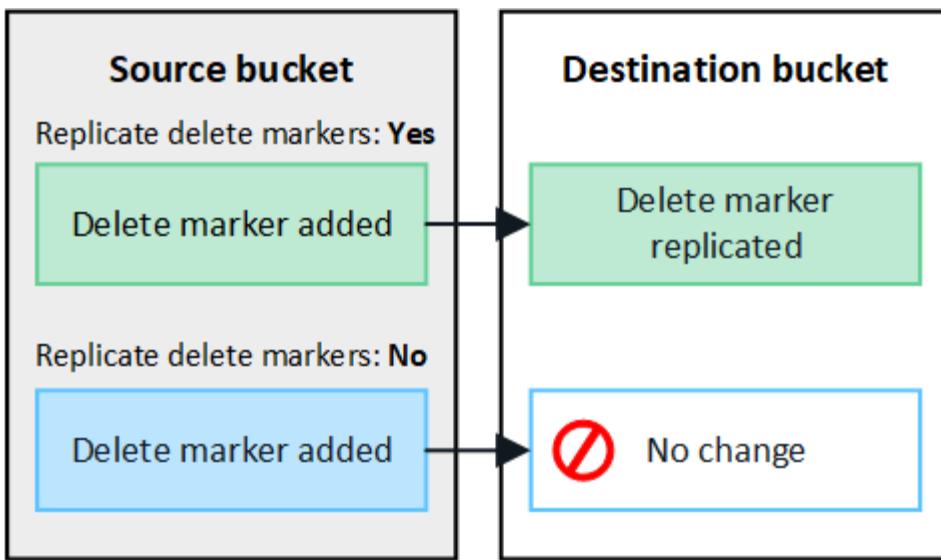
e. 사용 \* 을 선택하고 요구 사항 목록을 검토합니다.

f. 모든 요구 사항이 충족되면 사용할 그리드 페더레이션 연결을 선택합니다.

g. 선택적으로 \* Delete Markers 복제 \* 의 설정을 변경하여 S3 클라이언트가 버전 ID가 포함되지 않은 소스 그리드에 삭제 요청을 발행하는 경우 대상 그리드에 어떤 일이 발생하는지 확인합니다.

▪ \* 예 \* (기본값): 삭제 마커가 소스 버킷에 추가되고 대상 버킷에 복제됩니다.

▪ 아니요: 삭제 마커는 소스 버킷에 추가되지만 대상 버킷에 복제되지 않습니다.



삭제 요청에 버전 ID가 포함되어 있는 경우 해당 개체 버전은 소스 버킷에서 영구적으로 제거됩니다. StorageGRID 버전 ID가 포함된 삭제 요청을 복제하지 않으므로 대상에서 동일한 개체 버전이 삭제되지 않습니다.

참조하다 "[크로스 그리드 복제란](#)" 자세한 내용은.

- 필요에 따라 \* 교차 그리드 복제 \* 감사 범주의 설정을 변경하여 감사 메시지의 볼륨을 관리합니다.
  - \* Error \* (기본값): 실패한 교차 그리드 복제 요청만 감사 출력에 포함됩니다.
  - \* 정상 \*: 모든 교차 그리드 복제 요청이 포함되어 감사 출력 볼륨을 크게 높입니다.
- 선택 사항을 검토합니다. 두 버킷이 모두 비어 있지 않으면 이 설정을 변경할 수 없습니다.
- Enable and test \* 를 선택합니다.

잠시 후 성공 메시지가 나타납니다. 이 버킷에 추가된 객체는 이제 다른 그리드에 자동으로 복제됩니다. \*크로스 그리드 복제\*는 버킷 세부 정보 페이지에서 활성화된 기능으로 표시됩니다.

## 2. 필요에 따라 다른 그리드의 해당 버킷으로 이동하고 "[양방향 교차 그리드 복제를 활성화합니다](#)".

그리드 간의 복제를 테스트합니다

버킷에 대해 교차 그리드 복제가 활성화된 경우 연결 및 교차 그리드 복제가 올바르게 작동하고 소스 및 대상 버킷이 모든 요구 사항을 충족하는지 확인해야 할 수 있습니다(예: 버전 관리가 계속 활성화됨).

시작하기 전에

- 당신은 사용 중입니다 "[지원되는 웹 브라우저](#)".
- 이 있는 사용자 그룹에 속해 "[루트 액세스 권한](#)" 있습니다.

단계

- 버켓의 테넌트 계정에 로그인합니다.
- 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
- 버킷 세부 정보 페이지에 액세스하려면 테이블에서 버킷 이름을 선택합니다.
- Cross-grid replication \* 탭을 선택합니다.

## 5. Test connection \* 을 선택합니다.

연결이 정상적이면 성공 배너가 나타납니다. 그렇지 않으면 오류 메시지가 나타나는데, 이를 통해 그리드 관리자가 문제를 해결할 수 있습니다. 자세한 내용은 다음을 참조하세요. ["그리드 통합 오류 문제 해결"](#).

## 6. 양방향 복제가 수행되도록 구성된 경우 다른 그리드의 해당 버킷으로 이동하여 \* Test connection \* 을 선택하여 교차 그리드 복제가 다른 방향으로 작동하는지 확인합니다.

크로스 그리드 복제를 비활성화합니다

더 이상 다른 그리드에 객체를 복사하지 않으려는 경우 그리드 간 복제를 영구적으로 중지할 수 있습니다.

교차 그리드 복제를 사용하지 않도록 설정하기 전에 다음 사항에 유의하십시오.

- 그리드 간 복제를 비활성화해도 그리드 간에 이미 복사된 객체는 제거되지 않습니다. 예를 들어, 객체 `my-bucket` 복사된 그리드 1에 `my-bucket` 그리드 2의 버킷에 대해 크로스 그리드 복제를 비활성화하면 해당 버킷은 제거되지 않습니다. 이러한 객체를 삭제하려면 수동으로 제거해야 합니다.
- 각 버킷에 대해 교차 그리드 복제가 설정된 경우(즉, 양방향으로 복제가 발생하는 경우), 하나 또는 두 버킷에 대해 교차 그리드 복제를 비활성화할 수 있습니다. 예를 들어, 그리드 1에서 그리드 2 `my-bucket`로 객체를 복제하는 동시에 그리드 2에서 그리드 1 `my-bucket`로 객체를 계속 `my-bucket` 복제할 수 없도록 설정할 수 `my-bucket` 있습니다.
- 테넌트의 그리드 페더레이션 연결 사용 권한을 제거하려면 먼저 크로스 그리드 복제를 비활성화해야 합니다. ["허용된 테넌트 관리"](#).
- 객체가 포함된 버킷에 대해 크로스 그리드 복제를 비활성화하면 소스 버킷과 대상 버킷 모두에서 모든 객체를 삭제하지 않는 한 크로스 그리드 복제를 다시 활성화할 수 없습니다.



두 버킷이 모두 비어 있지 않으면 복제를 다시 설정할 수 없습니다.

시작하기 전에

- 당신은 사용 중입니다 ["지원되는 웹 브라우저"](#).
- 이 있는 사용자 그룹에 속해 ["루트 액세스 권한"](#)입니다.

단계

- 더 이상 복제할 객체가 없는 그리드에서 시작하여 버킷에 대한 교차 그리드 복제를 중지합니다.
  - 버킷의 테넌트 설정에 로그인합니다.
  - 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
  - 버킷 세부 정보 페이지에 액세스하려면 테이블에서 버킷 이름을 선택합니다.
  - Cross-grid replication \* 탭을 선택합니다.
  - 복제 비활성화 \* 를 선택합니다.
  - 이 버킷에 대한 크로스 그리드 복제를 비활성화하려면 텍스트 상자에 \*예\*를 입력하고 \*비활성화\*를 선택합니다.

잠시 후 성공 메시지가 나타납니다. 이 버킷에 추가된 새 오브젝트는 더 이상 다른 그리드에 자동으로 복제될 수 없습니다. \* 교차 그리드 복제 \* 는 버킷 페이지에서 더 이상 활성화 기능으로 표시되지 않습니다.

- 양방향 복제가 수행되도록 구성된 경우 다른 그리드의 해당 버킷으로 이동하여 다른 방향으로 크로스 그리드 복제를 중지합니다.

그리드 페더레이션 연결을 봅니다

테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 허용된 연결을 볼 수 있습니다.

시작하기 전에

- 테넌트 계정에는 \* 그리드 페더레이션 연결 사용 \* 권한이 있습니다.
- 를 사용하여 테넌트 관리자에 로그인되어 "지원되는 웹 브라우저"입니다.
- 이 있는 사용자 그룹에 속해 "루트 액세스 권한"있습니다.

단계

1. 스토리지(S3) \* > \* 그리드 페더레이션 연결 \* 을 선택합니다.

그리드 페더레이션 연결 페이지가 나타나고 다음 정보를 요약하는 테이블이 포함됩니다.

열	설명
연결 이름입니다	이 테넌트가 사용할 수 있는 그리드 페더레이션 연결이 있습니다.
크로스 그리드 복제를 포함한 버킷	각 그리드 페더레이션 연결에 대해 교차 그리드 복제가 활성화된 테넌트 버킷입니다. 이러한 버킷에 추가된 객체는 연결의 다른 그리드에 복제됩니다.
마지막 오류	각 그리드 페더레이션 연결에 대해 데이터가 다른 그리드에 복제되는 경우 가장 최근의 오류가 발생합니다. 을 <a href="#">마지막 오류를 지웁니다</a> 참조하십시오.

2. 필요한 경우 버킷 이름을 "버킷 세부 정보를 봅니다"선택합니다.

마지막 오류를 지웁니다

다음 이유 중 하나로 인해 \* Last error \* (마지막 오류 \*) 열에 오류가 나타날 수 있습니다.

- 소스 객체 버전을 찾을 수 없습니다.
- 소스 버킷을 찾을 수 없습니다.
- 대상 버킷이 삭제되었습니다.
- 대상 버킷이 다른 계정에 의해 다시 생성되었습니다.
- 대상 버킷에 버전 관리가 일시 중지되었습니다.
- 대상 버킷은 동일한 계정으로 다시 생성되었지만 현재는 버전이 지정되지 않았습니다.



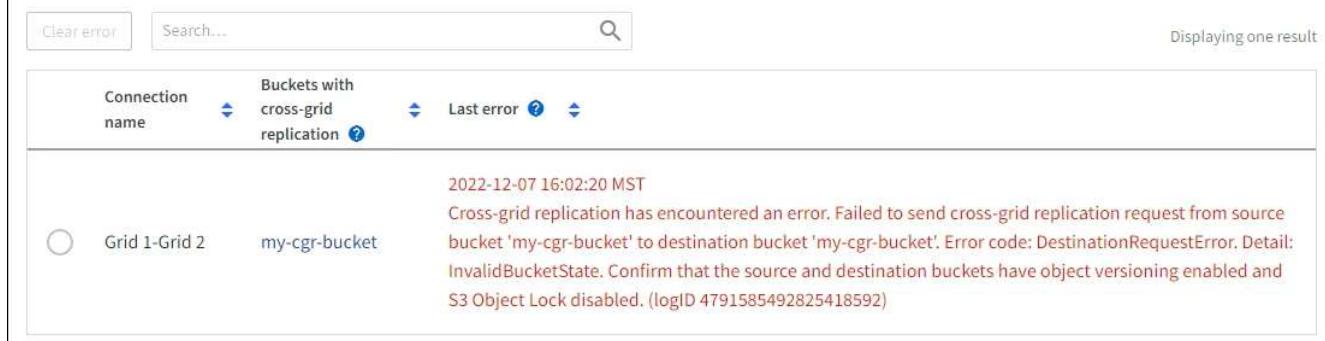
이 열에는 마지막으로 발생한 교차 그리드 복제 오류만 표시됩니다. 이전에 발생한 오류는 표시되지 않습니다.

단계

1. 마지막 오류 \* 열에 메시지가 나타나면 메시지 텍스트를 확인합니다.

예를 들어, 이 오류는 버전 관리가 일시 중단되었거나 S3 오브젝트 잠금이 설정되었기 때문에 교차 그리드 복제의 대상 버킷이 잘못된 상태에 있음을 나타냅니다.

# Grid federation connections



The screenshot shows a table with the following columns: Connection name, Buckets with cross-grid replication, and Last error. There is one row for 'Grid 1-Grid 2' with 'my-cgr-bucket'. The 'Last error' column contains the following text:

2022-12-07 16:02:20 MST  
Cross-grid replication has encountered an error. Failed to send cross-grid replication request from source bucket 'my-cgr-bucket' to destination bucket 'my-cgr-bucket'. Error code: DestinationRequestError. Detail: InvalidBucketState. Confirm that the source and destination buckets have object versioning enabled and S3 Object Lock disabled. (logID 4791585492825418592)

- 권장되는 작업을 수행합니다. 예를 들어 교차 그리드 복제를 위해 대상 버킷에서 버전 관리가 일시 중단된 경우 해당 버킷의 버전 관리를 다시 사용하도록 설정합니다.
- 테이블에서 연결을 선택합니다.
- Clear error \* 를 선택합니다.
- 메시지를 지우고 시스템 상태를 업데이트하려면 \* 예 \* 를 선택하십시오.
- 5-6분 정도 기다린 다음 새 오브젝트를 버킷에 넣습니다. 오류 메시지가 다시 나타나지 않는지 확인합니다.



오류 메시지가 지워졌는지 확인하려면 새 개체를 수신하기 전에 메시지의 타임스탬프가 나타난 후 5분 이상 기다립니다.

- 버킷 오류로 인해 객체를 복제하지 못했는지 확인하려면 을 참조하십시오."실패한 복제 작업을 식별하고 다시 시도하십시오"

## 그룹 및 사용자를 관리합니다

### ID 페더레이션을 사용합니다

ID 페더레이션을 사용하면 테넌트 그룹 및 사용자를 더 빠르게 설정할 수 있으며, 테넌트 사용자는 익숙한 자격 증명을 사용하여 테넌트 계정에 로그인할 수 있습니다.

테넌트 관리자에 대한 ID 페더레이션을 구성합니다

테넌트 그룹과 사용자를 Active Directory, Microsoft Entra ID, OpenLDAP 또는 Oracle Directory Server와 같은 다른 시스템에서 관리하려는 경우 테넌트 관리자에 대한 ID 페더레이션을 구성할 수 있습니다.

#### 시작하기 전에

- 귀하는 다음을 사용하여 테넌트 관리자에 로그인했습니다."지원되는 웹 브라우저" .
- 이 있는 사용자 그룹에 속해 "루트 액세스 권한"있습니다.
- ID 공급자로 Active Directory, Microsoft Entra ID, OpenLDAP 또는 Oracle Directory Server를 사용하고 있습니다.



나열되지 않은 LDAP v3 서비스를 사용하려면 기술 지원팀에 문의하세요.

- OpenLDAP를 사용하려면 OpenLDAP 서버를 구성해야 합니다. 을 OpenLDAP 서버 구성 지침 참조하십시오.

- LDAP 서버와의 통신에 TLS(Transport Layer Security)를 사용하려는 경우 ID 공급자는 TLS 1.2 또는 1.3을 사용해야 합니다. 을 "발신 TLS 연결에 지원되는 암호"참조하십시오.

이 작업에 대해

테넌트의 ID 페더레이션 서비스를 구성할 수 있는지 여부는 테넌트 계정 설정 방법에 따라 달라집니다. 테넌트가 Grid Manager용으로 구성된 ID 페더레이션 서비스를 공유할 수 있습니다. ID 페더레이션 페이지에 액세스할 때 이 메시지가 표시되면 이 테넌트에 대해 별도의 통합 ID 소스를 구성할 수 없습니다.

**i** This tenant account uses the LDAP server that is configured for the Grid Manager. Contact the grid administrator for information or to change this setting.

구성을 입력합니다

Identify 페더레이션을 구성할 때 StorageGRID가 LDAP 서비스에 연결하는 데 필요한 값을 제공합니다.

단계

- 액세스 관리 > \*ID 페더레이션\*을 선택합니다.
- ID 페더레이션 사용 \* 을 선택합니다.
- LDAP 서비스 유형 섹션에서 구성할 LDAP 서비스 유형을 선택합니다.

## LDAP service type

Select the type of LDAP service you want to configure.

Active Directory	Entra ID	OpenLDAP	Other
------------------	----------	----------	-------

Oracle Directory Server를 사용하는 LDAP 서버의 값을 구성하려면 \* 기타 \* 를 선택합니다.

- 기타 \* 를 선택한 경우 LDAP 속성 섹션의 필드를 작성합니다. 그렇지 않으면 다음 단계로 이동합니다.
  - 사용자 고유 이름: LDAP 사용자의 고유 식별자를 포함하는 속성의 이름입니다. 이 속성은 다음과 동일합니다. sAMAccountName Active Directory 및 uid OpenLDAP용. Oracle Directory Server를 구성하는 경우 다음을 입력하세요. uid .
  - 사용자 **UUID**: LDAP 사용자의 영구 고유 식별자를 포함하는 속성의 이름입니다. 이 속성은 다음과 동일합니다. objectGUID Active Directory 및 entryUUID OpenLDAP용. Oracle Directory Server를 구성하는 경우 다음을 입력하세요. nsuniqueid . 지정된 속성에 대한 각 사용자 값은 16바이트 또는 문자열 형식의 32자리 16진수여야 하며, 하이픈은 무시됩니다.
  - 그룹 고유 이름: LDAP 그룹의 고유 식별자를 포함하는 속성의 이름입니다. 이 속성은 다음과 동일합니다. sAMAccountName Active Directory 및 cn OpenLDAP용. Oracle Directory Server를 구성하는 경우 다음을 입력하세요. cn .
  - 그룹 **UUID**: LDAP 그룹의 영구 고유 식별자를 포함하는 속성의 이름입니다. 이 속성은 다음과 동일합니다. objectGUID Active Directory 및 entryUUID OpenLDAP용. Oracle Directory Server를 구성하는 경우 다음을 입력하세요. nsuniqueid . 지정된 속성에 대한 각 그룹의 값은 하이픈이 무시되는 16바이트 또는 문자열 형식의 32자리 16진수여야 합니다.

5. 모든 LDAP 서비스 유형에 대해 LDAP 서버 구성 섹션에 필요한 LDAP 서버 및 네트워크 연결 정보를 입력합니다.

- \* 호스트 이름 \*: LDAP 서버의 FQDN(정규화된 도메인 이름) 또는 IP 주소입니다.
- \* 포트 \*: LDAP 서버에 연결하는 데 사용되는 포트입니다.



STARTTLS의 기본 포트는 389이고 LDAPS의 기본 포트는 636입니다. 그러나 방화벽이 올바르게 구성된 경우 모든 포트를 사용할 수 있습니다.

- \* 사용자 이름 \*: LDAP 서버에 연결할 사용자의 DN(고유 이름)의 전체 경로입니다.

Active Directory의 경우 아래쪽 로그온 이름 또는 사용자 기본 이름을 지정할 수도 있습니다.

지정된 사용자는 그룹 및 사용자를 나열하고 다음 속성에 액세스할 수 있는 권한이 있어야 합니다.

- sAMAccountName 또는 uid
- objectGUID entryUUID, 또는 nsuniqueid
- cn
- memberOf 또는 isMemberOf
- \* Active Directory \*: objectSid, primaryGroupID, userAccountControl 및 userPrincipalName
- 입장 ID: accountEnabled 그리고 userPrincipalName
- \* 암호 \*: 사용자 이름과 연결된 암호입니다.



나중에 암호를 변경하는 경우 이 페이지에서 암호를 업데이트해야 합니다.

- \* Group Base DN \*: 그룹을 검색할 LDAP 하위 트리에 대한 DN(고유 이름)의 전체 경로입니다. Active Directory 예제(아래)에서 고유 이름이 기본 DN(DC=StorageGrid, DC=example, DC=com)과 관련된 모든 그룹을 통합 그룹으로 사용할 수 있습니다.



그룹 고유 이름 \* 값은 \* 그룹 기본 DN \* 내에서 고유해야 합니다.

- \* 사용자 기본 DN \*: 사용자를 검색할 LDAP 하위 트리의 고유 이름(DN)의 전체 경로입니다.



사용자 고유 이름 \* 값은 \* 사용자 기본 DN \* 내에서 고유해야 합니다.

- \* 사용자 이름 형식 바인딩 \* (선택 사항): 패턴을 자동으로 확인할 수 없는 경우 StorageGRID에서 기본 사용자 이름 패턴을 사용해야 합니다.

StorageGRID가 서비스 계정에 바인딩할 수 없는 경우 사용자가 로그인할 수 있으므로 \* 사용자 이름 형식 바인딩 \* 을 제공하는 것이 좋습니다.

다음 패턴 중 하나를 입력합니다.

- **UserPrincipalName 패턴(AD 및 Entra ID):** [USERNAME]@example.com
- 하위 수준 로그온 이름 패턴(AD 및 Entra ID): example\[USERNAME]

- \* 고유 이름 패턴 \*: CN=[USERNAME], CN=Users, DC=example, DC=com

[UserName] \* 을 서면 그대로 포함합니다.

## 6. TLS(전송 계층 보안) 섹션에서 보안 설정을 선택합니다.

- **STARTTLS** 사용: STARTTLS를 사용하여 LDAP 서버와의 통신을 보호합니다. 이 옵션은 Active Directory, OpenLDAP 또는 기타에 권장되지만 Microsoft Entra ID에서는 지원되지 않습니다.
- **LDAPS** 사용: LDAPS(SSL을 통한 LDAP) 옵션은 TLS를 사용하여 LDAP 서버에 연결합니다. Microsoft Entra ID의 경우 이 옵션을 선택해야 합니다.
- **TLS를 사용하지 마세요**: StorageGRID 시스템과 LDAP 서버 간의 네트워크 트래픽이 보호되지 않습니다. 이 옵션은 Microsoft Entra ID에서 지원되지 않습니다.



Active Directory 서버에서 LDAP 서명을 적용하는 경우 **TLS** 사용 안 함 옵션은 지원되지 않습니다. STARTTLS 또는 LDAPS를 사용해야 합니다.

## 7. STARTTLS 또는 LDAPS를 선택한 경우 연결 보안에 사용되는 인증서를 선택합니다.

- \* 운영 체제 CA 인증서 사용 \*: 운영 체제에 설치된 기본 그리드 CA 인증서를 사용하여 연결을 보호합니다.
- \* 사용자 지정 CA 인증서 사용 \*: 사용자 지정 보안 인증서를 사용합니다.

이 설정을 선택한 경우 사용자 지정 보안 인증서를 복사하여 CA 인증서 텍스트 상자에 붙여 넣습니다.

## 연결을 테스트하고 구성을 저장합니다

모든 값을 입력한 후 구성을 저장하기 전에 연결을 테스트해야 합니다. StorageGRID는 LDAP 서버에 대한 연결 설정과 바인딩 사용자 이름 형식(제공한 경우)을 확인합니다.

## 단계

1. Test connection \* 을 선택합니다.
2. 바인드 사용자 이름 형식을 제공하지 않은 경우:
  - 연결 설정이 유효한 경우 "Test connection successful(연결 테스트 성공)" 메시지가 나타납니다. Save \* 를 선택하여 설정을 저장합니다.
  - 연결 설정이 잘못된 경우 "테스트 연결을 설정할 수 없습니다." 메시지가 나타납니다. 닫기 \* 를 선택합니다. 그런 다음 문제를 해결하고 연결을 다시 테스트합니다.
3. 바인딩 사용자 이름 형식을 제공한 경우 유효한 통합 사용자의 사용자 이름과 암호를 입력합니다.

예를 들어 사용자 이름과 암호를 입력합니다. @ 또는 / 같은 특수 문자를 사용자 이름에 포함하지 마십시오.

### Test Connection

To test the connection and the bind username format, enter the username and password of a federated user. For example, enter your own federated username and password. The test values are not saved.

**Test username**

The username of a federated user.

**Test password**

?

Cancel
Test Connection

- 연결 설정이 유효한 경우 "Test connection successful(연결 테스트 성공)" 메시지가 나타납니다. Save \* 를 선택하여 설정을 저장합니다.
- 연결 설정, 바인딩 사용자 이름 형식 또는 테스트 사용자 이름과 암호가 올바르지 않으면 오류 메시지가 나타납니다. 모든 문제를 해결하고 연결을 다시 테스트합니다.

#### ID 소스와 강제로 동기화합니다

StorageGRID 시스템은 ID 소스에서 페더레이션 그룹과 사용자를 정기적으로 동기화합니다. 사용자 권한을 최대한 빨리 설정하거나 제한하려는 경우 동기화를 강제로 시작할 수 있습니다.

#### 단계

1. ID 페더레이션 페이지로 이동합니다.
2. 페이지 맨 위에서 \* 서버 동기화 \* 를 선택합니다.

동기화 프로세스는 환경에 따라 다소 시간이 걸릴 수 있습니다.



ID 소스에서 페더레이션 그룹과 사용자를 동기화하는 데 문제가 있는 경우 \* ID 페더레이션 동기화 실패 \* 경고가 트리거됩니다.

#### ID 페더레이션을 비활성화합니다

그룹 및 사용자에 대한 ID 페더레이션을 일시적 또는 영구적으로 비활성화할 수 있습니다. ID 페더레이션이 비활성화되면 StorageGRID 와 ID 소스 간에 통신이 이루어지지 않습니다. 하지만 구성한 설정은 모두 유지되므로 나중에 ID 페더레이션을 쉽게 다시 활성화할 수 있습니다.

#### 이 작업에 대해

ID 페더레이션을 사용하지 않도록 설정하기 전에 다음 사항을 확인해야 합니다.

- 페더레이션 사용자는 로그인할 수 없습니다.
- 현재 로그인한 페더레이션 사용자는 세션이 만료될 때까지 StorageGRID 시스템에 대한 액세스 권한을 유지하지만 세션이 만료된 후에는 로그인할 수 없습니다.
- StorageGRID 시스템과 ID 소스 간의 동기화가 발생하지 않으며, 동기화되지 않은 계정에 대한 경고는 발생하지

않습니다.

- 단일 로그인(SSO) 상태가 활성화 또는 샌드박스 모드\*인 경우 \*ID 페더레이션 사용 확인란은 비활성화됩니다. ID 페더레이션을 비활성화하려면 Single Sign-on 페이지에서 SSO 상태를 \*비활성화\*해야 합니다. 보다 "[SSO\(Single Sign-On\)를 비활성화합니다](#)".

단계

1. ID 페더레이션 페이지로 이동합니다.
2. ID 페더레이션 사용 \* 확인란의 선택을 취소합니다.

**OpenLDAP** 서버 구성 지침

OpenLDAP 서버를 ID 페더레이션에 사용하려면 OpenLDAP 서버에서 특정 설정을 구성해야 합니다.

 Active Directory 또는 Microsoft Entra ID가 아닌 ID 소스의 경우 StorageGRID 외부에서 비활성화된 사용자의 S3 액세스를 자동으로 차단하지 않습니다. S3 액세스를 차단하려면 해당 사용자의 S3 키를 삭제하거나 모든 그룹에서 사용자를 제거하세요.

**MemberOf** 및 구체화 오버레이

MemberOf 및 구체화 오버레이를 활성화해야 합니다. 자세한 내용은 [에서 역방향 그룹 구성원 유지 관리에 대한 지침을 참조하십시오](http://www.openldap.org/doc/admin24/index.html) ["OpenLDAP 설명서: 버전 2.4 관리자 가이드"].

인덱싱

지정된 인덱스 키워드를 사용하여 다음 OpenLDAP 속성을 구성해야 합니다.

- olcDbIndex: objectClass eq
- olcDbIndex: uid eq,pres,sub
- olcDbIndex: cn eq,pres,sub
- olcDbIndex: entryUUID eq

또한 최적의 성능을 위해 사용자 이름 도움말에 언급된 필드를 인덱싱해야 합니다.

에서 역방향 그룹 구성원 유지 관리에 대한 정보를 ["OpenLDAP 설명서: 버전 2.4 관리자 가이드"](#) 참조하십시오.

테넌트 그룹을 관리합니다

**S3** 테넌트에 대한 그룹을 생성합니다

통합 그룹을 가져오거나 로컬 그룹을 생성하여 S3 사용자 그룹에 대한 권한을 관리할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#) 있습니다.
- 이 있는 사용자 그룹에 속해 ["루트 액세스 권한"](#) 있습니다.
- 통합 그룹을 가져오려는 경우 이 ["ID 페더레이션을 구성했습니다"](#) 있으며 페더레이션 그룹이 구성된 ID 원본에 이미 있습니다.

- 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있으면 에 대한 워크플로 및 고려 사항을 검토한 후 "테넌트 그룹 및 사용자를 클론 생성합니다" 테넌트의 소스 그리드에 로그인됩니다.

그룹 생성 마법사에 액세스합니다

첫 번째 단계로 그룹 생성 마법사에 액세스합니다.

단계

1. 액세스 관리 > \*그룹\*을 선택합니다.
2. 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 이 그리드에 생성된 새 그룹이 연결의 다른 그리드에 있는 동일한 테넌트에 복제됨을 나타내는 파란색 배너가 나타나는지 확인합니다. 이 배너가 나타나지 않으면 테넌트의 대상 그리드에 로그인되었을 수 있습니다.

## Groups

Create and manage local and federated groups. Set group permissions to control access to specific pages and features.

3. Create group \* 을 선택합니다.

그룹 유형을 선택합니다

로컬 그룹을 생성하거나 통합 그룹을 가져올 수 있습니다.

단계

1. 로컬 그룹을 생성하려면 \* Local group \* 탭을 선택하고, 이전에 구성된 ID 소스에서 그룹을 가져오려면 \* Federated group \* 탭을 선택합니다.

StorageGRID 시스템에서 SSO(Single Sign-On)를 사용하는 경우 로컬 그룹에 속한 사용자는 그룹 권한에 따라 클라이언트 애플리케이션을 사용하여 테넌트의 리소스를 관리할 수 있지만 테넌트 관리자에 로그인할 수 없습니다.

2. 그룹의 이름을 입력합니다.

◦ \* 로컬 그룹 \*: 표시 이름과 고유 이름을 모두 입력합니다. 나중에 표시 이름을 편집할 수 있습니다.



테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 대상 그리드에 해당 테넌트에 대해 동일한 \* 고유 이름 \* 이 이미 있으면 클론 생성 오류가 발생합니다.

◦ \* 페더레이션 그룹 \*: 고유한 이름을 입력합니다. Active Directory의 경우 고유 이름은 속성과 연결된 sAMAccountName 이름입니다. OpenLDAP의 경우 고유 이름은 속성과 연결된 uid 이름입니다.

3. Continue \* 를 선택합니다.

## 그룹 권한을 관리합니다

그룹 권한은 테넌트 관리자 및 테넌트 관리 API에서 사용자가 수행할 수 있는 작업을 제어합니다.

### 단계

- 액세스 모드 \* 의 경우 다음 중 하나를 선택합니다.

- \* 읽기-쓰기 \* (기본값): 사용자는 테넌트 관리자에 로그인하여 테넌트 구성을 관리할 수 있습니다.
- \* 읽기 전용 \*: 사용자는 설정 및 기능만 볼 수 있습니다. 테넌트 관리자 또는 테넌트 관리 API에서 작업을 변경하거나 수행할 수 없습니다. 로컬 읽기 전용 사용자는 자신의 암호를 변경할 수 있습니다.



사용자가 여러 그룹에 속해 있고 모든 그룹이 읽기 전용으로 설정되어 있는 경우 사용자는 선택한 모든 설정 및 기능에 대해 읽기 전용 권한을 갖게 됩니다.

- 이 그룹에 대한 권한을 하나 이상 선택합니다.

을 "테넌트 관리 권한" 참조하십시오.

- Continue \* 를 선택합니다.

## S3 그룹 정책을 설정합니다

그룹 정책은 사용자가 가질 S3 액세스 권한을 결정합니다.

### 단계

- 이 그룹에 사용할 정책을 선택합니다.

그룹 정책	설명
S3 액세스 없음	기본값. 버킷 정책을 통해 액세스 권한이 부여되지 않은 한 이 그룹의 사용자는 S3 리소스에 액세스할 수 없습니다. 이 옵션을 선택하면 루트 사용자만 기본적으로 S3 리소스에 액세스할 수 있습니다.
읽기 전용 액세스	이 그룹의 사용자는 S3 리소스에 읽기 전용 권한을 가집니다. 예를 들어 이 그룹의 사용자는 개체를 나열하고 개체 데이터, 메타데이터 및 태그를 읽을 수 있습니다. 이 옵션을 선택하면 읽기 전용 그룹 정책의 JSON 문자열이 텍스트 상자에 나타납니다. 이 문자열을 편집할 수 없습니다.
전체 액세스	이 그룹의 사용자는 버킷을 포함하여 S3 리소스에 대한 모든 액세스 권한을 가집니다. 이 옵션을 선택하면 전체 액세스 그룹 정책의 JSON 문자열이 텍스트 상자에 나타납니다. 이 문자열을 편집할 수 없습니다.
랜섬웨어 완화	이 예제 정책은 이 테넌트의 모든 버킷에 적용됩니다. 이 그룹의 사용자는 일반적인 작업을 수행할 수 있지만 개체 버전 관리가 활성화된 버킷에서 개체를 영구적으로 삭제할 수는 없습니다.  Manage All Bucket * 권한이 있는 테넌트 관리자 사용자는 이 그룹 정책을 재정의할 수 있습니다. 모든 버킷 관리 권한을 신뢰할 수 있는 사용자로 제한하고 가능한 경우 MFA(Multi-Factor Authentication)를 사용합니다.

그룹 정책	설명
맞춤형	그룹의 사용자에게는 텍스트 상자에 지정한 사용 권한이 부여됩니다.

2. 사용자 정의 \* 를 선택한 경우 그룹 정책을 입력합니다. 각 그룹 정책은 크기 제한이 5,120바이트입니다. 올바른 JSON 형식 문자열을 입력해야 합니다.

언어 구문 및 예제를 포함한 그룹 정책에 대한 자세한 내용은 [그룹 정책의 예](#)를 참조하십시오.

3. 로컬 그룹을 만드는 경우 \* 계속 \* 을 선택합니다. 통합 그룹을 만드는 경우 \* 그룹 생성 \* 및 \* 마침 \* 을 선택합니다.

### 사용자 추가(로컬 그룹만 해당)

사용자를 추가하지 않고 그룹을 저장하거나 이미 존재하는 로컬 사용자를 선택적으로 추가할 수 있습니다.



테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 소스 그리드에 로컬 그룹을 생성할 때 선택한 모든 사용자는 대상 그리드에 그룹이 클론 생성될 때 포함되지 않습니다. 따라서 그룹을 만들 때 사용자를 선택하지 마십시오. 대신 사용자를 생성할 때 그룹을 선택합니다.

### 단계

- 필요에 따라 이 그룹에 대해 하나 이상의 로컬 사용자를 선택합니다.
- Create group \* 과 \* Finish \* 를 선택합니다.

생성한 그룹이 그룹 목록에 나타납니다.

테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 테넌트의 소스 그리드에 있는 경우 새 그룹이 테넌트의 대상 그리드에 복제됩니다. \* 성공 \* 은 그룹 세부 정보 페이지의 개요 섹션에 \* 클론 생성 상태 \* 로 표시됩니다.

### 테넌트 관리 권한

테넌트 그룹을 생성하기 전에 해당 그룹에 할당할 권한을 고려하십시오. 테넌트 관리 권한은 사용자가 테넌트 관리자 또는 테넌트 관리 API를 사용하여 수행할 수 있는 작업을 결정합니다. 사용자는 하나 이상의 그룹에 속할 수 있습니다. 사용자가 여러 그룹에 속한 경우 권한은 누적됩니다.

테넌트 관리자에 로그인하거나 테넌트 관리 API를 사용하려면 사용자가 하나 이상의 권한이 있는 그룹에 속해야 합니다. 로그인할 수 있는 모든 사용자는 다음 작업을 수행할 수 있습니다.

- 대시보드 보기
- 자신의 암호 변경(로컬 사용자의 경우)

모든 권한에 대해 그룹의 액세스 모드 설정은 사용자가 설정을 변경하고 작업을 수행할 수 있는지 또는 관련 설정과 기능만 볼 수 있는지 여부를 결정합니다.



사용자가 여러 그룹에 속해 있고 모든 그룹이 읽기 전용으로 설정되어 있는 경우 사용자는 선택한 모든 설정 및 기능에 대해 읽기 전용 권한을 갖게 됩니다.

그룹에 다음과 같은 권한을 할당할 수 있습니다.

권한	설명	세부 정보
루트 액세스	테넌트 관리자 및 테넌트 관리 API에 대한 전체 액세스를 제공합니다.	
자체 S3 자격 증명을 관리합니다	사용자가 자신의 S3 액세스 키를 생성하고 제거할 수 있습니다.	이 권한이 없는 사용자는 * storage(S3) * > * My S3 access keys * 메뉴 옵션을 볼 수 없습니다.
모든 버킷을 봅니다	사용자가 모든 버킷과 버킷 구성을 볼 수 있습니다.	모든 버킷 보기 또는 모든 버킷 관리 권한이 없는 사용자에게는 * Bucket * 메뉴 옵션이 표시되지 않습니다.  이 권한은 모든 버킷 관리 권한으로 대체됩니다. 이는 S3 클라이언트나 S3 콘솔에서 사용하는 S3 버킷이나 그룹 정책에는 영향을 미치지 않습니다.
모든 버킷을 관리합니다	사용자가 테넌트 관리자와 테넌트 관리 API를 사용하여 S3 버킷을 생성 및 삭제하고, S3 버킷이나 그룹 정책에 관계없이 테넌트 계정의 모든 S3 버킷에 대한 설정을 관리할 수 있도록 합니다.	모든 버킷 보기 또는 모든 버킷 관리 권한이 없는 사용자에게는 * Bucket * 메뉴 옵션이 표시되지 않습니다.  이 권한은 모든 버킷 보기 권한보다 우선합니다. 이는 S3 클라이언트나 S3 콘솔에서 사용하는 S3 버킷이나 그룹 정책에는 영향을 미치지 않습니다.
엔드포인트 관리	사용자가 테넌트 관리자 또는 테넌트 관리 API를 사용하여 플랫폼 서비스 엔드포인트를 생성하거나 편집할 수 있습니다. 이 엔드포인트는 StorageGRID 플랫폼 서비스의 대상으로 사용됩니다.	이 권한이 없는 사용자는 * 플랫폼 서비스 끝점 * 메뉴 옵션이 표시되지 않습니다.
S3 콘솔 탭을 사용합니다	모든 버킷 보기 또는 모든 버킷 관리 권한과 함께 사용하면 사용자가 버킷에 대한 세부 정보 페이지의 S3 콘솔 탭에서 오브젝트를 보고 관리할 수 있습니다.	

그룹을 관리합니다

필요에 따라 테넌트 그룹을 관리하여 그룹 보기, 편집 또는 복제 등을 수행할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["루트 액세스 권한"](#)입니다.

그룹을 보거나 편집합니다

각 그룹의 기본 정보와 세부 정보를 보고 편집할 수 있습니다.

단계

- 액세스 관리 > \*그룹\*을 선택합니다.
- 그룹 페이지에 제공된 정보를 검토하여 이 테넌트 계정의 모든 로컬 및 통합 그룹에 대한 기본 정보를 나열합니다.

테넌트 계정에 \* Use GRID Federation Connection \* 권한이 있고 테넌트의 소스 그리드에서 그룹을 보고 있는 경우:

- 그룹을 편집하거나 제거하면 변경 내용이 다른 그리드와 동기화되지 않음을 나타내는 배너 메시지가 표시됩니다.
- 필요에 따라 대상 그리드의 테넌트에 그룹이 복제되지 않았는지 여부를 나타내는 배너 메시지가 표시됩니다. 하지만 [그룹 클론을 재시도하십시오](#) 실패했어요.

- 그룹 이름을 변경하려면:
  - 그룹의 확인란을 선택합니다.
  - Actions \* > \* Edit group name \* 을 선택합니다.
  - 새 이름을 입력합니다.
  - 변경 사항 저장 \* 을 선택합니다
- 자세한 내용을 보거나 추가로 편집하려면 다음 중 하나를 수행합니다.
  - 그룹 이름을 선택합니다.
  - 그룹의 확인란을 선택하고 \* Actions \* > \* View group details \* 를 선택합니다.
- 각 그룹에 대해 다음 정보를 보여 주는 개요 섹션을 검토합니다.
  - 표시 이름
  - 고유한 이름입니다
  - 유형
  - 액세스 모드
  - 권한
  - S3 정책
  - 이 그룹의 사용자 수입니다
  - 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 테넌트의 소스 격자에서 그룹을 보고 있는 경우 추가 필드:
    - 복제 상태, \* 성공 \* 또는 \* 실패 \*
    - 이 그룹을 편집하거나 삭제하면 변경 내용이 다른 눈금과 동기화되지 않음을 나타내는 파란색 배너입니다.
- 필요에 따라 그룹 설정을 편집합니다. 참조하다 "[S3 테넌트에 대한 그룹을 생성합니다](#)" 입력해야 할 내용에 대한 자세한 내용은 다음을 참조하세요.
  - 개요 섹션에서 이름이나 편집 아이콘을 선택하여 표시 이름을 변경합니다 .
  - 그룹 권한 \* 탭에서 권한을 업데이트하고 \* 변경 사항 저장 \* 을 선택합니다.
  - 그룹 정책 \* 탭에서 변경을 수행하고 \* 변경 사항 저장 \* 을 선택합니다.

필요에 따라 다른 S3 그룹 정책을 선택하거나 사용자 지정 정책에 대한 JSON 문자열을 입력합니다.
- 그룹에 기존 로컬 사용자를 하나 이상 추가하려면 다음을 수행합니다.

- a. 사용자 탭을 선택합니다.

- b. 사용자 추가 \* 를 선택합니다.

- c. 추가할 기존 사용자를 선택하고 \* 사용자 추가 \* 를 선택합니다.

오른쪽 위에 성공 메시지가 나타납니다.

#### 8. 그룹에서 로컬 사용자 제거하기:

- 사용자 탭을 선택합니다.
- 사용자 제거 \* 를 선택합니다.
- 제거할 사용자를 선택하고 \* 사용자 제거 \* 를 선택합니다.

오른쪽 위에 성공 메시지가 나타납니다.

#### 9. 변경한 각 섹션에 대해 \* 변경 사항 저장 \* 을 선택했는지 확인합니다.

그룹이 중복되었습니다

기존 그룹을 복제하여 새 그룹을 더 빠르게 만들 수 있습니다.



테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 테넌트의 소스 그리드에서 그룹을 복제하는 경우 복제된 그룹은 테넌트의 대상 그리드에 복제됩니다.

단계

- 액세스 관리 > \*그룹\*을 선택합니다.
- 복제할 그룹의 확인란을 선택합니다.
- Actions \* > \* Duplicate group \* 을 선택합니다.
- 보다 "S3 테넌트에 대한 그룹을 생성합니다" 입력해야 할 내용에 대한 자세한 내용은 다음을 참조하세요.
- Create group \* 을 선택합니다.

그룹 클론을 다시 시도하십시오

실패한 클론을 재시도하려면 다음을 수행합니다.

- 그룹 이름 아래에 \_(클론 생성 실패) \_ 을(를) 나타내는 각 그룹을 선택합니다.
- Actions \* > \* Clone groups \* 를 선택합니다.
- 클론 생성 중인 각 그룹의 세부 정보 페이지에서 클론 작업의 상태를 봅니다.

자세한 내용은 [를 참조하십시오"클론 테넌트 그룹 및 사용자".](#)

하나 이상의 그룹을 삭제합니다

하나 이상의 그룹을 삭제할 수 있습니다. 삭제된 그룹에만 속하는 사용자는 더 이상 테넌트 관리자에 로그인하거나 테넌트 계정을 사용할 수 없습니다.



테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 그룹을 삭제하는 경우 StorageGRID는 다른 그리드에서 해당 그룹을 삭제하지 않습니다. 이 정보를 동기화해야 하는 경우 두 그리드에서 동일한 그룹을 삭제해야 합니다.

단계

- 액세스 관리 > \*그룹\*을 선택합니다.
  - 삭제할 각 그룹의 확인란을 선택합니다.
  - Actions \* > \* Delete group \* 또는 \* Actions \* > \* Delete groups \* 를 선택합니다.
- 확인 대화 상자가 나타납니다.
- 그룹 삭제 \* 또는 \* 그룹 삭제 \* 를 선택합니다.

## AssumeRole 설정

시작하기 전에

AssumeRole을 설정하려면 관리자여야 합니다.

이 작업에 대해

AssumeRole을 설정하려면 해당 그룹이 아직 없으면 맙을 대상 그룹을 만듭니다. 그룹의 S3 정책을 편집하여 이 그룹을 가정하여 허용되는 작업을 지정합니다. AssumeRole API를 사용하여 그룹을 맡을 수 있는 신뢰할 수 있는 사용자를 지정하기 위해 그룹의 S3 신뢰 정책을 편집합니다.

이 그룹을 가정하여 생성된 임시 보안 자격 증명은 제한된 기간 동안만 유효합니다. 세션은 15분에서 12시간까지이며, 기본 세션은 1시간입니다. 그룹의 S3 신뢰 정책에서 사용자를 제거하면 해당 사용자는 더 이상 이 그룹을 맡을 수 없습니다.

단계

- 액세스 관리 > \*그룹\*을 선택합니다.
- 그룹 이름을 클릭하세요.
- S3** 신뢰 정책 탭을 선택합니다.
- AssumeRole을 수행할 수 있는 사용자 목록을 포함하여 S3 신뢰 정책을 추가합니다.
- 변경 내용 저장 \* 을 선택합니다.
- S3** 그룹 정책 탭을 선택합니다.
- 이 그룹의 S3 신뢰 정책에 추가된 신뢰할 수 있는 사용자에 대해서만 필요한 S3 작업을 지정하도록 S3 정책을 편집합니다.
- 변경 내용 저장 \* 을 선택합니다.

## AssumeRole S3 신뢰 정책의 예

```
{  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Action": "sts:AssumeRole",  
      "Principal": {  
        "AWS": [  
          "urn:sgws:identity::1234567890:user/user1",  
          "arn:aws:iam::1234567890:user/user2"  
        ]  
      }  
    }  
  ]  
}
```

구성이 완료되면 S3 신뢰 정책에 나열된 사용자는 AssumeRole을 수행하고 자격 증명을 받을 수 있습니다. 최종 권한은 그룹 정책, 버킷 정책, 세션 정책에 따라 결정됩니다. 자세한 내용은 다음을 참조하십시오. ["접근 정책 사용"](#).

## 사용자 관리

로컬 사용자를 만들고 로컬 그룹에 할당하여 해당 사용자가 액세스할 수 있는 기능을 결정할 수 있습니다. 연합 사용자를 가져올 수도 있습니다. 테넌트 관리자에는 "root"라는 이름의 미리 정의된 로컬 사용자가 하나 포함됩니다. 로컬 사용자를 추가하거나 제거할 수 있지만 루트 사용자는 제거할 수 없습니다.



StorageGRID 시스템에서 SSO(Single Sign-On)가 활성화된 경우 로컬 사용자는 그룹 권한에 따라 클라이언트 애플리케이션을 사용하여 테넌트의 리소스에 액세스할 수 있지만 테넌트 관리자 또는 테넌트 관리 API에 로그인할 수 없습니다.

## 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["루트 액세스 권한"](#)입니다.
- 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있으면 에 대한 워크플로 및 고려 사항을 검토한 후 ["테넌트 그룹 및 사용자를 클론 생성합니다"](#) 테넌트의 소스 그리드에 로그인됩니다.

## 로컬 사용자를 생성합니다

로컬 사용자를 만들어 하나 이상의 로컬 그룹에 할당하여 액세스 권한을 제어할 수 있습니다.

그룹에 속하지 않은 S3 사용자는 관리 권한이나 S3 그룹 정책이 적용되지 않습니다. 이러한 사용자는 버킷 정책을 통해 S3 버킷 액세스가 부여될 수 있습니다.

사용자 생성 마법사에 액세스합니다

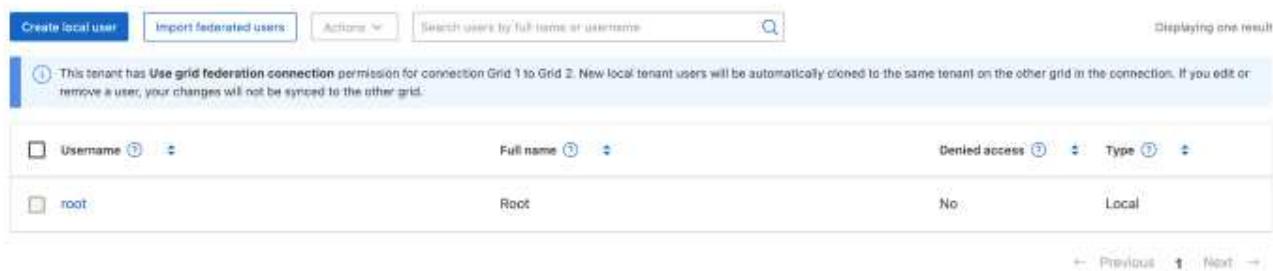
단계

- 액세스 관리 > \*사용자\*를 선택합니다.

테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우 파란색 배너는 해당 테넌트의 소스 그리드임을 나타냅니다. 이 그리드에서 만드는 모든 로컬 사용자는 연결의 다른 그리드에 복제됩니다.

## Users

View local and federated users. Edit properties and group membership of local users.



The screenshot shows a user management interface with the following details:

- Header buttons: Create local user, Import federated users, Actions, and a search bar.
- Message bar: "This tenant has Use grid federation connection permission for connection Grid 1 to Grid 2. New local tenant users will be automatically cloned to the same tenant on the other grid in the connection. If you edit or remove a user, your changes will not be synced to the other grid."
- Table headers: Username, Full name, Denied access, and Type.
- Table data: One row for 'root' with 'Root' in the Full name column, 'No' in Denied access, and 'Local' in Type.
- Page navigation: Previous and Next buttons.

- 사용자 생성 \* 을 선택합니다.

자격 증명을 입력합니다

단계

- 사용자 자격 증명 입력 \* 단계에 대해 다음 필드를 입력합니다.

필드에 입력합니다	설명
전체 이름	이 사용자의 전체 이름(예: 사용자의 이름 및 성 또는 응용 프로그램의 이름).
사용자 이름	이 사용자가 로그인하는 데 사용할 이름입니다. 사용자 이름은 고유해야 하며 변경할 수 없습니다. <ul style="list-style-type: none"><li>참고 *: 테넌트 계정에 * 그리드 페더레이션 연결 사용 * 권한이 있는 경우 대상 그리드에 해당 테넌트에 대해 동일한 * 사용자 이름 * 이 이미 있으면 클론 생성 오류가 발생합니다.</li></ul>
암호 및 암호 확인	사용자가 로그인할 때 처음 사용할 암호입니다.
액세스를 거부합니다	이 사용자가 하나 이상의 그룹에 속해 있더라도 테넌트 계정에 로그인하지 못하도록 하려면 * 예 * 를 선택합니다. 예를 들어, * 예 * 를 선택하여 사용자의 로그인 기능을 일시적으로 중단시킵니다.

- Continue \* 를 선택합니다.

그룹에 할당합니다

단계

1. 사용자를 하나 이상의 로컬 그룹에 할당하여 수행할 수 있는 작업을 결정합니다.

그룹에 사용자를 할당하는 것은 선택 사항입니다. 원하는 경우 그룹을 만들거나 편집할 때 사용자를 선택할 수 있습니다.

그룹에 속하지 않은 사용자에게는 관리 권한이 없습니다. 권한은 누적됩니다. 사용자는 자신이 속한 모든 그룹에 대한 모든 권한을 갖게 됩니다. [을 "테넌트 관리 권한" 참조하십시오.](#)

2. 사용자 생성 \* 을 선택합니다.

테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 테넌트의 소스 그리드에 있는 경우 새 로컬 사용자는 테넌트의 대상 그리드에 복제됩니다. \* 성공 \* 은 사용자 세부 정보 페이지의 개요 섹션에 \* 클론 생성 상태 \*로 표시됩니다.

3. 사용자 페이지로 돌아가려면 \* 마침 \* 을 선택합니다.

로컬 사용자를 보거나 편집합니다

단계

1. 액세스 관리 > \*사용자\*를 선택합니다.
2. 사용자 페이지에 제공된 정보를 검토하여 이 테넌트 계정에 대한 모든 로컬 및 통합 사용자의 기본 정보를 나열합니다.

테넌트 계정에 \* Use grid 페더레이션 연결 \* 권한이 있고 테넌트의 소스 그리드에서 사용자를 보고 있는 경우:

- 배너 메시지는 사용자를 편집하거나 제거하면 변경 내용이 다른 그리드와 동기화되지 않음을 나타냅니다.
- 필요에 따라 배너 메시지는 사용자가 대상 그리드의 테넌트에 복제되지 않았는지 여부를 나타냅니다. 할 수 [실패한 사용자 클론을 재시도합니다](#) 있습니다.

3. 사용자의 전체 이름을 변경하려면:

- a. 사용자의 확인란을 선택합니다.
- b. 작업 \* > \* 전체 이름 편집 \* 을 선택합니다.
- c. 새 이름을 입력합니다.
- d. 변경 사항 저장 \* 을 선택합니다

4. 자세한 내용을 보거나 추가로 편집하려면 다음 중 하나를 수행합니다.

- 사용자 이름을 선택합니다.
- 사용자의 확인란을 선택하고 \* Actions \* > \* View user details \* 를 선택합니다.

5. 각 사용자에 대해 다음 정보를 보여 주는 개요 섹션을 검토합니다.

- 전체 이름
- 사용자 이름
- 사용자 유형
- 액세스가 거부되었습니다

- 액세스 모드
- 그룹 구성원 자격
- 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 테넌트의 소스 격자에서 사용자를 보는 경우 추가 필드:
  - 복제 상태, \* 성공 \* 또는 \* 실패 \*
  - 이 사용자를 편집하면 변경 내용이 다른 눈금과 동기화되지 않음을 나타내는 파란색 배너입니다.

6. 필요에 따라 사용자 설정을 편집합니다. 입력할 내용에 대한 자세한 내용은 [을 로컬 사용자를 생성합니다](#) 참조하십시오.

a. 개요 섹션에서 이름 또는 편집 아이콘을 선택하여 전체 이름을 변경합니다 .

사용자 이름은 변경할 수 없습니다.

b. 암호 \* 탭에서 사용자 암호를 변경하고 \* 변경 사항 저장 \* 을 선택합니다.

c. 액세스 \* 탭에서 \* 아니요 \* 를 선택하여 사용자가 로그인할 수 있도록 하거나 \* 예 \* 를 선택하여 사용자가 로그인할 수 없도록 합니다. 그런 다음 \* 변경 사항 저장 \* 을 선택합니다.

d. 액세스 키 \* 탭에서 \* 키 생성 \* 을 선택하고 의 지침을 ["다른 사용자의 S3 액세스 키 생성"](#) 따릅니다.

e. 그룹 \* 탭에서 \* 그룹 편집 \* 을 선택하여 사용자를 그룹에 추가하거나 그룹에서 제거합니다. 그런 다음 \* 변경 사항 저장 \* 을 선택합니다.

7. 변경한 각 섹션에 대해 \* 변경 사항 저장 \* 을 선택했는지 확인합니다.

페더레이션 사용자 가져오기

최대 100명까지의 연합 사용자를 하나 이상 사용자 페이지로 직접 가져올 수 있습니다.

단계

1. 액세스 관리 > \*사용자\*를 선택합니다.
2. \*연합 사용자 가져오기\*를 선택합니다.
3. 하나 이상의 페더레이션 사용자에 대한 UUID 또는 사용자 이름을 입력하세요.

여러 항목이 있는 경우 각 UUID 또는 사용자 이름을 새 줄에 추가합니다.

4. \*가져오기\*를 선택하세요.

한 명 이상의 사용자에 대해 사용자 필드로의 가져오기가 실패하면 다음 단계를 수행하세요.

- a. \*사용자를 가져오지 않음\*을 확장하고 \*사용자 복사\*를 선택합니다.
- b. 이전\*을 선택하고 복사한 사용자를 \*연합 사용자 가져오기 대화 상자에 붙여넣어 가져오기를 다시 시도합니다.

연합 사용자 가져오기 대화 상자를 닫으면 성공적으로 가져온 사용자의 연합 사용자 정보가 사용자 페이지에 표시됩니다.

로컬 사용자를 복제하십시오

로컬 사용자를 복제하면 새 사용자를 보다 빠르게 만들 수 있습니다.



테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 테넌트의 소스 그리드에서 사용자를 복제하면 복제된 사용자는 테넌트의 대상 그리드에 복제됩니다.

#### 단계

1. 액세스 관리 > \*사용자\*를 선택합니다.
2. 복제할 사용자의 확인란을 선택합니다.
3. Actions \* > \* Duplicate user \* 를 선택합니다.
4. 입력할 내용에 대한 자세한 내용은 [로컬 사용자를 생성합니다](#) 참조하십시오.
5. 사용자 생성 \* 을 선택합니다.

사용자 클론을 다시 시도하십시오

실패한 클론을 재시도하려면 다음을 수행합니다.

1. 사용자 이름 아래에 \_ (클론 생성 실패) \_ 을(를) 나타내는 각 사용자를 선택합니다.
2. Actions \* > \* Clone users \* 를 선택합니다.
3. 클론 복제할 각 사용자의 세부 정보 페이지에서 클론 작업의 상태를 확인합니다.

자세한 내용은 [클론 테넌트 그룹 및 사용자](#)를 참조하십시오.

하나 이상의 로컬 사용자를 삭제합니다

StorageGRID 테넌트 계정에 더 이상 액세스할 필요가 없는 하나 이상의 로컬 사용자를 영구적으로 삭제할 수 있습니다.



테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있고 로컬 사용자를 삭제하는 경우 StorageGRID는 다른 그리드에서 해당 사용자를 삭제하지 않습니다. 이 정보를 동기화해야 하는 경우 두 그리드에서 동일한 사용자를 삭제해야 합니다.



통합 사용자를 삭제하려면 통합 ID 소스를 사용해야 합니다.

#### 단계

1. 액세스 관리 > \*사용자\*를 선택합니다.
2. 삭제할 각 사용자에 대한 확인란을 선택합니다.
3. Actions \* > \* Delete user \* 또는 \* Actions \* > \* Delete users \* 를 선택합니다.

확인 대화 상자가 나타납니다.

4. 사용자 삭제 \* 또는 \* 사용자 삭제 \* 를 선택합니다.

## S3 액세스 키를 관리합니다

### S3 액세스 키를 관리합니다

S3 테넌트 계정의 각 사용자는 StorageGRID 시스템에 오브젝트를 저장하고 검색하기 위한 액세스 키가 있어야 합니다. 액세스 키는 액세스 키 ID와 비밀 액세스 키로 구성됩니다.

S3 액세스 키는 다음과 같이 관리할 수 있습니다.

- 자신의 S3 자격 증명 관리 \* 권한이 있는 사용자는 자신의 S3 액세스 키를 생성하거나 제거할 수 있습니다.
- 루트 액세스 \* 권한이 있는 사용자는 S3 루트 계정 및 다른 모든 사용자의 액세스 키를 관리할 수 있습니다. 루트 액세스 키는 버킷 정책에 의해 명시적으로 비활성화되지 않는 한 테넌트의 모든 버킷과 객체에 대한 전체 액세스를 제공합니다.

StorageGRID는 서명 버전 2 및 서명 버전 4 인증을 지원합니다. 버킷 정책에 의해 명시적으로 활성화되지 않은 경우 교차 계정 액세스가 허용되지 않습니다.

자체 S3 액세스 키를 생성합니다

S3 테넌트를 사용 중이며 적절한 권한이 있는 경우 자체 S3 액세스 키를 생성할 수 있습니다. 버킷과 오브젝트에 액세스하려면 액세스 키가 있어야 합니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["자신의 S3 자격 증명 또는 루트 액세스 권한을 관리합니다"](#)입니다.

이 작업에 대해

하나 이상의 S3 액세스 키를 생성하여 테넌트 계정의 버킷을 생성하고 관리할 수 있습니다. 새 액세스 키를 생성한 후 새 액세스 키 ID와 비밀 액세스 키로 응용 프로그램을 업데이트합니다. 보안을 위해 필요한 것보다 더 많은 키를 생성하지 말고 사용하지 않는 키를 삭제하십시오. 하나의 키만 있고 만료되려고 하는 경우 이전 키가 만료되기 전에 새 키를 만든 다음 이전 키를 삭제합니다.

각 키에는 특정 만료 시간 또는 만료 기간이 있을 수 있습니다. 만료 시간에 대한 다음 지침을 따르십시오.

- 키의 만료 시간을 설정하여 특정 기간에 대한 액세스를 제한합니다. 만료 시간을 짧게 설정하면 액세스 키 ID 및 비밀 액세스 키가 실수로 노출되었을 경우 위험을 줄일 수 있습니다. 만료된 키는 자동으로 제거됩니다.
- 환경의 보안 위험이 낮으며 정기적으로 새 키를 만들 필요가 없는 경우 키에 대한 만료 시간을 설정할 필요가 없습니다. 나중에 새 키를 만들려면 이전 키를 수동으로 삭제합니다.



계정에 속한 S3 버킷 및 오브젝트는 테넌트 관리자에 계정에 대해 표시된 액세스 키 ID 및 비밀 액세스 키를 사용하여 액세스할 수 있습니다. 이러한 이유로 액세스 키를 암호로 보호해야 합니다. 액세스 키를 정기적으로 회전하고, 계정에서 사용되지 않는 키를 제거하며, 다른 사용자와 공유하지 마십시오.

단계

1. 스토리지(S3) \* > \* 내 액세스 키 \* 를 선택합니다.

내 액세스 키 페이지가 나타나고 기존 액세스 키가 나열됩니다.

2. Create key \* 를 선택합니다.

3. 다음 중 하나를 수행합니다.

- 만료 시간을 설정하지 않음 \* 을 선택하여 만료되지 않는 키를 생성합니다. (기본값)
- 만료 시간 설정 \* 을 선택하고 만료 날짜 및 시간을 설정합니다.



만료 날짜는 현재 날짜로부터 최대 5년일 수 있습니다. 만료 시간은 현재 시간에서 최소 1분이 될 수 있습니다.

#### 4. Create access key \* 를 선택합니다.

액세스 키 ID 및 비밀 액세스 키가 나열된 다운로드 액세스 키 대화 상자가 나타납니다.

#### 5. 액세스 키 ID와 비밀 액세스 키를 안전한 위치에 복사하거나 \* Download.csv \* 를 선택하여 액세스 키 ID와 비밀 액세스 키가 포함된 스프레드시트 파일을 저장합니다.



이 정보를 복사하거나 다운로드할 때까지 이 대화 상자를 닫지 마십시오. 대화 상자를 닫은 후에는 키를 복사하거나 다운로드할 수 없습니다.

#### 6. 마침 \* 을 선택합니다.

새 키가 내 액세스 키 페이지에 나열됩니다.

#### 7. 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우, 필요에 따라 테넌트 관리 API를 사용하여 소스 그리드의 테넌트에서 대상 그리드의 테넌트로 S3 액세스 키를 수동으로 복제합니다. 을 ["API를 사용하여 S3 액세스 키의 클론을 생성합니다" 참조하십시오.](#)

### S3 액세스 키를 봅니다

S3 테넌트를 사용 중이고 이 있는 경우 ["적절한 권한"](#) S3 액세스 키 목록을 볼 수 있습니다. 만료 시간을 기준으로 목록을 정렬할 수 있으므로 곧 만료되는 키를 확인할 수 있습니다. 필요에 따라 키를 사용할 수도 있고 ["키를 삭제합니다"](#) 더 이상 사용하지 않을 수도 ["새 키를 만듭니다"](#) 있습니다.



계정에 속한 S3 버킷 및 오브젝트는 테넌트 관리자에 계정에 대해 표시된 액세스 키 ID 및 비밀 액세스 키를 사용하여 액세스할 수 있습니다. 이러한 이유로 액세스 키를 암호로 보호해야 합니다. 액세스 키를 정기적으로 회전하고, 계정에서 사용되지 않는 키를 제거하며, 다른 사용자와 공유하지 마십시오.

### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#) 있습니다.
- 자신의 S3 자격 증명을 관리하는 사용자 그룹에 속해 ["권한"](#) 있습니다.

### 단계

1. 스토리지(S3) \* > \* 내 액세스 키 \* 를 선택합니다.
2. 내 액세스 키 페이지에서 \* 만료 시간 \* 또는 \* 액세스 키 ID \* 를 기준으로 기존 액세스 키를 정렬합니다.
3. 필요에 따라 새 키를 만들거나 더 이상 사용하지 않는 키를 삭제합니다.

기존 키가 만료되기 전에 새 키를 만들면 계정의 개체에 대한 액세스 권한을 일시적으로 잃지 않고 새 키를 사용할 수 있습니다.

만료된 키는 자동으로 제거됩니다.

## 자체 **S3** 액세스 키를 삭제합니다

S3 테넌트를 사용하는 경우 적절한 권한이 있으면 자신의 S3 액세스 키를 삭제할 수 있습니다. 액세스 키가 삭제된 후에는 더 이상 테넌트 계정의 객체와 버킷에 액세스할 수 없습니다.

### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 ["S3 자격 증명 사용 권한을 관리합니다"](#)입니다.

 계정에 속한 S3 버킷 및 오브젝트는 테넌트 관리자에 계정에 대해 표시된 액세스 키 ID 및 비밀 액세스 키를 사용하여 액세스할 수 있습니다. 이러한 이유로 액세스 키를 암호로 보호해야 합니다. 액세스 키를 정기적으로 회전하고, 계정에서 사용되지 않는 키를 제거하며, 다른 사용자와 공유하지 마십시오.

### 단계

1. 스토리지(S3) \* > \* 내 액세스 키 \* 를 선택합니다.
2. 내 액세스 키 페이지에서 제거할 각 액세스 키에 대한 확인란을 선택합니다.
3. Delete key \* 를 선택합니다.
4. 확인 대화 상자에서 \* 키 삭제 \* 를 선택합니다.

페이지의 오른쪽 상단에 확인 메시지가 나타납니다.

## 다른 사용자의 **S3** 액세스 키를 생성합니다

S3 테넌트를 사용하는 경우 적절한 권한이 있으면 버킷 및 오브젝트에 액세스해야 하는 애플리케이션 같은 다른 사용자를 위한 S3 액세스 키를 생성할 수 있습니다.

### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["루트 액세스 권한"](#)입니다.

### 이 작업에 대해

하나 이상의 다른 사용자를 위한 S3 액세스 키를 생성하여 해당 테넌트 계정에 대한 버킷을 생성하고 관리할 수 있습니다. 새 액세스 키를 생성한 후 새 액세스 키 ID와 비밀 액세스 키로 응용 프로그램을 업데이트합니다. 보안을 위해 사용자 요구 사항보다 더 많은 키를 생성하지 말고 사용하지 않는 키를 삭제하십시오. 하나의 키만 있고 만료되려고 하는 경우 이전 키가 만료되기 전에 새 키를 만든 다음 이전 키를 삭제합니다.

각 키에는 특정 만료 시간 또는 만료 기간이 있을 수 있습니다. 만료 시간에 대한 다음 지침을 따르십시오.

- 키의 만료 시간을 설정하여 사용자의 액세스를 특정 기간으로 제한합니다. 만료 시간을 짧게 설정하면 액세스 키 ID 및 비밀 액세스 키가 실수로 노출될 경우 위험을 줄일 수 있습니다. 만료된 키는 자동으로 제거됩니다.
- 환경의 보안 위험이 낮으며 정기적으로 새 키를 만들 필요가 없는 경우 키에 대한 만료 시간을 설정할 필요가 없습니다. 나중에 새 키를 만들려면 이전 키를 수동으로 삭제합니다.



사용자에게 속한 S3 버킷 및 오브젝트는 테넌트 관리자에서 해당 사용자에 대해 표시된 액세스 키 ID 및 비밀 액세스 키를 사용하여 액세스할 수 있습니다. 이러한 이유로 액세스 키를 암호로 보호해야 합니다. 액세스 키를 정기적으로 회전하고, 계정에서 사용되지 않는 키를 제거하며, 다른 사용자와 공유하지 마십시오.

#### 단계

- 액세스 관리 > \*사용자\*를 선택합니다.
- S3 액세스 키를 관리할 사용자를 선택합니다.

사용자 세부 정보 페이지가 나타납니다.

- 액세스 키 \* 를 선택한 다음 \* 키 만들기 \* 를 선택합니다.
- 다음 중 하나를 수행합니다.
  - 만료 시간을 설정하지 않음 \* 을 선택하여 만료되지 않는 키를 생성합니다. (기본값)
  - 만료 시간 설정 \* 을 선택하고 만료 날짜 및 시간을 설정합니다.



만료 날짜는 현재 날짜로부터 최대 5년일 수 있습니다. 만료 시간은 현재 시간에서 최소 1분이 될 수 있습니다.

- Create access key \* 를 선택합니다.

액세스 키 ID 및 비밀 액세스 키가 나열된 다운로드 액세스 키 대화 상자가 나타납니다.

- 액세스 키 ID와 비밀 액세스 키를 안전한 위치에 복사하거나 \* Download.csv \* 를 선택하여 액세스 키 ID와 비밀 액세스 키가 포함된 스프레드시트 파일을 저장합니다.



이 정보를 복사하거나 다운로드할 때까지 이 대화 상자를 닫지 마십시오. 대화 상자를 닫은 후에는 키를 복사하거나 다운로드할 수 없습니다.

- 마침 \* 을 선택합니다.

새 키가 사용자 세부 정보 페이지의 액세스 키 탭에 나열됩니다.

- 테넌트 계정에 \* 그리드 페더레이션 연결 사용 \* 권한이 있는 경우, 필요에 따라 테넌트 관리 API를 사용하여 소스 그리드의 테넌트에서 대상 그리드의 테넌트로 S3 액세스 키를 수동으로 복제합니다. 을 ["API를 사용하여 S3 액세스 키의 클론을 생성합니다"](#) 참조하십시오.

#### 다른 사용자의 S3 액세스 키를 볼니다

S3 테넌트를 사용하는 경우 적절한 권한이 있으면 다른 사용자의 S3 액세스 키를 볼 수 있습니다. 만료 시간을 기준으로 목록을 정렬하면 곧 만료되는 키를 확인할 수 있습니다. 필요에 따라 새 키를 생성하고 더 이상 사용하지 않는 키를 삭제할 수 있습니다.

#### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#) 있습니다.
- 이 ["루트 액세스 권한"](#) 있습니다.



사용자에게 속한 S3 버킷 및 오브젝트는 테넌트 관리자에서 해당 사용자에 대해 표시된 액세스 키 ID 및 비밀 액세스 키를 사용하여 액세스할 수 있습니다. 이러한 이유로 액세스 키를 암호로 보호해야 합니다. 액세스 키를 정기적으로 회전하고, 계정에서 사용되지 않는 키를 제거하며, 다른 사용자와 공유하지 마십시오.

#### 단계

- 액세스 관리 > \*사용자\*를 선택합니다.
- 사용자 페이지에서 S3 액세스 키를 보려는 사용자를 선택합니다.
- 사용자 세부 정보 페이지에서 \* 액세스 키 \* 를 선택합니다.
- 키를 \* Expiration Time \* 또는 \* Access key ID \* 로 정렬합니다.
- 필요에 따라 새 키를 생성하고 에서 더 이상 사용하지 않는 키를 수동으로 삭제합니다.

기존 키가 만료되기 전에 새 키를 만들면 사용자는 계정의 개체에 대한 액세스 권한을 일시적으로 잃지 않고 새 키를 사용할 수 있습니다.

만료된 키는 자동으로 제거됩니다.

#### 관련 정보

- ["다른 사용자의 S3 액세스 키를 생성합니다"](#)
- ["다른 사용자의 S3 액세스 키를 삭제합니다"](#)

#### 다른 사용자의 S3 액세스 키를 삭제합니다

S3 테넌트를 사용하는 경우 적절한 권한이 있으면 다른 사용자의 S3 액세스 키를 삭제할 수 있습니다. 액세스 키가 삭제된 후에는 더 이상 테넌트 계정의 객체와 버킷에 액세스할 수 없습니다.

#### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)있습니다.
- 이 ["루트 액세스 권한"](#)있습니다.



사용자에게 속한 S3 버킷 및 오브젝트는 테넌트 관리자에서 해당 사용자에 대해 표시된 액세스 키 ID 및 비밀 액세스 키를 사용하여 액세스할 수 있습니다. 이러한 이유로 액세스 키를 암호로 보호해야 합니다. 액세스 키를 정기적으로 회전하고, 계정에서 사용되지 않는 키를 제거하며, 다른 사용자와 공유하지 마십시오.

#### 단계

- 액세스 관리 > \*사용자\*를 선택합니다.
- 사용자 페이지에서 관리할 S3 액세스 키가 있는 사용자를 선택합니다.
- 사용자 세부 정보 페이지에서 \* 액세스 키 \* 를 선택한 다음 삭제할 각 액세스 키에 대한 확인란을 선택합니다.
- Actions \* > \* Delete Selected key \* 를 선택합니다.
- 확인 대화 상자에서 \* 키 삭제 \* 를 선택합니다.

페이지의 오른쪽 상단에 확인 메시지가 나타납니다.

## S3 버킷을 관리합니다

### S3 버킷을 생성합니다

테넌트 관리자를 사용하여 오브젝트 데이터용 S3 버킷을 생성할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 루트 액세스 또는 모든 버킷을 관리하는 사용자 그룹에 속해 ["권한"](#)있습니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.



버킷 또는 객체의 S3 객체 잠금 속성을 설정하거나 수정하는 권한은 다음을 통해 부여될 수 있습니다.["버킷 정책 또는 그룹 정책"](#).

- 버킷에 S3 오브젝트 잠금을 설정할 계획이라면 그리드 관리자가 StorageGRID 시스템에 대한 글로벌 S3 오브젝트 잠금 설정을 활성화했으며 S3 오브젝트 잠금 버킷 및 오브젝트 요구 사항을 검토했습니다.
- 각 테넌트에 5,000개의 버킷이 있는 경우 그리드의 각 스토리지 노드에 최소 64GB의 RAM이 있습니다.



각 그리드에는 최대 100,000개의 버킷이 포함될 수 있습니다.["가지 버킷"](#).

마법사에 액세스합니다

단계

- 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
- Create bucket \* 을 선택합니다.

세부 정보를 입력합니다

단계

- 버킷에 대한 세부 정보를 입력합니다.

필드에 입력합니다	설명
버킷 이름	<p>다음 규칙을 준수하는 버킷의 이름:</p> <ul style="list-style-type: none"> <li>각 StorageGRID 시스템에서 고유해야 합니다(테넌트 계정에서만 고유한 것은 아님).</li> <li>DNS를 준수해야 합니다.</li> <li>3자 이상 63자 이하여야 합니다.</li> <li>각 레이블은 소문자 또는 숫자로 시작하고 끝나야 하며 소문자, 숫자 및 하이픈만 사용할 수 있습니다.</li> <li>가상 호스팅 스타일 요청에는 기간을 포함할 수 없습니다. 마침표는 서버 와일드카드 인증서 확인에 문제를 일으킬 수 있습니다.</li> </ul> <p>자세한 내용은 <a href="#">"버킷 명명 규칙에 대한 AWS(Amazon Web Services) 문서입니다"</a> 참조하십시오.</p> <ul style="list-style-type: none"> <li>참고 *: 버킷을 만든 후에는 버킷 이름을 변경할 수 없습니다.</li> </ul>
지역	<p>버켓의 지역.</p> <p>StorageGRID 관리자가 사용 가능한 지역을 관리합니다. 버킷의 지역은 객체에 적용되는 데이터 보호 정책에 영향을 미칠 수 있습니다. 기본적으로 모든 버킷은 다음에서 생성됩니다. us-east-1 지역. 기본 지역이 다른 지역으로 구성된 경우 us-east-1, 이 다른 지역은 처음에 드롭다운에서 선택됩니다.</p> <ul style="list-style-type: none"> <li>참고 *: 버킷을 생성한 후에는 지역을 변경할 수 없습니다.</li> </ul>

## 2. Continue \* 를 선택합니다.

설정을 관리합니다

단계

### 1. 필요한 경우 버킷에 대한 오브젝트 버전 관리를 활성화합니다.

이 버킷에 각 오브젝트의 모든 버전을 저장하려면 오브젝트 버전 관리를 활성화하십시오. 그런 다음 필요에 따라 개체의 이전 버전을 검색할 수 있습니다.

다음과 같은 경우 개체 버전 관리를 활성화해야 합니다.

- 버킷은 그리드 간 복제에 사용됩니다.
- 당신은 만들고 싶어 **"가지 버킷"** 이 양동이에서.

### 2. 전역 S3 오브젝트 잠금 설정이 활성화된 경우 버킷에 WORM(Write-Once-Read-Many) 모델을 사용하여 오브젝트를 저장할 수 있도록 S3 오브젝트 잠금을 선택적으로 설정합니다.

특정 규정 요구 사항을 충족하기 위해 개체를 일정 시간 동안 유지해야 하는 경우에만 버킷에 대해 S3 오브젝트 잠금을 활성화합니다. S3 오브젝트 잠금은 고정된 시간 또는 무기한으로 오브젝트를 삭제 또는 덮어쓰는 것을 방지하는 영구 설정입니다.



버킷에 대해 S3 오브젝트 잠금 설정이 활성화된 후에는 비활성화할 수 없습니다. 올바른 권한이 있는 사용자는 변경할 수 없는 객체를 이 버킷에 추가할 수 있습니다. 이러한 오브젝트 또는 버킷 자체를 삭제하지 못할 수 있습니다.

버킷에 대해 S3 오브젝트 잠금을 설정하면 버킷 버전 관리가 자동으로 활성화됩니다.

3. S3 오브젝트 잠금 활성화 \* 를 선택한 경우 이 버킷에 대해 \* 기본 보존 \* 을 선택적으로 활성화합니다.



그리드 관리자가 사용자에게 권한을 부여해야 ["S3 오브젝트 잠금의 특정 기능 사용"](#)합니다.

기본 보존 \* 이 활성화되면 버킷에 추가된 새 오브젝트는 삭제 또는 덮어쓰기가 되지 않도록 자동으로 보호됩니다. 기본 보존 \* 설정은 고유한 보존 기간이 있는 개체에는 적용되지 않습니다.

- a. 기본 보존 \* 이 활성화된 경우 버킷에 대한 \* 기본 보존 모드 \* 를 지정합니다.

기본 보존 모드	설명
기대치를 설정합니다	<ul style="list-style-type: none"><li>있는 사용자는 <code>s3:BypassGovernanceRetention</code> 권한이 요청 헤더를 사용하여 보존 설정을 무시할 수 <code>x-amz-bypass-governance-retention: true</code> 있습니다.</li><li>이러한 사용자는 보존 기한이 되기 전에 개체 버전을 삭제할 수 있습니다.</li><li>이러한 사용자는 개체의 보존 기간(Retain-until-date)을 증가, 감소 또는 제거할 수 있습니다.</li></ul>
규정 준수	<ul style="list-style-type: none"><li>보존 기한에 도달할 때까지 개체를 삭제할 수 없습니다.</li><li>오브젝트의 보존 기한을 늘릴 수 있지만 줄일 수는 없습니다.</li><li>개체의 보존 기한은 해당 날짜에 도달할 때까지 제거할 수 없습니다.<ul style="list-style-type: none"><li>참고 *: 그리드 관리자가 규정 준수 모드를 사용할 수 있도록 허용해야 합니다.</li></ul></li></ul>

- b. 기본 보존 \* 이 활성화된 경우 버킷의 \* 기본 보존 기간 \* 을 지정합니다.

기본 보존 기간 \* 은 이 버킷에 추가된 새 오브젝트를 인제스트할 시간부터 보존 기간을 나타냅니다. 그리드 관리자가 설정한 테넌트의 최대 보존 기간보다 작거나 같은 값을 지정하십시오.

그리드 관리자가 테넌트를 생성할 때 1일에서 100년 사이의 값을 지정할 수 있는 `_MAXIMUM_RETENTION` 기간이 설정됩니다. `_default_retention` 기간을 설정하면 최대 보존 기간에 설정된 값을 초과할 수 없습니다. 필요한 경우 그리드 관리자에게 최대 보존 기간을 늘리거나 줄이도록 요청하십시오.

4. 선택적으로 \*용량 제한 활성화\*를 선택하고 값을 입력한 다음 용량 단위를 선택합니다.

용량 제한은 이 버킷의 오브젝트에 사용할 수 있는 최대 용량입니다. 이 값은 물리 크기(디스크의 크기)가 아닌 논리 양(개체 크기)을 나타냅니다.

제한을 설정하지 않으면 이 버킷의 용량은 무제한입니다. 자세한 내용은 ["용량 제한 사용량"](#) 참조하십시오.

5. 선택적으로 \*객체 개수 제한 사용\*을 선택합니다.

객체 수 제한은 이 버킷에 포함될 수 있는 최대 객체 수입니다. 이 값은 논리적인 양(객체 수)을 나타냅니다. 제한을 설정하지 않으면 객체 개수는 무제한입니다.

## 6. Create bucket \* 을 선택합니다.

버킷이 생성되어 버킷 페이지의 테이블에 추가됩니다.

## 7. 필요에 따라 \* 버킷 세부 정보 페이지로 이동 \* 을 선택하여 "버킷 세부 정보를 봅니다" 추가 구성을 수행합니다.

당신도 할 수 있습니다 "[브랜치 버킷 생성](#)" 필요에 따라.

버킷 세부 정보를 봅니다

테넌트 계정의 버킷을 볼 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 "[지원되는 웹 브라우저](#)"입니다.
- 이 있는 사용자 그룹에 속해 "[루트 액세스, 모든 버킷 관리 또는 모든 버킷 보기 권한](#)"입니다. 이러한 권한은 그룹 또는 버킷 정책의 권한 설정을 재정의합니다.

단계

### 1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

Bucket 페이지가 나타납니다.

### 2. 각 버킷에 대한 요약 표를 검토합니다.

필요에 따라 모든 열을 기준으로 정보를 정렬하거나 목록 앞뒤에 페이지를 표시할 수 있습니다.



표시되는 개체 수, 사용된 공간 및 사용 현황 값은 추정값입니다. 이러한 추정치는 베스트 타이밍, 네트워크 연결 및 노드 상태의 영향을 받습니다. 버킷에 버전 관리가 활성화된 경우 삭제된 개체 버전은 오브젝트 수에 포함됩니다.

이름

버킷의 고유한 이름이며 변경할 수 없습니다.

활성화된 기능

버킷에 대해 활성화된 기능 목록입니다.

**S3 오브젝트 잠금**

버킷에 대해 S3 오브젝트 잠금이 설정되었는지 여부

이 열은 그리드에 S3 오브젝트 잠금이 활성화된 경우에만 나타납니다. 또한 이 열에는 레거시 준수 버킷에 대한 정보도 표시됩니다.

지역

변경할 수 없는 버킷의 영역입니다. 이 열은 기본적으로 숨겨져 있습니다.

## 개체 수입니다

이 버킷의 오브젝트 수 버킷에 버전 관리가 활성화된 경우 현재 버전이 아닌 개체 버전이 이 값에 포함됩니다.

개체를 추가하거나 삭제할 때 이 값은 즉시 업데이트되지 않을 수 있습니다.

## 사용된 공간

버킷에 있는 모든 오브젝트의 논리적 크기입니다. 논리적 크기에는 복제 또는 삭제 코딩 복사본 또는 오브젝트 메타데이터에 필요한 실제 공간이 포함되지 않습니다.

이 값을 업데이트하는 데 최대 10분이 걸릴 수 있습니다.

## 사용

버킷의 용량 제한에 사용된 백분율(설정된 경우).

사용률은 내부 추정치를 기준으로 하며 경우에 따라 초과될 수 있습니다. 예를 들어, StorageGRID는 테넌트가 객체 업로드를 시작할 때 용량 제한을 확인하고 테넌트가 용량 제한을 초과하는 경우 이 버킷에 대한 새 수신을 거부합니다. 그러나 StorageGRID는 용량 제한을 초과했는지 확인할 때 현재 업로드 크기를 고려하지 않습니다. 객체가 삭제되면 용량 제한 사용량이 다시 계산될 때까지 테넌트가 이 버킷에 새 객체를 업로드하지 못할 수 있습니다. 계산에는 10분 이상이 걸릴 수 있습니다.

이 값은 객체 및 해당 메타데이터를 저장하는 데 필요한 물리적 크기가 아니라 논리적 크기를 나타냅니다.

## 용량

설정된 경우 버킷의 용량 제한입니다.

## 만든 날짜

버킷이 생성된 날짜 및 시간입니다. 이 열은 기본적으로 숨겨져 있습니다.

### 3. 특정 버킷의 세부 정보를 보려면 테이블에서 버킷 이름을 선택합니다.

- 웹 페이지 상단에 있는 요약 정보를 확인하여 지역 및 오브젝트 수와 같은 버킷의 세부 정보를 확인합니다.
- 용량 제한 사용량과 객체 수 제한 사용량 막대를 확인하세요. 사용량이 100%이거나 100%에 가까운 경우 한도를 늘리거나 일부 객체를 삭제하는 것을 고려하세요.
- 필요에 따라 \* Delete objects in bucket \* 과 \* Delete bucket \* 을 선택합니다.

각 옵션을 선택할 때 나타나는 주의 사항에 각별히 주의하십시오. 자세한 내용은 다음을 참조하십시오.



- "버킷의 모든 오브젝트를 삭제합니다"
- "버킷을 삭제합니다" (버킷은 비어 있어야 함)

### d. 필요에 따라 각 탭에서 버킷의 설정을 보거나 변경합니다.

- \* S3 콘솔 \* : 버킷에 대한 객체를 봅니다. 자세한 내용은 ["S3 콘솔 사용"](#) 참조하십시오.
- \* 버킷 옵션 \* : 옵션 설정을 보거나 변경합니다. 버킷이 생성된 후에는 S3 Object Lock과 같은 일부 설정을 변경할 수 없습니다.
  - "버킷 일관성 관리"
  - "마지막 액세스 시간 업데이트"

- "용량 제한"
- "객체 개수 제한"
- "오브젝트 버전 관리"
- "S3 오브젝트 잠금"
- "기본 버킷 보유"
- "교차 그리드 복제 관리" (테넌트에 대해 허용되는 경우)
- \* 플랫폼 서비스 \*["플랫폼 서비스 관리"](#): (테넌트에 대해 허용되는 경우)
- \* 버킷 액세스 \* : 옵션 설정을 보거나 변경합니다. 특정 액세스 권한이 있어야 합니다.
  - 구성["버킷 및 객체에 대한 CORS"](#) 따라서 버킷과 버킷 내의 객체는 다른 도메인의 웹 애플리케이션에서 접근할 수 있습니다.
  - ["사용자 액세스를 제어합니다"](#) S3 버킷과 해당 버킷의 오브젝트용.
- 지점: 버킷의 지점 버킷 목록을 봅니다. ["새로운 브랜치 버킷을 생성하거나 브랜치 버킷을 관리합니다."](#) .

브랜치 버킷이란 무엇인가요?

분기 버킷은 특정 시간에 존재했던 버킷의 객체에 대한 액세스를 제공합니다.

기존 버킷에서 브랜치 버킷을 만듭니다. 브랜치 버킷을 생성한 후, 해당 버킷이 생성된 원래 버킷을 기본 버킷이라고 합니다. 또한 다른 브랜치 버킷에서 브랜치 버킷을 만들 수도 있습니다.

지점 버킷은 보호된 데이터에 대한 액세스를 제공하지만 백업 역할을 하지는 않습니다. 데이터를 계속 보호하려면 기본 버킷에서 다음 기능을 사용하세요.

- ["S3 오브젝트 잠금"](#)
- ["교차 그리드 복제"](#)베이스 버킷용
- ["버킷 정책"](#)버전이 지정된 버킷에서 이전 객체 버전을 정리합니다.

지점 버킷의 다음 특성을 참고하세요.

- 다음을 사용하여 브랜치 버킷의 개체에 액세스할 수 있습니다. ["객체를 다운로드하기 위한 S3 콘솔"](#) .
- 클라이언트가 브랜치 버킷의 객체에 액세스하는 경우 브랜치 버킷의 ["액세스 정책"](#) 기본 버킷의 정책보다는 액세스가 허용되거나 거부되는지 여부를 결정합니다.
- 기본 버킷에 생성된 객체는 다음을 기준으로 평가됩니다. ["ILM 규칙"](#) 기본 버킷에 적용합니다. 지점 버킷에 생성된 개체는 ILM 규칙이 지점 버킷에 적용되는 방식에 따라 평가됩니다.
- 브랜치 버킷에서는 크로스 그리드 복제가 지원되지 않습니다.
- 플랫폼 서비스는 지점 버킷에서 지원되지 않습니다.

지점 버킷 사용에 대한 예

- 손상된 객체를 제거하려면 손상이 발생하기 전 시점부터 브랜치 버킷을 만든 다음 손상된 객체가 들어 있는 기본 버킷 대신 해당 브랜치 버킷을 애플리케이션에 가리키도록 하면 됩니다.
- 버전이 지정된 버킷에 데이터를 저장합니다. 시간  $T$  이후에 원치 않는 많은 물건이 섭취되는 우발적인 취약점이 있었습니다. Before 시간 값인  $_T$ 에 대한 분기 버킷을 생성하고 클라이언트 작업을 해당 분기 버킷으로

리디렉션할 수 있습니다. 그러면 Before time  $_T_$ 보다 일찍 수집된 객체만 클라이언트에 노출됩니다.

#### 브랜치 버킷의 객체에 대한 작업

- 브랜치 버킷에 대한 PUT 객체 작업은 브랜치에 객체를 생성합니다.
- 브랜치 버킷에 대한 GET 객체 작업은 브랜치에서 객체를 검색합니다. 객체가 브랜치 버킷에 없으면 객체는 기본 버킷에서 검색됩니다.
- 브랜치 버킷에서 객체 삭제는 다음과 같이 발생합니다.

작동	목표	결과	기본 버킷의 객체 가시성	브랜치 버킷의 객체 가시성
버전 ID 없이 삭제	베이스 버킷	삭제 마커는 기본 버킷에 대해서만 생성됩니다.	HEAD/GET은 객체가 존재하지 않지만 특정 버전에는 여전히 액세스할 수 있음을 반환합니다.	HEAD/GET은 객체가 존재하고 특정 버전에 계속 액세스할 수 있음을 반환합니다.  삭제 마커는 브랜치 버킷 이후에 생성되었을 것입니다. <code>beforeTime</code> .
버전 ID로 삭제	베이스 버킷	기본 버킷과 분기 버킷 모두에 대해 특정 개체 버전이 삭제됩니다.	HEAD/GET이 개체 버전이 존재하지 않는다는 것을 반환합니다.	HEAD/GET이 개체 버전이 존재하지 않는다는 것을 반환합니다.
버전 ID 없이 삭제	가지 버킷	삭제 마커는 브랜치 버킷에 대해서만 생성됩니다.	HEAD/GET은 객체를 반환합니다(기본 버킷 객체는 영향을 받지 않음)	HEAD/GET이 객체가 존재하지 않는다는 것을 반환합니다.
버전 ID로 삭제	가지 버킷	특정 객체 버전은 브랜치 버킷에 대해서만 삭제됩니다.	HEAD/GET은 특정 객체 버전을 반환합니다(기본 버킷 객체는 영향을 받지 않음)	HEAD/GET이 개체 버전이 존재하지 않는다는 것을 반환합니다.

또한 참조 "[S3 버전 오브젝트 삭제 방법](#)" .

#### 지점 버킷 관리

테넌트 관리자를 사용하여 지점 버킷에 대한 세부 정보를 만들고 확인하세요.

#### 시작하기 전에

- 귀하는 다음을 사용하여 테넌트 관리자에 로그인했습니다. "[지원되는 웹 브라우저](#)" .
- 귀하는 루트 액세스 권한이 있는 사용자 그룹에 속해 있습니다. "[모든 버킷 사용 권한을 관리합니다](#)" . 이러한 권한은 그룹 또는 버킷 정책의 권한 설정을 재정의합니다.
- 브랜치를 생성하려는 기본 버킷에는 다음이 있습니다. "[버전 관리 활성화됨](#)" .
- 당신은 기본 버킷의 소유자입니다.

이 작업에 대해

분기 버킷에 대한 다음 정보를 참고하세요.

- 버킷 또는 객체의 S3 객체 잠금 속성을 설정하는 권한은 다음을 통해 부여될 수 있습니다. ["버킷 정책 또는 그룹 정책"](#).
- 기본 버킷에서 버전 관리를 중단하면 기본 버킷의 내용이 더 이상 해당 분기 버킷에 표시되지 않습니다.



브랜치 버킷을 구성하고 생성한 후에는 구성은 변경할 수 없습니다.

브랜치 버킷 생성

단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
2. 브랜치를 만들려는 버킷("기본 버킷")을 선택합니다.
3. 버킷 세부 정보 페이지에서 지점 > \*지점 버킷 만들기\*를 선택합니다.

기본 버킷의 버전 관리가 활성화되어 있지 않으면 브랜치 버킷 만들기 버튼이 비활성화됩니다.

세부 정보를 입력합니다

단계

1. 지점 버킷에 대한 세부 정보를 입력하세요.

필드에 입력합니다	설명
지점 버킷 이름	<p>다음 규칙을 준수하는 브랜치 버킷의 이름:</p> <ul style="list-style-type: none"><li>• 각 StorageGRID 시스템에서 고유해야 합니다(테넌트 계정에서만 고유한 것은 아님).</li><li>• DNS를 준수해야 합니다.</li><li>• 3자 이상 63자 이하여야 합니다.</li><li>• 각 레이블은 소문자 또는 숫자로 시작하고 끝나야 하며 소문자, 숫자 및 하이픈만 사용할 수 있습니다.</li><li>• 가상 호스팅 스타일 요청에는 기간을 포함할 수 없습니다. 마침표는 서버 와일드카드 인증서 확인에 문제를 일으킬 수 있습니다.</li></ul> <p>자세한 내용은 <a href="#">"버킷 명명 규칙에 대한 AWS(Amazon Web Services) 문서입니다"</a> 참조하십시오.</p> <p>참고: 브랜치 버킷을 만든 후에는 이름을 변경할 수 없습니다.</p>
지역(지점 버킷에 대해 수정할 수 없음)	<p>지점 버킷의 지역입니다.</p> <p>지점 버킷의 지역은 기본 버킷의 지역과 일치해야 하므로 이 필드는 지점 버킷에 대해 비활성화됩니다.</p>

필드에 입력합니다	설명
시간 전에	<p>기본 버킷에서 생성된 객체 버전을 브랜치 버킷에서 액세스할 수 있게 되는 마감 시간입니다. 브랜치 버킷은 Before 시간보다 이전에 생성된 객체 버전에 대한 액세스를 제공합니다.</p> <p>시간 이전에는 이미 지나간 날짜와 시간이 있어야 합니다. 미래의 날짜일 수 없습니다.</p>
브랜치 버킷 유형	<ul style="list-style-type: none"> <li>읽기-쓰기: 브랜치 버킷에서 개체 또는 개체 버전을 추가하거나 삭제할 수 있습니다.</li> <li>읽기 전용: 브랜치 버킷의 객체를 수정할 수 없습니다.</li> </ul> <p>참고: 브랜치 버킷이 비어 있는 경우에만 브랜치 버킷 유형을 읽기 전용으로 설정할 수 있습니다. 기존 브랜치 버킷의 유형이 읽기-쓰기로 설정되어 있고 해당 버킷에 쓰지 않은 경우 유형을 읽기 전용으로 변경할 수 있습니다.</p>

## 2. Continue \* 를 선택합니다.

### 개체 설정 관리(선택 사항)

브랜치 버킷의 개체 설정은 기본 버킷의 개체 버전에 영향을 미치지 않습니다.

#### 단계

- 글로벌 S3 개체 잠금 설정이 활성화된 경우 선택적으로 브랜치 버킷에 대한 S3 개체 잠금을 활성화하려면 브랜치 버킷이 읽기-쓰기 버킷이어야 합니다.

예를 들어 특정 규정 요구 사항을 충족하기 위해 객체를 고정된 기간 동안 보관해야 하는 경우에만 지점 버킷에 대해 S3 객체 잠금을 활성화합니다. S3 객체 잠금은 특정 기간 또는 무기한으로 객체가 삭제되거나 덮어쓰이는 것을 방지하는 데 도움이 되는 영구 설정입니다.



버킷에 대해 S3 개체 잠금 설정을 활성화한 후에는 비활성화할 수 없습니다. 올바른 권한이 있는 사람은 누구나 변경할 수 없는 객체를 브랜치 버킷에 추가할 수 있습니다. 이러한 객체나 브랜치 버킷 자체를 삭제할 수 없을 수도 있습니다.

- \*S3 객체 잠금 활성화\*를 선택한 경우, 선택적으로 브랜치 버킷에 대해 \*기본 보존\*을 활성화합니다.



그리드 관리자가 사용자에게 권한을 부여해야 ["S3 오브젝트 잠금의 특정 기능 사용"](#)합니다.

기본 보존\*을 활성화하면 브랜치 버킷에 추가된 새 개체는 삭제되거나 덮어쓰이지 않도록 자동으로 보호됩니다.  
\*기본 보존 설정은 자체 보존 기간이 있는 개체에는 적용되지 않습니다.

- \*기본 보존\*이 활성화된 경우 브랜치 버킷에 대한 \*기본 보존 모드\*를 지정합니다.

기본 보존 모드	설명
기대치를 설정합니다	<ul style="list-style-type: none"> <li>있는 사용자는 <code>s3:BypassGovernanceRetention</code> 권한이 요청 헤더를 사용하여 보존 설정을 무시할 수 `x-amz-bypass-governance-retention: true` 있습니다.</li> <li>이러한 사용자는 보존 기한이 되기 전에 개체 버전을 삭제할 수 있습니다.</li> <li>이러한 사용자는 개체의 보존 기간(Retain-until-date)을 증가, 감소 또는 제거할 수 있습니다.</li> </ul>
규정 준수	<ul style="list-style-type: none"> <li>보존 기한에 도달할 때까지 개체를 삭제할 수 없습니다.</li> <li>오브젝트의 보존 기한을 늘릴 수 있지만 줄일 수는 없습니다.</li> <li>개체의 보존 기한은 해당 날짜에 도달할 때까지 제거할 수 없습니다. <ul style="list-style-type: none"> <li>참고 *: 그리드 관리자가 규정 준수 모드를 사용할 수 있도록 허용해야 합니다.</li> </ul> </li> </ul>

b. \*기본 보존\*이 활성화된 경우 지점 버킷의 \*기본 보존 기간\*을 지정합니다.

\*기본 보존 기간\*은 브랜치 버킷에 추가된 새 객체를 수집한 시점부터 보존해야 하는 기간을 나타냅니다. 그리드 관리자가 설정한 테넌트의 최대 보존 기간보다 작거나 같은 값을 지정합니다.

그리드 관리자가 테넌트를 생성할 때 1일에서 100년 사이의 값을 지정할 수 있는 `_MAXIMUM_RETENTION` 기간이 설정됩니다. `_default_retention` 기간을 설정하면 최대 보존 기간에 설정된 값을 초과할 수 없습니다. 필요한 경우 그리드 관리자에게 최대 보존 기간을 늘리거나 줄이도록 요청하십시오.

3. 선택적으로 \*용량 제한 활성화\*를 선택합니다.

용량 한도는 지점 버킷에 사용할 수 있는 최대 용량입니다. 이 값은 물리적인 양(디스크 크기)이 아닌 논리적인 양(객체 크기)을 나타냅니다.

제한을 설정하지 않으면 지점 버킷의 용량은 무제한입니다. . ["용량 제한 사용량"](#) 자세한 내용은.



이 설정은 브랜치 버킷에 직접 수집된 객체에만 적용되며, 브랜치 버킷을 통해 기본 버킷에서 볼 수 있는 객체에는 적용되지 않습니다.

4. 선택적으로 \*객체 개수 제한 사용\*을 선택합니다.

객체 수 제한은 브랜치 버킷이 포함할 수 있는 최대 객체 수입니다. 이 값은 논리적인 양(객체 수)을 나타냅니다. 제한을 설정하지 않으면 객체 개수는 무제한입니다.



이 설정은 브랜치 버킷에 직접 수집된 객체에만 적용되며, 브랜치 버킷을 통해 기본 버킷에서 볼 수 있는 객체에는 적용되지 않습니다.

5. Create bucket \* 을 선택합니다.

브랜치 버킷이 생성되어 버킷 페이지의 테이블에 추가됩니다.

6. 선택적으로 \*버킷 세부 정보 페이지로 이동\*을 선택하세요. ["지점 버킷 세부 정보 보기"](#) 추가 구성을 수행합니다.

버킷 세부 정보 페이지에서는 읽기 전용 버킷의 개체 수정과 관련된 일부 구성 옵션이 비활성화됩니다.

버킷에 ILM 정책 태그를 적용합니다

오브젝트 스토리지 요구사항에 따라 버킷에 적용할 ILM 정책 태그를 선택합니다.

ILM 정책은 오브젝트 데이터의 저장 위치와 일정 기간 후에 삭제되는지 여부를 제어합니다. 그리드 관리자는 ILM 정책을 만들고 활성 정책을 여러 개 사용할 때 ILM 정책 태그에 할당합니다.



버킷의 정책 태그를 자주 재할당하지 마십시오. 그렇지 않으면 성능 문제가 발생할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 "지원되는 웹 브라우저"입니다.
- 이 있는 사용자 그룹에 속해 "루트 액세스, 모든 버킷 관리 또는 모든 버킷 보기 권한"입니다. 이러한 권한은 그룹 또는 버킷 정책의 권한 설정을 재정의합니다.

단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

Bucket 페이지가 나타납니다. 필요에 따라 모든 열을 기준으로 정보를 정렬하거나 목록 앞뒤에 페이지를 표시할 수 있습니다.

2. ILM 정책 태그를 할당할 버킷의 이름을 선택합니다.

이미 태그가 할당된 버킷의 ILM 정책 태그 할당을 변경할 수도 있습니다.



표시된 개체 수와 사용된 공간 값은 추정값입니다. 이러한 추정치는 베스트 타이밍, 네트워크 연결 및 노드 상태의 영향을 받습니다. 버킷에 버전 관리가 활성화된 경우 삭제된 개체 버전은 오브젝트 수에 포함됩니다.

3. Bucket options 탭에서 ILM 정책 태그 아코디언을 확장합니다. 이 아코디언은 그리드 관리자가 사용자 지정 정책 태그를 사용하도록 설정한 경우에만 나타납니다.

4. 각 정책 태그에 대한 설명을 읽고 버킷에 적용할 태그를 결정합니다.



버킷의 ILM 정책 태그를 변경하면 버킷의 모든 오브젝트에 대한 ILM 재평가가 트리거됩니다. 새 정책에 따라 제한된 시간 동안 오브젝트를 유지하는 경우 이전 오브젝트는 삭제됩니다.

5. 버킷에 할당할 태그의 라디오 버튼을 선택합니다.

6. 변경 내용 저장 \* 을 선택합니다. 새 S3 버킷 태그는 키와 ILM 정책 태그 이름의 값을 사용하여 버킷에 NTAP-SG-ILM-BUCKET-TAG 설정됩니다.



S3 애플리케이션이 실수로 새 버킷 태그를 무시하거나 삭제하지 않도록 하십시오. 새 TagSet를 버킷에 적용할 때 이 태그를 생략하면 버킷의 객체가 기본 ILM 정책에 대해 평가되는 것으로 되돌아갑니다.



ILM 정책 태그가 검증된 테넌트 관리자 또는 테넌트 관리자 API만 사용하여 ILM 정책 태그를 설정하고 수정합니다. S3 PutBucketTagging API 또는 S3 DeleteBucketTagging API를 사용하여 ILM 정책 태그를 수정하지 마십시오 NTAP-SG-ILM-BUCKET-TAG.



버킷에 할당된 정책 태그를 변경하면 새로운 ILM 정책을 사용하여 오브젝트를 재평가하는 동안 성능에 일시적으로 영향을 미칩니다.

## 버킷 정책을 관리합니다

S3 버킷 및 해당 버킷의 오브젝트에 대한 사용자 액세스를 제어할 수 있습니다.

### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["루트 액세스 권한"](#)입니다. 모든 버킷 보기 및 모든 버킷 관리 권한만 볼 수 있습니다.
- 필요한 수의 스토리지 노드 및 사이트를 사용할 수 있는지 확인했습니다. 사이트 내에서 두 개 이상의 스토리지 노드를 사용할 수 없거나 사이트를 사용할 수 없는 경우 이러한 설정을 변경하지 못할 수 있습니다.

### 단계

1. Bucket \* 을 선택한 다음 관리하려는 버킷을 선택합니다.
2. 버킷 세부 정보 페이지에서 \* Bucket access \* > \* Bucket policy \* 를 선택합니다.
3. 다음 중 하나를 수행합니다.
  - Enable policy \* 확인란을 선택하여 버킷 정책을 입력합니다. 그런 다음 유효한 JSON 형식 문자열을 입력하십시오.

각 버킷 정책의 크기 제한은 20,480바이트입니다.

- 문자열을 편집하여 기존 정책을 수정합니다.
- 정책 활성화 \* 를 선택 취소하여 정책을 비활성화합니다.

언어 구문 및 예제를 포함한 버킷 정책에 대한 자세한 내용은 [를 참조하십시오"버킷 정책의 예"](#).

## 버킷 일관성 관리

일관성 값을 사용하여 버킷 설정 변경의 가용성을 지정하고 버킷 내의 오브젝트 가용성과 서로 다른 스토리지 노드 및 사이트에서 이러한 오브젝트의 일관성 간에 균형을 유지할 수 있습니다. 정합성 보장 값을 기본값과 다르게 변경하여 클라이언트 애플리케이션이 운영 요구사항을 충족할 수 있도록 할 수 있습니다.

### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["모든 버킷 또는 루트 액세스 권한을 관리합니다"](#)입니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.

## 버킷 일관성 지침

버킷 정합성 보장은 해당 S3 버킷 내의 오브젝트에 영향을 주는 클라이언트 애플리케이션의 정합성을 결정하는 데 사용됩니다. 일반적으로 버킷에는 \* Read-after-new-write \* 일관성을 사용해야 합니다.

버킷 일관성을 변경합니다

Read-after-new-write \* 일관성이 클라이언트 애플리케이션의 요구 사항을 충족하지 않는 경우 버킷 일관성을 설정하거나 헤더를 사용하여 일관성을 변경할 수 Consistency-Control 있습니다. 'Consistency-Control' 헤더가 버킷 일관성을 재정의합니다.



버킷의 일관성을 변경하면 변경 후 수집되는 객체만 수정된 설정을 충족할 수 있습니다.

### 단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
2. 테이블에서 버킷 이름을 선택합니다.
3. 버킷 세부 정보 페이지가 나타납니다.
4. 이 버킷의 객체에서 수행되는 작업에 대한 일관성을 선택합니다.
  - \* 모두 \*: 최고 수준의 일관성을 제공합니다. 모든 노드가 데이터를 즉시 수신하거나 요청이 실패합니다.
  - \* 강력한 글로벌 \*: 모든 사이트에서 모든 클라이언트 요청에 대해 쓰기 후 읽기 일관성을 보장합니다.
  - \* 강력한 사이트 \*: 사이트 내의 모든 클라이언트 요청에 대해 쓰기 후 읽기 일관성을 보장합니다.
  - \* Read-After-new-write \* (기본값): 새 객체에 대해 읽기-쓰기 후 정합성을 보장하고 객체 업데이트에 대한 최종 일관성을 제공합니다. 고가용성 및 데이터 보호 보장 제공 대부분의 경우에 권장됩니다.
  - \* 사용 가능 \*: 새 객체 및 객체 업데이트 모두에 대한 최종 일관성을 제공합니다. S3 버킷의 경우 필요한 경우에만 사용하십시오(예: 거의 읽지 않는 로그 값이 포함된 버킷의 경우 또는 존재하지 않는 키의 헤드 또는 GET 작업의 경우). S3 FabricPool 버킷은 지원되지 않습니다.
5. 변경 내용 저장 \* 을 선택합니다.

버킷 설정을 변경하면 어떻게 됩니까

버킷에는 버킷 및 해당 버킷 내의 객체의 동작에 영향을 주는 여러 설정이 있습니다.

다음 버킷 설정은 기본적으로 \* 강한 \* 일관성을 사용합니다. 사이트 내에서 두 개 이상의 스토리지 노드를 사용할 수 없거나 사이트를 사용할 수 없는 경우 이러한 설정을 변경하지 못할 수 있습니다.

- "배경이 비어 있는 버킷 삭제"
- "마지막 액세스 시간입니다"
- "버킷 수명 주기"
- "버킷 정책"
- "버킷 태그 지정"
- "버킷 버전 관리"
- "S3 오브젝트 잠금"

- "버킷 암호화"



버킷 버전 관리, S3 오브젝트 잠금 및 버킷 암호화에 대한 정합성 보장 값을 강력하게 일치하지 않는 값으로 설정할 수 없습니다.

다음 버킷 설정은 강력한 일관성을 사용하지 않으며 변경에 대한 가용성이 높습니다. 이러한 설정을 변경하는 데 시간이 걸릴 수 있습니다.

- "플랫폼 서비스 구성: 알림, 복제 또는 검색 통합"
- "버킷 및 객체에 대한 StorageGRID CORS 구성"
- 버킷 일관성을 변경합니다



버킷 설정 변경 시 사용되는 기본 일관성이 클라이언트 애플리케이션의 요구 사항을 충족하지 않는 경우의 헤더를 ["S3 REST API"](#) 사용하거나 또는 `force` 옵션을 사용하여 `reducedConsistency` 일관성을 변경할 수 있습니다. 'Consistency-Control' ["테넌트 관리 API"](#).

마지막 액세스 시간 업데이트를 사용하거나 사용하지 않도록 설정합니다

그리드 관리자가 StorageGRID 시스템에 대한 ILM(정보 수명 주기 관리) 규칙을 만들 때 오브젝트의 마지막 액세스 시간을 사용하여 해당 오브젝트를 다른 스토리지 위치로 이동할지 여부를 결정하도록 선택적으로 지정할 수 있습니다. S3 테넌트를 사용하는 경우 S3 버킷의 오브젝트에 대한 마지막 액세스 시간 업데이트를 활성화하여 이러한 규칙을 활용할 수 있습니다.

이 지침은 고급 필터 또는 참조 시간으로 \* 마지막 액세스 시간 \* 옵션을 사용하는 ILM 규칙을 하나 이상 포함하는 StorageGRID 시스템에만 적용됩니다. StorageGRID 시스템에 이러한 규칙이 포함되어 있지 않으면 이 지침을 무시할 수 있습니다. 자세한 내용은 ["ILM 규칙에서 마지막 액세스 시간을 사용합니다"](#) 참조하십시오.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["모든 버킷 또는 루트 액세스 권한을 관리합니다"](#)입니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.

이 작업에 대해

- 마지막 액세스 시간 \* 은 ILM 규칙에 대한 \* 참조 시간 \* 배치 명령에 사용할 수 있는 옵션 중 하나입니다. 규칙의 참조 시간을 마지막 액세스 시간으로 설정하면 그리드 관리자는 해당 개체를 마지막으로 검색한 시기(읽기 또는 보기)에 따라 특정 저장소 위치에 개체가 배치되도록 지정할 수 있습니다.

예를 들어, 최근에 본 오브젝트를 더 빠른 스토리지에 유지하기 위해 그리드 관리자는 다음을 지정하는 ILM 규칙을 생성할 수 있습니다.

- 지난 달 동안 검색된 객체는 로컬 스토리지 노드에 남아 있어야 합니다.
- 지난 달에 검색되지 않은 객체는 오프 사이트 위치로 이동해야 합니다.

기본적으로 마지막 액세스 시간에 대한 업데이트는 사용되지 않습니다. StorageGRID 시스템에 \* Last Access Time \* 옵션을 사용하는 ILM 규칙이 포함되어 있고 이 옵션이 이 버킷의 오브젝트에 적용되도록 하려면 해당 규칙에 지정된 S3 버킷에 대한 마지막 액세스 시간에 대한 업데이트를 활성화해야 합니다.



개체가 검색될 때 마지막 액세스 시간을 업데이트하면 특히 작은 개체의 StorageGRID 성능이 저하될 수 있습니다.

StorageGRID는 객체가 검색될 때마다 다음 단계를 수행해야 하므로 마지막 액세스 시간 업데이트 시 성능 영향이 발생합니다.

- 객체를 새 타임스탬프로 업데이트합니다
- ILM 대기열에 개체를 추가하여 현재 ILM 규칙 및 정책에 대해 다시 평가할 수 있습니다

이 표에는 마지막 액세스 시간이 비활성화되거나 활성화될 때 버킷의 모든 오브젝트에 적용되는 동작이 요약되어 있습니다.

요청 유형입니다	마지막 액세스 시간이 비활성화된 경우의 동작(기본값)		마지막 액세스 시간이 설정된 경우의 동작	
	마지막 액세스 시간이 업데이트되었습니까?	ILM 평가 대기열에 개체가 추가되었습니까?	마지막 액세스 시간이 업데이트되었습니까?	ILM 평가 대기열에 개체가 추가되었습니까?
HEAD 작업이 실행될 때 객체의 메타데이터를 검색하라는 요청	아니요	아니요	아니요	아니요
개체, 해당 액세스 제어 목록 또는 해당 메타데이터를 검색하는 요청입니다	아니요	아니요	예	예
개체의 메타데이터를 업데이트하도록 요청합니다	예	예	예	예
개체 또는 개체 버전을 나열하는 요청입니다	아니요	아니요	아니요	아니요
한 버킷에서 다른 버킷으로 오브젝트 복사 요청	<ul style="list-style-type: none"> <li>• 아니요, 소스 복제본입니다</li> <li>• 예, 대상 복사본에 대해입니다</li> </ul>	<ul style="list-style-type: none"> <li>• 아니요, 소스 복제본입니다</li> <li>• 예, 대상 복사본에 대해입니다</li> </ul>	<ul style="list-style-type: none"> <li>• 예. 소스 복제본에 대해 가능합니다</li> <li>• 예, 대상 복사본에 대해입니다</li> </ul>	<ul style="list-style-type: none"> <li>• 예. 소스 복제본에 대해 가능합니다</li> <li>• 예, 대상 복사본에 대해입니다</li> </ul>
여러 부분 업로드를 완료하도록 요청합니다	예. 조립된 개체에 대해 가능합니다	예. 조립된 개체에 대해 가능합니다	예. 조립된 개체에 대해 가능합니다	예. 조립된 개체에 대해 가능합니다

## 단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

2. 테이블에서 버킷 이름을 선택합니다.

버킷 세부 정보 페이지가 나타납니다.

3. Bucket options \* 탭에서 \* Last access time updates \* 아코디언을 선택합니다.

4. 마지막 액세스 시간 업데이트를 사용하거나 사용하지 않도록 설정합니다.

5. 변경 내용 저장 \* 을 선택합니다.

버킷의 오브젝트 버전 관리를 변경합니다

S3 테넌트를 사용하는 경우 S3 버킷의 버전 관리 상태를 변경할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["모든 버킷 또는 루트 액세스 권한을 관리합니다"](#)입니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.
- 필요한 수의 스토리지 노드 및 사이트를 사용할 수 있는지 확인했습니다. 사이트 내에서 두 개 이상의 스토리지 노드를 사용할 수 없거나 사이트를 사용할 수 없는 경우 이러한 설정을 변경하지 못할 수 있습니다.

이 작업에 대해

버킷에 대한 오브젝트 버전 관리를 설정하거나 일시 중지할 수 있습니다. 버켓에 대한 버전 관리를 활성화한 후에는 버전이 지정되지 않은 상태로 돌아갈 수 없습니다. 그러나 버킷의 버전 관리를 일시 중단할 수 있습니다.

- 사용 안 함: 버전 관리가 활성화되지 않았습니다
- 사용: 버전 관리가 활성화됩니다
- 일시 중단됨: 버전 관리가 이전에 활성화되었으며 일시 중단되었습니다

자세한 내용은 다음을 참조하십시오.

- ["오브젝트 버전 관리"](#)
- ["S3 버전 오브젝트 ILM 규칙 및 정책\(예 4\)"](#)
- ["오브젝트 삭제 방법"](#)

단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

2. 테이블에서 버킷 이름을 선택합니다.

버킷 세부 정보 페이지가 나타납니다.

3. Bucket options \* 탭에서 \* Object Version \* 아코디언을 선택합니다.

4. 이 버킷의 오브젝트에 대한 버전 관리 상태를 선택합니다.

오브젝트 버전 관리는 교차 그리드 복제에 사용되는 버킷에 대해 활성화된 상태를 유지해야 합니다. S3 오브젝트 잠금 또는 레거시 규정 준수를 활성화하면 \* 오브젝트 버전 관리 \* 옵션이 비활성화됩니다.

옵션을 선택합니다	설명
버전 관리를 활성화합니다	<p>이 버킷에 각 오브젝트의 모든 버전을 저장하려면 오브젝트 버전 관리를 활성화하십시오. 그런 다음 필요에 따라 개체의 이전 버전을 검색할 수 있습니다.</p> <p>버킷에 이미 있던 객체는 사용자가 수정할 때 버전이 적용됩니다.</p>
버전 관리를 일시 중단합니다	새 개체 버전을 더 이상 만들지 않으려면 개체 버전 관리를 일시 중단합니다. 기존 개체 버전을 검색할 수 있습니다.

5. 변경 내용 저장 \* 을 선택합니다.

### S3 오브젝트 잠금을 사용하여 오브젝트를 보존합니다

버킷과 오브젝트가 보존 규정 요구사항을 준수해야 하는 경우 S3 오브젝트 잠금을 사용할 수 있습니다.



그리드 관리자는 귀하에게 S3 오브젝트 잠금의 특정 기능을 사용할 수 있는 권한을 부여해야 합니다.

### S3 오브젝트 잠금이란 무엇입니까?

StorageGRID S3 오브젝트 잠금 기능은 Amazon S3(Amazon Simple Storage Service)의 S3 오브젝트 잠금과 동등한 오브젝트 보호 솔루션입니다.

StorageGRID 시스템에 대해 글로벌 S3 오브젝트 잠금 설정이 활성화된 경우 S3 테넌트 계정에서 S3 오브젝트 잠금이 활성화되어 있거나 사용되지 않고 버킷을 생성할 수 있습니다. 버킷에 S3 오브젝트 잠금이 활성화된 경우 버킷 버전 관리가 필요하며 자동으로 활성화됩니다.

\*S3 오브젝트 잠금이 없는 버킷은 보존 설정이 지정되지 않은 오브젝트만 가질 수 있습니다. 수집된 객체에는 보존 설정이 없습니다.

- S3 오브젝트 잠금이 있는 버킷은 S3 클라이언트 애플리케이션에 지정된 보존 설정이 있거나 없는 객체를 포함할 수 있습니다. 수집된 일부 객체에는 보존 설정이 있습니다.
- S3 오브젝트 잠금 및 기본 보존이 구성된 버킷 \*은 보존 설정이 지정된 객체와 보존 설정이 없는 새 객체를 업로드할 수 있습니다. 개체 수준에서 보존 설정이 구성되지 않았기 때문에 새 개체는 기본 설정을 사용합니다.

기본적으로 보존이 구성되면 새로 수집된 모든 객체에 보존 설정이 적용됩니다. 개체 보존 설정이 없는 기존 개체는 영향을 받지 않습니다.

### 보존 모드

StorageGRID S3 오브젝트 잠금 기능은 두 가지 보존 모드를 지원하여 오브젝트에 다양한 보호 수준을 적용합니다. 이러한 모드는 Amazon S3 보존 모드에 해당합니다.

- 규정 준수 모드:
  - 보존 기한에 도달할 때까지 개체를 삭제할 수 없습니다.
  - 오브젝트의 보존 기한을 늘릴 수 있지만 줄일 수는 없습니다.
  - 개체의 보존 기한은 해당 날짜에 도달할 때까지 제거할 수 없습니다.

- 거버넌스 모드:

- 특수 권한이 있는 사용자는 요청에서 우회 헤더를 사용하여 특정 보존 설정을 수정할 수 있습니다.
- 이러한 사용자는 보존 기한이 되기 전에 개체 버전을 삭제할 수 있습니다.
- 이러한 사용자는 개체의 보존 기간(Retain-until-date)을 증가, 감소 또는 제거할 수 있습니다.

### 개체 버전에 대한 보존 설정입니다

버킷이 S3 오브젝트 잠금이 설정된 상태로 생성된 경우 사용자는 S3 클라이언트 애플리케이션을 사용하여 버킷에 추가되는 각 오브젝트에 대해 다음 보존 설정을 선택적으로 지정할 수 있습니다.

- \* 보존 모드 \*: 규정 준수 또는 거버넌스 중 하나입니다.
- \* Retain-until-date \*: 개체 버전의 Retain-until-date가 미래인 경우 개체를 검색할 수 있지만 삭제할 수 없습니다.
- \* 법적 증거 자료 보관 \*: 개체 버전에 법적 증거 자료 보관 기능을 적용하면 해당 개체가 즉시 잠깁니다. 예를 들어 조사 또는 법적 분쟁과 관련된 객체에 법적 보류를 지정해야 할 수 있습니다. 법적 보류는 만료 날짜가 없지만 명시적으로 제거될 때까지 유지됩니다. 법적 보류는 보존 기한과 무관합니다.



개체가 법적 보류 중인 경우 보존 모드에 관계없이 개체를 삭제할 수 없습니다.

개체 설정에 대한 자세한 내용은 ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다."](#)를 참조하십시오.

### 버킷을 위한 기본 보존 설정입니다

버킷이 S3 오브젝트 잠금이 활성화된 상태로 생성된 경우 사용자는 버킷에 대해 다음 기본 설정을 선택적으로 지정할 수 있습니다.

- \* 기본 보존 모드 \*: 규정 준수 또는 거버넌스 중 하나입니다.
- \* 기본 보존 기간 \*: 이 버킷에 추가된 새 오브젝트 버전이 추가되는 날부터 보존되어야 하는 기간입니다.

기본 버킷 설정은 고유한 보존 설정이 없는 새 오브젝트에만 적용됩니다. 이러한 기본 설정을 추가하거나 변경할 때 기존 버킷 개체는 영향을 받지 않습니다.

["S3 버킷을 생성합니다"](#) 및 ["S3 오브젝트 잠금 기본 보존 업데이트"](#) 참조하십시오.

### S3 오브젝트 잠금 작업

그리드 관리자 및 테넌트 사용자를 위한 다음 목록에는 S3 오브젝트 잠금 기능을 사용하기 위한 상위 수준의 작업이 포함되어 있습니다.

#### 그리드 관리자

- 전체 StorageGRID 시스템에 대해 글로벌 S3 오브젝트 잠금 설정을 사용합니다.
- 정보 수명 주기 관리(ILM) 정책이 \_준수되는지 확인합니다. 즉, 이 정책이 ["S3 오브젝트 잠금이 설정된 버킷 요구사항"](#)(를) 충족하는지 확인합니다.
- 필요에 따라 테넌트가 규정 준수를 보존 모드로 사용할 수 있도록 허용합니다. 그렇지 않으면 거버넌스 모드만 허용됩니다.
- 필요에 따라 테넌트의 최대 보존 기간을 설정합니다.

## 테넌트 사용자입니다

- S3 오브젝트 잠금을 통해 버킷 및 오브젝트에 대한 고려 사항을 검토하십시오.
- 필요한 경우 그리드 관리자에게 문의하여 글로벌 S3 오브젝트 잠금 설정을 활성화하고 권한을 설정합니다.
- S3 오브젝트 잠금이 설정된 상태로 버킷을 생성합니다.
- 필요에 따라 버킷의 기본 보존 설정을 구성합니다.
  - 기본 보존 모드: 그리드 관리자가 허용하는 경우 거버넌스 또는 규정 준수
  - 기본 보존 기간: 그리드 관리자가 설정한 최대 보존 기간 이하여야 합니다.
- S3 클라이언트 애플리케이션을 사용하여 오브젝트를 추가하고 필요에 따라 오브젝트별 보존을 설정합니다.
  - 보존 모드. 거버넌스 또는 규정 준수(그리드 관리자가 허용하는 경우)
  - 보관 종료 날짜: 그리드 관리자가 설정한 최대 보존 기간 이하여야 합니다.

### S3 오브젝트 잠금이 설정된 버킷의 요구 사항

- StorageGRID 시스템에 대해 글로벌 S3 오브젝트 잠금 설정이 활성화된 경우 테넌트 관리자, 테넌트 관리 API 또는 S3 REST API를 사용하여 S3 오브젝트 잠금이 활성화된 버킷을 생성할 수 있습니다.
- S3 오브젝트 잠금을 사용하려는 경우 버킷을 생성할 때 S3 오브젝트 잠금을 활성화해야 합니다. 기존 버킷에 S3 오브젝트 잠금을 설정할 수 없습니다.
- 버킷에 대해 S3 오브젝트 잠금이 활성화된 경우 StorageGRID는 해당 버킷의 버전 관리를 자동으로 활성화합니다. 버킷의 S3 오브젝트 잠금을 비활성화하거나 버전 관리를 일시 중단할 수 없습니다.
- 필요에 따라 테넌트 관리자, 테넌트 관리 API 또는 S3 REST API를 사용하여 각 버킷의 기본 보존 모드 및 보존 기간을 지정할 수 있습니다. 버킷의 기본 보존 설정은 고유한 보존 설정이 없는 버킷에 추가된 새 오브젝트에만 적용됩니다. 이 기본 설정은 업로드할 때 각 개체 버전에 대해 보존 모드 및 보존 종료 날짜를 지정하여 재정의할 수 있습니다.
- S3 오브젝트 잠금이 설정된 버킷에 대해 버킷 라이프사이클 구성이 지원됩니다.
- S3 오브젝트 잠금이 설정된 버킷에는 CloudMirror 복제가 지원되지 않습니다.

### S3 오브젝트 잠금이 설정된 버킷의 오브젝트 요구사항

- 개체 버전을 보호하려면 버킷의 기본 보존 설정을 지정하거나 각 오브젝트 버전에 대한 보존 설정을 지정할 수 있습니다. 오브젝트 레벨의 보존 설정은 S3 클라이언트 애플리케이션 또는 S3 REST API를 사용하여 지정할 수 있습니다.
- 보존 설정은 개별 개체 버전에 적용됩니다. 개체 버전에는 보존 기한 및 법적 보류 설정이 둘 다 있을 수 있으며, 둘 중 하나만 설정할 수도 있고 둘 다 가질 수도 없습니다. 개체에 대한 보존 기한 또는 법적 보류 설정을 지정하면 요청에 지정된 버전만 보호됩니다. 이전 버전의 개체는 잠겨 있는 상태에서 새 버전의 개체를 만들 수 있습니다.

### S3 오브젝트 잠금이 설정된 버킷의 오브젝트 라이프사이클

S3 오브젝트 잠금이 설정된 버킷에 저장된 각 오브젝트는 다음 단계를 거칩니다.

#### 1. \* 오브젝트 수집 \*

오브젝트 버전이 S3 오브젝트 잠금이 설정된 버킷에 추가되면 보존 설정이 다음과 같이 적용됩니다.

- 개체에 대한 보존 설정이 지정된 경우 개체 수준 설정이 적용됩니다. 기본 버킷 설정은 무시됩니다.

- 개체에 대해 보존 설정을 지정하지 않으면 기본 버킷 설정이 적용됩니다(있는 경우).
  - 오브젝트 또는 버킷에 대해 보존 설정을 지정하지 않으면 S3 오브젝트 잠금으로 오브젝트가 보호되지 않습니다.
- 보존 설정이 적용되는 경우 오브젝트와 S3 사용자 정의 메타데이터는 모두 보호됩니다.

## 2. \* 개체 보존 및 삭제 \*

StorageGRID는 지정된 보존 기간 동안 보호된 각 개체의 복사본을 여러 개 저장합니다. 오브젝트 복사본 및 스토리지 위치의 정확한 수와 유형은 활성 ILM 정책의 규정 준수 규칙에 따라 결정됩니다. 보존 기한이 되기 전에 보호된 개체를 삭제할 수 있는지 여부는 보존 모드에 따라 다릅니다.

- 개체가 법적 보류 중인 경우 보존 모드에 관계없이 개체를 삭제할 수 없습니다.

레거시 준수 버킷을 계속 관리할 수 있습니까?

S3 오브젝트 잠금 기능은 이전 StorageGRID 버전에서 사용할 수 있었던 규정 준수 기능을 대체합니다. 이전 버전의 StorageGRID를 사용하여 준수 버킷을 생성한 경우 이러한 버켓의 설정을 계속 관리할 수 있지만, 더 이상 새로운 준수 버킷을 생성할 수 없습니다. 자세한 내용은 ["NetApp 기술 자료: StorageGRID 11.5에서 레거시 준수 버킷을 관리하는 방법"](#) 참조하십시오.

## S3 오브젝트 잠금 기본 보존 업데이트

버킷을 생성할 때 S3 오브젝트 잠금을 설정한 경우 버킷을 편집하여 기본 보존 설정을 변경할 수 있습니다. 기본 보존을 사용하거나 사용하지 않도록 설정하고 기본 보존 모드 및 보존 기간을 설정할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["모든 버킷 또는 루트 액세스 권한을 관리합니다"](#)입니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.
- S3 오브젝트 잠금은 StorageGRID 시스템에 대해 전역적으로 활성화되며 버킷을 생성할 때 S3 오브젝트 잠금을 활성화했습니다. [을 "S3 오브젝트 잠금을 사용하여 오브젝트를 보존합니다"](#) 참조하십시오.

단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
2. 테이블에서 버킷 이름을 선택합니다.
3. 버킷 세부 정보 페이지가 나타납니다.
4. Bucket options \* 탭에서 \* S3 Object Lock \* 아코디언을 선택합니다.
4. 이 버킷에 대해 \* 기본 보존 \* 을 활성화 또는 비활성화할 수 있습니다(선택 사항).

이 설정의 변경 사항은 버킷에 이미 있는 오브젝트 또는 자체 보존 기간이 있을 수 있는 오브젝트에는 적용되지 않습니다.

5. 기본 보존 \* 이 활성화된 경우 버킷에 대한 \* 기본 보존 모드 \* 를 지정합니다.

기본 보존 모드	설명
기대치를 설정합니다	<ul style="list-style-type: none"> <li>있는 사용자는 <code>s3:BypassGovernanceRetention</code> 권한이 요청 헤더를 사용하여 보존 설정을 무시할 수 `x-amz-bypass-governance-retention: true` 있습니다.</li> <li>이러한 사용자는 보존 기한이 되기 전에 개체 버전을 삭제할 수 있습니다.</li> <li>이러한 사용자는 개체의 보존 기간(Retain-until-date)을 증가, 감소 또는 제거할 수 있습니다.</li> </ul>
규정 준수	<ul style="list-style-type: none"> <li>보존 기한에 도달할 때까지 개체를 삭제할 수 없습니다.</li> <li>오브젝트의 보존 기한을 늘릴 수 있지만 줄일 수는 없습니다.</li> <li>개체의 보존 기한은 해당 날짜에 도달할 때까지 제거할 수 없습니다.</li> <li>참고*: 그리드 관리자가 규정 준수 모드를 사용할 수 있도록 허용해야 합니다.</li> </ul>

#### 6. 기본 보존 \* 이 활성화된 경우 버킷의 \* 기본 보존 기간 \* 을 지정합니다.

기본 보존 기간 \* 은 이 버킷에 추가된 새 오브젝트를 인제스트할 시간부터 보존 기간을 나타냅니다. 그리드 관리자가 설정한 테넌트의 최대 보존 기간보다 작거나 같은 값을 지정하십시오.

그리드 관리자가 테넌트를 생성할 때 1일에서 100년 사이의 값을 지정할 수 있는 `_MAXIMUM_RETENTION` 기간이 설정됩니다. `_default_retention` 기간을 설정하면 최대 보존 기간에 설정된 값을 초과할 수 없습니다. 필요한 경우 그리드 관리자에게 최대 보존 기간을 늘리거나 줄이도록 요청하십시오.

#### 7. 변경 내용 저장 \* 을 선택합니다.

#### 버킷 및 객체에 대한 StorageGRID CORS 구성

다른 도메인의 웹 애플리케이션에서 해당 버킷의 버킷 및 오브젝트에 액세스할 수 있도록 하려면 S3 버킷에 대해 CORS(Cross-Origin Resource Sharing)를 구성할 수 있습니다.

#### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- GET CORS 구성 요청의 경우 이 있는 사용자 그룹에 속해 ["모든 버킷을 관리하거나 모든 버킷을 봅니다"](#)입니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.
- PUT CORS 구성 요청의 경우 이 있는 사용자 그룹에 속해 ["모든 버킷 사용 권한을 관리합니다"](#)입니다. 이 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.
- 는 ["루트 액세스 권한"](#) 모든 CORS 구성 요청에 대한 액세스를 제공합니다.

#### 이 작업에 대해

CORS(Cross-origin Resource Sharing)는 한 도메인의 클라이언트 웹 애플리케이션이 다른 도메인의 리소스에 액세스할 수 있도록 하는 보안 메커니즘입니다. 예를 들어, 이라는 S3 버킷을 사용하여 그래픽을 저장한다고 가정해 `Images` 보겠습니다. 버킷에 대한 CORS를 구성하면 `Images` 해당 버킷의 이미지가 웹사이트에 표시되도록 할 수 `http://www.example.com` 있습니다.

버킷에 대해 **CORS**를 활성화합니다

단계

1. 텍스트 편집기를 사용하여 필요한 XML을 만듭니다. 이 예에서는 S3 버킷에 대해 CORS를 활성화하는 데 사용되는 XML을 보여 줍니다. 주요 내용은 다음과 같습니다.
  - 모든 도메인이 버킷으로 GET 요청을 보낼 수 있도록 허용합니다
  - 도메인에서만 GET, POST 및 삭제 요청을 보낼 수 있습니다 `http://www.example.com`
  - 모든 요청 헤더가 허용됩니다

```
<corsConfiguration
  xmlns="http://s3.amazonaws.com/doc/2020-10-22/">
  <corsRule>
    <allowedOrigin*></allowedOrigin>
    <allowedMethod>GET</allowedMethod>
    <allowedHeader*></allowedHeader>
  </corsRule>
  <corsRule>
    <allowedOrigin>http://www.example.com</allowedOrigin>
    <allowedMethod>GET</allowedMethod>
    <allowedMethod>POST</allowedMethod>
    <allowedMethod>DELETE</allowedMethod>
    <allowedHeader*></allowedHeader>
  </corsRule>
</corsConfiguration>
```

CORS 구성 XML에 대한 자세한 내용은 [을 참조하십시오 "AWS\(Amazon Web Services\) 설명서: Amazon Simple Storage Service 사용 설명서".](#)

2. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

3. 테이블에서 버킷 이름을 선택합니다.

버킷 세부 정보 페이지가 나타납니다.

4. Bucket access \* 탭에서 \* CORS(Cross-Origin Resource Sharing) \* 아코디언을 선택합니다.

5. CORS \* 활성화 확인란을 선택합니다.

6. 텍스트 상자에 CORS 구성 XML을 붙여 넣습니다.

7. 변경 내용 저장 \* 을 선택합니다.

**CORS** 설정을 수정합니다

단계

1. 텍스트 상자에서 CORS 구성 XML을 업데이트하거나 다시 시작하려면 \* Clear \* 를 선택합니다.
2. 변경 내용 저장 \* 을 선택합니다.

**CORS** 설정을 비활성화합니다

단계

1. CORS \* 활성화 확인란의 선택을 취소합니다.
2. 변경 내용 저장 \* 을 선택합니다.

관련 정보

["관리 인터페이스에 대한 StorageGRID CORS 구성"](#)

버킷에서 오브젝트를 삭제합니다

Tenant Manager를 사용하여 하나 이상의 버킷에서 오브젝트를 삭제할 수 있습니다.

고려 사항 및 요구 사항

이 단계를 수행하기 전에 다음 사항을 확인하십시오.

- 버킷에서 오브젝트를 삭제하면 StorageGRID은 StorageGRID 시스템의 모든 노드와 사이트에서 선택한 각 버킷의 모든 오브젝트 및 모든 오브젝트 버전을 영구적으로 제거합니다. StorageGRID는 관련된 개체 메타데이터도 제거합니다. 이 정보를 복구할 수 없습니다.
- 오브젝트, 오브젝트 복사본 및 동시 작업의 수에 따라 버킷의 모든 오브젝트를 삭제하는 데 몇 분, 일 또는 몇 주가 걸릴 수 있습니다.
- 버킷이 있는 경우 ["S3 오브젝트 잠금이 설정되었습니다"](#) \_years\_에 대해 \* 오브젝트 삭제: 읽기 전용 \* 상태로 남아 있을 수 있습니다.



S3 오브젝트 잠금을 사용하는 버킷은 모든 오브젝트에 대한 보존 날짜가 도달하고 법적 보관에 도달할 때까지 \* 오브젝트 삭제: 읽기 전용 \* 상태로 유지됩니다.

- 오브젝트가 삭제될 때 버킷의 상태는 \* 오브젝트 삭제: 읽기 전용 \*입니다. 이 상태에서는 버킷에 새 오브젝트를 추가할 수 없습니다.
- 모든 오브젝트가 삭제된 경우 버킷은 읽기 전용 상태로 유지됩니다. 다음 중 하나를 수행할 수 있습니다.
  - 쓰기 모드로 버킷을 되돌리십시오. 새 오브젝트에 이 버킷을 재사용하십시오
  - 버킷을 삭제합니다
  - 나중에 사용할 수 있도록 버킷을 읽기 전용 모드로 유지합니다
- 버킷에 오브젝트 버전 관리가 활성화된 경우 버킷 작업에서 오브젝트 삭제 작업을 사용하여 StorageGRID 11.8 이상에서 생성된 삭제 표식을 제거할 수 있습니다.
- 버킷에 개체 버전 관리가 활성화된 경우 오브젝트 삭제 작업은 StorageGRID 11.7 이하에서 생성된 삭제 표식을 제거하지 않습니다. 이 버킷의 오브젝트 삭제에 대한 정보를 ["S3 버전 오브젝트 삭제 방법"](#) 참조하십시오.
- 을 사용하는 경우 ["교차 그리드 복제"](#) 다음 사항에 유의하십시오.
  - 이 옵션을 사용해도 다른 그리드의 버킷에서 오브젝트가 삭제되지 않습니다.
  - 소스 버킷에 대해 이 옵션을 선택하면 다른 그리드의 대상 버킷에 오브젝트를 추가하면 \* 크로스 그리드 복제 실패 \* 경고가 트리거됩니다. 보장하지 못할 경우, 모든 버킷 오브젝트를 삭제하기 전에 다른 그리드의 버킷에 객체를 추가하지 ["크로스 그리드 복제를 비활성화합니다"](#) 않습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 "지원되는 웹 브라우저"입니다.
- 이 있는 사용자 그룹에 속해 "루트 액세스 권한"입니다. 이 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.

#### 단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

Bucket 페이지가 나타나고 기존의 모든 S3 버킷을 표시합니다.

2. 특정 버킷의 \* 작업 \* 메뉴 또는 세부 정보 페이지를 사용합니다.

#### 작업 메뉴

- a. 오브젝트를 삭제할 각 버킷의 확인란을 선택합니다.
- b. Actions \* > \* Delete objects in bucket \* 을 선택합니다.

#### 세부 정보 페이지

- a. 버킷 이름을 선택하여 세부 정보를 표시합니다.
- b. 버킷에서 오브젝트 삭제 \* 를 선택합니다.

3. 확인 대화 상자가 나타나면 세부 정보를 검토하고 \* 예 \* 를 입력한 다음 \* 확인 \* 을 선택합니다.

4. 삭제 작업이 시작될 때까지 기다립니다.

#### 몇 분 후:

- 버킷 세부 정보 페이지에 노란색 상태 배너가 나타납니다. 진행률 표시줄은 삭제된 개체의 비율을 나타냅니다.
- \* (읽기 전용) \* 버킷 세부 정보 페이지에서 버킷 이름 뒤에 나타납니다.
- \* (오브젝트 삭제: 읽기 전용) \* 버킷 페이지의 버킷 이름 옆에 표시됩니다.

Buckets > my-bucket

**Success** Starting to delete objects from one bucket. X

## my-bucket (read-only)

Region: us-east-1

Date created: 2022-12-14 10:09:50 MST

Object count: 3

[View bucket contents in Experimental S3 Console](#) 

[Delete bucket](#)

**⚠ All bucket objects are being deleted**  
StorageGRID is deleting all copies of the objects in this bucket, which might take days or weeks. While objects are being deleted, the bucket is read only. To stop the operation, select **Stop deleting objects**. You cannot restore objects that have already been deleted.

0% (0 of 3 objects deleted)

[Stop deleting objects](#)

- 작업이 실행되는 동안 필요에 따라 \* 개체 삭제 중지 \* 를 선택하여 프로세스를 중단합니다. 그런 다음 필요에 따라 \* 버킷 안의 오브젝트 삭제 \* 를 선택하여 프로세스를 재개합니다.

오브젝트 삭제 중지 \* 를 선택하면 버킷이 쓰기 모드로 돌아가지만 삭제된 오브젝트는 액세스하거나 복구할 수 없습니다.

- 작업이 완료될 때까지 기다립니다.

버킷이 비어 있으면 상태 배너가 업데이트되지만 버킷은 읽기 전용으로 유지됩니다.

Buckets > my-bucket

## my-bucket (read-only)

Region: us-east-1

Date created: 2022-12-14 10:09:50 MST

Object count: 0

[View bucket contents in Experimental S3 Console](#)

[Delete bucket](#)

Bucket is empty but is still read-only.

This bucket is now empty.

- To remove this bucket, select **Delete bucket**.
- To return this bucket to write mode so it can be reused, select **Stop deleting objects**.

[Stop deleting objects](#)

7. 다음 중 하나를 수행합니다.

- 버킷을 읽기 전용 모드로 유지하려면 페이지를 종료하십시오. 예를 들어, 빈 버킷을 읽기 전용 모드로 유지하여 나중에 사용할 수 있도록 버킷 이름을 예약할 수 있습니다.
- 버킷을 삭제합니다. 단일 버킷을 삭제하거나 버킷 페이지를 반환하려면 \* 버킷 삭제 \*를 선택하고 둘 이상의 버킷을 제거하려면 \* 작업 \* > \* 삭제 \* 버킷을 선택할 수 있습니다.



모든 오브젝트를 삭제한 후 버전이 있는 버킷을 삭제할 수 없는 경우 삭제 마커가 남아 있을 수 있습니다. 버킷을 삭제하려면 나머지 삭제 마커를 모두 제거해야 합니다.

- 쓰기 모드로 버킷을 되돌리십시오. 그리고 필요에 따라 새 오브젝트에 버킷을 다시 사용할 수 있습니다. 단일 버킷에 대해 \* 오브젝트 삭제 중지 \*를 선택하거나 버킷 페이지로 돌아가서 \* 작업 \* > \* 오브젝트 삭제 중지 \*를 선택하여 둘 이상의 버킷에 대해 삭제할 수 있습니다.

### S3 버킷을 삭제합니다

테넌트 관리자를 사용하여 비어 있는 하나 이상의 S3 버킷을 삭제할 수 있습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["모든 버킷 또는 루트 액세스 권한을 관리합니다"](#)입니다. 이러한 권한은 그룹 또는 버킷 정책의 사용 권한 설정을 재정의합니다.
- 삭제할 버킷이 비어 있습니다. 삭제하려는 버킷이 \_NOT\_EMIVE인 경우["버킷에서 오브젝트를 삭제합니다"](#)

이 작업에 대해

다음 지침은 Tenant Manager를 사용하여 S3 버킷을 삭제하는 방법을 설명합니다. 또는 를 사용하여 S3 버킷을 삭제할 수도 ["테넌트 관리 API"](#) ["S3 REST API"](#)입니다.

S3 버킷에 오브젝트, 비현재 오브젝트 버전이 포함되어 있거나 마커를 삭제할 수 없습니다. S3 버전 개체가 삭제되는 방법에 대한 자세한 내용은 [을 참조하십시오."오브젝트 삭제 방법"](#).

## 단계

1. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

Bucket 페이지가 나타나고 기존의 모든 S3 버킷을 표시합니다.

2. 특정 버킷의 \* 작업 \* 메뉴 또는 세부 정보 페이지를 사용합니다.

### 작업 메뉴

- a. 삭제할 각 버킷의 확인란을 선택합니다.
- b. Actions \* > \* Delete Bucket \* 을 선택합니다.

### 세부 정보 페이지

- a. 버킷 이름을 선택하여 세부 정보를 표시합니다.
- b. 버킷 삭제 \* 를 선택합니다.

3. 확인 대화 상자가 나타나면 \* 예 \* 를 선택합니다.

StorageGRID는 각 버킷이 비어 있음을 확인한 다음 각 버킷을 삭제합니다. 이 작업은 몇 분 정도 걸릴 수 있습니다.

버킷이 비어 있지 않으면 오류 메시지가 나타납니다. 버킷을 삭제하려면 먼저 해야 ["버킷의 모든 오브젝트와 삭제 표식을 삭제합니다"](#)합니다.

## S3 콘솔 사용

S3 콘솔을 사용하여 S3 버킷의 오브젝트를 보고 관리할 수 있다.

S3 콘솔을 통해 다음을 수행할 수 있습니다.

- 업로드, 다운로드, 이름 바꾸기, 복사, 이동, 개체를 삭제합니다
- 개체 버전을 보고, 되돌리고, 다운로드하고, 삭제합니다
- 접두사로 오브젝트를 검색합니다
- 오브젝트 태그 관리
- 개체 메타데이터를 봅니다
- 보기, 만들기, 이름 바꾸기, 복사, 이동, 폴더를 삭제합니다

S3 콘솔은 가장 일반적인 경우에 향상된 사용자 환경을 제공합니다. 모든 상황에서 CLI 또는 API 작업을 대체하도록 설계되지 않았습니다.



S3 콘솔을 사용할 때 작업 시간이 너무 오래 걸리는 경우(예: 분 또는 시간) 다음을 고려하십시오.

- 선택한 개체의 수를 줄입니다
- 비그래픽(API 또는 CLI) 방법을 사용하여 데이터에 액세스합니다

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 개체를 관리하려면 루트 액세스 권한이 있는 사용자 그룹에 속해 있어야 합니다. 또는 S3 콘솔 사용 탭 권한 및 모든 버킷 보기 권한 또는 모든 버킷 관리 권한이 있는 사용자 그룹에 속하게 됩니다. 을 ["테넌트 관리 권한"](#)참조하십시오.
- 사용자에 대해 S3 그룹 또는 버킷 정책이 구성되었습니다. 보다 ["버킷 및 그룹 액세스 정책을 사용합니다"](#).
- 사용자의 액세스 키 ID와 비밀 액세스 키를 알고 있습니다. 필요한 경우 이 정보가 포함된 파일이 있을 수도 `.csv` 있습니다. 를 ["액세스 키 생성에 대한 지침"](#)참조하십시오.

단계

- 저장소 > 버킷 > \*버킷 이름\*을 선택합니다.
- S3 콘솔 탭을 선택합니다.
- 액세스 키 ID와 비밀 액세스 키를 필드에 붙여 넣습니다. 그렇지 않으면 \* Upload access keys \* 를 선택하고 `.csv` 파일을 선택합니다.
- 로그인 \* 을 선택합니다.
- 버킷 객체의 테이블이 나타납니다. 필요에 따라 개체를 관리할 수 있습니다.

추가 정보

- \* 접두사로 검색 \*: 접두사 검색 기능은 현재 폴더와 관련된 특정 단어로 시작하는 오브젝트만 검색합니다. 검색에는 다른 위치에 단어가 포함된 개체가 포함되지 않습니다. 이 규칙은 폴더 내의 개체에도 적용됩니다. 예를 들어, 에 대한 검색은 `folder1/folder2/somefile-` 폴더 내에 있는 객체를 반환하고 `folder1/folder2/` 단어로 시작합니다 `somefile-`.
- \* 드래그 앤 드롭 \*: 컴퓨터의 파일 관리자에서 S3 콘솔로 파일을 끌어다 놓을 수 있습니다. 그러나 폴더를 업로드할 수는 없습니다.
- \* 폴더에 대한 작업 \*: 폴더를 이동, 복사 또는 이름 바꾸면 폴더의 모든 개체가 한 번에 하나씩 업데이트되어 시간이 걸릴 수 있습니다.
- \*버킷 버전 관리가 비활성화된 경우 영구 삭제 \*: 버전 관리가 비활성화된 상태로 버킷의 오브젝트를 덮어쓰거나 삭제하면 작업이 영구적입니다. 을 ["버킷의 오브젝트 버전 관리를 변경합니다"](#)참조하십시오.

## S3 플랫폼 서비스 관리

### S3 플랫폼 서비스

플랫폼 서비스 개요 및 고려 사항

플랫폼 서비스를 구현하기 전에 이러한 서비스 사용에 대한 개요 및 고려 사항을 검토하십시오.

S3에 대한 자세한 내용은 ["S3 REST API 사용"](#) 참조하십시오.

## 플랫폼 서비스 개요

StorageGRID 플랫폼 서비스를 사용하면 S3 오브젝트 및 오브젝트 메타데이터의 이벤트 알림과 복사본을 외부 대상에 보낼 수 있으므로 하이브리드 클라우드 전략을 구현할 수 있습니다.

플랫폼 서비스의 대상 위치는 일반적으로 StorageGRID 구축과 외부적이기 때문에 플랫폼 서비스는 데이터에 대한 외부 스토리지 리소스, 알림 서비스 및 검색 또는 분석 서비스를 사용하여 얻을 수 있는 성능과 유연성을 제공합니다.

단일 S3 버킷에 대해 모든 플랫폼 서비스 조합을 구성할 수 있습니다. 예를 들어, 및 ["알림"](#) StorageGRID S3 버킷에서 특정 오브젝트를 Amazon S3(Simple Storage Service)에 미러링하는 동시에 각 오브젝트에 대한 알림을 타사 모니터링 애플리케이션에 전송하여 AWS 비용을 추적할 수 있도록 할 ["CloudMirror 서비스"](#) 수 있습니다.



그리드 관리자 또는 그리드 관리 API를 사용하여 StorageGRID 관리자가 각 테넌트 계정에 대해 플랫폼 서비스 사용을 활성화해야 합니다.

## 플랫폼 서비스 구성 방법

플랫폼 서비스는 다음을 사용하여 구성하는 외부 엔드포인트와 통신합니다. ["테넌트 관리자"](#) 또는 ["테넌트 관리 API"](#). 각 엔드포인트는 StorageGRID S3 버킷, Amazon Web Services 버킷, Amazon SNS 주제, 웹훅 엔드포인트 또는 로컬, AWS 또는 다른 곳에서 호스팅되는 Elasticsearch 클러스터와 같은 외부 대상을 나타냅니다.

외부 끝점을 만든 후 버킷에 XML 구성은 추가하여 버킷에 대한 플랫폼 서비스를 활성화할 수 있습니다. XML 구성은 버킷이 작업해야 하는 오브젝트, 버킷이 취해야 하는 조치 및 버킷이 서비스에 사용해야 하는 엔드포인트를 식별합니다.

구성할 각 플랫폼 서비스에 대해 별도의 XML 구성은 추가해야 합니다. 예를 들면 다음과 같습니다.

- 키가 로 시작하는 모든 오브젝트를 Amazon S3 버킷에 복제하려면 복제 구성은 소스 버킷에 추가해야 합니다.  
`/images`
- 이러한 객체가 버킷에 저장될 때 알림을 보내려면 알림 구성은 추가해야 합니다.
- 이러한 객체의 메타데이터를 인덱싱하려면 검색 통합을 구현하는 데 사용되는 메타데이터 알림 구성은 추가해야 합니다.

구성 XML의 형식은 StorageGRID 플랫폼 서비스를 구현하는 데 사용되는 S3 REST API를 통해 제어됩니다.

플랫폼 서비스	S3 REST API	을 참조하십시오
CloudMirror 복제	<ul style="list-style-type: none"><li>• <code>GetBucketReplication</code> 을 참조하십시오</li><li>• <code>PutBucketReplication</code> 을 참조하십시오</li></ul>	<ul style="list-style-type: none"><li>• <a href="#">"CloudMirror 복제"</a></li><li>• <a href="#">"버킷 작업"</a></li></ul>
알림	<ul style="list-style-type: none"><li>• <code>GetBucketNotificationConfiguration</code> 을 참조하십시오</li><li>• <code>PutBucketNotificationConfiguration</code> 을 참조하십시오</li></ul>	<ul style="list-style-type: none"><li>• <a href="#">"알림"</a></li><li>• <a href="#">"버킷 작업"</a></li></ul>
검색 통합	<ul style="list-style-type: none"><li>• Bucket 메타데이터 알림 구성 가져오기</li><li>• Put Bucket 메타데이터 알림 구성</li></ul>	<ul style="list-style-type: none"><li>• <a href="#">"검색 통합"</a></li><li>• <a href="#">"StorageGRID 사용자 정의 작업"</a></li></ul>

## 플랫폼 서비스 사용에 대한 고려 사항

고려 사항	세부 정보
대상 엔드포인트 모니터링	각 대상 끝점의 가용성을 모니터링해야 합니다. 대상 끝점에 대한 연결이 오랜 시간 동안 손실되고 요청의 백로그가 많은 경우 StorageGRID에 대한 추가 클라이언트 요청(예: PUT 요청)이 실패합니다. 엔드포인트에 연결할 수 있게 되면 실패한 요청을 다시 시도해야 합니다.
대상 끝점 임계치 조절	<p>요청이 전송되는 속도가 대상 엔드포인트에서 요청을 수신할 수 있는 속도를 초과하는 경우 StorageGRID 소프트웨어는 버킷에 대한 수신 S3 요청을 스로틀할 수 있습니다. 임계치 조절은 대상 끝점으로 보내려고 기다리는 요청의 백로그가 있는 경우에만 발생합니다.</p> <p>단, 들어오는 S3 요청의 실행 시간이 더 오래 걸린다는 점을 알 수 있습니다. 속도가 현저히 느린 성능을 감지하기 시작하는 경우 수집 속도를 줄이거나 용량이 더 큰 엔드포인트를 사용해야 합니다. 요청 백로그가 계속 증가하는 경우 PUT 요청과 같은 클라이언트 S3 작업이 결국 실패합니다.</p> <p>CloudMirror 요청은 일반적으로 검색 통합 또는 이벤트 알림 요청보다 더 많은 데이터 전송을 포함하므로 대상 엔드포인트의 성능에 영향을 받을 가능성이 더 높습니다.</p>
주문 보증	<p>StorageGRID은 사이트 내의 개체에 대한 작업을 주문할 수 있도록 보장합니다. 객체에 대한 모든 작업이 동일한 사이트 내에 있는 한 최종 객체 상태(복제의 경우)는 항상 StorageGRID의 상태와 동일합니다.</p> <p>StorageGRID는 StorageGRID 사이트 전체에서 작업이 수행되는 경우 요청을 주문하기 위해 최선의 노력을 다하고 있습니다. 예를 들어 처음에 사이트 A에 오브젝트를 작성한 다음 나중에 사이트 B에서 동일한 오브젝트를 덮어쓰는 경우 CloudMirror에서 대상 버킷에 복제한 최종 오브젝트는 새로운 오브젝트일 수 없습니다.</p>
ILM 기반 오브젝트 삭제	<p>AWS CRR 및 Amazon Simple Notification Service의 삭제 동작과 일치시키기 위해 StorageGRID ILM 규칙으로 인해 소스 버킷의 오브젝트가 삭제될 때 CloudMirror 및 이벤트 알림 요청이 전송되지 않습니다. 예를 들어 ILM 규칙이 14일 후에 개체를 삭제하는 경우 CloudMirror 또는 이벤트 알림 요청이 전송되지 않습니다.</p> <p>반면, 검색 통합 요청은 ILM로 인해 객체가 삭제될 때 전송됩니다.</p>

고려 사항	세부 정보
Kafka 엔드포인트 사용	<p>Kafka 엔드포인트의 경우 상호 TLS는 지원되지 않습니다. 그 결과, Kafka 브로커 구성에서 로 설정한 <code>required</code> 경우 <code>ssl.client.auth</code> Kafka 엔드포인트 구성 문제가 발생할 수 있습니다.</p> <p>Kafka 엔드포인트 인증은 다음과 같은 인증 유형을 사용합니다. 이러한 유형은 Amazon SNS와 같은 다른 엔드포인트의 인증에 사용되는 유형과는 다르며 사용자 이름 및 암호 자격 증명이 필요합니다.</p> <ul style="list-style-type: none"> <li>• SASL/일반</li> <li>• SASL/SCRAM-SHA-256</li> <li>• SASL/SCRAM-SHA-512</li> <li>• 참고: * 구성된 스토리지 프록시 설정은 Kafka 플랫폼 서비스 엔드포인트에 적용되지 않습니다.</li> </ul>

#### CloudMirror 복제 서비스 사용에 대한 고려 사항

고려 사항	세부 정보
복제 상태입니다	StorageGRID는 헤더를 지원하지 <code>x-amz-replication-status</code> 않습니다.
개체 크기	<p>CloudMirror 복제 서비스를 통해 대상 버킷에 복제할 수 있는 개체의 최대 크기는 5TiB이며, 이는 <code>maximum_supported_object</code> 크기와 같습니다.</p> <ul style="list-style-type: none"> <li>• 참고 *: 단일 PutObject 작업의 <code>maximum_recommended_size</code>는 5GiB(5,368,709,120바이트)입니다. 5GiB보다 큰 객체가 있는 경우 대신 멀티파트 업로드를 사용합니다.</li> </ul>
버킷 버전 관리 및 버전 ID	<p>StorageGRID의 소스 S3 버킷에서 버전 관리가 활성화된 경우 대상 버킷의 버전 관리도 활성화해야 합니다.</p> <p>버전 관리를 사용할 때는 S3 프로토콜의 제한으로 인해 대상 버킷에서 오브젝트 버전 순서가 CloudMirror 서비스에 의해 보장되지 않는 것이 가장 좋습니다.</p> <ul style="list-style-type: none"> <li>• 참고 *: StorageGRID의 소스 버킷에 대한 버전 ID는 대상 버킷의 버전 ID와 관련이 없습니다.</li> </ul>
개체 버전에 태그 달기	<p>CloudMirror 서비스는 S3 프로토콜의 제한으로 인해 버전 ID를 제공하는 PutObjectTagging 또는 DeleteObjectTagging 요청을 복제하지 않습니다. 소스 및 대상의 버전 ID는 관련이 없으므로 특정 버전 ID에 대한 태그 업데이트를 복제할 수 없습니다.</p> <p>반면, CloudMirror 서비스는 버전 ID를 지정하지 않는 PutObjectTagging 요청이나 DeleteObjectTagging 요청을 복제합니다. 이러한 요청은 최신 키의 태그(또는 버킷의 버전이 지정된 경우 최신 버전)를 업데이트합니다. 태그가 있는 일반 베스트(업데이트 태그 지정 안 함)도 복제됩니다.</p>

고려 사항	세부 정보
멀티파트 업로드 및 ETag 값	여러 부분 업로드를 사용하여 업로드한 개체를 미러링할 때 CloudMirror 서비스는 해당 파트를 보존하지 않습니다. 따라서 ETag 대칭 복사된 개체의 값은 ETag 원래 개체의 값과 다릅니다.
SSE-C로 암호화된 오브젝트(고객이 제공한 키를 사용한 서버측 암호화)	CloudMirror 서비스는 SSE-C로 암호화된 객체를 지원하지 않습니다. CloudMirror 복제를 위해 소스 버킷으로 객체를 수집하려고 할 때 SSE-C 요청 헤더를 포함하면 작업이 실패합니다.
S3 오브젝트 잠금이 활성화된 버킷	S3 오브젝트 잠금이 활성화된 소스 또는 대상 버킷에는 복제가 지원되지 않습니다.

#### CloudMirror 복제 서비스를 이해합니다

StorageGRID가 버킷에 추가된 지정된 오브젝트를 하나 이상의 외부 대상 버킷에 복제하도록 하려는 경우 S3 버킷에 대해 CloudMirror 복제를 활성화할 수 있습니다.

예를 들어, CloudMirror 복제를 사용하여 특정 고객 레코드를 Amazon S3에 미러링한 다음 AWS 서비스를 활용하여 데이터에 대한 분석을 수행할 수 있습니다.



소스 버킷에 S3 오브젝트 잠금이 설정된 경우 CloudMirror 복제가 지원되지 않습니다.

#### CloudMirror 및 ILM

CloudMirror 복제는 그리드의 활성 ILM 정책과 독립적으로 작동합니다. CloudMirror 서비스는 소스 버킷에 저장된 객체를 복제하여 가능한 한 빨리 대상 버킷에 제공합니다. 오브젝트 수집의 성공 시 복제된 오브젝트 제공이 트리거됩니다.

#### CloudMirror 및 교차 그리드 복제

CloudMirror 복제는 교차 그리드 복제 기능과의 중요한 유사점과 차이점이 있습니다. 을 ["교차 그리드 복제와 CloudMirror 복제를 비교합니다"](#)참조하십시오.

#### CloudMirror 및 S3 버킷

CloudMirror 복제는 일반적으로 외부 S3 버킷을 대상으로 사용하도록 구성됩니다. 그러나 다른 StorageGRID 배포나 S3 호환 서비스를 사용하도록 복제를 구성할 수도 있습니다.

#### 기존 버킷

기존 버킷에 대해 CloudMirror 복제를 활성화하면 해당 버킷에 추가된 새 오브젝트만 복제됩니다. 버킷의 기존 객체는 복제되지 않습니다. 기존 오브젝트의 복제를 강제로 수행하려면 오브젝트 복사를 수행하여 기존 오브젝트의 메타데이터를 업데이트할 수 있습니다.



CloudMirror 복제를 사용하여 Amazon S3 대상으로 오브젝트를 복사하는 경우 Amazon S3는 각 PUT 요청 헤더 내의 사용자 정의 메타데이터 크기를 2KB로 제한합니다. 객체에 2KB보다 큰 사용자 정의 메타데이터가 있는 경우 해당 객체가 복제되지 않습니다.

## 다중 대상 버킷

단일 버킷의 오브젝트를 여러 대상 버킷으로 복제하려면 복제 구성 XML에서 각 규칙의 대상을 지정합니다. 객체를 둘 이상의 버킷에 동시에 복제할 수 없습니다.

## 버전 또는 버전 해제된 버킷

버전 또는 버전이 지정되지 않은 버킷에서 CloudMirror 복제를 구성할 수 있습니다. 대상 버킷의 버전을 지정하거나 버전을 취소할 수 있습니다. 버전 및 비버전 버킷의 모든 조합을 사용할 수 있습니다. 예를 들어 버전이 지정되지 않은 소스 버킷의 대상으로 버전 관리가 지정된 버킷을 지정하거나 그 반대로 지정할 수 있습니다. 버전이 지정되지 않은 버킷 간에 복제할 수도 있습니다.

## 삭제, 복제 루프 및 이벤트입니다

### 삭제 동작

는 Amazon S3 서비스, 지역 간 복제(CRR)의 삭제 동작과 동일합니다. 소스 버킷에서 오브젝트를 삭제하면 대상에서 복제된 오브젝트가 삭제되지 않습니다. 소스 및 대상 버킷의 버전이 모두 지정된 경우 삭제 마커가 복제됩니다. 대상 버킷의 버전이 지정되지 않은 경우 소스 버킷에서 오브젝트를 삭제해도 삭제 마커가 대상 버킷에 복제되거나 대상 오브젝트를 삭제하지 않습니다.

### 복제 루프로부터 보호합니다

오브젝트가 대상 버킷에 복제되면 StorageGRID는 객체를 "복제본"으로 표시합니다. 대상 StorageGRID 버킷은 복제본으로 표시된 객체를 다시 복제하지 않으므로 실수로 인한 복제 루프가 발생하지 않습니다. 이 복제 마킹은 StorageGRID 내부에 있으며 Amazon S3 버킷을 대상으로 사용할 때 AWS CRR을 활용할 수 없습니다.



복제본을 표시하는 데 사용되는 사용자 지정 헤더는입니다 `x-ntap-sg-replica`. 이 표시는 계단식 미러를 방지합니다. StorageGRID는 두 그리드 간의 양방향 CloudMirror를 지원합니다.

### 목적지 버킷의 이벤트

목적지 버킷에서 이벤트의 고유성과 순서는 보장되지 않습니다. 전송 성공을 보장하기 위해 수행된 작업의 결과로 소스 객체의 동일한 복제본이 두 개 이상 대상에 제공될 수 있습니다. 드물지만 둘 이상의 서로 다른 StorageGRID 사이트에서 동일한 객체가 동시에 업데이트되는 경우 대상 버킷의 작업 순서가 소스 버킷의 이벤트 순서와 일치하지 않을 수 있습니다.

### 버킷에 대한 알림을 이해합니다

StorageGRID 지정된 이벤트에 대한 알림을 대상 Kafka 클러스터, 웹훅 엔드포인트 또는 Amazon Simple Notification Service로 보내도록 하려면 S3 버킷에 대한 이벤트 알림을 활성화할 수 있습니다.

예를 들어, 버킷에 추가된 각 오브젝트에 대해 관리자에게 경고가 전송되도록 구성할 수 있습니다. 여기서 객체는 중요한 시스템 이벤트와 연결된 로그 파일을 나타냅니다.

이벤트 알림은 알림 구성에 지정된 대로 소스 버킷에서 생성되며 대상으로 전달됩니다. 개체와 관련된 이벤트가 성공하면 해당 이벤트에 대한 알림이 생성되고 배달 대기 상태가 됩니다.

알림의 고유성과 순서는 보장되지 않습니다. 전송 성공을 보장하기 위해 수행된 작업의 결과로 하나 이상의 이벤트 알림이 대상에 전달될 수 있습니다. 그리고 납품이 비동기식이기 때문에, 특히 서로 다른 StorageGRID 사이트에서 발생하는 작업의 경우, 대상에서 알림의 시간 순서가 소스 버킷의 이벤트 순서와 일치한다고 보장할 수 없습니다. 이벤트 메시지에서 키를 사용하여 Amazon S3 설명서에 설명된 대로 특정 객체에 대한 이벤트 순서를 결정할 수 `sequencer` 있습니다.

StorageGRID 이벤트 알림은 Amazon S3 API를 따르지만 몇 가지 제한 사항을 따릅니다.

- 지원되는 이벤트 유형은 다음과 같습니다.
  - S3:ObjectCreated:
    - S3:오브젝트 생성:PUT
    - S3:오브젝트 작성:우편
    - S3:오브젝트 생성:복사
  - S3:ObjectCreated:CompleteMultipartUpload
  - S3:ObjectRemoved:
    - S3:ObjectRemoved:Delete
    - S3:ObjectRemoved:DeleteMarkerCreated
  - S3:ObjectRestore:게시
- StorageGRID에서 보낸 이벤트 알림은 표준 JSON 형식을 사용하지만 일부 키는 포함하지 않으며 표에 나와 있는 대로 다른 키의 특정 값을 사용합니다.

키 이름	StorageGRID 값
이벤트 소스	sgws:s3
awsRegion	_ 포함되지 않음 _
X-amz-id-2	_ 포함되지 않음 _
ARN	urn:sgws:s3:::bucket_name

검색 통합 서비스를 이해합니다

오브젝트 메타데이터에 외부 검색 및 데이터 분석 서비스를 사용하려는 경우 S3 버킷에 대한 검색 통합을 활성화할 수 있습니다.

검색 통합 서비스는 개체가 생성 또는 삭제되거나 해당 메타데이터 또는 태그가 업데이트될 때마다 S3 오브젝트 메타데이터를 대상 엔드포인트에 자동으로 비동기적으로 전송하는 사용자 지정 StorageGRID 서비스입니다. 그런 다음 대상 서비스에서 제공하는 정교한 검색, 데이터 분석, 시각화 또는 머신 러닝 도구를 사용하여 오브젝트 데이터를 검색, 분석 및 분석할 수 있습니다.

예를 들어, S3 오브젝트 메타데이터를 원격 Elasticsearch 서비스로 전송하도록 버킷을 구성할 수 있습니다. 그런 다음 Elasticsearch를 사용하여 버킷에 대한 검색을 수행하고 객체 메타데이터에 있는 패턴에 대한 정교한 분석을 수행할 수 있습니다.

S3 오브젝트 잠금이 활성화된 버킷에서 Elasticsearch 통합을 구성할 수 있지만, 오브젝트의 S3 오브젝트 잠금 메타데이터(Retain To Date 및 Legal Hold 상태 포함)는 Elasticsearch로 전송되는 메타데이터에 포함되지 않습니다.



검색 통합 서비스가 개체 메타데이터를 대상으로 전송하도록 하기 때문에 해당 구성 XML을 "\_metadata\_notification" 구성 XML"이라고 합니다. 이 구성 XML은 enable\_event\_notifications에 사용되는 "notification configuration xml"과 다릅니다.

## 검색 통합 및 S3 버킷

버전 관리되거나 버전이 지정되지 않은 모든 버킷에 대해 검색 통합 서비스를 활성화할 수 있습니다. 검색 통합은 메타데이터 알림 구성 XML을 작업할 개체 및 개체 메타데이터에 대한 대상을 지정하는 버킷과 연결하여 구성됩니다.

메타데이터 알림은 버킷 이름, 개체 이름 및 버전 ID(있는 경우)로 명명된 JSON 문서 형식으로 생성됩니다. 각 메타데이터 알림에는 개체의 모든 태그 및 사용자 메타데이터 외에도 개체에 대한 표준 시스템 메타데이터 세트가 포함되어 있습니다.

 태그 및 사용자 메타데이터의 경우 StorageGRID는 날짜 및 숫자를 Elasticsearch에 문자열 또는 S3 이벤트 알림으로 전달합니다. 이러한 문자열을 날짜 또는 숫자로 해석하도록 Elasticsearch를 구성하려면 동적 필드 맵핑 및 날짜 형식 맵핑에 대한 Elasticsearch 지침을 따르십시오. 검색 통합 서비스를 구성하기 전에 인덱스에서 동적 필드 맵핑을 활성화해야 합니다. 문서가 인덱싱된 후에는 인덱스에서 문서의 필드 형식을 편집할 수 없습니다.

### 알림을 검색합니다

메타데이터 알림은 다음과 같은 경우에 생성되고 배달 대기 상태가 됩니다.

- 객체가 생성됩니다.
- 그리드의 ILM 정책 작업으로 인해 오브젝트가 삭제된 경우를 포함하여 오브젝트가 삭제됩니다.
- 오브젝트 메타데이터 또는 태그가 추가, 업데이트 또는 삭제됩니다. 메타데이터 및 태그의 전체 집합은 항상 변경된 값뿐만 아니라 업데이트 시 전송됩니다.

메타데이터 알림 구성 XML을 버킷에 추가하면 생성한 새 개체 및 데이터, 사용자 메타데이터 또는 태그를 업데이트하여 수정하는 모든 개체에 대한 알림이 전송됩니다. 하지만 이미 버킷에 있는 객체에 대해서는 알림이 전송되지 않습니다. 버킷의 모든 오브젝트에 대한 오브젝트 메타데이터가 대상으로 전송되도록 하려면 다음 중 하나를 수행해야 합니다.

- 버킷을 생성한 후 개체를 추가하기 전에 즉시 검색 통합 서비스를 구성합니다.
- 메타데이터 알림 메시지가 대상으로 전송되도록 버킷에 이미 있는 모든 객체에 대해 작업을 수행합니다.

## 검색 통합 서비스 및 Elasticsearch

StorageGRID 검색 통합 서비스는 Elasticsearch 클러스터를 대상으로 지원합니다. 다른 플랫폼 서비스와 마찬가지로 대상은 서비스의 구성 XML에서 URN이 사용되는 끝점에서 지정됩니다. 이를 사용하여 ["NetApp 상호 운용성 매트릭스 툴"](#) 지원되는 Elasticsearch 버전을 확인합니다.

### 플랫폼 서비스 끝점을 관리합니다

플랫폼 서비스 끝점을 구성합니다

버킷에 대한 플랫폼 서비스를 구성하려면 먼저 플랫폼 서비스의 대상으로 하나 이상의 엔드포인트를 구성해야 합니다.

플랫폼 서비스에 대한 액세스는 StorageGRID 관리자가 테넌트 단위로 사용하도록 설정합니다. 플랫폼 서비스 끝점을 만들거나 사용하려면 스토리지 노드가 외부 끝점 리소스에 액세스할 수 있도록 네트워킹이 구성된 그리드에서 끝점 관리 또는 루트 액세스 권한이 있는 테넌트 사용자여야 합니다. 단일 테넌트의 경우 최대 500개의 플랫폼 서비스 엔드포인트를 구성할 수 있습니다. 자세한 내용은 StorageGRID 관리자에게 문의하십시오.

## 플랫폼 서비스 엔드포인트란 무엇입니까?

플랫폼 서비스 끝점은 StorageGRID가 외부 대상에 액세스하는 데 필요한 정보를 지정합니다.

예를 들어, StorageGRID 버킷에서 Amazon S3 버킷으로 오브젝트를 복제하려는 경우 StorageGRID에서 Amazon의 대상 버킷에 액세스하는 데 필요한 정보 및 자격 증명이 포함된 플랫폼 서비스 엔드포인트를 생성합니다.

각 플랫폼 서비스 유형에는 고유한 엔드포인트가 필요하므로 사용하려는 각 플랫폼 서비스에 대해 하나 이상의 엔드포인트를 구성해야 합니다. 플랫폼 서비스 끝점을 정의한 후 서비스를 활성화하는 데 사용되는 구성 XML에서 끝점의 URN을 대상으로 사용합니다.

둘 이상의 소스 버킷에 대해 목적지와 동일한 끝점을 사용할 수 있습니다. 예를 들어, 여러 버킷에서 검색을 수행할 수 있도록 여러 소스 버킷을 구성하여 동일한 검색 통합 엔드포인트로 오브젝트 메타데이터를 보낼 수 있습니다. 하나 이상의 엔드포인트를 대상으로 사용하도록 소스 버킷을 구성할 수도 있습니다. 이를 통해 하나의 Amazon SNS(Simple Notification Service) 주제에 객체 생성에 대한 알림을 보내고 두 번째 Amazon SNS 주제에 대한 객체 삭제에 대한 알림을 보낼 수 있습니다.

### CloudMirror 복제용 엔드포인트

StorageGRID는 S3 버킷을 나타내는 복제 엔드포인트를 지원합니다. 이러한 버킷은 Amazon Web Services, 동일한 또는 원격 StorageGRID 구축 또는 다른 서비스에서 호스팅될 수 있습니다.

### 알림의 끝점입니다

StorageGRID Amazon SNS, Kafka 및 웹훅 엔드포인트를 지원합니다. SQS(Simple Queue Service) 및 AWS Lambda 엔드포인트는 지원되지 않습니다.

Kafka 엔드포인트의 경우 Mutual TLS가 지원되지 않습니다. 결과적으로, 만약 당신이 가지고 있다면 `ssl.client.auth`로 설정 `required` Kafka 브로커 구성에서 Kafka 엔드포인트 구성 문제가 발생할 수 있습니다.

### 검색 통합 서비스의 끝점입니다

StorageGRID는 Elasticsearch 클러스터를 나타내는 검색 통합 끝점을 지원합니다. 이러한 Elasticsearch 클러스터는 로컬 데이터 센터에 있거나 AWS 클라우드 또는 다른 곳에서 호스팅될 수 있습니다.

검색 통합 끝점은 특정 Elasticsearch 인덱스 및 유형을 참조합니다. StorageGRID에서 끝점을 만들기 전에 Elasticsearch에서 인덱스를 만들어야 합니다. 그렇지 않으면 끝점 생성이 실패합니다. 끝점을 만들기 전에 형식을 만들 필요가 없습니다. StorageGRID는 개체 메타데이터를 끝점으로 보낼 때 필요한 경우 형식을 만듭니다.

### 관련 정보

["StorageGRID 관리"](#)

플랫폼 서비스 끝점에 **URN**을 지정합니다

플랫폼 서비스 끝점을 만들 때는 고유한 URN(리소스 이름)을 지정해야 합니다. 플랫폼 서비스에 대한 구성 XML을 만들 때 URN을 사용하여 끝점을 참조합니다. 각 끝점의 URN은 고유해야 합니다.

StorageGRID에서는 플랫폼 서비스 엔드포인트를 만들 때 이를 검증합니다. 플랫폼 서비스 끝점을 만들기 전에 끝점에 지정된 리소스가 있고 해당 리소스에 도달할 수 있는지 확인합니다.

## urn 요소

플랫폼 서비스 끝점의 URN은 다음과 같이 또는 `urn:mysite`로 시작해야 합니다 `arn:aws`.

- 서비스가 AWS(Amazon Web Services)에서 호스팅되는 경우 를 사용합니다 `arn:aws`
- 서비스가 Google Cloud Platform(GCP)에서 호스팅되는 경우 를 사용합니다 `arn:aws`
- 서비스가 로컬로 호스팅되는 경우 를 사용합니다 `urn:mysite`

예를 들어, StorageGRID에서 호스팅되는 CloudMirror 끝점에 대해 URN을 지정하는 경우 URN은 로 시작할 수 `urn:sgws` 있습니다.

URN의 다음 요소는 다음과 같이 플랫폼 서비스의 유형을 지정합니다.

서비스	유형
CloudMirror 복제	s3
알림	sns, kafka , 또는 webhook
검색 통합	es

예를 들어, StorageGRID에서 호스팅되는 CloudMirror 끝점에 대해 URN을 계속 지정하려면 GET에 `urn:sgws:s3` 을 추가합니다 `s3`.

대부분의 엔드포인트의 경우 URN의 마지막 요소는 대상 URI에서 특정 대상 리소스를 식별합니다. 예를 들어, `sns-topic-name`.

웹훅 엔드포인트의 경우 대상 리소스는 대상 URI 자체입니다.

서비스	특정 리소스
CloudMirror 복제	bucket-name
알림	<code>sns-topic-name</code> 또는 <code>kafka-topic-name</code> 참고: 웹훅 엔드포인트의 경우 URN의 마지막 요소는 엔드포인트의 URN이 고유한 한 어떤 문자열이든 될 수 있습니다.
검색 통합	<code>domain-name/index-name/type-name</code> • 참고: * Elasticsearch 클러스터가 자동으로 인덱스를 만들도록 * 구성되지 * 인 경우 끝점을 만들기 전에 수동으로 인덱스를 만들어야 합니다.

## AWS 및 GCP에서 호스팅되는 서비스의 여관

AWS 및 GCP 엔터티의 경우 URN은 유효한 AWS ARN입니다. 예를 들면 다음과 같습니다.

- CloudMirror 복제:

```
arn:aws:s3:::bucket-name
```

- 알림:

```
arn:aws:sns:region:account-id:topic-name
```

- 검색 통합:

```
arn:aws:es:region:account-id:domain/domain-name/index-name/type-name
```



AWS 검색 통합 끝점의 경우에 domain-name 나와 있는 리터럴 문자열이 포함되어야 domain/ 합니다.

#### 현지 호스팅 서비스를 위한 여관

클라우드 서비스 대신 로컬로 호스팅된 서비스를 사용하는 경우 URN에 필요한 요소가 세 번째 및 최종 위치에 포함되어 있는 한 유효하고 고유한 URN을 만드는 방식으로 URN을 지정할 수 있습니다. 선택 사항으로 표시된 요소를 비워 두거나 자원을 식별하고 URN을 고유하게 만드는 데 도움이 되도록 원하는 방식으로 지정할 수 있습니다. 예를 들면 다음과 같습니다.

- CloudMirror 복제:

```
urn:mysite:s3:optional:optional:bucket-name
```

StorageGRID에 호스팅된 CloudMirror 엔드포인트의 경우 다음으로 시작하는 유효한 URN을 지정할 수 urn:sgws 있습니다.

```
urn:sgws:s3:optional:optional:bucket-name
```

- 알림:

Amazon Simple Notification Service 끝점 지정:

```
urn:mysite:sns:optional:optional:sns-topic-name
```

Kafka 끝점 지정:

```
urn:mysite:kafka:optional:optional:kafka-topic-name
```

웹훅 엔드포인트를 지정하세요.

```
urn:mysite:webhook:optional:optional:webhook-name
```

- 검색 통합:

```
urn:mysite:es:optional:optional:domain-name/index-name/type-name
```



로컬로 호스팅되는 검색 통합 끝점의 경우 domain-name 끝점의 URN이 고유하면 모든 문자열이 요소가 될 수 있습니다.

플랫폼 서비스 끝점을 만듭니다

플랫폼 서비스를 사용하려면 먼저 올바른 유형의 끝점을 하나 이상 만들어야 합니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- StorageGRID 관리자가 테넌트 계정에 대해 플랫폼 서비스를 활성화했습니다.
- 이 있는 사용자 그룹에 속해 ["끝점 또는 루트 액세스 권한을 관리합니다"](#)입니다.
- 플랫폼 서비스 엔드포인트에서 참조하는 리소스가 생성되었습니다.
  - CloudMirror 복제: S3 버킷
  - 이벤트 알림: Amazon Simple Notification Service(Amazon SNS) 주제, Kafka 주제 또는 웹훅 엔드포인트
  - 검색 알림: 대상 클러스터가 인덱스를 자동으로 생성하도록 구성되지 않은 경우 Elasticsearch 인덱스입니다.
- 대상 리소스에 대한 정보가 있습니다.
  - URI(Uniform Resource Identifier)의 호스트 및 포트



StorageGRID 시스템에서 호스팅되는 버킷을 CloudMirror 복제의 엔드포인트로 사용하려면 그리드 관리자에게 문의하여 입력해야 하는 값을 확인하십시오.

- 고유 리소스 이름(URN)

["플랫폼 서비스 끝점에 URN을 지정합니다"](#)

- 인증 자격 증명(필요한 경우):

## 검색 통합 끝점

검색 통합 끝점의 경우 다음 자격 증명을 사용할 수 있습니다.

- 액세스 키: 액세스 키 ID 및 비밀 액세스 키
- 기본 HTTP: 사용자 이름 및 암호

## CloudMirror 복제 엔드포인트

CloudMirror 복제 엔드포인트의 경우 다음 자격 증명을 사용할 수 있습니다.

- 액세스 키: 액세스 키 ID 및 비밀 액세스 키
- CAP(C2S Access Portal): 임시 자격 증명 URL, 서버 및 클라이언트 인증서, 클라이언트 키 및 선택적 클라이언트 개인 키 암호.

## Amazon SNS 엔드포인트입니다

Amazon SNS 엔드포인트의 경우 다음 자격 증명을 사용할 수 있습니다.

- 액세스 키: 액세스 키 ID 및 비밀 액세스 키

## Kafka 엔드포인트

Kafka 엔드포인트의 경우 다음 자격 증명을 사용할 수 있습니다.

- SASL /plain: 사용자 이름 및 암호
- SASL/SCRAM-SHA-256: 사용자 이름 및 암호
- SASL/SCRAM-SHA-512: 사용자 이름 및 암호

### ◦ 보안 인증서(인증서 검증이 필요한 경우)

- Elasticsearch 보안 기능이 활성화된 경우 연결 테스트에 대한 모니터 클러스터 권한, 쓰기 인덱스 권한 또는 문서 업데이트에 대한 인덱스 및 삭제 권한 모두가 있습니다.

## 단계

1. 스토리지(S3) \* > \* 플랫폼 서비스 엔드포인트 \* 를 선택합니다. 플랫폼 서비스 끝점 페이지가 나타납니다.
2. 끝점 만들기 \* 를 선택합니다.
3. 표시 이름을 입력하여 끝점과 그 용도를 간략하게 설명합니다.

엔드포인트가 지원하는 플랫폼 서비스 유형은 엔드포인트 페이지에 나열될 때 엔드포인트 이름 옆에 표시되므로 이름에 해당 정보를 포함할 필요가 없습니다.

4. URI \* 필드에서 끝점의 고유 URI(Resource Identifier)를 지정합니다.

다음 형식 중 하나를 사용합니다.

```
https://host:port
http://host:port
```

포트를 지정하지 않으면 다음과 같은 기본 포트가 사용됩니다.

- HTTPS URI의 경우 포트 443, HTTP URI의 경우 포트 80(대부분의 끝점)
- HTTPS 및 HTTP URI용 포트 9092(Kafka 엔드포인트만 해당)

예를 들어 StorageGRID에서 호스팅되는 버킷의 URI는 다음과 같습니다.

```
https://s3.example.com:10443
```

이 예에서는 s3.example.com StorageGRID HA(고가용성) 그룹의 VIP(가상 IP)에 대한 DNS 항목을 나타내고 10443 로드 밸런서 끝점에 정의된 포트를 나타냅니다.



가능하면 단일 장애 지점을 피하기 위해 로드 밸런싱 노드의 HA 그룹에 연결해야 합니다.

마찬가지로 AWS에서 호스팅되는 버킷의 URI는 다음과 같습니다.

```
https://s3-aws-region.amazonaws.com
```



엔드포인트가 CloudMirror 복제 서비스에 사용되는 경우 버킷 이름을 URI에 포함하지 마십시오. 버킷 이름을 \* URN \* 필드에 포함시킵니다.

5. 끝점에 대한 고유 URN(리소스 이름)을 입력합니다.



끝점이 생성된 후에는 끝점의 URN을 변경할 수 없습니다.

6. Continue \* 를 선택합니다.

7. 인증 유형 \* 의 값을 선택합니다.



웹후크 엔드포인트에 대한 인증을 원하는 경우 mTLS(Mutual Transport Layer Security)를 구성하세요.[9단계](#).

### 검색 통합 끝점

검색 통합 끝점에 대한 자격 증명을 입력하거나 업로드합니다.

제공하는 자격 증명에 대상 리소스에 대한 쓰기 권한이 있어야 합니다.

인증 유형입니다	설명	자격 증명
익명	대상에 대한 익명 액세스를 제공합니다. 보안이 비활성화된 끝점에서만 작동합니다.	인증이 없습니다.
액세스 키	AWS 스타일 자격 증명을 사용하여 대상과의 연결을 인증합니다.	<ul style="list-style-type: none"><li>액세스 키 ID입니다</li><li>비밀 액세스 키</li></ul>
기본 HTTP	사용자 이름과 암호를 사용하여 대상에 대한 연결을 인증합니다.	<ul style="list-style-type: none"><li>사용자 이름</li><li>암호</li></ul>

### CloudMirror 복제 엔드포인트

CloudMirror 복제 엔드포인트에 대한 자격 증명을 입력하거나 업로드합니다.

제공하는 자격 증명에 대상 리소스에 대한 쓰기 권한이 있어야 합니다.

인증 유형입니다	설명	자격 증명
익명	대상에 대한 익명 액세스를 제공합니다. 보안이 비활성화된 끝점에서만 작동합니다.	인증이 없습니다.
액세스 키	AWS 스타일 자격 증명을 사용하여 대상과의 연결을 인증합니다.	<ul style="list-style-type: none"><li>액세스 키 ID입니다</li><li>비밀 액세스 키</li></ul>
CAP(C2S 액세스 포털)	인증서 및 키를 사용하여 대상에 대한 연결을 인증합니다.	<ul style="list-style-type: none"><li>임시 자격 증명 URL입니다</li><li>서버 CA 인증서(PEM 파일 업로드)</li><li>클라이언트 인증서(PEM 파일 업로드)</li><li>클라이언트 개인 키(PEM 파일 업로드, OpenSSL 암호화 형식 또는 암호화되지 않은 개인 키 형식)</li><li>클라이언트 개인 키 암호 구문(선택 사항)</li></ul>

### Amazon SNS 엔드포인트입니다

Amazon SNS 끝점에 대한 자격 증명을 입력하거나 업로드합니다.

제공하는 자격 증명에 대상 리소스에 대한 쓰기 권한이 있어야 합니다.

인증 유형입니다	설명	자격 증명
익명	대상에 대한 익명 액세스를 제공합니다. 보안이 비활성화된 끝점에서만 작동합니다.	인증이 없습니다.
액세스 키	AWS 스타일 자격 증명을 사용하여 대상과의 연결을 인증합니다.	<ul style="list-style-type: none"> <li>액세스 키 ID입니다</li> <li>비밀 액세스 키</li> </ul>

### Kafka 엔드포인트

Kafka 엔드포인트에 대한 자격 증명을 입력하거나 업로드합니다.

제공하는 자격 증명에 대상 리소스에 대한 쓰기 권한이 있어야 합니다.

인증 유형입니다	설명	자격 증명
익명	대상에 대한 익명 액세스를 제공합니다. 보안이 비활성화된 끝점에서만 작동합니다.	인증이 없습니다.
SASL/일반	사용자 이름과 암호를 일반 텍스트로 사용하여 대상에 대한 연결을 인증합니다.	<ul style="list-style-type: none"> <li>사용자 이름</li> <li>암호</li> </ul>
SASL/SCRAM-SHA-256	Challenge-Response 프로토콜 및 SHA-256 해싱을 사용하여 사용자 이름과 암호를 사용하여 대상에 대한 연결을 인증합니다.	<ul style="list-style-type: none"> <li>사용자 이름</li> <li>암호</li> </ul>
SASL/SCRAM-SHA-512	Challenge-Response 프로토콜 및 SHA-512 해싱을 사용하여 사용자 이름과 암호를 사용하여 대상에 대한 연결을 인증합니다.	<ul style="list-style-type: none"> <li>사용자 이름</li> <li>암호</li> </ul>

사용자 이름과 암호가 Kafka 클러스터에서 가져온 위임 토큰에서 파생되는 경우 \* Use 위임 인증 사용 \* 을 선택합니다.

8. Continue \* 를 선택합니다.
9. 엔드포인트에 대한 TLS 연결이 어떻게 검증되는지 선택하려면 인증서 검증 라디오 버튼을 선택하세요.

## 대부분의 엔드포인트

검색 통합, CloudMirror 복제, Amazon SNS 또는 Kafka 엔드포인트에 대한 TLS 연결을 확인하세요.

인증서 확인 유형입니다	설명
TLS	엔드포인트 리소스에 대한 TLS 연결에 대한 서버 인증서를 검증합니다.
장애가 있는	인증서 검증이 비활성화되었습니다. 이 옵션은 안전하지 않습니다.
사용자 지정 CA 인증서를 사용합니다	사용자 지정 CA 인증서는 엔드포인트에 연결할 때 서버의 ID를 확인하는 데 사용됩니다.
운영 체제 CA 인증서를 사용합니다	운영 체제에 설치된 기본 그리드 CA 인증서를 사용하여 연결을 보호합니다.

## 웹훅 엔드포인트만

웹훅 엔드포인트에 대한 TLS 연결을 확인합니다.

인증서 확인 유형입니다	설명
TLS	엔드포인트 리소스에 대한 TLS 연결에 대한 서버 인증서를 검증합니다.
mTLS	엔드포인트 리소스에 대한 상호 TLS 연결에 대한 클라이언트 및 서버 인증서를 검증합니다.
장애가 있는	인증서 검증이 비활성화되었습니다. 이 옵션은 안전하지 않습니다.
사용자 지정 CA 인증서를 사용합니다	사용자 지정 CA 인증서는 엔드포인트에 연결할 때 서버의 ID를 확인하는 데 사용됩니다.

\*mTLS\*를 선택하면 이러한 옵션을 사용할 수 있습니다.

인증서 확인 유형입니다	설명
서버 인증서를 확인하지 마세요	서버 인증서 검증을 비활성화합니다. 즉, 서버의 ID가 검증되지 않습니다. 이 옵션은 안전하지 않습니다.
클라이언트 인증서	클라이언트 인증서는 엔드포인트에 연결할 때 클라이언트의 신원을 확인하는 데 사용됩니다.
클라이언트 개인 키	클라이언트 인증서의 개인 키입니다. 암호화된 경우 기존 형식인 PKCS #1을 사용해야 합니다(PKCS #8 형식은 지원되지 않음).
클라이언트 개인 키 암호문구	클라이언트 개인 키를 해독하기 위한 암호문구입니다. 개인 키가 암호화되지 않은 경우 이 부분을 비워 두세요.

## 10. 테스트를 선택하고 끝점 \* 을 작성합니다.

- 지정된 자격 증명을 사용하여 끝점에 도달할 수 있으면 성공 메시지가 나타납니다. 엔드포인트에 대한 연결은 각 사이트의 한 노드에서 검증됩니다.
- 끝점 유효성 검사에 실패하면 오류 메시지가 나타납니다. 오류를 수정하기 위해 끝점을 수정해야 하는 경우 \* 끝점 세부 정보로 돌아가기 \* 를 선택하고 정보를 업데이트합니다. 그런 다음 \* 테스트 를 선택하고 끝점 \* 을 만듭니다.



테넌트 계정에 플랫폼 서비스가 활성화되어 있지 않으면 엔드포인트 생성이 실패합니다.  
StorageGRID 관리자에게 문의하십시오.

끝점을 구성한 후 URN을 사용하여 플랫폼 서비스를 구성할 수 있습니다.

### 관련 정보

- ["플랫폼 서비스 끝점에 URN을 지정합니다"](#)
- ["CloudMirror 복제를 구성합니다"](#)
- ["이벤트 알림을 구성합니다"](#)
- ["검색 통합 서비스를 구성합니다"](#)

플랫폼 서비스 끝점에 대한 연결을 테스트합니다

플랫폼 서비스에 대한 연결이 변경된 경우 끝점에 대한 연결을 테스트하여 대상 리소스가 있는지 그리고 지정한 자격 증명을 사용하여 해당 리소스에 연결할 수 있는지 확인할 수 있습니다.

### 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["끝점 또는 루트 액세스 권한을 관리합니다"](#)입니다.

### 이 작업에 대해

StorageGRID는 자격 증명에 올바른 권한이 있는지 확인하지 않습니다.

### 단계

1. 스토리지(S3) \* > \* 플랫폼 서비스 엔드포인트 \* 를 선택합니다.

플랫폼 서비스 끝점 페이지가 나타나고 이미 구성된 플랫폼 서비스 끝점 목록이 표시됩니다.

2. 연결을 테스트할 끝점을 선택합니다.

끝점 세부 정보 페이지가 나타납니다.

3. Test connection \* 을 선택합니다.

- 지정된 자격 증명을 사용하여 끝점에 도달할 수 있으면 성공 메시지가 나타납니다. 엔드포인트에 대한 연결은 각 사이트의 한 노드에서 검증됩니다.
- 끝점 유효성 검사에 실패하면 오류 메시지가 나타납니다. 오류를 정정하기 위해 끝점을 수정해야 하는 경우 \* 구성 \* 을 선택하고 정보를 업데이트합니다. 그런 다음 \* 테스트 및 변경 내용 저장 \* 을 선택합니다.

플랫폼 서비스 끝점을 편집합니다

플랫폼 서비스 끝점의 구성을 편집하여 이름, URI 또는 기타 세부 정보를 변경할 수 있습니다. 예를 들어 만료된 자격 증명을 업데이트하거나 대체 작동을 위한 백업 Elasticsearch 인덱스를 가리키도록 URI를 변경해야 할 수 있습니다. 플랫폼 서비스 끝점의 URN은 변경할 수 없습니다.

시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 ["지원되는 웹 브라우저"](#)입니다.
- 이 있는 사용자 그룹에 속해 ["끝점 또는 루트 액세스 권한을 관리합니다"](#)입니다.

단계

1. 스토리지(S3) \* > \* 플랫폼 서비스 엔드포인트 \* 를 선택합니다.

플랫폼 서비스 끝점 페이지가 나타나고 이미 구성된 플랫폼 서비스 끝점 목록이 표시됩니다.

2. 편집할 끝점을 선택합니다.

끝점 세부 정보 페이지가 나타납니다.

3. Configuration \* 을 선택합니다.

4. 필요에 따라 끝점의 구성을 변경합니다.



끝점이 생성된 후에는 끝점의 URN을 변경할 수 없습니다.

- a. 끝점의 표시 이름을 변경하려면 편집 아이콘을 선택합니다.

- b. 필요에 따라 URI를 변경합니다.

- c. 필요에 따라 인증 유형을 변경합니다.

- 액세스 키 인증의 경우 \* S3 키 편집 \* 을 선택하고 새 액세스 키 ID 및 비밀 액세스 키를 붙여 넣어 필요에 따라 키를 변경합니다. 변경 사항을 취소하려면 \* S3 키 편집 되돌리기 \* 를 선택합니다.
- CAP(C2S Access Portal) 인증의 경우 임시 자격 증명 URL 또는 선택적 클라이언트 개인 키 암호를 변경하고 필요에 따라 새 인증서 및 키 파일을 업로드합니다.



클라이언트 개인 키는 OpenSSL 암호화 형식 또는 암호화되지 않은 개인 키 형식이어야 합니다.

- d. 필요에 따라 인증서 검증 방법을 변경하세요.

5. Test(테스트)를 선택하고 변경 내용을 저장합니다 \*.

- 지정된 자격 증명을 사용하여 끝점에 도달할 수 있으면 성공 메시지가 나타납니다. 엔드포인트에 대한 연결은 각 사이트의 한 노드에서 확인합니다.
- 끝점 유효성 검사에 실패하면 오류 메시지가 나타납니다. 끝점을 수정하여 오류를 수정한 다음 \* 테스트 및 변경 내용 저장 \* 을 선택합니다.

플랫폼 서비스 끝점을 삭제합니다

연결된 플랫폼 서비스를 더 이상 사용하지 않으려면 끝점을 삭제할 수 있습니다.

## 시작하기 전에

- 를 사용하여 테넌트 관리자에 로그인되어 "지원되는 웹 브라우저"입니다.
- 이 있는 사용자 그룹에 속해 "끝점 또는 루트 액세스 권한을 관리합니다"입니다.

## 단계

1. 스토리지(S3) \* > \* 플랫폼 서비스 엔드포인트 \* 를 선택합니다.

플랫폼 서비스 끝점 페이지가 나타나고 이미 구성된 플랫폼 서비스 끝점 목록이 표시됩니다.

2. 삭제할 각 끝점의 확인란을 선택합니다.



사용 중인 플랫폼 서비스 끝점을 삭제하면 해당 끝점을 사용하는 모든 버킷에 대해 연결된 플랫폼 서비스가 비활성화됩니다. 아직 완료되지 않은 요청은 삭제됩니다. 삭제된 URN을 더 이상 참조하지 않도록 버킷 구성을 변경할 때까지 새 요청은 계속 생성됩니다. StorageGRID는 이러한 요청을 복구할 수 없는 오류로 보고합니다.

3. 작업 \* > \* 끝점 삭제 \* 를 선택합니다.

확인 메시지가 나타납니다.

4. 끝점 삭제 \* 를 선택합니다.

## 플랫폼 서비스 끝점 오류 문제 해결

StorageGRID가 플랫폼 서비스 끝점과 통신하려고 할 때 오류가 발생하면 대시보드에 메시지가 표시됩니다. 플랫폼 서비스 끝점 페이지에서 마지막 오류 열은 오류가 발생한 시간을 나타냅니다. 끝점의 자격 증명과 연결된 권한이 올바르지 않으면 오류가 표시되지 않습니다.

## 오류가 발생했는지 확인합니다

지난 7일 이내에 플랫폼 서비스 끝점 오류가 발생한 경우 테넌트 관리자 대시보드에 경고 메시지가 표시됩니다. 플랫폼 서비스 끝점 페이지로 이동하여 오류에 대한 자세한 정보를 볼 수 있습니다.



One or more endpoints have experienced an error and might not be functioning properly. Go to the [Endpoints](#) page to view the error details. The last error occurred 2 hours ago.

대시보드에 나타나는 동일한 오류가 플랫폼 서비스 끝점 페이지 맨 위에도 나타납니다. 자세한 오류 메시지를 보려면:

## 단계

1. 끝점 목록에서 오류가 있는 끝점을 선택합니다.
2. 끝점 세부 정보 페이지에서 \* 연결 \* 을 선택합니다. 이 탭은 끝점에 대한 가장 최근 오류만 표시하고 오류가 발생한 시간을 표시합니다. 빨간색 X 아이콘이 포함된 오류는 지난 7일 내에 발생했습니다.

## 오류가 여전히 최신 상태인지 확인합니다

일부 오류는 해결된 후에도 \* 마지막 오류 \* 열에 계속 표시될 수 있습니다. 오류가 현재 오류인지 확인하거나 테이블에서 해결된 오류를 강제로 제거하려면 다음과 같이 하십시오.

## 단계

1. 끝점을 선택합니다.

끝점 세부 정보 페이지가 나타납니다.

2. 연결 \* > \* 연결 테스트 \* 를 선택합니다.

연결 테스트 \* 를 선택하면 StorageGRID가 플랫폼 서비스 끝점이 있는지, 그리고 현재 자격 증명으로 연결할 수 있는지 검증합니다. 엔드포인트에 대한 연결은 각 사이트의 한 노드에서 검증됩니다.

## 끝점 오류를 해결합니다

끝점 세부 정보 페이지의 \* 마지막 오류 \* 메시지를 사용하여 오류의 원인을 확인할 수 있습니다. 일부 오류에서는 문제를 해결하기 위해 끝점을 편집해야 할 수 있습니다. 예를 들어, 올바른 액세스 권한이 없거나 액세스 키가 만료되어 StorageGRID가 대상 S3 버킷을 액세스할 수 없는 경우 클라우드미러링 오류가 발생할 수 있습니다. 메시지는 "끝점 자격 증명 또는 대상 액세스를 업데이트해야 합니다."이고 세부 정보는 "AccessDenied" 또는 "InvalidAccessKeyId"입니다.

오류를 해결하기 위해 끝점을 편집해야 하는 경우 \* 테스트 및 변경 내용 저장 \* 을 선택하면 StorageGRID가 업데이트된 끝점을 검증하고 현재 자격 증명으로 연결할 수 있는지 확인합니다. 엔드포인트에 대한 연결은 각 사이트의 한 노드에서 검증됩니다.

## 단계

1. 끝점을 선택합니다.
2. 끝점 세부 정보 페이지에서 \* 구성 \* 을 선택합니다.
3. 필요에 따라 끝점 설정을 편집합니다.
4. 연결 \* > \* 연결 테스트 \* 를 선택합니다.

## 권한이 부족한 끝점 자격 증명

StorageGRID에서 플랫폼 서비스 끝점의 유효성을 검사할 때 끝점의 자격 증명을 사용하여 대상 리소스에 연결할 수 있는지 확인하고 기본적인 사용 권한 검사를 수행합니다. 그러나 StorageGRID는 특정 플랫폼 서비스 작업에 필요한 모든 사용 권한의 유효성을 검사하지 않습니다. 이러한 이유로 플랫폼 서비스를 사용하려고 할 때 오류(예: "403 금지됨")가 발생하면 끝점의 자격 증명과 연결된 사용 권한을 확인하십시오.

## 관련 정보

- ["StorageGRID 및 GT 관리, 플랫폼 서비스 문제 해결"](#)
- ["플랫폼 서비스 끝점을 만듭니다"](#)
- ["플랫폼 서비스 끝점에 대한 연결을 테스트합니다"](#)
- ["플랫폼 서비스 끝점을 편집합니다"](#)

## CloudMirror 복제를 구성합니다

버킷에 대해 CloudMirror 복제를 활성화하려면 유효한 버킷 복제 구성 XML을 생성하고 적용합니다.

## 시작하기 전에

- StorageGRID 관리자가 테넌트 계정에 대해 플랫폼 서비스를 활성화했습니다.

- 복제 소스로 사용할 버킷을 이미 생성했습니다.
- CloudMirror 복제의 대상으로 사용하려는 엔드포인트가 이미 있으며 URN이 있습니다.
- 이 있는 사용자 그룹에 속해 "모든 버킷 또는 루트 액세스 권한을 관리합니다" 있습니다. 이러한 권한은 테넌트 관리자를 사용하여 버킷을 구성할 때 그룹 또는 버킷 정책의 권한 설정을 재정의합니다.

#### 이 작업에 대해

CloudMirror 복제는 소스 버킷에서 엔드포인트에 지정된 대상 버킷으로 객체를 복제합니다.

버킷 복제에 대한 일반적인 정보와 이를 구성하는 방법은을 참조하십시오 ["Amazon S3\(Simple Storage Service\) 문서: 오브젝트 복제"](#). StorageGRID에서 GetBucketReplication, DeleteBucketReplication 및 PutBucketReplication을 구현하는 방법에 대한 자세한 내용은를 참조하십시오 ["버킷 작업"](#).



CloudMirror 복제는 교차 그리드 복제 기능과의 중요한 유사점과 차이점이 있습니다. 자세한 내용은을 참조하십시오 ["교차 그리드 복제와 CloudMirror 복제를 비교합니다"](#).

CloudMirror 복제를 구성할 때 다음과 같은 요구 사항 및 특성에 유의하십시오.

- 유효한 버킷 복제 구성 XML을 만들고 적용할 때 각 대상에 대해 S3 버킷 끝점의 URN을 사용해야 합니다.
- S3 오브젝트 잠금이 활성화된 소스 또는 대상 버킷에는 복제가 지원되지 않습니다.
- 객체가 포함된 버킷에서 CloudMirror 복제를 활성화하면 버킷에 추가된 새 객체가 복제되지만 버킷의 기존 객체는 복제되지 않습니다. 복제를 트리거하려면 기존 객체를 업데이트해야 합니다.
- 복제 구성 XML에서 스토리지 클래스를 지정하는 경우 StorageGRID는 대상 S3 끝점에 대해 작업을 수행할 때 해당 클래스를 사용합니다. 대상 끝점은 지정된 저장소 클래스도 지원해야 합니다. 대상 시스템 공급업체에서 제공하는 권장 사항을 따르십시오.

#### 단계

##### 1. 소스 버킷에 대한 복제 지원:

- 텍스트 편집기를 사용하여 S3 복제 API에 지정된 대로 복제를 활성화하는 데 필요한 복제 구성 XML을 생성합니다.
- XML을 구성할 때:
  - StorageGRID는 복제 구성의 V1만 지원합니다. 즉, StorageGRID는 규칙에 요소 사용을 지원하지 `Filter` 않으며 객체 버전을 삭제하기 위한 V1 규칙을 따릅니다. 자세한 내용은 복제 구성에 대한 Amazon 설명서를 참조하십시오.
  - S3 버킷 엔드포인트의 URN을 대상으로 사용합니다.
  - 선택적으로 `<StorageClass>` 요소를 추가하고 다음 중 하나를 지정합니다.
    - `STANDARD`: 기본 스토리지 클래스입니다. 개체를 업로드할 때 저장소 클래스를 지정하지 않으면 `STANDARD` 저장소 클래스가 사용됩니다.
    - `STANDARD_IA`: (표준 - 자주 액세스하지 않음) 자주 액세스하지 않지만 필요할 때 빠르게 액세스해야 하는 데이터에 이 스토리지 클래스를 사용합니다.
    - `REDUCED_REDUNDANCY`: 스토리지 클래스보다 낮은 중복성으로 저장할 수 있는 비중요하고 재현 가능한 데이터에 이 스토리지 클래스를 `STANDARD` 사용합니다.
  - 구성 XML에서 `Role` 지정하면 이 작업은 무시됩니다. 이 값은 StorageGRID에서 사용되지 않습니다.

```

<ReplicationConfiguration>
  <Role></Role>
  <Rule>
    <Status>Enabled</Status>
    <Prefix>2020</Prefix>
    <Destination>
      <Bucket>urn:sgws:s3:::2017-records</Bucket>
      <StorageClass>STANDARD</StorageClass>
    </Destination>
  </Rule>
</ReplicationConfiguration>

```

2. 대시보드에서 \* 버킷 보기 \* 를 선택하거나 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.
3. 소스 버킷의 이름을 선택합니다.

버킷 세부 정보 페이지가 나타납니다.

4. 플랫폼 서비스 \* > \* 복제 \* 를 선택합니다.
5. 복제 사용 \* 확인란을 선택합니다.
6. 복제 구성 XML을 텍스트 상자에 붙여 넣고 \* 변경 내용 저장 \* 을 선택합니다.



그리드 관리자 또는 그리드 관리 API를 사용하여 StorageGRID 관리자가 각 테넌트 계정에 대해 플랫폼 서비스를 활성화해야 합니다. 구성 XML을 저장할 때 오류가 발생하면 StorageGRID 관리자에게 문의하십시오.

7. 복제가 올바르게 구성되었는지 확인합니다.
  - a. 복제 구성에 지정된 대로 복제 요구 사항을 충족하는 객체를 소스 버킷에 추가합니다.  
앞에 나와 있는 예에서 접두사 "2020"과 일치하는 객체가 복제됩니다.
  - b. 객체가 대상 버킷에 복제되었는지 확인합니다.  
오브젝트 크기가 작은 경우 복제가 빠르게 수행됩니다.

#### 관련 정보

["플랫폼 서비스 끝점을 만듭니다"](#)

#### 이벤트 알림을 구성합니다

알림 구성 XML을 생성하고 테넌트 관리자를 사용하여 XML을 버킷에 적용하여 버킷에 대한 알림을 활성화합니다.

#### 시작하기 전에

- StorageGRID 관리자가 테넌트 계정에 대해 플랫폼 서비스를 활성화했습니다.

- 알림 소스로 사용할 버킷을 이미 생성했습니다.
- 이벤트 알림 대상으로 사용하려는 엔드포인트가 이미 있으며 URN이 있습니다.
- 이 있는 사용자 그룹에 속해 "[모든 버킷 또는 루트 액세스 권한을 관리합니다](#)"입니다. 이러한 권한은 테넌트 관리자를 사용하여 버킷을 구성할 때 그룹 또는 버킷 정책의 권한 설정을 재정의합니다.

#### 이 작업에 대해

이벤트 알림을 구성하려면 알림 구성 XML을 소스 버킷과 연결합니다. 알림 구성 XML은 버킷 알림을 구성하기 위한 S3 규칙을 따르며, 대상 Amazon SNS 주제, Kafka 주제 또는 웹훅 엔드포인트가 엔드포인트의 URN으로 지정됩니다.

이벤트 알림에 대한 일반적인 정보와 이벤트 알림을 구성하는 방법은 ["아마존 문서" 참조하십시오.](#) StorageGRID에서 S3 버킷 알림 구성 API를 구현하는 방법에 대한 자세한 내용은 ["S3 클라이언트 애플리케이션 구현 지침" 참조하십시오.](#)

버킷에 대한 이벤트 알림을 구성할 때는 다음 요구 사항 및 특성을 참고하십시오.

- 유효한 알림 구성 XML을 만들고 적용할 때 각 대상에 대한 이벤트 알림 끝점의 URN을 사용해야 합니다.
- S3 오브젝트 잠금이 설정된 버킷에서 이벤트 알림을 구성할 수 있지만, 오브젝트의 S3 오브젝트 잠금 메타데이터 (보존 기한 및 법적 증거 자료 보관 상태 포함)는 알림 메시지에 포함되지 않습니다.
- 이벤트 알림을 구성한 후에는 소스 버킷의 객체에 대해 지정된 이벤트가 발생할 때마다 알림이 생성되어 대상으로 사용되는 Amazon SNS 주제, Kafka 주제 또는 웹훅 엔드포인트로 전송됩니다.
- 객체가 포함된 버킷에 대해 이벤트 알림을 활성화하면 알림 구성이 저장된 후 수행되는 작업에 대해서만 알림이 전송됩니다.

#### 단계

##### 1. 소스 버킷에 대한 알림 활성화:

- 텍스트 편집기를 사용하여 S3 알림 API에 지정된 대로 이벤트 알림을 활성화하는 데 필요한 알림 구성 XML을 생성합니다.
- XML을 구성할 때는 이벤트 알림 끝점의 URN을 대상 항목으로 사용합니다.

```

<NotificationConfiguration>
  <TopicConfiguration>
    <Id>Image-created</Id>
    <Filter>
      <S3Key>
        <FilterRule>
          <Name>prefix</Name>
          <Value>images/</Value>
        </FilterRule>
      </S3Key>
    </Filter>
    <Topic>arn:aws:sns:us-east-1:050340950352:sgws-topic</Topic>
    <Event>s3:ObjectCreated:*</Event>
  </TopicConfiguration>
</NotificationConfiguration>

```

2. 테넌트 관리자에서 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

3. 소스 버킷의 이름을 선택합니다.

버킷 세부 정보 페이지가 나타납니다.

4. 플랫폼 서비스 \* > \* 이벤트 알림 \* 을 선택합니다.

5. 이벤트 알림 사용 \* 확인란을 선택합니다.

6. 알림 구성 XML을 텍스트 상자에 붙여 넣고 \* 변경 내용 저장 \* 을 선택합니다.



그리드 관리자 또는 그리드 관리 API를 사용하여 StorageGRID 관리자가 각 테넌트 계정에 대해 플랫폼 서비스를 활성화해야 합니다. 구성 XML을 저장할 때 오류가 발생하면 StorageGRID 관리자에게 문의하십시오.

7. 이벤트 알림이 올바르게 구성되었는지 확인합니다.

a. 구성 XML에 구성된 알림을 트리거하기 위한 요구 사항을 충족하는 소스 버킷의 객체에 대한 작업을 수행합니다.

이 예제에서는 접두사를 사용하여 개체를 만들 때마다 이벤트 알림이 `images/` 전송됩니다.

b. 알림이 대상 Amazon SNS 주제, Kafka 주제 또는 웹훅 엔드포인트에 전달되었는지 확인합니다.

예를 들어 대상 주제가 Amazon SNS에 호스팅되어 있는 경우 알림이 배달될 때 이메일을 보내도록 서비스를 구성할 수 있습니다.

```
{
  "Records": [
    {
      "eventVersion": "2.0",
      "eventSource": "sgws:s3",
      "eventTime": "2017-08-08T23:52:38Z",
      "eventName": "ObjectCreated:Put",
      "userIdentity": {
        "principalId": "11111111111111111111"
      },
      "requestParameters": {
        "sourceIPAddress": "193.51.100.20"
      },
      "responseElements": {
        "x-amz-request-id": "122047343"
      },
      "s3": {
        "s3SchemaVersion": "1.0",
        "configurationId": "Image-created",
        "bucket": {
          "name": "test1",
          "ownerIdentity": {
            "principalId": "11111111111111111111"
          },
          "arn": "arn:sgws:s3:::test1"
        },
        "object": {
          "key": "images/cat.jpg",
          "size": 0,
          "eTag": "d41d8cd98f00b204e9800998ecf8427e",
          "sequencer": "14D90402421461C7"
        }
      }
    }
  ]
}
```

+

대상 항목에서 알림이 수신되면 StorageGRID 알림에 대한 소스 버킷을 성공적으로 구성한 것입니다.

#### 관련 정보

- ["버킷에 대한 알림을 이해합니다"](#)
- ["S3 REST API 사용"](#)
- ["플랫폼 서비스 끝점을 만듭니다"](#)

검색 통합 서비스를 구성합니다

검색 통합 XML을 만들고 테넌트 관리자를 사용하여 XML을 버킷에 적용하여 버킷에 대한 검색 통합을 활성화합니다.

시작하기 전에

- StorageGRID 관리자가 테넌트 계정에 대해 플랫폼 서비스를 활성화했습니다.
- 인덱싱할 콘텐츠가 있는 S3 버킷을 이미 생성했습니다.
- 검색 통합 서비스의 대상으로 사용하려는 엔드포인트가 이미 있으며 URN이 있습니다.
- 이 있는 사용자 그룹에 속해 ["모든 버킷 또는 루트 액세스 권한을 관리합니다"](#) 있습니다. 이러한 권한은 테넌트 관리자를 사용하여 버킷을 구성할 때 그룹 또는 버킷 정책의 권한 설정을 재정의합니다.

이 작업에 대해

소스 버킷에 대한 검색 통합 서비스를 구성한 후 객체를 만들거나 객체의 메타데이터 또는 태그를 업데이트하면 대상 엔드포인트로 객체 메타데이터가 전송됩니다.

이미 객체가 포함된 버킷에 대해 검색 통합 서비스를 활성화하면 기존 객체에 대한 메타데이터 알림이 자동으로 전송되지 않습니다. 이러한 기존 개체를 업데이트하여 해당 메타데이터가 대상 검색 인덱스에 추가되도록 합니다.

단계

#### 1. 버킷에 대한 검색 통합 활성화:

- 텍스트 편집기를 사용하여 검색 통합을 활성화하는 데 필요한 메타데이터 알림 XML을 만듭니다.
- XML을 구성할 때는 검색 통합 끝점의 URN을 대상으로 사용합니다.

개체 이름의 접두어를 기준으로 개체를 필터링할 수 있습니다. 예를 들어, 접두사가 있는 개체의 메타데이터는 한 대상에, 접두사가 있는 개체의 메타데이터는 다른 대상에 `videos` 보낼 수 `images` 있습니다. 접두사가 겹치는 구성은 유효하지 않으며 제출될 때 거부됩니다. 예를 들어, 접두사가 있는 객체에 대한 규칙 하나와 접두사가 있는 객체에 대한 두 번째 규칙이 `test2` 포함된 구성은 `test` 허용되지 않습니다.

필요한 경우 [을 메타데이터 구성 XML의 예](#) 참조하십시오.

```
<MetadataNotificationConfiguration>
  <Rule>
    <Status>Enabled</Status>
    <Prefix></Prefix>
    <Destination>
      <Urn>/Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>
```

메타데이터 알림 구성 XML의 요소:

이름	설명	필수 요소입니다
MetadataNotificationConfiguration을 참조하십시오	<p>메타데이터 알림의 개체 및 대상을 지정하는 데 사용되는 규칙의 컨테이너 태그입니다.</p> <p>하나 이상의 규칙 요소가 포함되어 있습니다.</p>	예
규칙	<p>메타데이터를 지정된 인덱스에 추가해야 하는 개체를 식별하는 규칙의 컨테이너 태그입니다.</p> <p>접두사가 겹치는 규칙은 거부됩니다.</p> <p>MetadataNotificationConfiguration 요소에 포함되어 있습니다.</p>	예
ID입니다	<p>규칙의 고유 식별자입니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	아니요
상태	<p>상태는 '활성화' 또는 '비활성화'가 될 수 있습니다. 비활성화된 규칙에 대해 어떠한 작업도 수행되지 않습니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
접두어	<p>접두사와 일치하는 개체는 규칙의 영향을 받으며 해당 메타데이터는 지정된 대상으로 전송됩니다.</p> <p>모든 오브젝트를 일치시키려면 빈 접두사를 지정합니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
목적지	<p>규칙의 대상에 대한 컨테이너 태그입니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예

이름	설명	필수 요소입니다
urn	<p>객체 메타데이터가 전송되는 대상의 urn입니다. 다음 속성을 가진 StorageGRID 끝점의 URN이어야 합니다.</p> <ul style="list-style-type: none"> <li>• es 세 번째 요소여야 합니다.</li> <li>• URN은 메타데이터가 저장되는 인덱스 및 형식으로 끝나야 domain-name/myindex/mytype 합니다.</li> </ul> <p>엔드포인트는 테넌트 관리자 또는 테넌트 관리 API를 사용하여 구성됩니다. 다음과 같은 형식을 취합니다.</p> <ul style="list-style-type: none"> <li>• arn:aws:es:region:account-ID:domain/mydomain/myindex/mytype</li> <li>• urn:mysite:es:::mydomain/myindex/mytype</li> </ul> <p>구성 XML을 제출하기 전에 끝점을 구성해야 합니다. 그렇지 않으면 404 오류로 인해 구성이 실패합니다.</p> <p>urn은 Destination 요소에 포함됩니다.</p>	예

2. 테넌트 관리자에서 \* 스토리지(S3) \* > \* 버킷 \* 을 선택합니다.

3. 소스 버킷의 이름을 선택합니다.

버킷 세부 정보 페이지가 나타납니다.

4. 플랫폼 서비스 \* > \* 통합 검색 \* 을 선택합니다

5. 검색 통합 사용 \* 확인란을 선택합니다.

6. 메타데이터 알림 구성을 텍스트 상자에 붙여 넣고 \* 변경 내용 저장 \* 을 선택합니다.



그리드 관리자 또는 관리 API를 사용하여 StorageGRID 관리자가 각 테넌트 계정에 대해 플랫폼 서비스를 활성화해야 합니다. 구성 XML을 저장할 때 오류가 발생하면 StorageGRID 관리자에게 문의하십시오.

7. 검색 통합 서비스가 올바르게 구성되었는지 확인합니다.

a. 구성 XML에 지정된 대로 메타데이터 알림을 트리거하기 위한 요구 사항을 충족하는 객체를 소스 버킷에 추가합니다.

앞의 예제에서 버킷에 추가된 모든 오브젝트는 메타데이터 알림을 트리거합니다.

b. 개체의 메타데이터와 태그가 포함된 JSON 문서가 끝점에 지정된 검색 인덱스에 추가되었는지 확인합니다.

작업을 마친 후

필요에 따라 다음 방법 중 하나를 사용하여 버킷에 대한 검색 통합을 비활성화할 수 있습니다.

- 스토리지(S3) \* > \* 버킷 \* 을 선택하고 \* 검색 통합 활성화 \* 확인란의 선택을 취소합니다.
- S3 API를 직접 사용하는 경우 Delete Bucket 메타데이터 알림 요청을 사용합니다. S3 클라이언트 애플리케이션

구현 지침을 참조하십시오.

예: 모든 개체에 적용되는 메타데이터 알림 구성입니다

이 예제에서 모든 오브젝트의 오브젝트 메타데이터는 동일한 대상으로 전송됩니다.

```
<MetadataNotificationConfiguration>
  <Rule>
    <ID>Rule-1</ID>
    <Status>Enabled</Status>
    <Prefix></Prefix>
    <Destination>
      <Urn>urn:myes:es:::sgws-notifications/test1/all</Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>
```

예: 두 개의 규칙이 있는 메타데이터 알림 구성

이 예에서는 접두사와 일치하는 개체의 개체 메타데이터가 /images 한 대상으로 전송되고, 접두사와 일치하는 개체의 개체 메타데이터가 /videos 두 번째 대상으로 전송됩니다.

```
<MetadataNotificationConfiguration>
  <Rule>
    <ID>Images-rule</ID>
    <Status>Enabled</Status>
    <Prefix>/images</Prefix>
    <Destination>
      <Urn>arn:aws:es:us-east-1:3333333:domain/es-
domain/graphics/imagetype</Urn>
    </Destination>
  </Rule>
  <Rule>
    <ID>Videos-rule</ID>
    <Status>Enabled</Status>
    <Prefix>/videos</Prefix>
    <Destination>
      <Urn>arn:aws:es:us-west-1:2222222:domain/es-
domain/graphics/videotype</Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>
```

메타데이터 알림 형식입니다

버킷에 대한 검색 통합 서비스를 활성화하면 개체 메타데이터 또는 태그를 추가, 업데이트 또는 삭제할 때마다 JSON 문서가 생성되어 대상 끝점으로 전송됩니다.

이 예제는 키가 있는 개체가 라는 버킷에 `test` 생성될 때 생성될 수 있는 JSON의 예를 보여줍니다  
SGWS/Tagging.txt.test`버킷이 버전이 아니므로 `versionId 태그가 비어 있습니다.

```
{  
  "bucket": "test",  
  "key": "SGWS/Tagging.txt",  
  "versionId": "",  
  "accountId": "86928401983529626822",  
  "size": 38,  
  "md5": "3d6c7634a85436eee06d43415012855",  
  "region": "us-east-1",  
  "metadata": {  
    "age": "25"  
  },  
  "tags": {  
    "color": "yellow"  
  }  
}
```

**JSON** 문서에 포함된 필드입니다

문서 이름에는 버킷 이름, 오브젝트 이름 및 버전 ID(있는 경우)가 포함됩니다.

버킷 및 오브젝트 정보

`bucket`: 버킷의 이름입니다

`key`: 개체 키 이름입니다

`versionID`: 버전 버킷의 객체에 대한 객체 버전입니다

`region`: 버킷 영역(예: us-east-1)

시스템 메타데이터

`size`: HTTP 클라이언트에서 볼 수 있는 개체 크기(바이트)

`md5`: 개체 해시입니다

사용자 메타데이터

`metadata`: 객체에 대한 모든 사용자 메타데이터, 키-값 쌍

`key:value`

## 태그

tags: 객체에 대해 키-값 쌍으로 정의된 모든 객체 태그

key:value

### Elasticsearch에서 결과를 보는 방법

태그 및 사용자 메타데이터의 경우 StorageGRID는 날짜 및 숫자를 Elasticsearch에 문자열 또는 S3 이벤트 알림으로 전달합니다. 이러한 문자열을 날짜 또는 숫자로 해석하도록 Elasticsearch를 구성하려면 동적 필드 매핑 및 날짜 형식 매핑에 대한 Elasticsearch 지침을 따르십시오. 검색 통합 서비스를 구성하기 전에 인덱스에서 동적 필드 매핑을 사용하도록 설정합니다. 문서가 인덱싱된 후에는 인덱스에서 문서의 필드 형식을 편집할 수 없습니다.

## S3 REST API 사용

### S3 REST API 지원 버전 및 업데이트

StorageGRID는 REST(Representational State Transfer) 웹 서비스 세트로 구현되는 S3(Simple Storage Service) API를 지원합니다.

S3 REST API를 지원하므로 StorageGRID 시스템을 사용하는 사내 오브젝트 스토리지를 통해 S3 웹 서비스용으로 개발된 서비스 지향 애플리케이션을 연결할 수 있습니다. 클라이언트 애플리케이션의 현재 S3 REST API 호출 사용에 대한 최소 변경 사항이 필요합니다.

#### 지원되는 버전

StorageGRID 다음과 같은 특정 버전의 STS, S3 및 HTTP를 지원합니다.

항목	버전
STS AssumeRole API 사양	<a href="#">"Amazon Web Services(AWS) 설명서: Amazon Assumerole API 참조"</a> AssumeRole에 대한 자세한 내용은 다음을 참조하세요. <a href="#">"AssumeRole 설정"</a> .
S3 API 사양	<a href="#">"AWS 설명서: Amazon Simple Storage Service API 참조"</a>
HTTP	1.1 HTTP에 대한 자세한 내용은 HTTP/1.1(RFC 7230-35)을 참조하십시오. <a href="#">"IETF RFC 2616:HTTP/1.1(Hypertext Transfer Protocol)"</a> <ul style="list-style-type: none"><li>참고 *: StorageGRID는 HTTP/1.1 파이프라인을 지원하지 않습니다.</li></ul>

## S3 REST API 지원 업데이트

놓습니다	설명
12.0	<ul style="list-style-type: none"> <li>관리 API를 사용하여 다른 도메인이 StorageGRID의 데이터에 액세스할 수 있도록 하는 관리 인터페이스에 대한 CORS(교차 출처 리소스 공유) 지원이 추가되었습니다. <a href="#">"자세한 정보"</a>.</li> <li>STS AssumeRole 및 세션 정책에 대한 지원이 추가되었습니다. 보다 <a href="#">"세션 정책의 예"</a>. 테넌트 그룹에서 AssumeRole을 설정할 수 있습니다.</li> </ul>
11.9	<ul style="list-style-type: none"> <li>다음 요청 및 지원되는 헤더에 대해 미리 계산된 SHA-256 체크섬 값에 대한 지원이 추가되었습니다. 이 기능을 사용하여 업로드된 개체의 무결성을 확인할 수 있습니다. <ul style="list-style-type: none"> <li>CompleteMultipartUpload: x-amz-checksum-sha256</li> <li>CreateMultipartUpload: x-amz-checksum-algorithm</li> <li>GetObject(개체 가져오기): x-amz-checksum-mode</li> <li>제목 개체: x-amz-checksum-mode</li> <li>목록 파트</li> <li>Put개체: x-amz-checksum-sha256</li> <li>업로드 부품: x-amz-checksum-sha256</li> </ul> </li> <li>그리드 관리자가 테넌트 레벨 보존 및 규정 준수 설정을 제어할 수 있는 기능이 추가되었습니다. 이러한 설정은 S3 오브젝트 잠금 설정에 영향을 줍니다. <ul style="list-style-type: none"> <li>버킷 기본 보존 모드 및 객체 보존 모드: 그리드 관리자가 허용하는 경우 거버넌스 또는 규정 준수</li> <li>버킷 기본 보존 기간 및 객체 유지 기한: 그리드 관리자가 설정한 최대 보존 기간에 허용되는 값보다 작거나 같아야 합니다.</li> </ul> </li> <li>콘텐츠 인코딩 및 스트리밍 x-amz-content-sha256 값에 대한 지원이 aws-chunked 향상되었습니다. 제한 사항: <ul style="list-style-type: none"> <li>있는 경우 chunk-signature는 선택 사항이며 검증되지 않았습니다</li> <li>있는 경우 x-amz-trailer 콘텐츠는 무시됩니다</li> </ul> </li> </ul>
11.8	에 사용된 이름과 일치하도록 S3 작업의 이름을 업데이트했습니다. <a href="#">"AWS(Amazon Web Services) 문서: Amazon Simple Storage Service API Reference"를 참조하십시오"</a>
11.7	<ul style="list-style-type: none"> <li>"빠른 참조: 지원되는 S3 API 요청" 추가되었습니다.</li> <li>S3 Object Lock을 통한 거버넌스 모드 사용을 지원합니다.</li> <li>Get Object 및 Head Object 요청에 대한 StorageGRID별 응답 헤더에 대한 지원이 추가되었습니다 x-ntap-sg-cgr-replication-status. 이 헤더는 크로스 그리드 복제를 위한 객체의 복제 상태를 제공합니다.</li> <li>SelectObjectContent 요청이 이제 Parquet 객체를 지원합니다.</li> </ul>

놓습니다	설명
11.6	<ul style="list-style-type: none"> <li>• GET Object 및 HEAD Object 요청에서 REQUEST 매개 변수 사용에 대한 지원이 추가되었습니다 <code>partNumber</code>.</li> <li>• S3 오브젝트 잠금의 버킷 레벨에서 기본 보존 모드 및 기본 보존 기간에 대한 지원이 추가되었습니다.</li> <li>• 객체에 대해 허용되는 보존 기간의 범위를 설정하기 위해 정책 조건 키에 대한 지원이 추가되었습니다 <code>s3:object-lock-remaining-retention-days</code>.</li> <li>• 단일 PUT 객체 작업의 <code>Maximum_Recommended_size</code>를 5GiB(5,368,709,120바이트)로 변경했습니다. 5GiB보다 큰 객체가 있는 경우 대신 멀티파트 업로드를 사용합니다.</li> </ul>
11.5	<ul style="list-style-type: none"> <li>• 버킷 암호화 관리에 대한 지원이 추가되었습니다.</li> <li>• S3 오브젝트 잠금 및 더 이상 사용되지 않는 레거시 규정 준수 요청에 대한 지원 추가</li> <li>• 버전이 있는 버킷에서 여러 오브젝트 삭제 사용에 대한 지원이 추가되었습니다.</li> <li>• `Content-MD5` 이제 요청 헤더가 올바르게 지원됩니다.</li> </ul>
11.4	<ul style="list-style-type: none"> <li>• 버킷 태그 삭제, 버킷 태그 지정 가져오기 및 버킷 태그 지정을 위한 지원이 추가되었습니다. 비용 할당 태그는 지원되지 않습니다.</li> <li>• StorageGRID 11.4에서 만든 버킷의 경우 성능 모범 사례에 맞게 개체 키 이름을 제한하는 것이 더 이상 필요하지 않습니다.</li> <li>• 이벤트 유형에 버킷 알림에 대한 지원이 <code>s3:ObjectRestore:Post</code> 추가되었습니다.</li> <li>• 이제 여러 파트에 대한 AWS 크기 제한이 적용됩니다. 멀티파트 업로드의 각 파트는 5MiB에서 5GiB 사이여야 합니다. 마지막 부분은 5MiB보다 작을 수 있습니다.</li> <li>• TLS 1.3에 대한 지원이 추가되었습니다</li> </ul>
11.3	<ul style="list-style-type: none"> <li>• 고객이 제공한 키(SSE-C)를 사용하여 오브젝트 데이터의 서버측 암호화에 대한 지원이 추가되었습니다.</li> <li>• 삭제, 가져오기 및 넣기 버킷 라이프사이클 작업(만료 작업에만 해당) 및 응답 헤더에 대한 지원이 추가되었습니다 <code>x-amz-expiration</code>.</li> <li>• 수집 시 동기식 배치를 사용하는 ILM 규칙의 영향을 설명하기 위해 PUT 개체, Put Object-Copy 및 MultiPart Upload가 업데이트되었습니다.</li> <li>• TLS 1.1 암호가 더 이상 지원되지 않습니다.</li> </ul>
11.2	<p>클라우드 스토리지 풀과 함께 사용할 POST 오브젝트 복원에 대한 지원이 추가되었습니다. 그룹 및 버킷 정책에서 ARN, 정책 조건 키 및 정책 변수에 대해 AWS 구문 사용을 지원합니다. StorageGRID 구문을 사용하는 기존 그룹 및 버킷 정책은 계속 지원됩니다.</p> <ul style="list-style-type: none"> <li>• 참고: * 사용자 지정 StorageGRID 기능에 사용되는 것을 포함하여 다른 구성 JSON/XML에서 ARN/URN을 사용하는 것은 변경되지 않았습니다.</li> </ul>
11.1	CORS(Cross-Origin Resource Sharing), 그리드 노드에 대한 S3 클라이언트 연결을 위한 HTTP 및 버킷에 대한 규정 준수 설정에 대한 지원이 추가되었습니다.

놓습니다	설명
11.0	버킷에 대한 플랫폼 서비스(CloudMirror 복제, 알림 및 Elasticsearch 검색 통합) 구성 지원 추가 또한 버킷에 대한 객체 태그 위치 제약 조건 및 사용 가능한 정합성 보장에 대한 지원이 추가되었습니다.
10.4	버전 관리, 끝점 도메인 이름 페이지 업데이트, 정책, 정책 예제 및 PutOverwriteObject 권한에 대한 ILM 검색 변경 사항에 대한 지원이 추가되었습니다.
10.3	버전 관리 지원 추가.
10.2	그룹 및 버킷 액세스 정책 및 다중 파트 복제본(업로드 부분 복사)에 대한 지원이 추가되었습니다.
10.1	멀티파트 업로드, 가상 호스팅 스타일 요청 및 v4 인증에 대한 지원이 추가되었습니다.
10.0	StorageGRID 시스템에서 S3 REST API를 처음 지원합니다. 현재 지원되는 <i>Simple Storage Service API Reference</i> 버전은 2006-03-01입니다.

## 빠른 참조: 지원되는 **S3 API** 요청

이 페이지에서는 StorageGRID에서 Amazon S3(Simple Storage Service) API를 지원하는 방법을 요약합니다.

이 페이지에는 StorageGRID에서 지원하는 S3 작업만 포함됩니다.



각 작업에 대한 AWS 설명서를 보려면 제목에서 링크를 선택합니다.

### 공통 **URI** 쿼리 매개 변수 및 요청 헤더

별도로 지정하지 않으면 다음과 같은 공통 URI 쿼리 매개 변수가 지원됩니다.

- `versionId` (개체 작업에 필요한 경우)

별도로 명시되지 않은 경우 다음과 같은 일반적인 요청 헤더가 지원됩니다.

- `Authorization`
- `Connection`
- `Content-Length`
- `Content-MD5`
- `Content-Type`
- `Date`
- `Expect`
- `Host`

- x-amz-date

관련 정보

- "S3 REST API 구현 세부 정보"
- "Amazon Simple Storage Service API 참조: 일반 요청 헤더"

#### **"AbortMultipartUpload 를 클릭합니다"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두 지원하며공통 매개 변수 및 머리글, 이 추가 URI 쿼리 매개 변수는 다음과 같습니다.

- uploadId

요청 본문

없음

**StorageGRID** 설명서

"멀티파트 업로드 작업"

#### **"CompleteMultipartUpload를 클릭합니다"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두 지원하며공통 매개 변수 및 머리글, 이 추가 URI 쿼리 매개 변수는 다음과 같습니다.

- uploadId
- x-amz-checksum-sha256

본문 **XML** 태그를 요청합니다

StorageGRID는 다음과 같은 요청 본문 XML 태그를 지원합니다.

- ChecksumSHA256
- CompleteMultipartUpload
- ETag
- Part
- PartNumber

**StorageGRID** 설명서

"CompleteMultipartUpload를 클릭합니다"

#### **"CopyObject 를 선택합니다"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 공통 매개 변수 및 머리글항목과 다음 추가 헤더를 지원합니다.

- x-amz-copy-source
- x-amz-copy-source-if-match
- x-amz-copy-source-if-modified-since
- x-amz-copy-source-if-none-match
- x-amz-copy-source-if-unmodified-since
- x-amz-copy-source-server-side-encryption-customer-algorithm
- x-amz-copy-source-server-side-encryption-customer-key
- x-amz-copy-source-server-side-encryption-customer-key-MD5
- x-amz-metadata-directive
- x-amz-object-lock-legal-hold
- x-amz-object-lock-mode
- x-amz-object-lock-retain-until-date
- x-amz-server-side-encryption
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5
- x-amz-storage-class
- x-amz-tagging
- x-amz-tagging-directive
- x-amz-meta-<metadata-name>

요청 본문

없음

**StorageGRID** 설명서

["CopyObject" 를 선택합니다](#)

["CreateBucket"](#)

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 항목과 다음 추가 헤더를 지원합니다.

- x-amz-bucket-object-lock-enabled

요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

**StorageGRID** 설명서

["버킷 작업"](#)

## "CreateMultiUpload" 를 클릭합니다"

### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 공통 매개 변수 및 머리글 항목과 다음 추가 헤더를 지원합니다.

- Cache-Control
- Content-Disposition
- Content-Encoding
- Content-Language
- Expires
- x-amz-checksum-algorithm
- x-amz-server-side-encryption
- x-amz-storage-class
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5
- x-amz-tagging
- x-amz-object-lock-mode
- x-amz-object-lock-retain-until-date
- x-amz-object-lock-legal-hold
- x-amz-meta-<metadata-name>

요청 본문

없음

### StorageGRID 설명서

## "CreateMultiUpload" 를 클릭합니다"

### "삭제 버킷"

### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 공통 매개 변수 및 머리글 지원합니다.

### StorageGRID 설명서

## "버킷 작업"

### "DeleteBucketCors"

### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 공통 매개 변수 및 머리글 지원합니다.

요청 본문

없음

### StorageGRID 설명서

"버킷 작업"

#### "DeleteBucketEncryption"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

### StorageGRID 설명서

"버킷 작업"

#### "DeleteBucketLifecycle"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

### StorageGRID 설명서

- "버킷 작업"
- "S3 라이프사이클 구성을 생성합니다"

#### "DeleteBucketPolicy를 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

### StorageGRID 설명서

"버킷 작업"

#### "DeleteBucketReplication 을 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

### StorageGRID 설명서

"버킷 작업"

### "삭제 BucketTagging"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

"버킷 작업"

### "DeleteObject" 를 클릭합니다"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 지원과 다음 추가 요청 헤더를 지원합니다.

- x-amz-bypass-governance-retention

요청 본문

없음

**StorageGRID** 설명서

"객체에 대한 작업"

### "DeleteObjects" 를 클릭합니다"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 지원과 다음 추가 요청 헤더를 지원합니다.

- x-amz-bypass-governance-retention

요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

**StorageGRID** 설명서

"객체에 대한 작업"

### "DeleteObjectTagging" 을 선택합니다"

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

"객체에 대한 작업"

## "GetBucketAcl"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBucketCors" 를 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBucketEncryption"을 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBucketLifecycleConfiguration" 을 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

- ["버킷 작업"](#)
- ["S3 라이프사이클 구성의 생성"](#)

## "GetBucketLocation 을 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBuckNotificationConfiguration 을 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBucketPolicy를 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBucketReplication 을 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## "GetBucketTagging"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

"버킷 작업"

**"GetBucketVersioning" 을 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

"버킷 작업"

**"GetObject" 를 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두 지원하며 [공통 매개 변수 및 머리글](#) 다음과 같은 추가 URI 쿼리 매개 변수를 지원합니다.

- x-amz-checksum-mode
- partNumber
- response-cache-control
- response-content-disposition
- response-content-encoding
- response-content-language
- response-content-type
- response-expires

그리고 이러한 추가 요청 헤더는 다음과 같습니다.

- Range
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5
- If-Match
- If-Modified-Since
- If-None-Match
- If-Unmodified-Since

요청 본문

없음

**StorageGRID** 설명서

["GetObject"를 참조하십시오"](#)

**"GetObjectAcl"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["객체에 대한 작업"](#)

**"GetObjectLegalHold"를 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)

**"GetObjectLockConfiguration"을 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)

**"GetObjectRetention"을 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"

### "GetObjectTagging"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["객체에 대한 작업"](#)

### "머리버킷"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

### "HeadObject 를 선택합니다"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 항목과 다음 추가 헤더를 지원합니다.

- x-amz-checksum-mode
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5
- If-Match
- If-Modified-Since
- If-None-Match
- If-Unmodified-Since
- Range

요청 본문

없음

**StorageGRID** 설명서

["HeadObject 를 선택합니다"](#)

## "ListBucket"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

없음

**StorageGRID** 설명서

["서비스 및 GT, ListBucket에 대한 작업"](#)

## "ListMultipartUploads 를 참조하십시오"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 정보와 다음과 같은 추가 매개 변수를 지원합니다.

- encoding-type
- key-marker
- max-uploads
- prefix
- upload-id-marker

요청 본문

없음

**StorageGRID** 설명서

["ListMultipartUploads 를 참조하십시오"](#)

## "ListObjects 를 선택합니다"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 정보와 다음과 같은 추가 매개 변수를 지원합니다.

- delimiter
- encoding-type
- marker
- max-keys
- prefix

요청 본문

없음

**StorageGRID** 설명서

["버킷 작업"](#)

## ["ListObjectsV2 를 참조하십시오"](#)

### **Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 공통 매개 변수 및 머리글정보와 다음과 같은 추가 매개 변수를 지원합니다.

- continuation-token
- delimiter
- encoding-type
- fetch-owner
- max-keys
- prefix
- start-after

요청 본문

없음

### **StorageGRID** 설명서

#### ["버킷 작업"](#)

## ["ListObjectVersions 를 선택합니다"](#)

### **Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 공통 매개 변수 및 머리글정보와 다음과 같은 추가 매개 변수를 지원합니다.

- delimiter
- encoding-type
- key-marker
- max-keys
- prefix
- version-id-marker

요청 본문

없음

### **StorageGRID** 설명서

#### ["버킷 작업"](#)

## ["목록 파트"](#)

### **Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 공통 매개 변수 및 머리글정보와 다음과 같은 추가 매개 변수를 지원합니다.

- max-parts

- part-number-marker

- uploadId

요청 본문

없음

**StorageGRID** 설명서

["ListMultipartUploads 를 참조하십시오"](#)

**"BucketCors의"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

**StorageGRID** 설명서

["버킷 작업"](#)

**"PutBucketEncryption을 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

본문 **XML** 태그를 요청합니다

StorageGRID는 다음과 같은 요청 본문 XML 태그를 지원합니다.

- ApplyServerSideEncryptionByDefault
- Rule
- ServerSideEncryptionConfiguration
- SSEAlgorithm

**StorageGRID** 설명서

["버킷 작업"](#)

**"PutBucketLifecycleConfiguration을 참조하십시오"**

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

본문 **XML** 태그를 요청합니다

StorageGRID는 다음과 같은 요청 본문 XML 태그를 지원합니다.

- And
- Days

- `Expiration`
- `ExpiredObjectDeleteMarker`
- `Filter`
- `ID`
- `Key`
- `LifecycleConfiguration`
- `NewerNoncurrentVersions`
- `NoncurrentDays`
- `NoncurrentVersionExpiration`
- `Prefix`
- `Rule`
- `Status`
- `Tag`
- `Value`

#### **StorageGRID** 설명서

- "버킷 작업"
- "S3 라이프사이클 구성을 생성합니다"

#### ["PutBucketNotificationConfiguration을 참조하십시오"](#)

##### **Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

##### 본문 **XML** 태그를 요청합니다

StorageGRID는 다음과 같은 요청 본문 XML 태그를 지원합니다.

- `Event`
- `Filter`
- `FilterRule`
- `Id`
- `Name`
- `NotificationConfiguration`
- `Prefix`
- `S3Key`
- `Suffix`
- `Topic`
- `TopicConfiguration`

- Value

## StorageGRID 설명서

### "버킷 작업"

#### "BucketPolicy를 참조하십시오"

##### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

##### 요청 본문

지원되는 JSON 본문 필드에 대한 자세한 내용은 다음을 참조하세요. ["버킷 및 그룹 액세스 정책을 사용합니다"](#).

#### "PutBucketReplication을 참조하십시오"

##### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

##### 본문 XML 태그를 요청합니다

- Bucket
- Destination
- Prefix
- ReplicationConfiguration
- Rule
- Status
- StorageClass

## StorageGRID 설명서

### "버킷 작업"

#### "BucketTagging"

##### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

##### 요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

## StorageGRID 설명서

### "버킷 작업"

#### "PutBucketVersioning을 참조하십시오"

##### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

바디 매개 변수를 요청합니다

StorageGRID는 다음과 같은 요청 본문 매개 변수를 지원합니다.

- VersioningConfiguration
- Status

## StorageGRID 설명서

"버킷 작업"

### "PutObject"를 선택합니다"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 공통 매개 변수 및 머리글 항목과 다음 추가 헤더를 지원합니다.

- Cache-Control
- Content-Disposition
- Content-Encoding
- Content-Language
- Expires
- x-amz-checksum-sha256
- x-amz-server-side-encryption
- x-amz-storage-class
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5
- x-amz-tagging
- x-amz-object-lock-mode
- x-amz-object-lock-retain-until-date
- x-amz-object-lock-legal-hold
- x-amz-meta-`<metadata-name>`

요청 본문

- 개체의 이진 데이터입니다

## StorageGRID 설명서

"PutObject"를 선택합니다"

### "PutObjectLegalHold"를 선택합니다"

**Uri** 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 공통 매개 변수 및 머리글 지원합니다.

## 요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

### StorageGRID 설명서

["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)

["PutObjectLockConfiguration 을 참조하십시오"](#)

#### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

## 요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

### StorageGRID 설명서

["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)

["PutObjectRetention"](#)

#### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대한 모든 [공통 매개 변수 및 머리글](#) 항목과 다음 추가 헤더를 지원합니다.

- x-amz-bypass-governance-retention

## 요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

### StorageGRID 설명서

["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)

["PutObjectTagging"](#)

#### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

## 요청 본문

StorageGRID는 구현 시 Amazon S3 REST API에 의해 정의된 모든 요청 본문 매개 변수를 지원합니다.

### StorageGRID 설명서

["객체에 대한 작업"](#)

["RestoreObject 를 선택합니다"](#)

#### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

## 요청 본문

지원되는 본문 필드에 대한 자세한 내용은 ["RestoreObject 를 선택합니다" 참조하십시오](#).

## "SelectObjectContent 를 선택합니다"

### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두를 [공통 매개 변수 및 머리글](#) 지원합니다.

#### 요청 본문

지원되는 본문 필드에 대한 자세한 내용은 다음을 참조하십시오.

- ["S3 Select를 사용합니다"](#)
- ["SelectObjectContent 를 선택합니다"](#)

## "업로드 파트"

### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두 지원하며 [공통 매개 변수 및 머리글](#) 다음과 같은 추가 URI 쿼리 매개 변수를 지원합니다.

- partNumber
- uploadId

그리고 이러한 추가 요청 헤더는 다음과 같습니다.

- x-amz-checksum-sha256
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5

#### 요청 본문

- 파트의 이진 데이터

### StorageGRID 설명서

## "업로드 파트"

## "업로드파트 복사"

### Uri 쿼리 매개 변수 및 요청 헤더

StorageGRID는 이 요청에 대해 모두 지원하며 [공통 매개 변수 및 머리글](#) 다음과 같은 추가 URI 쿼리 매개 변수를 지원합니다.

- partNumber
- uploadId

그리고 이러한 추가 요청 헤더는 다음과 같습니다.

- x-amz-copy-source
- x-amz-copy-source-if-match

- x-amz-copy-source-if-modified-since
- x-amz-copy-source-if-none-match
- x-amz-copy-source-if-unmodified-since
- x-amz-copy-source-range
- x-amz-server-side-encryption-customer-algorithm
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-key-MD5
- x-amz-copy-source-server-side-encryption-customer-algorithm
- x-amz-copy-source-server-side-encryption-customer-key
- x-amz-copy-source-server-side-encryption-customer-key-MD5

요청 본문

없음

**StorageGRID** 설명서

["업로드파트 복사"](#)

## S3 REST API 구성을 테스트합니다

AWS CLI(Amazon Web Services 명령줄 인터페이스)를 사용하여 시스템에 대한 연결을 테스트하고 개체를 읽고 쓸 수 있는지 확인할 수 있습니다.

시작하기 전에

- 에서 AWS CLI를 다운로드하여 "[aws.amazon.com/cli](https://aws.amazon.com/cli)" 설치했습니다.
- 필요한 경우 가 "["로드 벨런서 끝점을 생성했습니다"](#) 있습니다. 그렇지 않으면 연결할 스토리지 노드의 IP 주소와 사용할 포트 번호를 알 수 있습니다. 을 "[클라이언트 연결용 IP 주소 및 포트](#)" 참조하십시오.
- 있습니다. "[S3 테넌트 계정을 생성했습니다](#)"
- 테넌트 및 에 로그인했습니다 "[선택키를 만들었습니다](#)".

이러한 단계에 대한 자세한 내용은 을 "[클라이언트 연결을 구성합니다](#)" 참조하십시오.

단계

1. StorageGRID 시스템에서 생성한 계정을 사용하도록 AWS CLI 설정을 구성합니다.
  - a. 구성 모드로 전환: `aws configure`
  - b. 생성한 계정의 액세스 키 ID를 입력합니다.
  - c. 생성한 계정의 암호 액세스 키를 입력합니다.
  - d. 사용할 기본 영역을 입력합니다. `us-east-1` 예를 들어,
  - e. 사용할 기본 출력 형식을 입력하거나 \* Enter \* 를 눌러 JSON을 선택합니다.
2. 버킷을 만듭니다.

이 예에서는 IP 주소 10.96.101.17 및 포트 10443을 사용하도록 로드 밸런서 끝점을 구성했다고 가정합니다.

```
aws s3api --endpoint-url https://10.96.101.17:10443  
--no-verify-ssl create-bucket --bucket testbucket
```

버킷이 성공적으로 생성되면 다음 예와 같이 버킷의 위치가 반환됩니다.

```
"Location": "/testbucket"
```

### 3. 개체를 업로드합니다.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl  
put-object --bucket testbucket --key s3.pdf --body C:\s3-  
test\upload\s3.pdf
```

객체가 성공적으로 업로드되면 객체 데이터의 해시인 Etag가 반환됩니다.

### 4. 버킷의 내용을 나열하여 객체가 업로드되었는지 확인합니다.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl  
list-objects --bucket testbucket
```

### 5. 개체를 삭제합니다.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl  
delete-object --bucket testbucket --key s3.pdf
```

### 6. 버킷을 삭제합니다.

```
aws s3api --endpoint-url https://10.96.101.17:10443 --no-verify-ssl  
delete-bucket --bucket testbucket
```

## StorageGRID에서 S3 REST API를 구현하는 방법

### 클라이언트 요청 충돌

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다.

"Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을

완료하는 시점을 기준으로 합니다.

## 정합성

정합성 보장은 서로 다른 스토리지 노드 및 사이트에서 객체의 가용성과 객체 일관성 간의 균형을 제공합니다. 애플리케이션에 필요한 만큼 일관성을 변경할 수 있습니다.

기본적으로 StorageGRID 새로 생성된 객체에 대해 읽기 후 쓰기 일관성을 보장합니다. 성공적으로 완료된 PUT 다음에 오는 GET은 새로 작성된 데이터를 읽을 수 있습니다. 기존 객체의 덮어쓰기, 메타데이터 업데이트 및 삭제는 결국 일관성을 유지합니다.

개체 작업을 다른 일관성으로 수행하려는 경우 다음을 수행할 수 있습니다.

- 일관성을 지정하세요 [각 버킷](#) .
- 에 대한 일관성을 [각 API 작동](#) 지정합니다.
- 다음 작업 중 하나를 수행하여 그리드 전체의 기본 일관성을 변경합니다.
  - 그리드 관리자에서 구성 > 시스템 > 저장소 설정 > \*기본 버킷 일관성\*으로 이동합니다.
  - ..



그리드 전체의 일관성에 대한 변경은 설정이 변경된 후에 생성된 버킷에만 적용됩니다. 변경에 대한 세부 정보를 확인하려면 에 있는 감사 로그를 참조하십시오 `/var/local/log(* consistencyLevel* 검색)`.

## 일관성 값

일관성은 StorageGRID 객체를 추적하는 데 사용하는 메타데이터가 노드 간에 분산되는 방식에 영향을 미칩니다. 일관성은 클라이언트 요청에 대한 객체의 가용성에 영향을 미칩니다.

버킷 또는 API 작업의 정합성을 다음 값 중 하나로 설정할 수 있습니다.

- \* ALL \*: 모든 노드가 즉시 객체 메타데이터를 수신하거나 요청이 실패합니다.
- 강력한 글로벌: 모든 사이트의 모든 클라이언트 요청에 대해 읽기-쓰기 일관성을 보장합니다. Quorum 의미 체계가 구성되면 다음 동작이 적용됩니다.
  - 그리드에 사이트가 3개 이상 있는 경우 클라이언트 요청에 대한 사이트 장애 허용을 허용합니다. 2개 사이트 그리드는 사이트 장애 허용 범위를 갖지 않습니다.
  - 다음 S3 작업은 한 사이트가 다운되면 성공할 수 없습니다.
    - DeleteBucketEncryption
    - PutBucketBranch
    - PutBucketEncryption을 참조하십시오
    - PutBucketVersioning을 참조하십시오
    - PutObjectLegalHold를 선택합니다
    - PutObjectLockConfiguration 을 참조하십시오
    - PutObjectRetention

필요한 경우 다음을 수행할 수 있습니다. ["강력한 글로벌 일관성을 위해 StorageGRID Quorum 의미 체계를 구성합니다."](#) .

- **Strong-site**: 개체 메타데이터가 사이트의 다른 노드에 즉시 배포됩니다. 사이트 내의 모든 클라이언트 요청에 대해 쓰기 후 읽기 일관성을 보장합니다.
- **Read-after-new-write**: 새 개체에 대해 읽기-쓰기 후 일관성을 제공하고 개체 업데이트에 대한 최종 일관성을 제공합니다. 고가용성 및 데이터 보호 보장 대부분의 경우에 권장됩니다.
- \* 사용 가능 \*: 새 객체 및 객체 업데이트 모두에 대한 최종 일관성을 제공합니다. S3 버킷의 경우 필요한 경우에만 사용하십시오(예: 거의 읽지 않는 로그 값이 포함된 버킷의 경우 또는 존재하지 않는 키의 헤드 또는 GET 작업의 경우). S3 FabricPool 버킷은 지원되지 않습니다.

"Read-after-new-write" 및 "Available" 정합성 보장을 사용합니다

HEAD 또는 GET 작업에서 "Read-after-new-write" 일관성을 사용하는 경우 StorageGRID는 다음과 같이 여러 단계로 조회를 수행합니다.

- 먼저 낮은 일관성을 사용하여 오브젝트를 찾습니다.
- 이 조회가 실패하면 다음 일관성 값에서 조회를 반복하여 강력한 글로벌 동작과 동일한 일관성을 유지합니다.

HEAD 또는 GET 작업에서 "Read-after-new-write" 일관성을 사용하지만 객체가 존재하지 않는 경우 객체 조회는 항상 강력한 글로벌 동작과 동일한 일관성을 유지합니다. 이 일관성을 유지하기 위해서는 각 사이트에서 개체 메타데이터의 여러 복사본을 사용할 수 있어야 하므로, 같은 사이트에 있는 두 개 이상의 스토리지 노드를 사용할 수 없는 경우 500개의 내부 서버 오류가 발생할 수 있습니다.

Amazon S3와 유사한 일관성 보장이 필요하지 않은 경우 일관성을 "사용 가능"으로 설정하여 헤드 및 가져오기 작업에 대한 이러한 오류를 방지할 수 있습니다. 두부 또는 GET 작업에서 "사용 가능한" 일관성을 사용하는 경우 StorageGRID는 최종 일관성을 제공합니다. 일관성 향상을 위해 실패한 작업을 다시 시도하지 않으므로 개체 메타데이터의 여러 복사본을 사용할 필요가 없습니다.

**API** 작업에 대한 일관성을 지정합니다

개별 API 작업에 대한 일관성을 설정하려면 작업에 대해 정합성 보장 값을 지원해야 하며 요청 헤더에서 일관성을 지정해야 합니다. 이 예제에서는 GetObject 작업의 일관성을 "Strong-site"로 설정합니다.

```
GET /bucket/object HTTP/1.1
Date: date
Authorization: authorization name
Host: host
Consistency-Control: strong-site
```



PutObject 및 GetObject 작업 모두에 대해 동일한 일관성을 사용해야 합니다.

버킷의 일관성을 지정합니다.

버킷의 일관성을 설정하려면 StorageGRID 요청을 사용할 수 ["버킷 일관성을 유지합니다"](#) 있습니다. 또는 Tenant Manager에서 수행할 수 ["버킷의 일관성을 변경합니다"](#) 있습니다.

버킷의 일관성을 설정할 때 다음 사항에 유의하십시오.

- 버킷의 일관성을 설정하면 버킷의 오브젝트나 버킷 구성에 수행되는 S3 작업에 사용되는 일관성이 결정됩니다. 버킷 자체의 작동에는 영향을 미치지 않습니다.
- 개별 API 작업의 일관성이 버킷의 일관성을 재정의합니다.
- 일반적으로 버킷은 "Read-after-new-write"라는 기본 일관성을 사용해야 합니다. 요청이 올바르게 작동하지 않는 경우 가능한 경우 응용 프로그램 클라이언트 동작을 변경합니다. 또는 각 API 요청의 일관성을 지정하도록 클라이언트를 구성합니다. 버킷 수준의 일관성을 마지막 수단으로 설정합니다.

일관성과 **ILM** 규칙이 상호 작용하여 데이터 보호에 영향을 미치는 방식

일관성을 선택하고 ILM 규칙을 따르는 것은 오브젝트가 보호되는 방식에 영향을 미칩니다. 이러한 설정은 상호 작용할 수 있습니다.

예를 들어, 오브젝트가 저장될 때 사용되는 일관성은 오브젝트 메타데이터의 초기 배치에 영향을 주고, ILM 규칙에 대해 선택된 수집 동작은 오브젝트 복사본의 초기 배치에 영향을 미칩니다. StorageGRID에서는 클라이언트 요청을 이행하기 위해 오브젝트의 메타데이터와 해당 데이터에 모두 액세스해야 하므로 일관성 및 수집 동작에 대해 일치하는 보호 수준을 선택하면 초기 데이터 보호 수준을 높이고 시스템 응답을 보다 예측 가능하게 할 수 있습니다.

ILM 규칙에 사용할 수 있는 항목은 다음과 ["수집 옵션"](#) 같습니다.

#### 이중 커밋

StorageGRID는 즉시 개체의 중간 복사본을 만들고 클라이언트에 성공을 반환합니다. ILM 규칙에 지정된 복사본은 가능한 경우 만들어집니다.

#### 엄격한

ILM 규칙에 지정된 모든 복제본이 클라이언트에 반환되기 전에 만들어져야 합니다.

#### 균형

StorageGRID는 수집 시 ILM 규칙에 지정된 모든 복제본을 만들려고 합니다. 이 작업이 불가능할 경우 중간 복제본이 만들어지고 성공이 클라이언트에 반환됩니다. ILM 규칙에 지정된 복사본은 가능한 경우 만들어집니다.

일관성과 **ILM** 규칙이 상호 작용하는 방법의 예

다음 ILM 규칙과 일관성을 갖춘 3개 사이트 그리드가 있다고 가정해 보겠습니다.

- **ILM 규칙:** 로컬 사이트에 하나씩, 원격 사이트마다 하나씩, 총 세 개의 개체 복사본을 만듭니다. 엄격한 수집 동작을 사용합니다.
- **일관성:** 강력한 글로벌(객체 메타데이터가 여러 사이트에 즉시 배포됨).

클라이언트가 그리드에 객체를 저장하면 StorageGRID 세 개의 객체 사본을 모두 만들고 메타데이터를 여러 사이트에 배포한 후 클라이언트에 성공을 반환합니다.

해당 객체는 성공적인 메시지 수집 시점에 손실로부터 완벽하게 보호됩니다. 예를 들어, 로컬 사이트가 수집 직후 손실되더라도 개체 데이터와 개체 메타데이터의 사본은 모두 원격 사이트에 남아 있습니다. 해당 객체는 다른 사이트에서도 완전히 검색 가능합니다.

대신 동일한 ILM 규칙과 강력한 사이트 일관성을 사용하면 개체 데이터가 원격 사이트에 복제되고 개체 메타데이터가 해당 사이트에 배포되기 전에 클라이언트가 성공 메시지를 받을 수 있습니다. 이 경우, 객체 메타데이터의 보호 수준이 객체 데이터의 보호 수준과 일치하지 않습니다. 수집 직후 로컬 사이트가 손실되면 개체 메타데이터도 손실됩니다. 해당 객체를 검색할 수 없습니다.

일관성과 ILM 규칙 간의 상호 관계는 복잡할 수 있습니다. 도움이 필요하면 NetApp에 문의하십시오.

## 오브젝트 버전 관리

각 오브젝트의 여러 버전을 유지하려면 버킷의 버전 관리 상태를 설정할 수 있습니다. 버킷에 대한 버전 관리를 사용하면 실수로 개체가 삭제되지 않도록 보호하고 이전 버전의 개체를 검색 및 복원할 수 있습니다.

StorageGRID 시스템은 대부분의 기능을 지원하는 버전 관리를 구현하지만 몇 가지 제한 사항이 있습니다. StorageGRID는 각 오브젝트의 버전을 최대 10,000개까지 지원합니다.

오브젝트 버전 관리를 StorageGRID ILM(정보 라이프사이클 관리) 또는 S3 버킷 라이프사이클 구성과 결합할 수 있습니다. 각 버킷에 대해 버전 관리를 명시적으로 설정해야 합니다. 버킷에 대해 버전 관리를 사용하도록 설정하면 버킷에 추가된 각 오브젝트에 버전 ID가 할당되며, StorageGRID 시스템에서 생성됩니다.

MFA(다중 요소 인증) 삭제 사용은 지원되지 않습니다.



버전 관리는 StorageGRID 버전 10.3 이상으로 생성된 버킷에서만 사용할 수 있습니다.

### ILM 및 버전 관리

ILM 정책은 개체의 각 버전에 적용됩니다. ILM 스캔 프로세스는 모든 개체를 지속적으로 스캔하고 현재 ILM 정책에 대해 다시 평가합니다. ILM 정책에 대한 모든 변경 사항은 이전에 수집된 모든 개체에 적용됩니다. 여기에는 버전 관리가 활성화된 경우 이전에 수집된 버전이 포함됩니다. ILM 스캐닝은 이전에 수집된 개체에 새로운 ILM 변경 사항을 적용합니다.

버전 관리 지원 버킷의 S3 객체의 경우 버전 관리 지원을 통해 "현재 시간"을 참조 시간으로 사용하는 ILM 규칙을 만들 수 있습니다(의 "이전 객체 버전에만 이 규칙을 적용하시겠습니까?"라는 질문에 대해 \* 예 \* 를 선택 "[ILM 규칙 만들기](#) [마법사의 1단계](#)"). 개체가 업데이트되면 이전 버전은 업데이트되지 않습니다. "비현재 시간" 필터를 사용하면 이전 버전의 객체가 스토리지에 미치는 영향을 줄이는 정책을 만들 수 있습니다.



다중파트 업로드 작업을 사용하여 새 버전의 개체를 업로드할 때 개체의 원래 버전에 대한 비현재 시간은 다중파트 업로드가 완료될 때가 아닌 새 버전에 대해 다중파트 업로드가 생성된 시점을 반영합니다. 제한된 경우 원래 버전의 비현재 시간이 현재 버전의 시간보다 몇 시간 또는 며칠 빨라질 수 있습니다.

### 관련 정보

- ["S3 버전 오브젝트 삭제 방법"](#)
- ["S3 버전 오브젝트 ILM 규칙 및 정책\(예 4\)"..](#)

## S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다

StorageGRID 시스템에서 글로벌 S3 오브젝트 잠금 설정이 활성화된 경우 S3 오브젝트 잠금이 활성화된 버킷을 생성할 수 있습니다. 각 오브젝트 버전에 대해 각 버킷의 기본 보존 또는 보존 설정을 지정할 수 있습니다.

### 버킷에 대해 S3 오브젝트 잠금을 활성화하는 방법

StorageGRID 시스템에 대해 글로벌 S3 오브젝트 잠금 설정이 활성화된 경우 각 버킷을 생성할 때 선택적으로 S3 오브젝트 잠금을 활성화할 수 있습니다.

S3 오브젝트 잠금은 버킷을 생성할 때만 활성화할 수 있는 영구 설정입니다. 버킷을 생성한 후에는 S3 오브젝트 잠금을

추가하거나 비활성화할 수 없습니다.

버킷에 대해 S3 오브젝트 잠금을 설정하려면 다음 방법 중 하나를 사용하십시오.

- 테넌트 관리자를 사용하여 버킷을 생성합니다. 을 ["S3 버킷을 생성합니다"](#) 참조하십시오.
- 요청 헤더가 있는 CreateBucket 요청을 사용하여 버킷을 x-amz-bucket-object-lock-enabled 만듭니다. 을 ["버킷 작업"](#) 참조하십시오.

S3 오브젝트 잠금에서는 버킷 버전 관리가 필요하며, 이 버전은 버킷을 생성할 때 자동으로 활성화됩니다. 버킷의 버전 관리는 일시 중단할 수 없습니다. 을 ["오브젝트 버전 관리"](#) 참조하십시오.

버킷의 기본 보존 설정입니다

버킷에 대해 S3 오브젝트 잠금이 활성화된 경우 버킷에 대한 기본 보존을 선택적으로 설정하고 기본 보존 모드 및 기본 보존 기간을 지정할 수 있습니다.

### 기본 보존 모드

- 규정 준수 모드:
  - 보존 기한에 도달할 때까지 개체를 삭제할 수 없습니다.
  - 오브젝트의 보존 기한을 늘릴 수 있지만 줄일 수는 없습니다.
  - 개체의 보존 기한은 해당 날짜에 도달할 때까지 제거할 수 없습니다.
- 거버넌스 모드:
  - 있는 사용자는 s3:BypassGovernanceRetention 권한이 요청 헤더를 사용하여 보존 설정을 무시할 수 x-amz-bypass-governance-retention: true 있습니다.
  - 이러한 사용자는 보존 기한이 되기 전에 개체 버전을 삭제할 수 있습니다.
  - 이러한 사용자는 개체의 보존 기간(Retain-until-date)을 증가, 감소 또는 제거할 수 있습니다.

### 기본 보존 기간

각 버킷에는 년 또는 일 단위로 지정된 기본 보존 기간이 있을 수 있습니다.

버킷의 기본 보존 설정 방법

버킷의 기본 보존을 설정하려면 다음 방법 중 하나를 사용합니다.

- 테넌트 관리자에서 버킷 설정을 관리합니다. ["S3 버킷을 생성합니다"](#) 및 을 ["S3 오브젝트 잠금 기본 보존 업데이트"](#) 참조하십시오.
- 버킷에 대한 PutObjectLockConfiguration 요청을 실행하여 기본 모드와 기본 일 또는 년 수를 지정합니다.

### PutObjectLockConfiguration 을 참조하십시오

PutObjectLockConfiguration 요청을 사용하면 S3 오브젝트 잠금이 설정된 버킷의 기본 보존 모드 및 기본 보존 기간을 설정하고 수정할 수 있습니다. 이전에 구성한 기본 보존 설정을 제거할 수도 있습니다.

새 오브젝트 버전이 버킷에 수집되면 및 x-amz-object-lock-retain-until-date 이 지정되지 않은 경우 기본 보존 모드가 x-amz-object-lock-mode 적용됩니다. 이 지정되지 않은 경우 기본 보존 기간은 유지 종료 날짜를 계산하는 데 x-amz-object-lock-retain-until-date 사용됩니다.

오브젝트 버전을 수집한 후 기본 보존 기간을 수정하면 오브젝트 버전의 보존 기한은 그대로 유지되고 새 기본 보존 기간을 사용하여 다시 계산되지 않습니다.

이 작업을 완료하려면 권한이 있거나 계정 루트여야 `s3:PutBucketObjectLockConfiguration` 합니다.

`Content-MD5` PUT 요청에 요청 헤더를 지정해야 합니다.

## 요청 예

이 예에서는 버킷에 대해 S3 Object Lock을 설정하고 기본 보존 모드를 규정 준수로 설정하고 기본 보존 기간을 6년으로 설정합니다.

```
PUT /bucket?object-lock HTTP/1.1
Accept-Encoding: identity
Content-Length: 308
Host: host
Content-MD5: request header
User-Agent: s3sign/1.0.0 requests/2.24.0 python/3.8.2
X-Amz-Date: date
X-Amz-Content-SHA256: authorization-string
Authorization: authorization-string

<ObjectLockConfiguration>
  <ObjectLockEnabled>Enabled</ObjectLockEnabled>
  <Rule>
    <DefaultRetention>
      <Mode>COMPLIANCE</Mode>
      <Years>6</Years>
    </DefaultRetention>
  </Rule>
</ObjectLockConfiguration>
```

## 버킷의 기본 보존 결정 방법

버킷에 대해 S3 오브젝트 잠금이 설정되었는지 확인하고 기본 보존 모드 및 보존 기간을 확인하려면 다음 방법 중 하나를 사용하십시오.

- 테넌트 관리자에서 버킷을 확인합니다. 을 ["S3 버킷을 봅니다"](#) 참조하십시오.
- `GetObjectLockConfiguration` 요청을 실행합니다.

## `GetObjectLockConfiguration` 을 참조하십시오

`GetObjectLockConfiguration` 요청을 사용하면 버킷에 대해 S3 오브젝트 잠금이 설정되어 있는지 확인하고, 사용하도록 설정되어 있는 경우 버킷에 대해 구성된 기본 보존 모드 및 보존 기간이 있는지 확인할 수 있습니다.

새 오브젝트 버전이 버킷에 수집되면 이 지정되지 않은 경우 기본 모드가 `x-amz-object-lock-mode`

적용됩니다. 이 지정되지 않은 경우 기본 보존 기간은 유지 종료 날짜를 계산하는 데 `x-amz-object-lock-retain-until-date` 사용됩니다.

이 작업을 완료하려면 권한이 있거나 계정 루트여야 `s3:GetBucketObjectLockConfiguration` 합니다.

## 요청 예

```
GET /bucket?object-lock HTTP/1.1
Host: host
Accept-Encoding: identity
User-Agent: aws-cli/1.18.106 Python/3.8.2 Linux/4.4.0-18362-Microsoft
botocore/1.17.29
x-amz-date: date
x-amz-content-sha256: authorization-string
Authorization: authorization-string
```

## 응답 예

```
HTTP/1.1 200 OK
x-amz-id-2:
iVmcB70XXJRkRH1Fivq1151/T24gRfpwpuZrEG11Bb9ImOMAAe98oxSpX1knabA0LTvBYJpSIX
k=
x-amz-request-id: B34E94CACB2CEF6D
Date: Fri, 04 Sep 2020 22:47:09 GMT
Transfer-Encoding: chunked
Server: AmazonS3

<?xml version="1.0" encoding="UTF-8"?>
<ObjectLockConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <ObjectLockEnabled>Enabled</ObjectLockEnabled>
  <Rule>
    <DefaultRetention>
      <Mode>COMPLIANCE</Mode>
      <Years>6</Years>
    </DefaultRetention>
  </Rule>
</ObjectLockConfiguration>
```

## 개체의 보존 설정을 지정하는 방법

S3 오브젝트 잠금이 활성화된 버킷에는 S3 오브젝트 잠금 보존 설정이 있는 오브젝트와 없는 오브젝트의 조합이 포함될 수 있습니다.

오브젝트 레벨의 보존 설정은 S3 REST API를 사용하여 지정됩니다. 객체에 대한 보존 설정은 버킷의 기본 보존 설정보다 우선합니다.

각 개체에 대해 다음 설정을 지정할 수 있습니다.

- \* 보존 모드 \*: 규정 준수 또는 거버넌스 중 하나입니다.
- \* Retain-until-date \*: StorageGRID에서 개체 버전을 유지해야 하는 기간을 지정하는 날짜입니다.
  - 준수 모드에서 보존 기한이 미래인 경우 오브젝트를 검색할 수 있지만 수정하거나 삭제할 수 없습니다. 보관 기한을 늘릴 수 있지만 이 날짜는 감소 또는 제거할 수 없습니다.
  - 거버넌스 모드에서 특별 권한이 있는 사용자는 보존 기한 설정을 무시할 수 있습니다. 보존 기간이 경과하기 전에 객체 버전을 삭제할 수 있습니다. 또한 보존 기간을 늘리거나 줄이거나 제거할 수도 있습니다.
- \* 법적 증거 자료 보관 \*: 개체 버전에 법적 증거 자료 보관 기능을 적용하면 해당 개체가 즉시 잠깁니다. 예를 들어 조사 또는 법적 분쟁과 관련된 객체에 법적 보류를 지정해야 할 수 있습니다. 법적 보류는 만료 날짜가 없지만 명시적으로 제거될 때까지 유지됩니다.

개체에 대한 법적 보류 설정은 보존 모드 및 보존 기한과 무관합니다. 개체 버전이 법적 증거 자료 보관 중인 경우 해당 버전을 삭제할 수 없습니다.

오브젝트 버전을 버킷에 추가할 때 S3 오브젝트 잠금 설정을 지정하려면 "[PutObject 를 선택합니다](#)", 를 실행하거나 "[CopyObject 를 선택합니다](#)" "[CreateMultipartUpload 를 클릭합니다](#)" 요청을 실행하십시오.

다음을 사용할 수 있습니다.

- `x-amz-object-lock-mode` 규정 준수 또는 거버넌스일 수 있습니다(대소문자 구분).



을 지정한 `x-amz-object-lock-mode` 경우에는 도 지정해야 `x-amz-object-lock-retain-until-date`합니다.

- x-amz-object-lock-retain-until-date
  - 유지 기한 값은 형식이어야 '2020-08-10T21:46:00Z'입니다. 소수 자릿수는 허용되지만 소수점 이하 자릿수는 3자리만 유지됩니다(밀리초 단위). 다른 ISO 8601 형식은 허용되지 않습니다.
  - 보존 종료 날짜는 미래여야 합니다.
- x-amz-object-lock-legal-hold

법적 증거 자료 보관(대소문자 구분)이 켜져 있는 경우, 해당 물체는 법적 증거 자료 보관 하에 배치됩니다. 법적 증거 자료 보관 기능이 꺼져 있는 경우 법적 증거 자료 보관 작업이 없습니다. 다른 값을 사용하면 400개의 잘못된 요청(InvalidArgument) 오류가 발생합니다.

이러한 요청 헤더를 사용하는 경우 다음과 같은 제한 사항에 유의하십시오.

- Content-MD5 요청 헤더가 PutObject 요청에 있는 경우 요청 `x-amz-object-lock-\*` 헤더가 필요합니다. Content-MD5 CopyObject 또는 CreateMultipartUpload에는 필요하지 않습니다.
- 버킷에 S3 오브젝트 잠금이 설정되어 있지 않고 요청 헤더가 있는 경우 x-amz-object-lock-\* 400 Bad Request(InvalidRequest) 오류가 반환됩니다.
- PutObject 요청에서는 AWS 동작을 일치시키기 위해 의 사용을 x-amz-storage-class: REDUCED\_REDUNDANCY 지원합니다. 하지만 오브젝트가 S3 오브젝트 잠금이 설정된 버킷으로 수집되면 StorageGRID는 항상 이중 커밋 수집을 수행합니다.
- 이후의 Get 또는 HeadObject 버전 응답에는 헤더 x-amz-object-lock-mode, x-amz-object-lock-retain-until-date 및 x-amz-object-lock-legal-hold, 구성된 경우 요청 보낸 사람에게 올바른

권한이 있는지 여부가 s3:Get\* 포함됩니다.

정책 조건 키를 사용하여 개체에 대해 허용되는 최소 및 최대 보존 기간을 제한할 수 s3:object-lock-remaining-retention-days 있습니다.

개체의 보존 설정을 업데이트하는 방법

기존 개체 버전에 대한 법적 증거 자료 보관 또는 보존 설정을 업데이트해야 하는 경우 다음 개체 하위 리소스 작업을 수행할 수 있습니다.

- PutObjectLegalHold

새 법적 증거 자료 보관 값이 커져 있으면 해당 개체는 법적 증거 자료 보관 아래에 배치됩니다. 법적 증거 자료 보관 가치가 없는 경우 법적 구속이 해제됩니다.

- PutObjectRetention

- 모드 값은 규정 준수 또는 거버넌스(대/소문자 구분)일 수 있습니다.
- 유지 기한 값은 형식이어야 '2020-08-10T21:46:00Z'입니다. 소수 자릿수는 허용되지만 소수점 이하 자릿수는 3자리만 유지됩니다(밀리초 단위). 다른 ISO 8601 형식은 허용되지 않습니다.
- 개체 버전에 기존 보존 기한이 있는 경우 개체 버전을 늘릴 수만 있습니다. 새 값은 미래여야 합니다.

거버넌스 모드 사용 방법

권한이 있는 사용자는 s3:BypassGovernanceRetention 거버넌스 모드를 사용하는 개체의 활성 보존 설정을 무시할 수 있습니다. 삭제 또는 PutObjectRetention 작업은 요청 헤더를 포함해야 x-amz-bypass-governance-retention:true 합니다. 이러한 사용자는 다음과 같은 추가 작업을 수행할 수 있습니다.

- DeleteObject 또는 DeleteObjects 작업을 수행하여 보존 기간이 경과하기 전에 개체 버전을 삭제합니다.

법적 증거 자료 보관 중인 객체는 삭제할 수 없습니다. 법적 증거 자료 보관 기능을 해제해야 합니다.

- 개체의 보존 기간이 경과하기 전에 개체 버전의 모드를 거버넌스에서 규정 준수로 변경하는 PutObjectRetention 작업을 수행합니다.

규정 준수 모드를 거버넌스로 변경하는 것은 허용되지 않습니다.

- PutObjectRetention 작업을 수행하여 개체 버전의 보존 기간을 증가, 감소 또는 제거합니다.

관련 정보

- "[S3 오브젝트 잠금으로 오브젝트 관리](#)"
- "[S3 오브젝트 잠금을 사용하여 오브젝트를 보존합니다](#)"
- "[Amazon Simple Storage Service 사용자 가이드: 오브젝트 잠금](#)"

**S3** 라이프사이클 구성을 생성합니다

S3 라이프사이클 구성을 생성하여 StorageGRID 시스템에서 특정 오브젝트 삭제 시기를 제어할 수 있습니다.

이 섹션의 간단한 예는 S3 라이프사이클 구성에서 특정 S3 버킷에서 특정 객체가 삭제(만료)되는 시기를 제어하는

방법을 보여줍니다. 이 섹션의 예제는 설명을 위한 것입니다. S3 라이프사이클 구성 생성에 대한 자세한 내용은 [를 참조하십시오 "Amazon Simple Storage Service 사용자 가이드: 객체 수명 주기 관리"](#). StorageGRID는 만료 작업만 지원하며 전환 작업은 지원하지 않습니다.

문서 수정 상태 설정은 무엇입니까

라이프사이클 구성은 특정 S3 버킷의 오브젝트에 적용되는 규칙 세트입니다. 각 규칙은 영향을 받는 개체와 해당 개체가 만료되는 시기(특정 날짜 또는 특정 일 수 이후)를 지정합니다.

StorageGRID는 수명 주기 구성에서 최대 1,000개의 수명 주기 규칙을 지원합니다. 각 규칙에는 다음 XML 요소가 포함될 수 있습니다.

- 만료: 지정된 날짜에 도달하거나 지정된 일 수에 도달할 때 개체를 인제스트할 때로부터 개체를 삭제합니다.
- NoncurrentVersionExpiration: 지정된 일 수에 도달할 때 개체가 비전류가 되었을 때부터 개체를 삭제합니다.
- 필터(접두사, 태그)
- 상태
- ID입니다

각 오브젝트는 S3 버킷 라이프사이클 또는 ILM 정책의 보존 설정을 따릅니다. S3 버킷 라이프사이클이 구성되면 라이프사이클 만료 작업이 버킷 라이프사이클 필터와 일치하는 오브젝트에 대한 ILM 정책을 재정의합니다. 버킷 수명 주기 필터와 일치하지 않는 객체는 ILM 정책의 보존 설정을 사용합니다. 객체가 버킷 수명 주기 필터와 일치하고 만료 작업이 명시적으로 지정되지 않은 경우 ILM 정책의 보존 설정이 사용되지 않으며 객체 버전이 영구적으로 보존됩니다. [을 "S3 버킷 라이프사이클 및 ILM 정책의 우선순위 예" 참조하십시오.](#)

따라서 ILM 규칙의 배치 지침이 개체에 계속 적용되더라도 그리드에서 개체를 제거할 수 있습니다. 또는 개체에 대한 ILM 배치 지침이 만료된 후에도 개체가 그리드에 남아 있을 수 있습니다. 자세한 내용은 [을 참조하십시오 "ILM이 개체 수명 전반에 걸쳐 작동하는 방식"](#).



버킷 수명 주기 구성은 S3 오브젝트 잠금이 활성화된 버킷과 함께 사용할 수 있지만 버킷 수명 주기 구성은 레거시 준수 버킷에서 지원되지 않습니다.

StorageGRID는 다음 버킷 작업을 사용하여 라이프사이클 구성을 관리합니다.

- `DeleteBucketLifecycle`
- `GetBucketLifecycleConfiguration` 을 참조하십시오
- `PutBucketLifecycleConfiguration`을 참조하십시오

문서 수정 상태 설정 작성

라이프사이클 구성은 첫 번째 단계에서는 하나 이상의 규칙이 포함된 JSON 파일을 만듭니다. 예를 들어 이 JSON 파일에는 다음과 같은 세 가지 규칙이 포함되어 있습니다.

1. 규칙 1은 접두사/ `key2` 와 일치하고 값이 인 `tag2` 개체에만 `category1` 적용됩니다. `'Expiration'` 매개 변수는 필터와 일치하는 개체가 2020년 8월 22일 자정에 만료되도록 지정합니다.
2. 규칙 2는 접두사/ 과 일치하는 개체에만 `category2` 적용됩니다. `'Expiration'` 매개 변수는 필터와 일치하는 개체가 수집된 후 100일 후에 만료되도록 지정합니다.



일 수를 지정하는 규칙은 오브젝트가 수집된 시점을 기준으로 합니다. 현재 날짜가 수집 날짜와 일 수를 더한 값을 초과하면 라이프사이클 구성이 적용되는 즉시 일부 객체가 버킷에서 제거될 수 있습니다.

3. 규칙 3은 접두사/과 일치하는 개체에만 category3 적용됩니다. `Expiration` 매개 변수는 일치하는 개체의 현재 버전이 아닌 경우 50일 후에 만료되도록 지정합니다.

```
{  
    "Rules": [  
        {  
            "ID": "rule1",  
            "Filter": {  
                "And": {  
                    "Prefix": "category1/",  
                    "Tags": [  
                        {  
                            "Key": "key2",  
                            "Value": "tag2"  
                        }  
                    ]  
                }  
            },  
            "Expiration": {  
                "Date": "2020-08-22T00:00:00Z"  
            },  
            "Status": "Enabled"  
        },  
        {  
            "ID": "rule2",  
            "Filter": {  
                "Prefix": "category2/"  
            },  
            "Expiration": {  
                "Days": 100  
            },  
            "Status": "Enabled"  
        },  
        {  
            "ID": "rule3",  
            "Filter": {  
                "Prefix": "category3/"  
            },  
            "NoncurrentVersionExpiration": {  
                "NoncurrentDays": 50  
            },  
            "Status": "Enabled"  
        }  
    ]  
}
```

버킷에 라이프사이클 구성 적용합니다

문서 수정 상태 구성 파일을 만든 후 PutBucketLifecycleConfiguration 요청을 실행하여 버킷에 적용합니다.

이 요청은 예제 파일의 수명주기 구성 이름이 인 버킷의 객체에 testbucket 적용합니다.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-bucket-lifecycle-configuration  
--bucket testbucket --lifecycle-configuration file://bktjson.json
```

수명 주기 구성이 버킷에 성공적으로 적용되었는지 확인하려면 GetBucketLifecycleConfiguration 요청을 실행합니다. 예를 들면 다음과 같습니다.

```
aws s3api --endpoint-url <StorageGRID endpoint> get-bucket-lifecycle-configuration  
--bucket testbucket
```

성공적으로 응답하면 방금 적용한 문서 수정 상태 설정이 나열됩니다.

버킷 수명 주기 만료가 객체에 적용되는지 확인합니다

PutObject, HeadObject 또는 GetObject 요청을 실행할 때 수명 주기 구성의 만료 규칙이 특정 객체에 적용되는지 여부를 확인할 수 있습니다. 규칙이 적용되는 경우 응답에는 `Expiration` 객체가 만료되는 시점과 일치하는 만료 규칙을 나타내는 매개 변수가 포함됩니다.



버킷 수명 주기가 ILM을 재정의하기 때문에 `expiry-date` 표시된 날짜는 오브젝트가 삭제될 실제 날짜입니다. 자세한 내용은 ["객체 보존이 결정되는 방식"](#)을 참조하십시오.

예를 들어, 이 PutObject 요청은 2020년 6월 22일에 발행되었으며 버킷에 객체를 testbucket 배치합니다.

```
aws s3api --endpoint-url <StorageGRID endpoint> put-object  
--bucket testbucket --key obj2test2 --body bktjson.json
```

성공 응답은 객체가 100일(2020년 10월 1일) 내에 만료되고 라이프사이클 구성의 규칙 2와 일치함을 나타냅니다.

```
{  
  * "Expiration": "expiry-date=\\"Thu, 01 Oct 2020 09:07:49 GMT\\", rule-id=\\"rule2\\",  
  "ETag": "\\"9762f8a803bc34f5340579d4446076f7\\"  
}
```

예를 들어, 이 HeadObject 요청은 testbucket의 동일한 객체에 대한 메타데이터를 가져오는 데 사용되었습니다.

```
aws s3api --endpoint-url <StorageGRID endpoint> head-object  
--bucket testbucket --key obj2test2
```

성공 응답에는 개체의 메타데이터가 포함되며 개체가 100일 후에 만료되고 규칙 2와 일치함을 나타냅니다.

```
{  
    "AcceptRanges": "bytes",  
    * "Expiration": "expiry-date=\\"Thu, 01 Oct 2020 09:07:48 GMT\\", rule-  
    id=\\"rule2\\",  
    "LastModified": "2020-06-23T09:07:48+00:00",  
    "ContentLength": 921,  
    "ETag": "\\"9762f8a803bc34f5340579d4446076f7\\"  
    "ContentType": "binary/octet-stream",  
    "Metadata": {}  
}
```



버전 관리를 사용하는 버킷의 경우 `x-amz-expiration` 응답 헤더는 현재 버전의 객체에만 적용됩니다.

## S3 REST API 구현을 위한 권장사항

StorageGRID와 함께 사용할 S3 REST API를 구현할 때는 다음 권장 사항을 따라야 합니다.

### 존재하지 않는 객체에 대한 헤드 권장 사항

응용 프로그램이 실제로 존재하지 않을 것으로 예상되는 경로에 객체가 있는지 정기적으로 확인하는 경우 "사용 가능"을 사용해야 합니다. ["정합성"](#). 예를 들어, 애플리케이션이 PUT하기 전에 위치를 HEAD하는 경우 "사용 가능" 일관성을 사용해야 합니다.

그렇지 않으면 헤드 작업에서 객체를 찾지 못할 경우 같은 사이트에 있는 두 개 이상의 스토리지 노드를 사용할 수 없거나 원격 사이트에 연결할 수 없는 경우 500개의 내부 서버 오류가 발생할 수 있습니다.

요청을 사용하여 각 버킷에 대해 "사용 가능한" 일관성을 설정하거나 개별 API 작업에 대한 요청 헤더에서 일관성을 지정할 수 ["버킷 일관성을 유지합니다"](#) 있습니다.

### 개체 키에 대한 권장 사항

버킷이 처음 생성된 시점을 기준으로 오브젝트 키 이름에 대한 다음 권장 사항을 따르십시오.

### StorageGRID 11.4 또는 이전 버전에서 생성된 버킷

- 개체 키의 처음 네 문자로 임의의 값을 사용하지 마십시오. 이는 이전 AWS에서 권장하는 키 접두사와 다릅니다. 대신 와 같이 고유하지 않은 임의의 접두사를 사용합니다 `image`.
- 이전 AWS 권장 사항에 따라 키 접두사에 랜덤 및 고유 문자를 사용하려면 오브젝트 키에 디렉토리 이름이 접두사로 지정됩니다. 즉, 다음 형식을 사용합니다.

`mybucket/mydir/f8e3-image3132.jpg`

이 형식 대신:

mybucket/f8e3-image3132.jpg

### StorageGRID 11.4 이상에서 생성된 버킷

성능 모범 사례에 맞게 개체 키 이름을 제한하는 것은 필요하지 않습니다. 대부분의 경우 개체 키 이름의 처음 4개 문자에 임의의 값을 사용할 수 있습니다.



단, 짧은 시간 내에 모든 오브젝트를 지속적으로 제거하는 S3 워크로드가 예외입니다. 이 사용 사례에 대한 성능 영향을 최소화하려면 키와 같은 1000개의 오브젝트마다 주요 이름의 앞부분을 다르게 지정해야 합니다. 예를 들어, S3 클라이언트가 일반적으로 초당 2,000개의 오브젝트를 기록하고 ILM 또는 버킷 라이프사이클 정책에 따라 3일 후에 모든 오브젝트를 제거한다고 가정해 보겠습니다. 성능에 미치는 영향을 최소화하기 위해 다음과 같은 패턴을 사용하여 키의 이름을 지정할 수 있습니다.

/mybucket/mydir/yyyyymmddhhmmss-random\_UUID.jpg

"범위 읽기"에 대한 권장 사항

이 설정된 경우 **"저장된 개체를 압축하는 전역 옵션"** S3 클라이언트 응용 프로그램은 반환되는 바이트 범위를 지정하는 GetObject 작업을 수행하지 않아야 합니다. 이러한 "범위 읽기" 작업은 StorageGRID에서 요청된 바이트에 액세스하기 위해 개체의 압축을 효과적으로 해제해야 하기 때문에 비효율적입니다. 매우 큰 개체에서 작은 범위의 바이트를 요청하는 GetObject 작업은 특히 비효율적입니다. 예를 들어, 50GB의 압축된 개체에서 10MB 범위를 읽는 것은 비효율적입니다.

압축된 개체에서 범위를 읽으면 클라이언트 요청이 시간 초과될 수 있습니다.



개체를 압축해야 하고 클라이언트 응용 프로그램에서 범위 읽기를 사용해야 하는 경우 응용 프로그램의 읽기 시간 초과를 늘리십시오.

## Amazon S3 REST API 지원

### S3 REST API 구현 세부 정보

StorageGRID 시스템은 대부분의 작업을 지원하고 몇 가지 제한 사항이 있는 간단한 스토리지 서비스 API(API 버전 2006-03-01)를 구현합니다. S3 REST API 클라이언트 애플리케이션을 통합할 때 구현 세부 정보를 이해해야 합니다.

StorageGRID 시스템은 가상 호스팅 방식의 요청과 경로 스타일 요청을 모두 지원합니다.

날짜 처리

S3 REST API의 StorageGRID 구현은 유효한 HTTP 날짜 형식만 지원합니다.

StorageGRID 시스템은 날짜 값을 허용하는 모든 헤더에 대해 유효한 HTTP 날짜 형식만 지원합니다. 날짜의 시간 부분은 그리니치 표준시(GMT) 형식 또는 표준 시간대 오프셋 없이 UTC(국제 표준시) 형식으로 지정할 수 있습니다(+0000을 지정해야 함). 요청에 헤더를 포함하면 x-amz-date 날짜 요청 헤더에 지정된 모든 값이 재정의됩니다. AWS Signature 버전 4를 사용할 x-amz-date 경우 날짜 헤더가 지원되지 않으므로 서명된 요청에 헤더가 있어야 합니다.

## 공통 요청 헤더

StorageGRID 시스템은 한 가지 예외를 제외하고 에서 정의한 일반 요청 헤더를 ["Amazon Simple Storage Service API 참조: 일반 요청 헤더"](#) 지원합니다.

요청 헤더	구축
권한 부여	AWS Signature 버전 2에 대한 전체 지원  다음 경우를 제외하고 AWS Signature 버전 4 지원: <ul style="list-style-type: none"><li>에서 실제 페이로드 체크섬 값을 <code>x-amz-content-sha256</code> 제공하면 헤더에 값이 제공된 것처럼 값이 유효성 검사 없이 수락됩니다.</li><li>UNSIGNED-PAYLOAD 스트리밍을 의미하는 헤더 값 <code>aws-chunked</code>(예: <code>streaming-AWS4-HMAC-SHA256</code> 페이로드)을 제공하면 <code>x-amz-content-sha256</code> 청크 데이터에 대해 청크 서명이 확인되지 않습니다.</li></ul>

## 공통 응답 헤더

StorageGRID 시스템은 한 가지 예외를 제외하고 [\\_Simple Storage Service API Reference\\_](#)에 의해 정의된 모든 공통 응답 헤더를 지원합니다.

응답 헤더	구축
<code>X-amz-id-2</code>	사용 안 합니다

## 요청을 인증합니다

StorageGRID 시스템은 S3 API를 사용하여 오브젝트에 대한 인증된 액세스와 익명 액세스를 모두 지원합니다.

S3 API는 S3 API 요청을 인증하는 데 서명 버전 2 및 서명 버전 4를 지원합니다.

인증된 요청은 액세스 키 ID 및 비밀 액세스 키를 사용하여 서명해야 합니다.

StorageGRID 시스템은 HTTP 헤더와 쿼리 매개 변수 사용이라는 두 가지 인증 방법을 `Authorization` 지원합니다.

### HTTP 인증 헤더를 사용합니다

HTTP `Authorization` 헤더는 버킷 정책에서 허용하는 익명 요청을 제외하고 모든 S3 API 작업에서 사용됩니다. `'Authorization'` 헤더에는 요청을 인증하는 데 필요한 모든 서명 정보가 들어 있습니다.

### 쿼리 매개 변수를 사용합니다

쿼리 매개 변수를 사용하여 URL에 인증 정보를 추가할 수 있습니다. 이를 URL 사전 서명 이라고 하며, 이 URL을 사용하여 특정 리소스에 대한 임시 액세스 권한을 부여할 수 있습니다. 미리 지정된 URL을 가진 사용자는 리소스에 액세스하기 위해 비밀 액세스 키를 알 필요가 없으며, 이를 통해 타사에 리소스에 대한 제한된 액세스를 제공할 수 있습니다.

## 서비스에 대한 작업

StorageGRID 시스템은 서비스에 대해 다음 작업을 지원합니다.

작동	구축
ListBucket (이전 명칭: Get Service)	모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.
스토리지 사용량을 가져옵니다	StorageGRID <a href="#">"스토리지 사용량을 가져옵니다"</a> 요청에서는 계정에 의해 사용 중인 총 저장소 양과 계정에 연결된 각 버킷에 대해 알려줍니다. 이 작업은 /의 경로와 사용자 지정 쿼리 매개 변수(?x-ntap-sg-usage)가 추가된 서비스에 대한 작업입니다.
옵션 /	클라이언트 애플리케이션은 스토리지 노드의 사용 가능 여부를 확인하기 위해 S3 인증 자격 증명을 제공하지 않고 스토리지 노드의 S3 포트에 요청을 전송할 수 있습니다 <code>OPTIONS /</code> . 이 요청을 사용하여 모니터링을 수행하거나, 외부 로드 밸런서가 스토리지 노드가 다운된 시점을 식별하도록 할 수 있습니다.

## 버킷 작업

StorageGRID 시스템은 각 S3 테넌트 계정에 대해 최대 5,000개의 버킷을 지원합니다.

각 그리드에는 최대 100,000개의 버킷을 포함할 수 있습니다.

5,000개의 버킷을 지원하려면 그리드의 각 스토리지 노드에 최소 64GB의 RAM이 있어야 합니다.

버킷 이름 제한은 AWS US 표준 지역 제한을 따르지만, S3 가상 호스팅 스타일 요청을 지원하기 위해 DNS 명명 규칙으로 제한해야 합니다.

자세한 내용은 다음을 참조하십시오.

- ["Amazon Simple Storage Service 사용자 가이드: 버킷 할당량, 제한 및 제한 사항"](#)
- ["S3 끝점 도메인 이름을 구성합니다"](#)

ListObjects(GET Bucket) 및 ListObjectVersions(GET Bucket 개체 버전) 작업은 StorageGRID 지원합니다. ["일관성 값"](#).

개별 버킷에 대해 마지막 액세스 시간에 대한 업데이트가 설정되었는지 여부를 확인할 수 있습니다. 을 ["버킷 최종 액세스 시간 가져오기"](#) 참조하십시오.

다음 표에서는 StorageGRID에서 S3 REST API 버킷 작업을 구축하는 방법을 설명합니다. 이러한 작업을 수행하려면 계정에 필요한 액세스 자격 증명을 제공해야 합니다.

작동	구축
CreateBucket	<p>새 버킷을 생성합니다. 버킷을 만들면 버켓 소유자가 됩니다.</p> <ul style="list-style-type: none"> <li>버킷 이름은 다음 규칙을 준수해야 합니다. <ul style="list-style-type: none"> <li>각 StorageGRID 시스템에서 고유해야 합니다(테넌트 계정에서만 고유한 것은 아님).</li> <li>DNS를 준수해야 합니다.</li> <li>3자 이상 63자 이하여야 합니다.</li> <li>인접한 레이블이 마침표로 구분된 하나 이상의 레이블일 수 있습니다. 각 레이블은 소문자 또는 숫자로 시작하고 끝나야 하며 소문자, 숫자 및 하이픈만 사용할 수 있습니다.</li> <li>텍스트 형식의 IP 주소처럼 보이지 않아야 합니다.</li> <li>가상 호스팅 스타일 요청에서 기간을 사용하지 않아야 합니다. 마침표는 서버 와일드카드 인증서 확인에 문제를 일으킬 수 있습니다.</li> </ul> </li> <li>기본적으로 버킷은 지역에 생성되지만 us-east-1 요청 본문의 요청 요소를 사용하여 다른 영역을 지정할 수 LocationConstraint 있습니다. 요소를 사용할 때는 LocationConstraint 그리드 관리자 또는 그리드 관리 API를 사용하여 정의된 영역의 정확한 이름을 지정해야 합니다. 사용할 지역 이름을 모르는 경우 시스템 관리자에게 문의하십시오.</li> </ul> <p>참고: CreateBucket 요청이 StorageGRID에 정의되지 않은 영역을 사용하는 경우 오류가 발생합니다.</p> <ul style="list-style-type: none"> <li>요청 헤더를 포함하여 S3 오브젝트 잠금이 설정된 버킷을 생성할 수 x-amz-bucket-object-lock-enabled 있습니다. 을 <a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a>참조하십시오.</li> </ul> <p>버킷을 생성할 때 S3 오브젝트 잠금을 활성화해야 합니다. 버킷을 생성한 후에는 S3 오브젝트 잠금을 추가하거나 비활성화할 수 없습니다. S3 오브젝트 잠금에는 버킷 버전 관리가 필요하며, 이 버전은 버킷을 생성할 때 자동으로 활성화됩니다.</p>
삭제 버킷	버킷을 삭제합니다.
DeleteBucketCors	버킷에 대한 CORS 구성을 삭제합니다.
DeleteBucketEncryption	버킷에서 기본 암호화를 삭제합니다. 암호화된 기존 개체는 암호화된 상태로 유지되지만 버킷에 추가된 새 개체는 암호화되지 않습니다.
DeleteBucketLifecycle	버킷에서 문서 수정 상태 설정을 삭제합니다. 을 <a href="#">"S3 라이프사이클 구성을 생성합니다"</a> 참조하십시오.
DeleteBucketPolicy를 참조하십시오	버킷에 연결된 정책을 삭제합니다.

작동	구축
DeleteBuckReplication 을 참조하십시오	버킷에 연결된 복제 구성을 삭제합니다.
삭제 BucketTagging	<p>하위 리소스를 사용하여 tagging 버킷에서 모든 태그를 제거합니다.</p> <p>주의: 이 버킷에 대해 기본값이 아닌 ILM 정책 태그가 설정된 경우 이 버킷에 NTAP-SG-ILM-BUCKET-TAG 할당된 값이 있는 버킷 태그가 있습니다. 버킷 태그가 있는 경우 DeleteBucketTagging 요청을 NTAP-SG-ILM-BUCKET-TAG 실행하지 마십시오. 대신 태그 및 할당된 값만 사용하여 PutBucketTagging 요청을 NTAP-SG-ILM-BUCKET-TAG 실행하여 버킷에서 다른 모든 태그를 제거합니다. 버킷 태그를 수정하거나 제거하지 NTAP-SG-ILM-BUCKET-TAG 마십시오.</p>
GetBuckketAcl	양수 응답과 ID, DisplayName 및 버킷 소유자의 사용 권한을 반환합니다. 이는 소유자가 버킷에 대한 모든 액세스 권한을 가지고 있음을 나타냅니다.
GetBuckketCors 를 참조하십시오	버킷의 구성을 반환합니다 cors.
GetBuckketEncryption을 참조하십시오	버킷의 기본 암호화 구성을 반환합니다.
GetBuckLifecycleConfiguration 을 참조하십시오  (이전 명칭 Get Bucket 수명주기)	버킷의 수명주기 구성을 반환합니다. 을 <a href="#">"S3 라이프사이클 구성을 생성합니다"</a> 참조하십시오.
GetBucketLocation 을 참조하십시오  (이전 명칭 Get Bucket 수명주기)	CreateBucket 요청에서 요소를 사용하여 설정한 영역을 반환합니다 LocationConstraint. 버킷의 영역이 인 경우 us-east-1 영역에 대해 빈 문자열이 반환됩니다.
GetBuckNotificationConfiguration 을 참조하십시오  (이전 명칭 Get Bucket 알림)	버킷에 연결된 알림 구성을 반환합니다.
GetBuckketPolicy를 참조하십시오	버킷에 연결된 정책을 반환합니다.
GetBucketReplication 을 참조하십시오	버킷에 연결된 복제 구성을 반환합니다.

작동	구축
GetBucketTagging	<p>하위 리소스를 사용하여 <code>tagging</code> 버킷의 모든 태그를 반환합니다.</p> <p>주의: 이 버킷에 대해 기본값이 아닌 ILM 정책 태그가 설정된 경우 이 버킷에 NTAP-SG-ILM-BUCKET-TAG 할당된 값이 있는 버킷 태그가 있습니다. 이 태그를 수정하거나 제거하지 마십시오.</p>
GetBucketVersioning 을 참조하십시오	<p>이 구현에서는 하위 리소스를 사용하여 <code>versioning</code> 버킷의 버전 관리 상태를 반환합니다.</p> <ul style="list-style-type: none"> <li>• <code>blank</code>: 버전 관리가 활성화되지 않았습니다(버킷이 "버전 없음").</li> <li>• <code> 사용</code>: 버전 관리가 활성화됩니다</li> <li>• <code>일시 중단됨</code>: 버전 관리가 이전에 활성화되었으며 일시 중단되었습니다</li> </ul>
GetObjectLockConfiguration 을 참조하십시오	<p>구성된 경우 버킷 기본 보존 모드와 기본 보존 기간을 반환합니다.</p> <p>을 "<a href="#">S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다</a>" 참조하십시오.</p>
머리버킷	<p>버킷이 존재하는지, 버킷에 액세스할 수 있는 권한이 있는지 확인합니다.</p> <p>이 작업은 다음을 반환합니다.</p> <ul style="list-style-type: none"> <li>• <code>x-ntap-sg-bucket-id</code>: UUID 형식의 버킷의 UUID입니다.</li> <li>• <code>x-ntap-sg-trace-id</code>: 연결된 요청의 고유한 추적 ID입니다.</li> </ul>

작동	구축
ListObjects 및 ListObjectsV2 를 참조하십시오  (이전 명칭 Get Bucket)	<p>버킷에 있는 오브젝트의 일부 또는 전체(최대 1,000개)를 반환합니다. 객체에 대한 스토리지 클래스는 스토리지가 스토리지 클래스 옵션으로 수집된 경우에도 두 값 중 하나를 가질 수 있습니다. REDUCED_REDUNDANCY</p> <ul style="list-style-type: none"> <li>‘STANDARD’ 객체가 스토리지 노드로 구성된 스토리지 풀에 저장되었음을 나타냅니다.</li> <li>GLACIER- 클라우드 스토리지 풀에 지정된 외부 버킷으로 오브젝트가 이동되었음을 나타냅니다.</li> </ul> <p>버킷에 접두사가 같은 삭제된 키가 많은 경우 키가 포함되지 않은 키가 응답에 포함될 수 CommonPrefixes 있습니다.</p> <p>HeadObject 및 ListObject 요청의 경우 StorageGRID 다른 정밀도로 LastModified 타임스탬프를 반환하는 반면 AWS는 다음 예에서 볼 수 있듯이 동일한 정밀도로 타임스탬프를 반환합니다.</p> <ul style="list-style-type: none"> <li>StorageGRID HeadObject: “마지막 수정일”: “2024-09-26T16:43:24+00:00”</li> <li>StorageGRID ListObject: “마지막 수정일”: “2024-09-26T16:43:24.931000+00:00”</li> <li>AWS HeadObject: “마지막 수정일”: “2023-10-17T00:19:54+00:00”</li> <li>AWS ListObject: “마지막 수정일”: “2023-10-17T00:19:54+00:00”</li> </ul>
ListObjectVersions 를 선택합니다  (이전에 명명된 Get Bucket Object 버전)	버킷에서 읽기 권한을 가진 경우 이 작업을 하위 리소스와 함께 versions 사용하면 버킷에 있는 모든 오브젝트 버전의 메타데이터가 나열됩니다.
BucketCors의	버킷이 오리진 간 요청을 처리할 수 있도록 버킷에 대한 CORS 구성을 설정합니다. CORS(Cross-origin Resource Sharing)는 한 도메인의 클라이언트 웹 애플리케이션이 다른 도메인의 리소스에 액세스할 수 있도록 하는 보안 메커니즘입니다. 예를 들어, 이라는 S3 버킷을 사용하여 그래픽을 저장한다고 가정해 images 보겠습니다. 버킷에 대한 CORS 구성을 설정하면 images 해당 버킷의 이미지가 웹사이트에 표시되도록 할 수 <a href="http://www.example.com">http://www.example.com</a> 있습니다.
PutBucketEncryption을 참조하십시오	<p>기존 버킷의 기본 암호화 상태를 설정합니다. 버킷 수준 암호화가 활성화된 경우 버킷에 추가된 모든 새 오브젝트는 암호화됩니다. StorageGRID는 StorageGRID 관리 키로 서버 측 암호화를 지원합니다. 서버측 암호화 구성 규칙을 지정할 때 매개 변수를로 AES256 설정하고 SSEAlgorithm 매개 변수를 사용하지 마십시오 KMSMasterKeyID.</p> <p>객체 업로드 요청이 이미 암호화를 지정한 경우(즉, 요청에 요청 헤더가 포함된 경우) 버킷 기본 암호화 구성은 x-amz-server-side-encryption-* 무시됩니다.</p>

작동	구축
PutBucketLifecycleConfiguration을 참조하십시오  (이전에 명명된 Put Bucket 수명 주기)	<p>버킷에 대한 새 수명 주기 구성을 생성하거나 기존 수명 주기 구성을 대체합니다. StorageGRID는 수명 주기 구성에서 최대 1,000개의 수명 주기 규칙을 지원합니다. 각 규칙에는 다음 XML 요소가 포함될 수 있습니다.</p> <ul style="list-style-type: none"> <li>만료(일, 날짜, ExpiredObjectDeleteMarker)</li> <li>비 currentVersionExpiration(NewerNoncurrentVersions, NoncurrentDays)</li> <li>필터(접두사, 태그)</li> <li>상태</li> <li>ID입니다</li> </ul> <p>StorageGRID는 다음 작업을 지원하지 않습니다.</p> <ul style="list-style-type: none"> <li>AbortIncompleteMultipartUpload를 중단합니다</li> <li>전환</li> </ul> <p>을 "S3 라이프사이클 구성을 생성합니다" 참조하십시오. 버킷 수명 주기의 만료 작업이 ILM 배치 지침과 상호 작용하는 방식을 이해하려면 을 참조하십시오."ILM이 개체 수명 전반에 걸쳐 작동하는 방식"</p> <ul style="list-style-type: none"> <li>참고 *: 버킷 수명 주기 구성은 S3 오브젝트 잠금이 활성화된 버킷과 함께 사용할 수 있지만 레거시 준수 버킷에서는 버킷 수명 주기 구성이 지원되지 않습니다.</li> </ul>

작동	구축
PutBucketNotificationConfiguration을 참조하십시오 (이전에 명명된 Put Bucket 알림)	<p>요청 본문에 포함된 알림 구성 XML을 사용하여 버킷에 대한 알림을 구성합니다. 다음과 같은 구현 세부 사항에 유의해야 합니다.</p> <ul style="list-style-type: none"> <li>StorageGRID Amazon Simple Notification Service(Amazon SNS) 주제, Kafka 주제 또는 웹훅 엔드포인트를 대상으로 지원합니다. SQS(Simple Queue Service) 또는 AWS Lambda 엔드포인트는 지원되지 않습니다.</li> <li>알림 대상은 StorageGRID 엔드포인트의 URN으로 지정해야 합니다. 테넌트 관리자 또는 테넌트 관리 API를 사용하여 엔드포인트를 생성할 수 있습니다.</li> </ul> <p>알림 설정을 성공적으로 하려면 끝점이 있어야 합니다. 끝점이 없으면 400 Bad Request 코드와 함께 오류가 InvalidArgument 반환됩니다.</p> <ul style="list-style-type: none"> <li>다음 이벤트 유형에 대한 알림을 구성할 수 없습니다. 이러한 이벤트 유형은 * 지원되지 않습니다 *. <ul style="list-style-type: none"> <li>s3:ReducedRedundancyLostObject</li> <li>s3:ObjectRestore:Completed</li> </ul> </li> <li>StorageGRID에서 보낸 이벤트 알림은 다음 목록에 표시된 것처럼 일부 키를 포함하지 않고 다른 키에 대해 특정 값을 사용한다는 점을 제외하고 표준 JSON 형식을 사용합니다. <ul style="list-style-type: none"> <li>* eventSource * 를 선택합니다</li> <li>sgws:s3</li> <li>* awsRegion *</li> </ul> <p>포함되지 않음</p> <ul style="list-style-type: none"> <li>x-amz-id-2 *</li> </ul> <p>포함되지 않음</p> <ul style="list-style-type: none"> <li>* 표시 *</li> </ul> <p>urn:sgws:s3:::bucket_name</p> </li> </ul>
BucketPolicy를 참조하십시오	버킷에 첨부된 정책을 설정합니다. 보다 <a href="#">"버킷 및 그룹 액세스 정책을 사용합니다"</a> .

작동	구축
PutBucketReplication을 참조하십시오	<p>"<a href="#">StorageGRID CloudMirror 복제</a>" 요청 본문에 제공된 복제 구성 XML을 사용하여 버킷을 구성합니다. CloudMirror 복제의 경우 다음과 같은 구축 세부 정보를 알고 있어야 합니다.</p> <ul style="list-style-type: none"> <li>StorageGRID는 복제 구성의 V1만 지원합니다. 즉, StorageGRID는 규칙에 요소 사용을 지원하지 Filter 않으며 객체 버전을 삭제하기 위한 V1 규칙을 따릅니다. 자세한 내용은 <a href="#">Amazon Simple Storage Service 사용 설명서: 복제 구성</a>.</li> <li>버킷 복제는 버전 관리되거나 버전이 지정되지 않은 버킷에서 구성할 수 있습니다.</li> <li>복제 구성 XML의 각 규칙에서 다른 대상 버킷을 지정할 수 있습니다. 소스 버킷은 둘 이상의 대상 버킷에 복제할 수 있습니다.</li> <li>대상 버킷은 테넌트 관리자 또는 테넌트 관리 API에 지정된 StorageGRID 엔드포인트의 URN으로 지정해야 합니다. 을 <a href="#">"CloudMirror 복제를 구성합니다" 참조하십시오.</a></li> </ul> <p>복제 구성이 성공하려면 엔드포인트가 있어야 합니다. 끝점이 없으면 요청이 A로 실패합니다 400 Bad Request. 오류 메시지는 다음과 같습니다. <i>Unable to save the replication policy. The specified endpoint URN does not exist: URN.</i></p> <ul style="list-style-type: none"> <li>구성 XML에서 을 지정할 Role 필요는 없습니다. 이 값은 StorageGRID에서 사용되지 않으며 제출될 경우 무시됩니다.</li> <li>구성 XML에서 스토리지 클래스를 생략하면 StorageGRID에서는 STANDARD 기본적으로 스토리지 클래스를 사용합니다.</li> <li>소스 버킷에서 객체를 삭제하거나 소스 버킷 자체를 삭제하는 경우 지역 간 복제 동작은 다음과 같습니다. <ul style="list-style-type: none"> <li>복제되기 전에 오브젝트 또는 버킷을 삭제하면 객체/버킷이 복제되지 않으므로 사용자에게 통지되지 않습니다.</li> <li>복제된 후 오브젝트 또는 버킷을 삭제하면 StorageGRID는 지역 간 복제 V1에 대한 표준 Amazon S3 삭제 동작을 따릅니다.</li> </ul> </li> </ul>

작동	구축
BucketTagging	<p>에서는 하위 리소스를 사용하여 <code>tagging</code> 버킷의 태그 집합을 추가하거나 업데이트합니다. 버킷 태그를 추가할 때 다음과 같은 제한 사항을 숙지하십시오.</p> <ul style="list-style-type: none"> <li>StorageGRID 및 Amazon S3 모두 각 버킷당 최대 50개의 태그를 지원합니다.</li> <li>버킷과 연결된 태그에는 고유한 태그 키가 있어야 합니다. 태그 키의 길이는 최대 128자의 유니코드 문자일 수 있습니다.</li> <li>태그 값의 길이는 최대 256자의 유니코드 문자일 수 있습니다.</li> <li>키와 값은 대/소문자를 구분합니다.</li> </ul> <p>주의: 이 버킷에 대해 기본값이 아닌 ILM 정책 태그가 설정된 경우 이 버킷에 <code>NTAP-SG-ILM-BUCKET-TAG</code> 할당된 값이 있는 버킷 태그가 있습니다. <code>'NTAP-SG-ILM-BUCKET-TAG'</code> 버킷 태그가 모든 <code>PutBucketTagging</code> 요청에 할당된 값과 함께 포함되어 있는지 확인합니다. 이 태그를 수정하거나 제거하지 마십시오.</p> <p>참고: 이 작업은 버킷에 이미 있는 현재 태그를 덮어씁니다. 기존 태그를 세트에서 생략하면 해당 태그가 버킷에 대해 제거됩니다.</p>
PutBucketVersioning을 참조하십시오	<p>하위 리소스를 사용하여 <code>versioning</code> 기존 버킷의 버전 관리 상태를 설정합니다. 다음 값 중 하나를 사용하여 버전 관리 상태를 설정할 수 있습니다.</p> <ul style="list-style-type: none"> <li><code>Enabled</code>(사용): 버킷의 오브젝트에 대한 버전 관리를 활성화합니다. 버킷에 추가된 모든 오브젝트는 고유한 버전 ID를 받습니다.</li> <li><code>Suspended</code>(일시 중지됨): 버킷의 오브젝트에 대한 버전 관리를 비활성화합니다. 버킷에 추가된 모든 객체는 버전 ID를 <code>'null'</code> 받습니다.</li> </ul>
PutObjectLockConfiguration을 참조하십시오	<p>버킷 기본 보존 모드 및 기본 보존 기간을 구성하거나 제거합니다.</p> <p>기본 보존 기간이 수정되면 기존 개체 버전의 보존 기한은 그대로 유지되며 새 기본 보존 기간을 사용하여 다시 계산되지 않습니다.</p> <p>자세한 내용은 <a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a> 참조하십시오.</p>

## 객체에 대한 작업

### 객체에 대한 작업

이 섹션에서는 StorageGRID 시스템이 객체에 대해 S3 REST API 작업을 구축하는 방법에 대해 설명합니다.

다음 조건은 모든 개체 작업에 적용됩니다.

- StorageGRID ["일관성 값"](#) 다음을 제외한 모든 객체 작업은 지원됩니다.
  - `GetObjectAcl`
  - `OPTIONS /`

- PutObjectLegalHold를 선택합니다
  - PutObjectRetention
  - SelectObjectContent 를 선택합니다
- 동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.
  - StorageGRID 버킷의 모든 오브젝트는 익명 사용자 또는 다른 계정에서 만든 오브젝트를 포함하여 버킷 소유자가 소유합니다.

다음 표에서는 StorageGRID에서 S3 REST API 오브젝트 작업을 구현하는 방법을 설명합니다.

작동	구축
DeleteObject 를 클릭합니다	<p>MFA(다중 요소 인증) 및 응답 헤더는 x-amz-mfa 지원되지 않습니다.</p> <p>DeleteObject 요청을 처리할 때 StorageGRID는 저장된 모든 위치에서 개체의 모든 복사본을 즉시 제거하려고 시도합니다. 성공하면 StorageGRID는 즉시 클라이언트에 응답을 반환합니다. 위치를 일시적으로 사용할 수 없기 때문에 30초 이내에 모든 복사본을 제거할 수 없는 경우 StorageGRID는 제거할 복사본을 대기시킨 다음 클라이언트에 성공 여부를 표시합니다.</p> <p><b>버전 관리</b></p> <p>특정 버전을 제거하려면 요청자가 버킷 소유자여야 하고 하위 리소스를 사용해야 versionId 합니다. 이 하위 리소스를 사용하면 버전이 영구적으로 삭제됩니다. 삭제 마커에 해당하는 경우 versionId 응답 헤더는 x-amz-delete-marker 로 설정된 상태로 'true' 반환됩니다.</p> <ul style="list-style-type: none"> <li>• 버전 관리가 활성화된 버킷에서 하위 리소스 없이 오브젝트를 삭제하면 versionId 삭제 마커가 생성됩니다. 응답 헤더를 x-amz-delete-marker 사용하여 삭제 마커에 대한 를 versionId 반환하고 x-amz-version-id 응답 헤더를 로 반환한다. true</li> <li>• 버전 관리가 일시 중지된 버킷에서 하위 리소스 없이 개체를 삭제하면 versionId 기존 'null' 버전 또는 'null' 삭제 마커가 영구적으로 삭제되고 새 'null' 삭제 마커가 생성됩니다. x-amz-delete-marker 응답 헤더가 로 설정된 `true 상태로 반환됩니다.</li> <li>• 참고 *: 경우에 따라 객체에 대해 여러 개의 삭제 마커가 존재할 수 있습니다.</li> </ul> <p>거버넌스 모드에서 개체 버전을 삭제하는 방법은 을 <a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a> 참조하십시오.</p>
DeleteObjects 를 클릭합니다 (이전에 이름이 여러 개체 삭제)	<p>MFA(다중 요소 인증) 및 응답 헤더는 x-amz-mfa 지원되지 않습니다.</p> <p>동일한 요청 메시지에서 여러 객체를 삭제할 수 있습니다.</p> <p>거버넌스 모드에서 개체 버전을 삭제하는 방법은 을 <a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a> 참조하십시오.</p>

작동	구축
DeleteObjectTagging 을 선택합니다	<p>에서는 하위 리소스를 사용하여 tagging 개체에서 모든 태그를 제거합니다.</p> <p>버전 관리</p> <p>쿼리 매개 변수가 요청에 지정되지 않은 경우 <code>versionId</code> 작업은 버전 버킷의 가장 최신 버전의 객체에서 모든 태그를 삭제합니다. 개체의 현재 버전이 삭제 표식이면 응답 헤더가 <code>true</code> 상태로 "MethodNotAllowed" 상태가 <code>x-amz-delete-marker</code> 반환됩니다.</p>
GetObject 를 참조하십시오	<a href="#">"GetObject 를 참조하십시오"</a>
GetObjectAcl	계정에 필요한 액세스 자격 증명이 제공된 경우 이 작업은 개체 소유자의 ID, DisplayName 및 사용 권한과 함께 긍정적인 응답을 반환합니다. 이는 소유자가 개체에 대한 모든 액세스 권한을 가지고 있음을 나타냅니다.
GetObjectLegalHold 를 참조하십시오	<a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a>
GetObjectRetention을 참조하십시오	<a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a>
GetObjectTagging	<p>에서는 하위 리소스를 사용하여 tagging 개체의 모든 태그를 반환합니다.</p> <p>버전 관리</p> <p>쿼리 매개 변수가 요청에 지정되지 않은 경우 <code>versionId</code> 작업은 버전이 지정된 버킷에 있는 가장 최신 버전의 객체에서 모든 태그를 반환합니다. 개체의 현재 버전이 삭제 표식이면 응답 헤더가 <code>true</code> 상태로 "MethodNotAllowed" 상태가 <code>x-amz-delete-marker</code> 반환됩니다.</p>
HeadObject 를 선택합니다	<a href="#">"HeadObject 를 선택합니다"</a>
RestoreObject 를 선택합니다	<a href="#">"RestoreObject 를 선택합니다"</a>
PutObject 를 선택합니다	<a href="#">"PutObject 를 선택합니다"</a>
CopyObject 를 선택합니다 (이전에 명명된 Put Object - Copy)	<a href="#">"CopyObject 를 선택합니다"</a>
PutObjectLegalHold를 선택합니다	<a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a>
PutObjectRetention	<a href="#">"S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"</a>

작동	구축
PutObjectTagging	<p>에서는 하위 리소스를 사용하여 tagging 기준 개체에 태그 집합을 추가합니다.</p> <p><b>개체 태그 제한</b></p> <p>새 개체를 업로드할 때 태그를 추가하거나 기존 개체에 태그를 추가할 수 있습니다. StorageGRID 및 Amazon S3 모두 각 오브젝트에 대해 최대 10개의 태그를 지원합니다. 개체와 관련된 태그에는 고유한 태그 키가 있어야 합니다. 태그 키의 길이는 최대 128자의 유니코드 문자이고 태그 값의 길이는 최대 256자의 유니코드 문자일 수 있습니다. 키와 값은 대/소문자를 구분합니다.</p> <p><b>태그 업데이트 및 수집 동작</b></p> <p>PutObjectTagging을 사용하여 개체의 태그를 업데이트하는 경우 StorageGRID는 개체를 다시 수집하지 않습니다. 즉, 일치하는 ILM 규칙에 지정된 Ingest 동작 옵션이 사용되지 않습니다. ILM이 정상적인 백그라운드 ILM 프로세스에 의해 다시 평가될 때 업데이트로 인해 트리거되는 개체 배치에 대한 모든 변경 사항이 발생합니다.</p> <p>즉, ILM 규칙이 수집 동작에 Strict 옵션을 사용하는 경우 필요한 개체 배치를 만들 수 없는 경우(예: 새로 필요한 위치를 사용할 수 없음) 작업이 수행되지 않습니다. 업데이트된 오브젝트는 필요한 배치가 가능할 때까지 현재 위치를 유지합니다.</p> <p><b>충돌 해결</b></p> <p>동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.</p> <p><b>버전 관리</b></p> <p>쿼리 매개 변수가 요청에 지정되지 않은 경우 <code>versionId</code> 작업은 버전이 지정된 버킷에서 가장 최신 버전의 객체에 태그를 추가합니다. 개체의 현재 버전이 삭제 표식이면 응답 헤더가로 설정된 <code>true</code> 상태로 "MethodNotAllowed" 상태가 <code>x-amz-delete-marker</code> 반환됩니다.</p>
SelectObjectContent 를 선택합니다	" <a href="#">SelectObjectContent 를 선택합니다</a> "

**S3 Select**를 사용합니다

StorageGRID는 에 대해 다음과 같은 Amazon S3 Select 절, 데이터 형식 및 연산자를 ["SelectObjectContent 명령"](#) 지원합니다.



목록에 없는 항목은 지원되지 않습니다.

구문은 을 참조하십시오 "[SelectObjectContent 를 선택합니다](#)". S3 Select에 대한 자세한 내용은 를 ["S3 Select용 AWS 문서"](#) 참조하십시오.

S3 Select가 활성화된 테넌트 계정만 SelectObjectContent 쿼리를 실행할 수 있습니다. 를 ["S3 Select 사용에 대한](#)

## 고려 사항 및 요구 사항"참조하십시오.

### 절을 참조하십시오

- 목록을 선택합니다
- FROM 절
- WHERE 절
- Limit 절

### 데이터 유형

- 불입니다
- 정수
- 문자열
- 부동
- 십진수, 숫자
- 타임 스탬프입니다

### 연산자

#### 논리 연산자

- 및
- 아닙니다
- 또는

#### 비교 연산자

- 를 누릅니다
- 를 누릅니다
- lt;=.(&L
- GT;=.(&G
- =
- =
- 를 누릅니다
- !=
- 사이
- 인치

#### 패턴 일치 연산자

- 좋아요

- –
- %

#### 단일 작업자

- NULL입니다
- NULL이 아닙니다

#### 수학 연산자

- 를 누릅니다
- –
- \*
- /
- %

StorageGRID는 Amazon S3 Select 운영자 우선권을 따릅니다.

#### 집계 함수

- 평균()
- 개수(\*)
- 최대()
- 최소()
- 합계()

#### 조건부 함수

- 케이스
- 합체
- 노NIF

#### 변환 함수

- 캐스트(지원되는 데이터 형식용)

#### 날짜 함수

- date\_add
- Date\_DIFF(날짜/시간)
- 압축 풀기
- to\_string(대상 문자열)
- 를 \_TIMESTAMP로 설정합니다

- UTCNOW

## 문자열 함수

- char\_length, character\_length
- 낮음
- 부분 문자열
- 잘라내기
- 위쪽

서버측 암호화를 사용합니다

서버측 암호화를 통해 유휴 개체 데이터를 보호할 수 있습니다. StorageGRID는 개체를 쓸 때 데이터를 암호화하고 개체에 액세스할 때 데이터를 해독합니다.

서버측 암호화를 사용하려면 암호화 키가 관리되는 방식에 따라 상호 배타적인 두 가지 옵션 중 하나를 선택할 수 있습니다.

- \* SSE(StorageGRID 관리 키를 사용한 서버 측 암호화) \*: S3 요청을 발행하여 오브젝트를 저장할 때 StorageGRID는 고유 키를 사용하여 오브젝트를 암호화합니다. S3 요청을 통해 오브젝트를 검색할 때 StorageGRID는 저장된 키를 사용하여 오브젝트를 해독합니다.
- \* SSE-C(고객이 제공한 키를 사용한 서버측 암호화) \*: S3 요청을 발행하여 오브젝트를 저장할 때 사용자는 자신만의 암호화 키를 제공합니다. 오브젝트를 검색할 때 요청의 일부로 동일한 암호화 키를 제공합니다. 두 암호화 키가 일치하면 해당 개체는 해독되고 개체 데이터는 반환됩니다.

StorageGRID는 모든 개체 암호화 및 암호 해독 작업을 관리하지만 사용자가 제공하는 암호화 키를 관리해야 합니다.



제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다.



개체가 SSE 또는 SSE-C로 암호화된 경우 버킷 수준 또는 그리드 수준 암호화 설정은 무시됩니다.

## SSE를 사용합니다

StorageGRID에서 관리하는 고유 키를 사용하여 개체를 암호화하려면 다음 요청 헤더를 사용합니다.

x-amz-server-side-encryption

SSE 요청 헤더는 다음 오브젝트 작업에서 지원됩니다.

- "PutObject 를 선택합니다"
- "CopyObject 를 선택합니다"
- "CreateMultipartUpload 를 클릭합니다"

## SSE-C를 사용합니다

관리하는 고유 키로 개체를 암호화하려면 다음 세 가지 요청 헤더를 사용합니다.

요청 헤더	설명
x-amz-server-side-encryption-customer-algorithm	암호화 알고리즘을 지정합니다. 헤더 값은 AES256 이어야 합니다.
x-amz-server-side-encryption-customer-key	개체를 암호화하거나 해독하는 데 사용할 암호화 키를 지정합니다. 키의 값은 256비트 base64로 인코딩되어야 합니다.
x-amz-server-side-encryption-customer-key-MD5	RFC 1321에 따라 암호화 키의 MD5 다이제스트를 지정합니다. RFC 1321은 암호화 키가 오류 없이 전송되도록 하는 데 사용됩니다. MD5 다이제스트 값은 base64로 인코딩된 128비트여야 합니다.

SSE-C 요청 헤더는 다음 개체 작업에서 지원됩니다.

- "GetObject 를 참조하십시오"
- "HeadObject 를 선택합니다"
- "PutObject 를 선택합니다"
- "CopyObject 를 선택합니다"
- "CreateMultptUpload 를 클릭합니다"
- "업로드 파트"
- "업로드파트 복사"

고객이 제공한 키(**SSE-C**)와 함께 서버측 암호화 사용 시 고려 사항

SSE-C를 사용하기 전에 다음 사항을 고려하십시오.

- https를 사용해야 합니다.



StorageGRID는 SSE-C를 사용할 때 http를 통해 이루어진 모든 요청을 거부합니다. 보안을 위해 http를 사용하여 실수로 보낸 모든 키가 손상되는 것을 고려해야 합니다. 키를 폐기하고 필요에 따라 회전합니다.

- 응답의 ETag는 객체 데이터의 MD5가 아닙니다.
- 암호화 키를 개체에 매핑하는 작업을 관리해야 합니다. StorageGRID는 암호화 키를 저장하지 않습니다. 각 개체에 대해 제공하는 암호화 키를 추적할 책임은 사용자에게 있습니다.
- 버킷을 버전 관리가 활성화된 경우 각 오브젝트 버전에는 고유한 암호화 키가 있어야 합니다. 각 개체 버전에 사용되는 암호화 키를 추적할 책임은 사용자에게 있습니다.
- 클라이언트 측에서 암호화 키를 관리하기 때문에 클라이언트 측에서 키 회전과 같은 추가 보호 수단을 관리해야 합니다.



제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다.

- 버킷에 대해 교차 그리드 복제 또는 CloudMirror 복제가 구성된 경우 SSE-C 객체를 수집할 수 없습니다. 수집 작업이 실패합니다.

#### 관련 정보

["Amazon S3 사용자 가이드: SSE-C\(고객 제공 키\)와 함께 서버측 암호화 사용"](#)

**CopyObject** 를 선택합니다

S3 CopyObject 요청을 사용하여 이미 S3에 저장된 개체의 복사본을 만들 수 있습니다. CopyObject 작업은 GetObject 를 수행한 다음 PutObject 를 수행하는 작업과 같습니다.

#### 충돌 해결

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.

#### 개체 크기

단일 PutObject 작업의 maximum\_recommended\_size는 5GiB(5,368,709,120바이트)입니다. 5GiB보다 큰 오브젝트가 있는 경우 대신 ["멀티파트 업로드"](#) 사용하십시오.

단일 PutObject 작업의 maximum\_supported\_size는 5TiB(5,497,558,138,880바이트)입니다.



StorageGRID 11.6 이하에서 업그레이드한 경우 5GiB를 초과하는 객체를 업로드하려고 하면 S3 PUT 개체 크기가 너무 큼 경고가 트리거됩니다. StorageGRID 11.7 또는 11.8을 새로 설치한 경우 경고가 트리거되지 않습니다. 하지만 StorageGRID의 향후 릴리즈에서는 AWS S3 표준에 맞춰 5GiB보다 큰 오브젝트 업로드를 지원하지 않습니다.

#### 사용자 메타데이터의 **UTF-8** 문자

요청에 사용자 정의 메타데이터의 키 이름이나 값에 UTF-8 값이 포함되어 있으면 StorageGRID 동작이 정의되지 않습니다.

StorageGRID는 사용자 정의 메타데이터의 키 이름이나 값에 포함된 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 이스케이프된 UTF-8 문자는 ASCII 문자로 처리됩니다.

- 사용자 정의 메타데이터에 이스케이프된 UTF-8 문자가 포함된 경우 요청이 성공합니다.
- 키 이름 또는 값의 해석된 값에 인쇄할 수 없는 문자가 포함된 경우 StorageGRID는 헤더를 반환하지 x-amz-missing-meta 않습니다.

#### 지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- Content-Type
- x-amz-copy-source

- x-amz-copy-source-if-match
- x-amz-copy-source-if-none-match
- x-amz-copy-source-if-unmodified-since
- x-amz-copy-source-if-modified-since
- x-amz-meta-, 그 뒤에 사용자 정의 메타데이터를 포함하는 이름-값 쌍이 옵니다
- x-amz-metadata-directive: 기본값은 로 COPY, 개체 및 관련 메타데이터를 복사할 수 있습니다.

개체를 복사할 때 기존 메타데이터를 덮어쓰도록 지정하거나 개체 메타데이터를 업데이트하도록 지정할 수 REPLACE 있습니다.

- x-amz-storage-class
- x-amz-tagging-directive: 기본값은 로 COPY, 개체 및 모든 태그를 복사할 수 있습니다.

개체를 복사할 때 기존 태그를 덮어쓰도록 지정하거나 태그를 업데이트하도록 지정할 수 REPLACE 있습니다.

- S3 오브젝트 잠금 요청 헤더:

- x-amz-object-lock-mode
- x-amz-object-lock-retain-until-date
- x-amz-object-lock-legal-hold

이러한 헤더 없이 요청이 이루어지면 버킷 기본 보존 설정을 사용하여 오브젝트 버전 모드와 보존 기간을 계산합니다. 을 ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)참조하십시오.

- SSE 요청 헤더:

- x-amz-copy-source-server-side-encryption-customer-algorithm
- x-amz-copy-source-server-side-encryption-customer-key
- x-amz-copy-source-server-side-encryption-customer-key-MD5
- x-amz-server-side-encryption
- x-amz-server-side-encryption-customer-key-MD5
- x-amz-server-side-encryption-customer-key
- x-amz-server-side-encryption-customer-algorithm

을 참조하십시오 [서버측 암호화에 대한 요청 헤더](#)

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- Cache-Control
- Content-Disposition
- Content-Encoding

- Content-Language
- Expires
- x-amz-checksum-algorithm

개체를 복사할 때 소스 개체에 체크섬이 있으면 StorageGRID에서 해당 체크섬 값을 새 개체에 복사하지 않습니다. 이 동작은 개체 요청에서 을 사용하려고 하는지 여부에 관계없이 x-amz-checksum-algorithm 적용됩니다.

- x-amz-website-redirect-location

## 스토리지 클래스 옵션

`x-amz-storage-class` 요청 헤더가 지원되며 일치하는 ILM 규칙이 Dual Commit 또는 Balanced를 사용할 경우 StorageGRID에서 생성하는 객체 복제본 수에 영향을 xref:{relative\_path}.../ilm/data-protection-options-for-ingest.html["수집 옵션"] 줍니다.

- STANDARD

(기본값) ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 이중 커밋 수집 작업을 지정합니다.

- REDUCED\_REDUNDANCY

ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 단일 커밋 수집 작업을 지정합니다.



S3 오브젝트 잠금이 설정된 상태에서 오브젝트를 버킷에 수집하면 REDUCED\_REDUNDANCY 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷에 수집하는 경우 REDUCED\_REDUNDANCY 옵션은 오류를 반환합니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

## CopyObject에서 x-amz-copy-source 사용

헤더에 지정된 소스 버킷 및 키가 대상 버킷 및 키와 다른 경우 x-amz-copy-source 소스 오브젝트 데이터의 복사본이 대상에 기록됩니다.

소스와 대상이 일치하고 헤더가로 지정된 REPLACE 경우 x-amz-metadata-directive 객체의 메타데이터는 요청에 제공된 메타데이터 값으로 업데이트됩니다. 이 경우 StorageGRID는 오브젝트를 다시 수집하지 않습니다. 여기에는 두 가지 중요한 결과가 있습니다.

- 기존 개체를 현재 위치에서 암호화하거나 기존 개체의 암호화를 변경하는 데 CopyObject 를 사용할 수 없습니다. 헤더나 x-amz-server-side-encryption-customer-algorithm 헤더를 제공하면 x-amz-server-side-encryption StorageGRID는 요청을 거부하고 반환합니다 XNotImplemented.
- 일치하는 ILM 규칙에 지정된 Ingest 동작 옵션은 사용되지 않습니다. ILM이 정상적인 백그라운드 ILM 프로세스에 의해 다시 평가될 때 업데이트로 인해 트리거되는 개체 배치에 대한 모든 변경 사항이 발생합니다.

즉, ILM 규칙이 수집 동작에 Strict 옵션을 사용하는 경우 필요한 개체 배치를 만들 수 없는 경우(예: 새로 필요한 위치를 사용할 수 없음) 작업이 수행되지 않습니다. 업데이트된 오브젝트는 필요한 배치가 가능할 때까지 현재

위치를 유지합니다.

## 서버측 암호화에 대한 요청 헤더

사용자가 ["서버측 암호화를 사용합니다"](#) 제공하는 요청 헤더는 소스 개체가 암호화되어 있는지 여부와 대상 개체를 암호화할 계획인지에 따라 달라집니다.

- 소스 객체가 SSE-C(고객 제공 키)를 사용하여 암호화되는 경우 CopyObject 요청에 다음 세 개의 헤더를 포함해야 객체를 해독한 후 복사할 수 있습니다.
  - x-amz-copy-source-server-side-encryption-customer-algorithm: AES256 지정합니다.
  - x-amz-copy-source-server-side-encryption-customer-key: 원본 개체를 만들 때 제공한 암호화 키를 지정합니다.
  - x-amz-copy-source-server-side-encryption-customer-key-MD5: 원본 개체를 만들 때 제공한 MD5 다이제스트를 지정합니다.
- 제공 및 관리하는 고유 키를 사용하여 대상 개체(복사본)를 암호화하려면 다음 세 개의 머리글을 포함합니다.
  - x-amz-server-side-encryption-customer-algorithm: AES256 지정합니다.
  - x-amz-server-side-encryption-customer-key: 대상 객체에 대한 새 암호화 키를 지정합니다.
  - x-amz-server-side-encryption-customer-key-MD5: 새 암호화 키의 MD5 다이제스트를 지정합니다.



제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 대한 고려 사항을 ["서버 측 암호화 사용 검토하십시오."](#)

- SSE(StorageGRID)에서 관리하는 고유 키를 사용하여 대상 개체(복사본)를 암호화하려는 경우 CopyObject 요청에 다음 헤더를 포함합니다.

- x-amz-server-side-encryption



server-side-encryption`개체 값을 업데이트할 수 없습니다. 대신 다음을 사용하여 새 값으로 `x-amz-metadata-directive 복사본을 server-side-encryption 만듭니다 REPLACE.

## 버전 관리

소스 버킷의 버전이 관리되는 경우 헤더를 사용하여 개체의 최신 버전을 복사할 수 x-amz-copy-source 있습니다. 개체의 특정 버전을 복사하려면 하위 리소스를 사용하여 복사할 버전을 명시적으로 versionId 합니다. 대상 버킷의 버전이 지정된 경우 생성된 버전이 x-amz-version-id 응답 헤더에 반환됩니다. 대상 버킷에 대한 버전 관리가 일시 중단된 경우 x-amz-version-id "null" 값이 반환됩니다.

**GetObject** 를 참조하십시오

S3 GetObject 요청을 사용하여 S3 버킷에서 객체를 검색할 수 있습니다.

## GetObject 및 multipart 개체

요청 매개변수를 사용하여 멀티파트나 분할된 오브젝트의 특정 부분을 검색할 수 `partNumber` 있습니다. `x-amz-mp-parts-count` 응답 요소는 개체의 파트 수를 나타냅니다.

분할/다중 파트 오브젝트 및 비분할/비다중 파트 오브젝트 모두에 대해 1로 설정할 수 `partNumber` 있지만 `x-amz-mp-parts-count` 응답 요소는 분할된 오브젝트 또는 다중 파트 오브젝트에 대해서만 반환됩니다.

## 사용자 메타데이터의 UTF-8 문자

StorageGRID는 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자가 있는 개체에 대한 요청 가져오기 키 이름 또는 값에 인쇄할 수 없는 문자가 포함된 경우 헤더를 반환하지 `x-amz-missing-meta` 않습니다.

## 지원되는 요청 헤더

다음 요청 헤더가 지원됩니다.

- `x-amz-checksum-mode`: 지정합니다 `ENABLED`

`Range` 헤더는 `GetObject`에 대해 지원되지 `x-amz-checksum-mode` 않습니다.  
활성화된 상태로 요청에 `x-amz-checksum-mode` 포함하면 `Range` StorageGRID는 응답에 체크섬 값을 반환하지 않습니다.

## 지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않으며 `xNotImplemented` 반환됩니다.

- `x-amz-website-redirect-location`

## 버전 관리

하위 리소스가 지정되지 않은 경우 `versionId`이 작업은 버전이 지정된 버킷에서 가장 최신 버전의 개체를 가져옵니다. 객체의 현재 버전이 삭제 마커인 경우 응답 헤더가 `true`로 설정된 `true` 상태에서 "찾을 수 없음" 상태가 `x-amz-delete-marker` 반환됩니다.

## 고객이 제공한 암호화 키(**SSE-C**)를 사용하여 서버측 암호화를 위한 요청 헤더

사용자가 제공한 고유 키로 개체를 암호화한 경우 머리를 3개를 모두 사용합니다.

- `x-amz-server-side-encryption-customer-algorithm`: AES256 지정합니다.
- `x-amz-server-side-encryption-customer-key`: 개체의 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: 개체의 암호화 키의 MD5 다이제스트를 지정합니다.

 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 ["서버측 암호화를 사용합니다"](#).

## 클라우드 스토리지 풀 객체에 대한 **GetObject**의 동작입니다

개체가 예 저장된 경우 "클라우드 스토리지 풀"GetObject 요청의 동작은 개체의 상태에 따라 달라집니다. 자세한 내용은 ["HeadObject 를 선택합니다"](#) 참조하십시오.



오브젝트가 클라우드 스토리지 풀에 저장되고 하나 이상의 오브젝트 복제본이 그리드에 있는 경우  
GetObject 요청은 클라우드 스토리지 풀에서 검색하기 전에 그리드에서 데이터 검색을 시도합니다.

개체의 상태입니다	GetObject 의 동작입니다
StorageGRID로 수집되었지만 아직 ILM에 의해 평가되지 않은 오브젝트 또는 기존 스토리지 풀에 저장된 오브젝트 또는 삭제 코딩 사용	200 OK 개체의 복사본이 검색됩니다.
Cloud Storage Pool의 개체이지만 아직 검색할 수 없는 상태로 전환되지 않았습니다	200 OK 개체의 복사본이 검색됩니다.
개체가 검색할 수 없는 상태로 전환되었습니다	403 Forbidden, InvalidObjectState 요청을 사용하여 <a href="#">"RestoreObject 를 선택합니다"</a> 객체를 검색 가능한 상태로 복원합니다.
복구할 수 없는 상태에서 복원 중인 개체입니다	403 Forbidden, InvalidObjectState RestoreObject 요청이 완료될 때까지 기다립니다.
객체가 클라우드 스토리지 풀에 완전히 복구되었습니다	200 OK 개체의 복사본이 검색됩니다.

## 클라우드 스토리지 풀에서 다중 또는 분할 오브젝트

여러 부분 개체를 업로드했거나 StorageGRID가 큰 개체를 세그먼트로 분할한 경우 StorageGRID는 개체의 부분 또는 세그먼트의 하위 집합을 샘플링하여 클라우드 저장소 풀에서 개체를 사용할 수 있는지 여부를 결정합니다. 개체의 일부 부분이 이미 검색할 수 없는 상태로 전환되었거나 개체의 일부 부분이 아직 복원되지 않은 경우 GetObject 요청이 잘못 반환될 수 200 OK 있습니다.

다음과 같은 경우:

- GetObject 요청에서 일부 데이터를 반환하지만 전송 도중에 중지됩니다.
- 이후의 GetObject 요청이 반환될 수 403 Forbidden 있습니다.

## GetObject 및 교차 그리드 복제

를 사용 중이며 ["교차 그리드 복제"](#) 버킷에 대해 활성화된 경우 ["그리드 통합"](#) S3 클라이언트는 GetObject 요청을 실행하여 개체의 복제 상태를 확인할 수 있습니다. 응답에는 다음 값 중 하나가 있는 StorageGRID 관련 x-ntap-sg-cgr-replication-status 응답 헤더가 포함됩니다.

그리드	복제 상태입니다
출처	<ul style="list-style-type: none"> <li>* 완료됨 *: 복제가 성공했습니다.</li> <li>* 보류 중 *: 객체가 아직 복제되지 않았습니다.</li> <li>* 실패 *: 영구적인 장애로 인해 복제에 실패했습니다. 사용자가 오류를 해결해야 합니다.</li> </ul>
목적지	<ul style="list-style-type: none"> <li>replica *: 객체가 소스 그리드에서 복제되었습니다.</li> </ul>



StorageGRID는 헤더를 지원하지 `x-amz-replication-status` 않습니다.

#### HeadObject 를 선택합니다

S3 HeadObject 요청을 사용하여 개체 자체를 반환하지 않고 개체에서 메타데이터를 검색할 수 있습니다. 객체가 클라우드 스토리지 풀에 저장된 경우 HeadObject 를 사용하여 객체의 전환 상태를 확인할 수 있습니다.

#### HeadObject 및 multipart 개체

요청 매개 변수를 사용하여 멀티파트 또는 분할된 개체의 특정 부분에 대한 메타데이터를 검색할 수 `partNumber` 있습니다. `x-amz-mp-parts-count` 응답 요소는 개체의 파트 수를 나타냅니다.

분할/다중 파트 오브젝트 및 비분할/비다중 파트 오브젝트 모두에 대해 1로 설정할 수 `partNumber` 있지만 `x-amz-mp-parts-count` 응답 요소는 분할된 오브젝트 또는 다중 파트 오브젝트에 대해서만 반환됩니다.

#### 사용자 메타데이터의 UTF-8 문자

StorageGRID는 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 사용자 정의 메타데이터에서 이스케이프된 UTF-8 문자가 있는 개체에 대한 헤드 요청은 키 이름이나 값에 인쇄할 수 없는 문자가 포함된 경우 헤더를 반환하지 `x-amz-missing-meta` 않습니다.

#### 지원되는 요청 헤더

다음 요청 헤더가 지원됩니다.

- `x-amz-checksum-mode`

`partNumber` 매개 변수 및 `Range` 머리글은 HeadObject 에 대해 지원되지 `x-amz-checksum-mode` 않습니다. 활성화된 상태로 요청에 포함하면 `x-amz-checksum-mode` StorageGRID는 응답에 체크섬 값을 반환하지 않습니다.

#### 지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않으며 `xNotImplemented` 반환됩니다.

- `x-amz-website-redirect-location`

## 버전 관리

하위 리소스가 지정되지 않은 경우 `versionId` 이 작업은 버전이 지정된 버킷에서 가장 최신 버전의 개체를 가져옵니다. 객체의 현재 버전이 삭제 마커인 경우 응답 헤더가 `true` 상태에서 "찾을 수 없음" 상태가 `x-amz-delete-marker` 반환됩니다.

고객이 제공한 암호화 키(**SSE-C**)를 사용하여 서버측 암호화를 위한 요청 헤더

사용자가 제공한 고유 키로 개체를 암호화한 경우 이 헤더 3개를 모두 사용합니다.

- `x-amz-server-side-encryption-customer-algorithm`: AES256 지정합니다.
- `x-amz-server-side-encryption-customer-key`: 개체의 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: 개체의 암호화 키의 MD5 다이제스트를 지정합니다.

 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 ["서버측 암호화를 사용합니다"](#).

클라우드 스토리지 풀 객체에 대한 **HeadObject** 응답입니다

개체가 예 저장된 경우 ["클라우드 스토리지 풀"](#) 다음 응답 헤더가 반환됩니다.

- `x-amz-storage-class`: GLACIER
- `x-amz-restore`

응답 헤더는 클라우드 스토리지 풀로 이동되는 오브젝트의 상태에 대한 정보를 제공하며, 선택적으로 검색할 수 없는 상태로 전환된 후 복구됩니다.

개체의 상태입니다	<b>HeadObject</b> 에 대한 응답입니다
StorageGRID로 수집되었지만 아직 ILM에 의해 평가되지 않은 오브젝트 또는 기존 스토리지 풀에 저장된 오브젝트 또는 삭제 코딩 사용	200 OK (특수 응답 헤더가 반환되지 않음)
Cloud Storage Pool의 개체이지만 아직 검색할 수 없는 상태로 전환되지 않았습니다	200 OK  <code>x-amz-storage-class: GLACIER</code>  <code>x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"</code>  오브젝트가 검색 불가능한 상태로 전환될 때까지의 값은 <code>expiry-date</code> 나중에 먼 시간으로 설정됩니다. 정확한 전환 시간은 StorageGRID 시스템에 의해 제어되지 않습니다.

개체의 상태입니다	HeadObject에 대한 응답입니다
개체가 검색할 수 없는 상태로 전환되었지만 하나 이상의 복사본이 그리드에 있습니다	<p>200 OK</p> <p>x-amz-storage-class: GLACIER</p> <p>x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2030 00:00:00 GMT"</p> <p>의 값은 expiry-date 나중에 먼 시간으로 설정됩니다.</p> <ul style="list-style-type: none"> <li>참고 *: 그리드의 복제본을 사용할 수 없는 경우(예: 스토리지 노드가 다운된 경우), 객체를 성공적으로 검색하기 전에 클라우드 스토리지 풀에서 복제본을 복원하도록 요청을 해야 <a href="#">"RestoreObject를 선택합니다"</a>합니다.</li> </ul>
개체가 검색할 수 없는 상태로 전환되었으며 그리드에 복사본이 없습니다	<p>200 OK</p> <p>x-amz-storage-class: GLACIER</p>
복구할 수 없는 상태에서 복원 중인 개체입니다	<p>200 OK</p> <p>x-amz-storage-class: GLACIER</p> <p>x-amz-restore: ongoing-request="true"</p>
객체가 클라우드 스토리지 풀에 완전히 복구되었습니다	<p>200 OK</p> <p>x-amz-storage-class: GLACIER</p> <p>x-amz-restore: ongoing-request="false", expiry-date="Sat, 23 July 20 2018 00:00:00 GMT"</p> <p>는 expiry-date 클라우드 스토리지 풀의 객체가 검색할 수 없는 상태로 반환되는 시점을 나타냅니다.</p>

### Cloud Storage Pool에서 다중 또는 분할 오브젝트 지원

여러 부분 개체를 업로드했거나 StorageGRID가 큰 개체를 세그먼트로 분할한 경우 StorageGRID는 개체의 부분 또는 세그먼트의 하위 집합을 샘플링하여 클라우드 저장소 풀에서 개체를 사용할 수 있는지 여부를 결정합니다. 개체의 일부 부분이 이미 검색할 수 없는 상태로 전환되었거나 개체의 일부 부분이 아직 복원되지 않은 경우 HeadObject 요청이 잘못 반환될 수 x-amz-restore: ongoing-request="false" 있습니다.

### HeadObject 및 교차 그리드 복제

를 사용 중이며 "교차 그리드 복제"버킷에 대해 활성화된 경우 "그리드 통합" S3 클라이언트는 HeadObject 요청을 실행하여 객체의 복제 상태를 확인할 수 있습니다. 응답에는 다음 값 중 하나가 있는 StorageGRID 관련 x-ntap-sg-

cgr-replication-status 응답 헤더가 포함됩니다.

그리드	복제 상태입니다
출처	<ul style="list-style-type: none"><li>* 완료됨 *: 복제가 성공했습니다.</li><li>* 보류 중 *: 객체가 아직 복제되지 않았습니다.</li><li>* 실패 *: 영구적인 장애로 인해 복제에 실패했습니다. 사용자가 오류를 해결해야 합니다.</li></ul>
목적지	<ul style="list-style-type: none"><li>replica *: 객체가 소스 그리드에서 복제되었습니다.</li></ul>



StorageGRID는 헤더를 지원하지 x-amz-replication-status 않습니다.

**PutObject**를 선택합니다

S3 PutObject 요청을 사용하여 객체를 버킷에 추가할 수 있습니다.

충돌 해결

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.

객체 크기

단일 PutObject 작업의 maximum\_recommended\_size는 5GiB(5,368,709,120바이트)입니다. 5GiB보다 큰 오브젝트가 있는 경우 대신 ["멀티파트 업로드"](#) 사용하십시오.

단일 PutObject 작업의 maximum\_supported\_size는 5TiB(5,497,558,138,880바이트)입니다.



StorageGRID 11.6 이하에서 업그레이드한 경우 5GiB를 초과하는 객체를 업로드하려고 하면 S3 PUT 개체 크기가 너무 큼 경고가 트리거됩니다. StorageGRID 11.7 또는 11.8을 새로 설치한 경우 경고가 트리거되지 않습니다. 하지만 StorageGRID의 향후 릴리즈에서는 AWS S3 표준에 맞춰 5GiB보다 큰 오브젝트 업로드를 지원하지 않습니다.

사용자 메타데이터 크기입니다

Amazon S3는 각 PUT 요청 헤더 내의 사용자 정의 메타데이터 크기를 2KB로 제한합니다. StorageGRID는 사용자 메타데이터를 24KiB로 제한합니다. 사용자 정의 메타데이터의 크기는 각 키와 값의 UTF-8 인코딩에서 바이트 수의 합계를 구하여 측정됩니다.

사용자 메타데이터의 **UTF-8** 문자

요청에 사용자 정의 메타데이터의 키 이름이나 값에 UTF-8 값이 포함되어 있으면 StorageGRID 동작이 정의되지 않습니다.

StorageGRID는 사용자 정의 메타데이터의 키 이름이나 값에 포함된 이스케이프된 UTF-8 문자를 구문 분석하거나 해석하지 않습니다. 이스케이프된 UTF-8 문자는 ASCII 문자로 처리됩니다.

- 사용자 정의 메타데이터에 이스케이프된 UTF-8 문자가 포함된 경우 PutObject, CopyObject, GetObject 및 HeadObject 요청이 성공합니다.
- 키 이름 또는 값의 해석된 값에 인쇄할 수 없는 문자가 포함된 경우 StorageGRID는 헤더를 반환하지 x-amz-missing-meta 않습니다.

## 개체 태그 제한

새 개체를 업로드할 때 태그를 추가하거나 기존 개체에 태그를 추가할 수 있습니다. StorageGRID 및 Amazon S3 모두 각 오브젝트에 대해 최대 10개의 태그를 지원합니다. 개체와 관련된 태그에는 고유한 태그 키가 있어야 합니다. 태그 키의 길이는 최대 128자의 유니코드 문자이고 태그 값의 길이는 최대 256자의 유니코드 문자일 수 있습니다. 키와 값은 대/소문자를 구분합니다.

## 개체 소유권

StorageGRID에서는 소유자가 아닌 계정 또는 익명 사용자가 만든 개체를 포함하여 모든 개체가 버킷 소유자 계정에 의해 소유됩니다.

## 지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- Cache-Control
- Content-Disposition
- Content-Encoding

Content-Encoding StorageGRID에 대해 를 aws-chunked 지정하면 다음 항목이 검증되지 않습니다.

- StorageGRID는 청크 데이터에 대해 를 확인하지 chunk-signature 않습니다.
- StorageGRID는 사용자가 개체에 대해 제공한 값을 확인하지 x-amz-decoded-content-length 않습니다.

- Content-Language
- Content-Length
- Content-MD5
- Content-Type
- Expires
- Transfer-Encoding

페이지로드 서명도 사용되는 경우 청크 전송 인코딩이 aws-chunked 지원됩니다.

- x-amz-checksum-sha256
- x-amz-meta-, 그 뒤에 사용자 정의 메타데이터를 포함하는 이름-값 쌍이 옵니다.

사용자 정의 메타데이터에 대한 이름 값 쌍을 지정할 때 다음 일반 형식을 사용합니다.

```
x-amz-meta-name: value
```

사용자 정의 생성 시간 \* 옵션을 ILM 규칙의 참조 시간으로 사용하려면 객체를 생성할 때 기록하는 메타데이터의 이름으로 를 사용해야 `creation-time` 합니다. 예를 들면 다음과 같습니다.

```
x-amz-meta-creation-time: 1443399726
```

의 값은 `creation-time` 1970년 1월 1일 이후 초로 계산됩니다.



ILM 규칙은 참조 시간에 \* 사용자 정의 생성 시간 \* 과 Balanced 또는 Strict 수집 옵션을 모두 사용할 수 없습니다. ILM 규칙을 만들면 오류가 반환됩니다.

- `x-amz-tagging`
- S3 오브젝트 잠금 요청 헤더
  - `x-amz-object-lock-mode`
  - `x-amz-object-lock-retain-until-date`
  - `x-amz-object-lock-legal-hold`

이러한 헤더 없이 요청이 이루어지면 버킷 기본 보존 설정을 사용하여 오브젝트 버전 모드와 보존 기간을 계산합니다. 을 ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#) 참조하십시오.

- SSE 요청 헤더:
  - `x-amz-server-side-encryption`
  - `x-amz-server-side-encryption-customer-key-MD5`
  - `x-amz-server-side-encryption-customer-key`
  - `x-amz-server-side-encryption-customer-algorithm`

을 참조하십시오 [서버측 암호화에 대한 요청 헤더](#)

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- `If-Match`
- `If-None-Match`
- `x-amz-acl`
- `x-amz-sdk-checksum-algorithm`
- `x-amz-trailer`
- `x-amz-website-redirect-location`

`x-amz-website-redirect-location` 헤더가 를 `XNotImplemented` 반환합니다.

## 스토리지 클래스 옵션

`x-amz-storage-class` 요청 헤더가 지원됩니다. 예 제출된 `x-amz-storage-class` 값은 StorageGRID 시스템 (ILM에 의해 결정됨)에 저장되는 오브젝트의 영구 사본의 수가 아니라 수집 중에 StorageGRID가 오브젝트 데이터를 보호하는 방법에 영향을 줍니다.

수집된 객체와 일치하는 ILM 규칙이 Strict 수집 옵션을 사용하는 경우 `x-amz-storage-class` 헤더는 영향을 미치지 않습니다.

다음 값을 사용할 수 `x-amz-storage-class` 있습니다.

- STANDARD (기본값)

- \* 이중 커밋 \*: ILM 규칙이 Ingest 동작에 대한 이중 커밋 옵션을 지정하는 경우, 개체가 수집되는 즉시 해당 개체의 두 번째 복사본이 생성되어 다른 스토리지 노드(이중 커밋)에 배포됩니다. ILM을 평가할 때 StorageGRID는 이러한 초기 중간 복사본이 규칙의 배치 지침을 충족하는지 여부를 결정합니다. 그렇지 않으면 새 오브젝트 복사본을 다른 위치에 만들어야 하고 초기 중간 복사본을 삭제해야 할 수 있습니다.
- \* 균형 \*: ILM 규칙이 균형 옵션을 지정하고 StorageGRID이 규칙에 지정된 모든 복사본을 즉시 만들 수 없는 경우 StorageGRID은 다른 스토리지 노드에 두 개의 임시 복사본을 만듭니다.

StorageGRID가 ILM 규칙(동기식 배치)에 지정된 모든 오브젝트 복사본을 즉시 생성할 수 있다면 `x-amz-storage-class` 헤더는 효과가 없습니다.

- REDUCED\_REDUNDANCY

- \* 이중 커밋 \*: ILM 규칙이 Ingest 동작에 대한 이중 커밋 옵션을 지정하는 경우 StorageGRID는 오브젝트가 수집될 때(단일 커밋) 단일 임시 복사본을 만듭니다.
- \* 균형 \*: ILM 규칙이 균형 옵션을 지정하는 경우 StorageGRID은 시스템에서 규칙에 지정된 모든 사본을 즉시 만들 수 없는 경우에만 단일 중간 복사본을 만듭니다. StorageGRID에서 동기 배치를 수행할 수 있는 경우 이 머리글은 영향을 주지 않습니다. 이 REDUCED\_REDUNDANCY 옵션은 오브젝트와 일치하는 ILM 규칙이 복제된 단일 복사본을 생성할 때 가장 적합합니다. 이 경우를 REDUCED\_REDUNDANCY 사용하면 모든 수집 작업에서 불필요한 추가 오브젝트 복사본을 생성하여 삭제할 필요가 없습니다.

다른 상황에서는 이 옵션을 사용하지 REDUCED\_REDUNDANCY 않는 것이 좋습니다. REDUCED\_REDUNDANCY 수집 중 오브젝트 데이터가 손실될 위험이 증가함 예를 들어, ILM 평가가 발생하기 전에 실패한 스토리지 노드에 단일 복사본이 처음 저장되는 경우 데이터가 손실될 수 있습니다.

 복제된 복사본이 항상 하나만 있으면 데이터가 영구적으로 손실될 위험이 있습니다. 복제된 객체 복제본이 하나만 있는 경우 스토리지 노드에 장애가 발생하거나 심각한 오류가 발생한 경우 해당 객체가 손실됩니다. 또한 업그레이드와 같은 유지보수 절차 중에는 개체에 대한 액세스가 일시적으로 중단됩니다.

`REDUCED\_REDUNDANCY` 지정하면 오브젝트를 처음 수집할 때 생성되는 복사본의 개수만 영향을 줍니다. 활성 ILM 정책에 따라 오브젝트를 평가할 때 생성되는 오브젝트 복사본 수에 영향을 미치지 않으며 StorageGRID 시스템에서 더 낮은 수준의 이중화로 데이터가 저장되지는 않습니다.

 S3 오브젝트 잠금이 설정된 상태에서 오브젝트를 버킷에 수집하면 REDUCED\_REDUNDANCY 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷에 수집하는 경우 REDUCED\_REDUNDANCY 옵션은 오류를 반환합니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

#### 서버측 암호화에 대한 요청 헤더

다음 요청 헤더를 사용하여 서버측 암호화를 사용하여 개체를 암호화할 수 있습니다. SSE 및 SSE-C 옵션은 상호 배타적입니다.

- \* SSE \*: StorageGRID에서 관리하는 고유 키를 사용하여 오브젝트를 암호화하려면 다음 헤더를 사용하십시오.
  - x-amz-server-side-encryption

머리글이 PutObject 요청에 포함되어 있지 않으면 x-amz-server-side-encryption PutObject 응답에서 그리드 전체의 머리글이 ["저장된 개체 암호화 설정입니다"](#) 생략됩니다.

- \* SSE-C \*: 사용자가 제공 및 관리하는 고유 키로 객체를 암호화하려면 이 헤더 세 개를 모두 사용합니다.
  - x-amz-server-side-encryption-customer-algorithm: AES256 지정합니다.
  - x-amz-server-side-encryption-customer-key: 새 개체에 대한 암호화 키를 지정합니다.
  - x-amz-server-side-encryption-customer-key-MD5: 새 개체의 암호화 키의 MD5 다이제스트를 지정합니다.

 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 해당 고려 사항을 ["서버 측 암호화 사용"](#) 검토하십시오.

 개체가 SSE 또는 SSE-C로 암호화된 경우 버킷 수준 또는 그리드 수준 암호화 설정은 무시됩니다.

#### 버전 관리

버킷에 대해 버전 관리를 사용하도록 설정하면 저장되는 오브젝트 버전에 대해 고유한 `versionId` 버전이 자동으로 생성됩니다. `versionId` 응답 헤더를 사용하여 응답에서도 `x-amz-version-id` 반환됩니다.

버전 관리가 일시 중단되면 개체 버전이 null로 저장되고 null 버전이 이미 있는 경우 해당 버전을 `versionId` 덮어씁니다.

#### 승인 헤더의 서명 계산

헤더를 사용하여 요청을 인증하는 경우 Authorization StorageGRID는 다음과 같은 점에서 AWS와 다릅니다.

- StorageGRID에서는 머리글이 에 포함될 CanonicalHeaders 필요가 host 없습니다.
- StorageGRID는 Content-Type에 포함되지 CanonicalHeaders 않아도 됩니다.

- StorageGRID에서는 머리글이 포함될 CanonicalHeaders 필요가 x-amz-\* 없습니다.



일반적으로 이러한 헤더를 항상 포함시켜 유효성을 확인하지만 이러한 헤더를 CanonicalHeaders 제외하면 StorageGRID에서 오류를 반환하지 않습니다.

자세한 내용은 ["승인 헤더에 대한 서명 계산: 단일 청크\(AWS 서명 버전 4\)로 페이로드 전송"](#) 참조하십시오.

#### 관련 정보

- ["ILM을 사용하여 개체를 관리합니다"](#)
- ["Amazon Simple Storage Service API 참조: PutObject"](#)

**RestoreObject** 를 선택합니다

S3 RestoreObject 요청을 사용하여 클라우드 스토리지 풀에 저장된 개체를 복원할 수 있습니다.

지원되는 요청 유형입니다

StorageGRID에서는 객체를 복원하기 위한 RestoreObject 요청만 지원합니다. 복원 유형을 지원하지 SELECT 않습니다. 요청 반환을 선택합니다 XNotImplemented.

#### 버전 관리

선택적으로 versionId 버전 지정된 버킷에서 특정 버전의 개체를 복원하도록 지정합니다. 을 지정하지 `versionId` 않으면 개체의 최신 버전이 복원됩니다

클라우드 스토리지 풀 객체에서 **RestoreObject**의 동작입니다

개체가에 저장된 경우 ["클라우드 스토리지 풀"](#) RestoreObject 요청에는 개체의 상태에 따라 다음과 같은 동작이 발생합니다. 자세한 내용은 ["HeadObject 를 선택합니다"](#) 참조하십시오.



객체가 클라우드 스토리지 풀에 저장되어 있고 하나 이상의 객체 복제본도 그리드에 있는 경우 RestoreObject 요청을 실행하여 객체를 복구할 필요가 없습니다. 대신 GetObject 요청을 사용하여 로컬 복사본을 직접 검색할 수 있습니다.

개체의 상태입니다	<b>RestoreObject</b> 의 동작입니다
StorageGRID로 수집되었지만 ILM에서 아직 평가되지 않은 오브젝트 또는 클라우드 스토리지 풀에 없는 오브젝트	403 Forbidden, InvalidObjectState
Cloud Storage Pool의 개체이지만 아직 검색할 수 없는 상태로 전환되지 않았습니다	200 OK 변경되지 않았습니다. <ul style="list-style-type: none"> <li>참고 * : 개체가 검색 불가능한 상태로 전환되기 전에 개체를 변경할 수 expiry-date 없습니다.</li> </ul>

개체의 상태입니다	RestoreObject의 동작입니다
개체가 검색할 수 없는 상태로 전환되었습니다	<p>202 Accepted 요청 본문에 지정된 일 수 동안 객체의 검색 가능한 복제본을 클라우드 스토리지 풀에 복구합니다. 이 기간이 끝나면 객체는 복구할 수 없는 상태로 돌아갑니다.</p> <p>필요한 경우, request 요소를 사용하여 Tier 복원 작업이 완료되는 데 걸리는 시간을 (Expedited Standard`결정합니다, 또는 `Bulk). 지정하지 않으면 Tier Standard 계층이 사용됩니다.</p> <ul style="list-style-type: none"> <li>중요 *: 개체가 S3 Glacier Deep Archive로 전환되었거나 클라우드 스토리지 풀이 Azure Blob 스토리지를 사용하는 경우 계층을 사용하여 복원할 수 없습니다 Expedited. 다음 오류가 403 Forbidden 반환됩니다 InvalidTier Retrieval option is not supported by this storage class.</li> </ul>
복구할 수 없는 상태에서 복원 중인 개체입니다	409 Conflict, RestoreAlreadyInProgress
객체가 클라우드 스토리지 풀에 완전히 복구되었습니다	<p>200 OK</p> <ul style="list-style-type: none"> <li>참고: * 오브젝트가 검색 가능한 상태로 복원된 경우에 대한 새 값으로 RestoreObject 요청을 다시 실행하여 오브젝트를 Days 변경할 수 expiry-date 있습니다. 복원 날짜는 요청 시간을 기준으로 업데이트됩니다.</li> </ul>

#### SelectObjectContent 를 선택합니다

S3 SelectObjectContent 요청을 사용하여 간단한 SQL 문을 기반으로 S3 개체의 내용을 필터링할 수 있습니다.

자세한 내용은 ["Amazon Simple Storage Service API 참조: SelectObjectContent"](#) 참조하십시오.

#### 시작하기 전에

- 테넌트 계정에 S3 Select 권한이 있습니다.
- `s3:GetObject` 쿼리할 개체에 대한 사용 권한이 있습니다.
- 쿼리할 객체는 다음 형식 중 하나여야 합니다.
  - CSV \*. GZIP 또는 BZIP2 보관 파일로 압축하거나 그대로 사용할 수 있습니다.
  - \* 파케 \*. Parquet 객체에 대한 추가 요구 사항:
    - S3 Select는 GZIP 또는 Snappy를 사용한 컬럼 압축만 지원합니다. S3 Select는 Parquet 오브젝트에 대한 전체 오브젝트 압축을 지원하지 않습니다.
    - S3 Select는 Parquet 출력을 지원하지 않습니다. 출력 형식을 CSV 또는 JSON으로 지정해야 합니다.
    - 압축되지 않은 최대 행 그룹 크기는 512MB입니다.
    - 개체의 스키마에 지정된 데이터 형식을 사용해야 합니다.
    - 간격, JSON, 목록, 시간 또는 UUID 논리적 유형은 사용할 수 없습니다.

- SQL 식의 최대 길이는 256KB입니다.
- 입력 또는 결과에 있는 모든 레코드의 최대 길이는 1MiB입니다.

## CSV 요청 구문 예

```

POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.abc-company.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Expression>string</Expression>
  <ExpressionType>string</ExpressionType>
  <RequestProgress>
    <Enabled>boolean</Enabled>
  </RequestProgress>
  <InputSerialization>
    <CompressionType>GZIP</CompressionType>
    <CSV>
      <AllowQuotedRecordDelimiter>boolean</AllowQuotedRecordDelimiter>
      <Comments>#</Comments>
      <FieldDelimiter>\t</FieldDelimiter>
      <FileHeaderInfo>USE</FileHeaderInfo>
      <QuoteCharacter>'"</QuoteCharacter>
      <QuoteEscapeCharacter>\\</QuoteEscapeCharacter>
      <RecordDelimiter>\n</RecordDelimiter>
    </CSV>
  </InputSerialization>
  <OutputSerialization>
    <CSV>
      <FieldDelimiter>string</FieldDelimiter>
      <QuoteCharacter>string</QuoteCharacter>
      <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
      <QuoteFields>string</QuoteFields>
      <RecordDelimiter>string</RecordDelimiter>
    </CSV>
  </OutputSerialization>
  <ScanRange>
    <End>long</End>
    <Start>long</Start>
  </ScanRange>
</SelectObjectContentRequest>

```

## 쪽모이 세공 요청 구문 예

```
POST /{Key+}?select&select-type=2 HTTP/1.1
Host: Bucket.s3.abc-company.com
x-amz-expected-bucket-owner: ExpectedBucketOwner
<?xml version="1.0" encoding="UTF-8"?>
<SelectObjectContentRequest xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
  <Expression>string</Expression>
  <ExpressionType>string</ExpressionType>
  <RequestProgress>
    <Enabled>boolean</Enabled>
  </RequestProgress>
  <InputSerialization>
    <CompressionType>GZIP</CompressionType>
    <PARQUET>
      </PARQUET>
    </InputSerialization>
    <OutputSerialization>
      <CSV>
        <FieldDelimiter>string</FieldDelimiter>
        <QuoteCharacter>string</QuoteCharacter>
        <QuoteEscapeCharacter>string</QuoteEscapeCharacter>
        <QuoteFields>string</QuoteFields>
        <RecordDelimiter>string</RecordDelimiter>
      </CSV>
    </OutputSerialization>
    <ScanRange>
      <End>long</End>
      <Start>long</Start>
    </ScanRange>
  </SelectObjectContentRequest>
```

## SQL 쿼리의 예

이 쿼리는 시/도 이름, 2010년 인구, 2015년 예상 인구, 미국 인구 조사 데이터의 변경 비율을 가져옵니다. 상태가 아닌 파일의 레코드는 무시됩니다.

```
SELECT STNAME, CENSUS2010POP, POPESTIMATE2015, CAST((POPESTIMATE2015 -
CENSUS2010POP) AS DECIMAL) / CENSUS2010POP * 100.0 FROM S3Object WHERE
NAME = STNAME
```

쿼리할 파일의 처음 몇 줄은 SUB-EST2020\_ALL.csv 다음과 같습니다.

```

SUMLEV,STATE,COUNTY,PLACE,COUSUB,CONCIT,PRIMGEO_FLAG,FUNCSTAT,NAME,STNAME,
CENSUS2010POP,
ESTIMATESBASE2010,POPESTIMATE2010,POPESTIMATE2011,POPESTIMATE2012,POPESTIM
ATE2013,POPESTIMATE2014,
POPESTIMATE2015,POPESTIMATE2016,POPESTIMATE2017,POPESTIMATE2018,POPESTIMAT
E2019,POPESTIMATE042020,
POPESTIMATE2020
040,01,000,00000,00000,00000,0,A,Alabama,Alabama,4779736,4780118,4785514,4
799642,4816632,4831586,
4843737,4854803,4866824,4877989,4891628,4907965,4920706,4921532
162,01,000,00124,00000,00000,0,A,Abbeville
city,Alabama,2688,2705,2699,2694,2645,2629,2610,2602,
2587,2578,2565,2555,2555,2553
162,01,000,00460,00000,00000,0,A,Adamsville
city,Alabama,4522,4487,4481,4474,4453,4430,4399,4371,
4335,4304,4285,4254,4224,4211
162,01,000,00484,00000,00000,0,A,Addison
town,Alabama,758,754,751,750,745,744,742,734,734,728,
725,723,719,717

```

### AWS-CLI 사용 예(CSV)

```

aws s3api select-object-content --endpoint-url https://10.224.7.44:10443
--no-verify-ssl --bucket 619c0755-9e38-42e0-a614-05064f74126d --key SUB-
EST2020_ALL.csv --expression-type SQL --input-serialization '{"CSV":'
{"FileHeaderInfo": "USE", "Comments": "#", "QuoteEscapeCharacter": "\"",
"RecordDelimiter": "\n", "FieldDelimiter": ",", "QuoteCharacter": "\"",
"AllowQuotedRecordDelimiter": false}, "CompressionType": "NONE"}' --output
-serialization '{"CSV": {"QuoteFields": "ASNEEDED",
"QuoteEscapeCharacter": "#", "RecordDelimiter": "\n", "FieldDelimiter": ",",
"QuoteCharacter": "\""} }' --expression "SELECT STNAME, CENSUS2010POP,
POPESTIMATE2015, CAST((POPESTIMATE2015 - CENSUS2010POP) AS DECIMAL) /
CENSUS2010POP * 100.0 FROM S3Object WHERE NAME = STNAME" changes.csv

```

출력 파일의 처음 몇 줄은 changes.csv 다음과 같습니다.

```

Alabama,4779736,4854803,1.5705260708959658022953568983726297854
Alaska,710231,738430,3.9703983633493891424057806544631253775
Arizona,6392017,6832810,6.8959922978928247531256565807005832431
Arkansas,2915918,2979732,2.1884703204959810255295244928012378949
California,37253956,38904296,4.4299724839960620557988526104449148971
Colorado,5029196,5454328,8.4532796097030221132761578590295546246

```

## AWS-CLI 사용 예(Parquet)

```
aws s3api select-object-content --endpoint-url https://10.224.7.44:10443
--bucket 619c0755-9e38-42e0-a614-05064f74126d --key SUB-
EST2020_ALL.parquet --expression "SELECT STNAME, CENSUS2010POP,
POPESTIMATE2015, CAST((POPESTIMATE2015 - CENSUS2010POP) AS DECIMAL) /
CENSUS2010POP * 100.0 FROM S3Object WHERE NAME = STNAME" --expression-type
'SQL' --input-serialization '{"Parquet":{}}' --output-serialization
'{"CSV": {}}' changes.csv
```

출력 파일의 처음 몇 줄인 changes.csv는 다음과 같습니다.

```
Alabama,4779736,4854803,1.5705260708959658022953568983726297854
Alaska,710231,738430,3.9703983633493891424057806544631253775
Arizona,6392017,6832810,6.8959922978928247531256565807005832431
Arkansas,2915918,2979732,2.1884703204959810255295244928012378949
California,37253956,38904296,4.4299724839960620557988526104449148971
Colorado,5029196,5454328,8.4532796097030221132761578590295546246
```

## 멀티파트 업로드 작업

### 멀티파트 업로드 작업

이 섹션에서는 StorageGRID가 멀티파트 업로드 작업을 지원하는 방법에 대해 설명합니다.

다음 조건 및 참고 사항은 모든 다중파트 업로드 작업에 적용됩니다.

- 해당 버킷에 대한 ListMultipartUploads 쿼리의 결과가 불완전한 결과를 반환할 수 있으므로 단일 버킷에 대한 동시 다중파트 업로드가 1,000개를 초과해서는 안 됩니다.
- StorageGRID는 여러 파트에 대해 AWS 크기 제한을 적용합니다. S3 클라이언트는 다음 지침을 따라야 합니다.
  - 멀티파트 업로드의 각 파트는 5MiB(5,242,880바이트)와 5GiB(5,368,709,120바이트) 사이여야 합니다.
  - 마지막 부분은 5MiB(5,242,880바이트)보다 작을 수 있습니다.
  - 일반적으로 파트 크기는 가능한 한 커야합니다. 예를 들어, 100GiB 개체의 경우 5GiB의 파트 크기를 사용합니다. 각 파트는 고유한 개체로 간주되므로 큰 파트 크기를 사용하면 StorageGRID 메타데이터 오버헤드가 줄어듭니다.
  - 5GiB보다 작은 오브젝트의 경우 대신 비다중파트 업로드를 사용하는 것이 좋습니다.
- ILM 규칙이 Balanced 또는 Strict를 사용하는 경우, ILM은 수집 시 멀티파트 개체의 각 부분과 다중파트 업로드가 완료될 때 개체 전체에 대해 평가됩니다. ["수집 옵션"](#) 이 사항이 개체 및 파트 배치에 미치는 영향에 대해 알고 있어야 합니다.
  - S3 다중파트 업로드가 진행되는 동안 ILM이 변경되면 다중파트 업로드가 완료될 때 개체의 일부 부분이 현재 ILM 요구사항을 충족하지 못할 수 있습니다. 올바르게 배치되지 않은 모든 부품은 ILM 재평가를 위해 대기열에 추가되고 나중에 올바른 위치로 이동됩니다.
  - 파트에 대한 ILM을 평가할 때 StorageGRID은 개체의 크기가 아닌 파트 크기를 필터링합니다. 즉, 개체의

일부를 개체에 대한 ILM 요구 사항을 전체가 충족하지 않는 위치에 저장할 수 있습니다. 예를 들어, 모든 작은 오브젝트가 DC2에 저장되지만 10GB 이상의 오브젝트는 모두 DC1에 저장되도록 규칙이 지정된 경우 10부분 다중 부분 업로드의 각 1GB 부분은 인제스트 시 DC2에 저장됩니다. 그러나 개체 전체에 대해 ILM을 평가하면 개체의 모든 부분이 DC1로 이동됩니다.

- 모든 다중파트 업로드 작업은 StorageGRID 지원합니다. ["일관성 값"](#).
- 멀티파트 업로드를 사용하여 개체를 수집하는 경우 이 ["객체 분할 임계값\(1GiB\)"](#) 적용되지 않습니다.
- 필요에 따라 멀티파트 업로드와 함께을 사용할 수 ["서버 측 암호화"](#)입니다. SSE(StorageGRID 관리 키가 있는 서버측 암호화)를 사용하려면 CreateMultipartUpload 요청에만 요청 헤더를 포함합니다 x-amz-server-side-encryption. SSE-C(고객 제공 키를 사용한 서버측 암호화)를 사용하려면 CreateMultipartUpload 요청과 이후의 각 UploadPart 요청에서 동일한 3개의 암호화 키 요청 헤더를 지정합니다.
- 다중파트 업로드된 객체는 다음에 포함됩니다. ["가지 버킷"](#) 업로드가 완료되는 시점과 관계없이, 기본 버킷의 Before 타임스탬프 이전에 수집이 시작된 경우입니다.

작동	구축
AbortMultipartUpload 를 클릭합니다	모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.
CompleteMultipartUpload를 클릭합니다	을 참조하십시오 <a href="#">"CompleteMultipartUpload를 클릭합니다"</a>
CreateMultipartUpload 를 클릭합니다 (이전에 명명된 다중파트 업로드 시작)	을 참조하십시오 <a href="#">"CreateMultipartUpload 를 클릭합니다"</a>
ListMultipartUploads 를 참조하십시오	을 참조하십시오 <a href="#">"ListMultipartUploads 를 참조하십시오"</a>
목록 파트	모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.
업로드 파트	을 참조하십시오 <a href="#">"업로드 파트"</a>
업로드파트 복사	을 참조하십시오 <a href="#">"업로드파트 복사"</a>

**CompleteMultipartUpload**를 클릭합니다

CompleteMultipartUpload 작업은 이전에 업로드한 부품을 조립하여 객체의 다중 부분 업로드를 완료합니다.



StorageGRID는 CompleteMultipartUpload를 사용하여 요청 매개 변수에 대해 비연속 값들 오름차순으로 partNumber 지원합니다. 매개 변수는 어떤 값으로든 시작할 수 있습니다.

## 충돌 해결

동일한 키에 쓰는 두 클라이언트 등의 충돌하는 클라이언트 요청은 "최신 성공" 기준으로 해결됩니다. "Latest-WINS" 평가 시기는 S3 클라이언트가 작업을 시작할 때가 아니라 StorageGRID 시스템이 지정된 요청을 완료하는 시점을 기준으로 합니다.

지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- x-amz-checksum-sha256
- x-amz-storage-class

‘x-amz-storage-class’ 일치하는 ILM 규칙이 을 지정한 경우 머리글에 StorageGRID에서 생성하는 개체 복사본 수에 영향을 xref:{relative\_path}../ilm/data-protection-options-for-ingest.html [“이중 커밋 또는 Balanced 인제스트 옵션”] 줍니다.

- STANDARD

(기본값) ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 이중 커밋 수집 작업을 지정합니다.

- REDUCED\_REDUNDANCY

ILM 규칙이 이중 커밋 옵션을 사용하거나 균형 옵션이 중간 복사본 생성으로 돌아올 때 단일 커밋 수집 작업을 지정합니다.



S3 오브젝트 잠금이 설정된 상태에서 오브젝트를 버킷에 수집하면 REDUCED\_REDUNDANCY 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷에 수집하는 경우 REDUCED\_REDUNDANCY 옵션은 오류를 반환합니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.



15일 이내에 여러 부분 업로드가 완료되지 않으면 작업이 비활성으로 표시되고 모든 관련 데이터가 시스템에서 삭제됩니다.



ETag` 반환되는 값은 데이터의 MD5 합계가 아니라 다중 개체 값의 Amazon S3 API 구현을 `ETag 따릅니다.

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- If-Match
- If-None-Match
- x-amz-sdk-checksum-algorithm
- x-amz-trailer

버전 관리

이 작업은 여러 부분 업로드를 완료합니다. 버켓에 대해 버전 관리가 활성화된 경우 다중 파트 업로드가 완료된 후 개체 버전이 생성됩니다.

버킷에 대해 버전 관리를 사용하도록 설정하면 저장되는 오브젝트 버전에 대해 고유한 `versionId` 버전이 자동으로 생성됩니다. `versionId` 응답 헤더를 사용하여 응답에서도 `x-amz-version-id` 반환됩니다.

버전 관리가 일시 중단되면 개체 버전이 `null`로 저장되고 `null` 버전이 이미 있는 경우 해당 버전을 `versionId` 덮어씁니다.



버켓에 대해 버전 관리가 활성화된 경우, 같은 개체 키에서 동시 다중 파트 업로드가 완료된 경우에도 다중 파트 업로드를 완료하면 항상 새 버전이 생성됩니다. 버켓에 대해 버전 관리를 사용하지 않으면 다중 파트 업로드를 시작한 다음 다른 다중 파트 업로드를 시작하여 동일한 개체 키에서 먼저 완료할 수 있습니다. 비버전 버킷에서는 마지막으로 완료한 다중 파트 업로드가 우선 적용됩니다.

복제, 알림 또는 메타데이터 알림에 실패했습니다

플랫폼 서비스에 대해 다중 파트 업로드가 발생하는 버킷이 구성된 경우 연결된 복제 또는 알림 작업이 실패한 경우에도 다중 파트 업로드가 성공합니다.

테넌트는 개체의 메타데이터 또는 태그를 업데이트하여 실패한 복제 또는 알림을 트리거할 수 있습니다. 테넌트는 불필요한 변경을 방지하기 위해 기존 값을 다시 제출할 수 있습니다.

을 ["플랫폼 서비스 문제 해결"](#) 참조하십시오.

**CreateMultipartUpload**를 클릭합니다

**CreateMultipartUpload**(이전에 이름이 Multipart Upload 시작) 작업은 개체에 대한 다중 부분 업로드를 시작하고 업로드 ID를 반환합니다.

`x-amz-storage-class` 요청 헤더가 지원됩니다. 예 제출된 `x-amz-storage-class` 값은 StorageGRID 시스템(ILM에 의해 결정됨)에 저장되는 오브젝트의 영구 사본의 수가 아니라 수집 중에 StorageGRID가 오브젝트 데이터를 보호하는 방법에 영향을 줍니다.

수집된 객체와 일치하는 ILM 규칙이 Strict 를 사용하는 경우 ["수집 옵션"](#) `x-amz-storage-class` 헤더는 영향을 미치지 않습니다.

다음 값을 사용할 수 `x-amz-storage-class` 있습니다.

- STANDARD (기본값)
  - \* Dual Commit \*: ILM 규칙이 Dual Commit Ingest 옵션을 지정하는 경우 오브젝트가 수집되는 즉시 해당 오브젝트의 두 번째 복사본이 생성되어 다른 스토리지 노드(Dual Commit)로 배포됩니다. ILM을 평가할 때 StorageGRID는 이러한 초기 중간 복사본이 규칙의 배치 지침을 충족하는지 여부를 결정합니다. 그렇지 않으면 새 오브젝트 복사본을 다른 위치에 만들어야 하고 초기 중간 복사본을 삭제해야 할 수 있습니다.
  - \* 균형 \*: ILM 규칙이 균형 옵션을 지정하고 StorageGRID이 규칙에 지정된 모든 복사본을 즉시 만들 수 없는 경우 StorageGRID은 다른 스토리지 노드에 두 개의 임시 복사본을 만듭니다.
- REDUCED\_REDUNDANCY

- \* Dual Commit \*: ILM 규칙이 Dual Commit 옵션을 지정하는 경우 StorageGRID는 개체가 수집될 때(단일 커밋) 하나의 중간 복사본을 생성합니다.
- \* 균형 \*: ILM 규칙이 균형 옵션을 지정하는 경우 StorageGRID은 시스템에서 규칙에 지정된 모든 사본을 즉시 만들 수 없는 경우에만 단일 중간 복사본을 만듭니다. StorageGRID에서 동기 배치를 수행할 수 있는 경우 이 머리글은 영향을 주지 않습니다. 이 REDUCED\_REDUNDANCY 옵션은 오브젝트와 일치하는 ILM 규칙이 복제된 단일 복사본을 생성할 때 가장 적합합니다. 이 경우를 REDUCED\_REDUNDANCY 사용하면 모든 수집 작업에서 불필요한 추가 오브젝트 복사본을 생성하여 삭제할 필요가 없습니다.

다른 상황에서는 이 옵션을 사용하지 REDUCED\_REDUNDANCY 않는 것이 좋습니다. REDUCED\_REDUNDANCY 수집 중 오브젝트 데이터가 손실될 위험이 증가함 예를 들어, ILM 평가가 발생하기 전에 실패한 스토리지 노드에 단일 복사본이 처음 저장되는 경우 데이터가 손실될 수 있습니다.

 복제된 복사본이 항상 하나만 있으면 데이터가 영구적으로 손실될 위험이 있습니다. 복제된 객체 복제본이 하나만 있는 경우 스토리지 노드에 장애가 발생하거나 심각한 오류가 발생한 경우 해당 객체가 손실됩니다. 또한 업그레이드와 같은 유지보수 절차 중에는 개체에 대한 액세스가 일시적으로 중단됩니다.

‘REDUCED\_REDUNDANCY’ 지정하면 오브젝트를 처음 수집할 때 생성되는 복사본의 개수만 영향을 줍니다. 활성 ILM 정책에 따라 오브젝트를 평가할 때 생성되는 오브젝트 복사본 수에 영향을 미치지 않으며 StorageGRID 시스템에서 더 낮은 수준의 이중화로 데이터가 저장되지는 않습니다.

 S3 오브젝트 잠금이 설정된 상태에서 오브젝트를 버킷에 수집하면 REDUCED\_REDUNDANCY 옵션이 무시됩니다. 오브젝트를 레거시 준수 버킷에 수집하는 경우 REDUCED\_REDUNDANCY 옵션은 오류를 반환합니다. StorageGRID은 규정 준수 요구 사항이 충족될 수 있도록 항상 이중 커밋 수집을 수행합니다.

지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- Content-Type
- x-amz-checksum-algorithm

현재는 SHA256 값만 x-amz-checksum-algorithm 지원됩니다.

- x-amz-meta-, 그 뒤에 사용자 정의 메타데이터를 포함하는 이름-값 쌍이 옵니다

사용자 정의 메타데이터에 대한 이름-값 쌍을 지정할 때 다음 일반 형식을 사용합니다.

```
x-amz-meta-name: `value`
```

사용자 정의 생성 시간 \* 옵션을 ILM 규칙의 참조 시간으로 사용하려면 객체를 생성할 때 기록하는 메타데이터의 이름으로를 사용해야 creation-time 합니다. 예를 들면 다음과 같습니다.

```
x-amz-meta-creation-time: 1443399726
```

의 값은 `creation-time` 1970년 1월 1일 이후 초로 계산됩니다.



`'creation-time'` 기준 Compliance가 활성화된 버킷에 오브젝트를 추가하는 경우 사용자 정의 메타데이터로 추가할 수 없습니다. 오류가 반환됩니다.

- S3 오브젝트 잠금 요청 헤더:

- `x-amz-object-lock-mode`
- `x-amz-object-lock-retain-until-date`
- `x-amz-object-lock-legal-hold`

이러한 헤더 없이 요청이 이루어지면 버킷 기본 보존 설정을 사용하여 개체 버전 보존 기간을 계산합니다.

#### ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)

- SSE 요청 헤더:

- `x-amz-server-side-encryption`
- `x-amz-server-side-encryption-customer-key-MD5`
- `x-amz-server-side-encryption-customer-key`
- `x-amz-server-side-encryption-customer-algorithm`

#### [서버측 암호화에 대한 요청 헤더](#)



StorageGRID에서 UTF-8 문자를 처리하는 방법에 대한 자세한 내용은 [을 참조하십시오 "PutObject 를 선택합니다"](#).

#### 서버측 암호화에 대한 요청 헤더

다음 요청 헤더를 사용하여 서버측 암호화를 사용하여 다중 파트 개체를 암호화할 수 있습니다. SSE 및 SSE-C 옵션은 상호 배타적입니다.

- \* `sse` \*: StorageGRID에서 관리하는 고유 키로 개체를 암호화하려면 `CreateMultipartUpload` 요청에서 다음 헤더를 사용합니다. `UploadPart` 요청에는 이 헤더를 지정하지 마십시오.
  - `x-amz-server-side-encryption`
- \* `SSE-C` \*: 제공 및 관리하는 고유 키로 개체를 암호화하려면 `CreateMultipartUpload` 요청(및 이후의 각 `UploadPart` 요청)에 이 헤더 세 개를 모두 사용하십시오.
  - `x-amz-server-side-encryption-customer-algorithm`: AES256 지정합니다.
  - `x-amz-server-side-encryption-customer-key`: 새 개체에 대한 암호화 키를 지정합니다.
  - `x-amz-server-side-encryption-customer-key-MD5`: 새 개체의 암호화 키의 MD5 다이제스트를 지정합니다.



제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 예상 고려 사항을 ["서버 측 암호화 사용"](#) 검토하십시오.

지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- x-amz-website-redirect-location

‘x-amz-website-redirect-location’ 헤더가 를 ‘XNotImplemented’ 반환합니다.

## 버전 관리

멀티파트 업로드는 업로드 시작, 리스트 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. CompleteMultipartUpload 작업이 수행될 때 객체가 생성되고 해당되는 경우 버전이 지정됩니다.

**ListMultipartUploads** 를 참조하십시오

**ListMultipartUploads** 작업은 버킷에 대해 진행 중인 다중 파트 업로드를 나열합니다.

지원되는 요청 매개 변수는 다음과 같습니다.

- encoding-type
- key-marker
- max-uploads
- prefix
- upload-id-marker
- Host
- Date
- Authorization

## 버전 관리

멀티파트 업로드는 업로드 시작, 리스트 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. CompleteMultipartUpload 작업이 수행될 때 객체가 생성되고 해당되는 경우 버전이 지정됩니다.

## 업로드 파트

**UploadPart** 작업은 객체에 대한 다중 부분 업로드의 파트를 업로드합니다.

지원되는 요청 헤더입니다

지원되는 요청 헤더는 다음과 같습니다.

- x-amz-checksum-sha256
- Content-Length
- Content-MD5

## 서버측 암호화에 대한 요청 헤더

CreateMultipartUpload 요청에 대해 SSE-C 암호화를 지정한 경우 각 UploadPart 요청에 다음 요청 머리글도 포함해야 합니다.

- `x-amz-server-side-encryption-customer-algorithm`: AES256 지정합니다.
- `x-amz-server-side-encryption-customer-key`: CreateMultipartUpload 요청에서 제공한 것과 동일한 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: CreateMultipartUpload 요청에 제공한 것과 동일한 MD5 다이제스트를 지정합니다.



제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 개체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 ["서버측 암호화를 사용합니다"](#).

CreateMultipartUpload 요청 중에 SHA-256 체크섬을 지정한 경우 각 UploadPart 요청에 다음 요청 헤더도 포함해야 합니다.

- `x-amz-checksum-sha256`: 이 부품에 대한 SHA-256 체크섬을 지정합니다.

## 지원되지 않는 요청 헤더입니다

다음 요청 헤더는 지원되지 않습니다.

- `x-amz-sdk-checksum-algorithm`
- `x-amz-trailer`

## 버전 관리

멀티파트 업로드는 업로드 시작, 리스트업 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. CompleteMultipartUpload 작업이 수행될 때 객체가 생성되고 해당되는 경우 버전이 지정됩니다.

## 업로드파트 복사

UploadPartCopy 작업은 기존 개체의 데이터를 데이터 소스로 복사하여 개체의 일부를 업로드합니다.

UploadPartCopy 작업은 모든 Amazon S3 REST API 동작으로 구현됩니다. 예고 없이 변경될 수 있습니다.

이 요청은 StorageGRID 시스템 내에 지정된 오브젝트 데이터를 읽고 `x-amz-copy-source-range` 씁니다.

지원되는 요청 헤더는 다음과 같습니다.

- `x-amz-copy-source-if-match`
- `x-amz-copy-source-if-none-match`
- `x-amz-copy-source-if-unmodified-since`
- `x-amz-copy-source-if-modified-since`

## 서버측 암호화에 대한 요청 헤더

CreateMultipartUpload 요청에 대해 SSE-C 암호화를 지정한 경우 각 UploadPartCopy 요청에 다음 요청 머리글도 포함해야 합니다.

- `x-amz-server-side-encryption-customer-algorithm`: AES256 지정합니다.
- `x-amz-server-side-encryption-customer-key`: CreateMultipartUpload 요청에서 제공한 것과 동일한 암호화 키를 지정합니다.
- `x-amz-server-side-encryption-customer-key-MD5`: CreateMultipartUpload 요청에 제공한 것과 동일한 MD5 다이제스트를 지정합니다.

소스 객체가 SSE-C(고객 제공 키)를 사용하여 암호화되는 경우 UploadPartCopy 요청에 다음 세 개의 헤더를 포함해야 객체를 해독한 후 복사할 수 있습니다.

- `x-amz-copy-source-server-side-encryption-customer-algorithm`: AES256 지정합니다.
- `x-amz-copy-source-server-side-encryption-customer-key`: 원본 객체를 만들 때 제공한 암호화 키를 지정합니다.
- `x-amz-copy-source-server-side-encryption-customer-key-MD5`: 원본 객체를 만들 때 제공한 MD5 다이제스트를 지정합니다.

 제공한 암호화 키는 저장되지 않습니다. 암호화 키를 분실하면 해당 객체가 손실됩니다. 고객이 제공한 키를 사용하여 개체 데이터를 보호하기 전에 의 고려 사항을 검토하십시오 ["서버측 암호화를 사용합니다"](#).

## 버전 관리

멀티파트 업로드는 업로드 시작, 리스트 업로드, 파트 업로드, 업로드된 파트 조립 및 업로드 완료를 위한 별도의 작업으로 구성됩니다. CompleteMultipartUpload 작업이 수행될 때 객체가 생성되고 해당되는 경우 버전이 지정됩니다.

## 오류 응답

StorageGRID 시스템은 적용되는 모든 표준 S3 REST API 오류 응답을 지원합니다. 또한 StorageGRID 구현에는 여러 개의 사용자 지정 응답이 추가됩니다.

지원되는 **S3 API** 오류 코드입니다

이름	HTTP 상태입니다
액세스가 거부되었습니다	403 사용 금지
배다이제스트	400 잘못된 요청
BucketAlreadyExists를 참조하십시오	409 충돌
BucketNotEmpty	409 충돌
IncompleteBody	400 잘못된 요청

이름	HTTP 상태입니다
내부 오류입니다	500 내부 서버 오류입니다
InvalidAccessKeyId 입니다	403 사용 금지
InvalidArgument 를 선택합니다	400 잘못된 요청
InvalidBuckName입니다	400 잘못된 요청
InvalidBuckketState입니다	409 충돌
InvalidDigest 를 선택합니다	400 잘못된 요청
InvalidEncryptionAlgorithmError 가 발생합니다	400 잘못된 요청
InvalidPart 를 선택합니다	400 잘못된 요청
InvalidPartOrder를 선택합니다	400 잘못된 요청
InvalidRange 를 선택합니다	416 요청된 범위가 충분하지 않습니다
InvalidRequest 입니다	400 잘못된 요청
InvalidStorageClass 의 값을 반환합니다	400 잘못된 요청
InvalidTag 를 선택합니다	400 잘못된 요청
InvalidURI입니다	400 잘못된 요청
키투롱	400 잘못된 요청
MalformedXML을 참조하십시오	400 잘못된 요청
MetadataTooLarge를 참조하십시오	400 잘못된 요청
MethodNotAllowed 를 참조하십시오	405 메서드를 사용할 수 없습니다
MissingContentLength를 참조하십시오	411 길이 필요
MissingRequestBodyError가 발생합니다	400 잘못된 요청
MissingSecurityHeader 를 참조하십시오	400 잘못된 요청

이름	HTTP 상태입니다
NoSuchBucket	404를 찾을 수 없습니다
NoSuchKey를 클릭합니다	404를 찾을 수 없습니다
NoSuchUpload 를 클릭합니다	404를 찾을 수 없습니다
구현되지 않았습니다	501 구현되지 않음
NoSuchBucketPolicy를 참조하십시오	404를 찾을 수 없습니다
ObjectLockConfigurationNotFoundError 가 발생합니다	404를 찾을 수 없습니다
사전 조건에 실패했습니다	412 전제 조건 실패
RequestTimeTooSkewed 를 참조하십시오	403 사용 금지
서비스를 사용할 수 없습니다	503 서비스를 사용할 수 없습니다
SignatureDoesNotMatch 를 참조하십시오	403 사용 금지
투만이버킷	400 잘못된 요청
UserKeyMustBeSpecified 를 선택합니다	400 잘못된 요청

#### StorageGRID 사용자 지정 오류 코드

이름	설명	HTTP 상태입니다
XBucketLifecycleNotAllowed를 참조하십시오	버킷 수명 주기 구성은 레거시 준수 버킷에서 허용되지 않습니다	400 잘못된 요청
XBucketPolicyParseException 을 참조하십시오	수신된 버킷 정책 JSON을 구문 분석하지 못했습니다.	400 잘못된 요청
XComplianceConflict	레거시 준수 설정으로 인해 작업이 거부되었습니다.	403 사용 금지
XComplianceRedundancyForbidden을 선택합니다	레거시 준수 버킷에서는 감소된 중복성이 허용되지 않습니다	400 잘못된 요청
XMaxBucketPolicyLengthExceeded 를 참조하십시오	정책이 허용되는 최대 버킷 정책 길이를 초과합니다.	400 잘못된 요청

이름	설명	HTTP 상태입니다
XMissingInternalRequestHeader를 참조하십시오	내부 요청의 헤더가 누락되었습니다.	400 잘못된 요청
XNoSuchBucketCompliance	지정된 버킷에 레거시 준법 기능이 설정되어 있지 않습니다.	404를 찾을 수 없습니다
XNotAcceptable(X 허용 가능)	요청에 총족되지 않은 하나 이상의 수락 헤더가 있습니다.	406 허용되지 않습니다
XNotImplemented(XNotImplemented)	제공한 요청은 구현되지 않은 기능을 의미합니다.	501 구현되지 않음

## StorageGRID 사용자 정의 작업

### StorageGRID 사용자 정의 작업

StorageGRID 시스템은 S3 REST API에 추가된 사용자 지정 작업을 지원합니다.

다음 표에서는 StorageGRID에서 지원하는 사용자 지정 작업을 보여 줍니다.

작동	설명
"버킷 일관성 확보"	특정 버킷에 적용되는 일관성을 반환합니다.
"버킷 일관성을 유지합니다"	특정 버킷에 적용되는 일관성을 설정합니다.
"버킷 최종 액세스 시간 가져오기"	특정 버킷에 대해 마지막 액세스 시간 업데이트를 사용할 수 있는지 여부를 반환합니다.
"버킷 최종 접근 시간"	특정 버킷에 대한 마지막 액세스 시간 업데이트를 활성화하거나 비활성화할 수 있습니다.
"버킷 메타데이터 알림 구성을 삭제합니다"	특정 버킷과 연결된 메타데이터 알림 구성 XML을 삭제합니다.
"Bucket 메타데이터 알림 구성을 가져오기"	특정 버킷과 연결된 메타데이터 알림 구성 XML을 반환합니다.
"Put Bucket 메타데이터 알림 구성"	버킷에 대한 메타데이터 알림 서비스를 구성합니다.
"스토리지 사용량을 가져옵니다"	계정과 연결된 각 버킷에서 사용 중인 총 저장소 양을 나타냅니다.
"사용되지 않음: 규정 준수 설정이 있는 CreateBucket"	더 이상 사용되지 않으며 지원되지 않음: Compliance를 사용하는 새 버킷을 더 이상 생성할 수 없습니다.

작동	설명
"사용되지 않음: 버킷 준수 가져오기"	더 이상 사용되지 않지만 지원됨: 기존 레거시 준수 버킷에 대해 현재 적용되는 규정 준수 설정을 반환합니다.
"사용되지 않음: 버킷 준수"	사용되지 않지만 지원됨: 기존 레거시 준수 버킷의 준수 설정을 수정할 수 있습니다.

## 버킷 일관성 확보

Get Bucket Consistency 요청을 사용하면 특정 버킷에 적용되는 일관성을 확인할 수 있습니다.

기본 정합성 보장은 새로 생성된 개체에 대해 쓰기 후 읽기를 보장하도록 설정됩니다.

이 작업을 완료하려면 S3:GetBucketConsistency 권한이 있거나 계정 루트여야 합니다.

### 요청 예

```
GET /bucket?x-ntap-sg-consistency HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

### 응답

응답 XML에서 은 <Consistency> 다음 값 중 하나를 반환합니다.

정합성	설명
모두	모든 노드가 데이터를 즉시 수신하거나 요청이 실패합니다.
강함 - 글로벌	모든 사이트에서 모든 클라이언트 요청에 대해 쓰기 후 읽기 정합성을 보장합니다.
강력한 사이트	사이트 내의 모든 클라이언트 요청에 대해 쓰기 후 읽기 일관성을 보장합니다.
읽기-후-새로-쓰기	(기본값) 새 객체에 대한 읽기 후 쓰기 정합성을 보장하고 객체 업데이트에 대한 최종 일관성을 제공합니다. 고가용성 및 데이터 보호 보장 대부분의 경우에 권장됩니다.
사용 가능	새 개체 및 개체 업데이트 모두에 대한 최종 일관성을 제공합니다. S3 버킷의 경우 필요한 경우에만 사용하십시오(예: 거의 읽지 않는 로그 값이 포함된 버킷의 경우 또는 존재하지 않는 키의 헤드 또는 GET 작업의 경우). S3 FabricPool 버킷은 지원되지 않습니다.

## 응답 예

```
HTTP/1.1 200 OK
Date: Fri, 18 Sep 2020 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/11.5.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<Consistency xmlns="http://s3.storagegrid.com/doc/2015-02-01/">read-after-new-write</Consistency>
```

## 관련 정보

### ["정합성"](#)

버킷 일관성을 유지합니다

Put Bucket Consistency 요청을 사용하면 버킷에서 수행된 작업에 적용할 일관성을 지정할 수 있습니다.

기본 정합성 보장은 새로 생성된 개체에 대해 쓰기 후 읽기를 보장하도록 설정됩니다.

시작하기 전에

이 작업을 완료하려면 S3:PutBucketConsistency 권한이 있거나 계정 루트여야 합니다.

요청하십시오

매개 변수에는 x-ntap-sg-consistency 다음 값 중 하나가 포함되어야 합니다.

정합성	설명
모두	모든 노드가 데이터를 즉시 수신하거나 요청이 실패합니다.
강함 - 글로벌	모든 사이트에서 모든 클라이언트 요청에 대해 쓰기 후 읽기 정합성을 보장합니다.
강력한 사이트	사이트 내의 모든 클라이언트 요청에 대해 쓰기 후 읽기 일관성을 보장합니다.
읽기-후-새로-쓰기	(기본값) 새 개체에 대한 읽기 후 쓰기 정합성을 보장하고 개체 업데이트에 대한 최종 일관성을 제공합니다. 고가용성 및 데이터 보호 보장 제공 대부분의 경우에 권장됩니다.

정합성	설명
사용 가능	새 개체 및 개체 업데이트 모두에 대한 최종 일관성을 제공합니다. S3 버킷의 경우 필요한 경우에만 사용하십시오(예: 거의 읽지 않는 로그 값이 포함된 버킷의 경우 또는 존재하지 않는 키의 헤드 또는 GET 작업의 경우). S3 FabricPool 버킷은 지원되지 않습니다.

- 참고: \* 일반적으로 "Read-after-new-write" 일관성을 사용해야 합니다. 요청이 올바르게 작동하지 않는 경우 가능한 경우 응용 프로그램 클라이언트 동작을 변경합니다. 또는 각 API 요청의 일관성을 지정하도록 클라이언트를 구성합니다. 버킷 수준의 일관성을 마지막 수단으로 설정합니다.

요청 예

```
PUT /bucket?x-ntap-sg-consistency=strong-global HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

관련 정보

["정합성"](#)

버킷 최종 액세스 시간 가져오기

[버킷 최종 액세스 시간 가져오기(Get Bucket Last Access Time) 요청을 사용하면 개별 버킷에 대해 마지막 액세스 시간 업데이트가 활성화되었거나 비활성화되었는지 확인할 수 있습니다.

이 작업을 완료하려면 S3:GetBucketLastAccessTime 권한이 있거나 계정 루트여야 합니다.

요청 예

```
GET /bucket?x-ntap-sg-lastaccesstime HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

응답 예

이 예에서는 버킷에 대해 마지막 액세스 시간 업데이트가 활성화되어 있음을 보여 줍니다.

```

HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 01:02:18 GMT
Connection: CLOSE
Server: StorageGRID/10.3.0
x-amz-request-id: 12345
Content-Length: 127
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<LastAccessTime xmlns="http://s3.storagegrid.com/doc/2015-02-01/">enabled
</LastAccessTime>

```

## 버킷 최종 접근 시간

Put Bucket Last Access Time 요청을 사용하면 개별 버킷에 대한 마지막 액세스 시간 업데이트를 활성화하거나 비활성화할 수 있습니다. 마지막 액세스 시간 업데이트를 비활성화하면 성능이 향상되고 버전 10.3.0 이상으로 생성된 모든 버킷의 기본 설정이 됩니다.

이 작업을 완료하려면 버킷에 대한 S3:PutBucketLastAccessTime 권한이 있거나 계정 루트여야 합니다.

StorageGRID 버전 10.3부터는 모든 새 버킷에 대해 마지막 액세스 시간에 대한 업데이트가 기본적으로 비활성화됩니다. 이전 버전의 StorageGRID를 사용하여 만든 버킷이 있고 새 기본 동작과 일치시키려면 이전의 각 버킷에 대해 마지막 액세스 시간 업데이트를 명시적으로 비활성화해야 합니다. Put Bucket 마지막 액세스 시간 요청을 사용하거나 Tenant Manager의 버킷에 대한 세부 정보 페이지에서 마지막 액세스 시간에 대한 업데이트를 활성화 또는 비활성화할 수 있습니다. ["마지막 액세스 시간 업데이트를 사용하거나 사용하지 않도록 설정합니다"](#) 참조하십시오.

버킷에 대해 마지막 액세스 시간 업데이트가 비활성화된 경우 버킷의 작업에 다음 동작이 적용됩니다.

- GetObject, GetObjectAcl, GetObjectTagging 및 HeadObject 요청은 마지막 액세스 시간을 업데이트하지 않습니다. ILM(정보 수명 주기 관리) 평가를 위해 객체가 대기열에 추가되지 않습니다.
- 메타데이터만 업데이트하는 CopyObject 및 PutObjectTagging 요청도 마지막 액세스 시간을 업데이트합니다. ILM 평가를 위해 오브젝트가 대기열에 추가됩니다.
- 소스 버킷에 대해 마지막 액세스 시간에 대한 업데이트를 사용할 수 없는 경우 CopyObject 요청이 소스 버킷의 마지막 액세스 시간을 업데이트하지 않습니다. 복사된 객체는 소스 버킷에 대한 ILM 평가를 위해 대기열에 추가되지 않습니다. 그러나 대상의 경우 CopyObject 요청은 항상 마지막 액세스 시간을 업데이트합니다. ILM 평가를 위해 개체의 복사본이 대기열에 추가됩니다.
- CompleteMultipartUpload 요청이 마지막 액세스 시간을 업데이트합니다. 완료된 객체가 ILM 평가를 위해 대기열에 추가됩니다.

예를 요청하십시오

이 예제에서는 버킷의 마지막 액세스 시간을 설정합니다.

```
PUT /bucket?x-ntap-sg-lastaccesstime=enabled HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

이 예제에서는 버킷의 마지막 액세스 시간을 비활성화합니다.

```
PUT /bucket?x-ntap-sg-lastaccesstime=disabled HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

버킷 메타데이터 알림 구성을 삭제합니다

Delete Bucket 메타데이터 알림 구성 요청을 사용하면 구성 XML을 삭제하여 개별 버킷에 대한 검색 통합 서비스를 비활성화할 수 있습니다.

이 작업을 완료하려면 버킷에 대한 S3:DeleteBucketMetadataNotification 권한이 있거나 계정 루트여야 합니다.

요청 예

이 예제에서는 버킷에 대한 검색 통합 서비스를 비활성화하는 방법을 보여 줍니다.

```
DELETE /test1?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

**Bucket** 메타데이터 알림 구성 가져오기

Get Bucket 메타데이터 알림 구성 요청을 사용하면 개별 버킷에 대한 검색 통합을 구성하는 데 사용되는 구성 XML을 검색할 수 있습니다.

이 작업을 완료하려면 S3:GetBucketMetadataNotification 권한이 있거나 계정 루트여야 합니다.

요청 예

이 요청은 이름이 인 버킷에 대한 메타데이터 알림 구성을 bucket 검색합니다.

```
GET /bucket?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

## 응답

응답 본문에는 버킷에 대한 메타데이터 알림 구성이 포함됩니다. 메타데이터 알림 구성은 버킷이 검색 통합을 위해 구성되는 방식을 결정할 수 있습니다. 즉, 인덱싱된 개체와 해당 개체 메타데이터가 전송되는 끝점을 확인할 수 있습니다.

```
<MetadataNotificationConfiguration>
  <Rule>
    <ID>Rule-1</ID>
    <Status>rule-status</Status>
    <Prefix>key-prefix</Prefix>
    <Destination>
      <Urn>arn:aws:es:_region:account-
ID_:domain/_mydomain/myindex/mytype_</Urn>
    </Destination>
  </Rule>
  <Rule>
    <ID>Rule-2</ID>
    ...
  </Rule>
  ...
</MetadataNotificationConfiguration>
```

각 메타데이터 알림 구성에는 하나 이상의 규칙이 포함됩니다. 각 규칙은 적용되는 오브젝트와 StorageGRID가 오브젝트 메타데이터를 보내야 하는 대상을 지정합니다. 대상은 StorageGRID 끝점의 URN을 사용하여 지정해야 합니다.

이름	설명	필수 요소입니다
MetadataNotificationConfiguration을 참조하십시오	메타데이터 알림의 개체 및 대상을 지정하는 데 사용되는 규칙의 컨테이너 태그입니다.  하나 이상의 규칙 요소가 포함되어 있습니다.	예
규칙	메타데이터를 지정된 인덱스에 추가해야 하는 개체를 식별하는 규칙의 컨테이너 태그입니다.  접두사가 겹치는 규칙은 거부됩니다.  MetadataNotificationConfiguration 요소에 포함되어 있습니다.	예
ID입니다	규칙의 고유 식별자입니다.  Rule 요소에 포함되어 있습니다.	아니요

이름	설명	필수 요소입니다
상태	<p>상태는 '활성화' 또는 '비활성화'가 될 수 있습니다. 비활성화된 규칙에 대해 어떠한 작업도 수행되지 않습니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
접두어	<p>접두사와 일치하는 개체는 규칙의 영향을 받으며 해당 메타데이터는 지정된 대상으로 전송됩니다.</p> <p>모든 오브젝트를 일치시키려면 빈 접두사를 지정합니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
목적지	<p>규칙의 대상에 대한 컨테이너 태그입니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
urn	<p>객체 메타데이터가 전송되는 대상의 urn입니다. 다음 속성을 가진 StorageGRID 끝점의 URN이어야 합니다.</p> <ul style="list-style-type: none"> <li>• es 세 번째 요소여야 합니다.</li> <li>• URN은 메타데이터가 저장되는 인덱스 및 형식으로 끝나야 domain-name/myindex/mytype 합니다.</li> </ul> <p>엔드포인트는 테넌트 관리자 또는 테넌트 관리 API를 사용하여 구성됩니다. 다음과 같은 형식을 취합니다.</p> <ul style="list-style-type: none"> <li>• arn:aws:es:_region:account-ID_:domain/mydomain/myindex/mytype</li> <li>• urn:mysite:es:::mydomain/myindex/mytype</li> </ul> <p>구성 XML을 제출하기 전에 끝점을 구성해야 합니다. 그렇지 않으면 404 오류로 인해 구성이 실패합니다.</p> <p>urn은 Destination 요소에 포함됩니다.</p>	예

응답 예

태그 사이에 포함된 XML은

<MetadataNotificationConfiguration></MetadataNotificationConfiguration>`검색 통합 끝점과의 통합이 버킷에 대해 구성되는 방법을 보여 줍니다. 이 예제에서 개체 메타데이터는 이름이 인 AWS 도메인에서 호스팅되는 `records 이름이 2017 인 Elasticsearch 인덱스로 보내집니다. current

```

HTTP/1.1 200 OK
Date: Thu, 20 Jul 2017 18:24:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/11.0.0
x-amz-request-id: 3832973499
Content-Length: 264
Content-Type: application/xml

<MetadataNotificationConfiguration>
  <Rule>
    <ID>Rule-1</ID>
    <Status>Enabled</Status>
    <Prefix>2017</Prefix>
    <Destination>
      <Urn>arn:aws:es:us-east-
1:3333333:domain/records/current/2017</Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>

```

#### 관련 정보

["테넌트 계정을 사용합니다"](#)

#### Put Bucket 메타데이터 알림 구성

Put Bucket 메타데이터 알림 구성 요청을 사용하면 개별 버킷에 대한 검색 통합 서비스를 활성화할 수 있습니다. 요청 본문에 제공하는 메타데이터 알림 구성 XML은 대상 검색 인덱스에 메타데이터가 전송되는 개체를 지정합니다.

이 작업을 완료하려면 버킷에 대한 S3:PutBucketMetadataNotification 권한이 있거나 계정 루트여야 합니다.

#### 요청하십시오

요청 본문에는 메타데이터 알림 구성이 포함되어야 합니다. 각 메타데이터 알림 구성에는 하나 이상의 규칙이 포함됩니다. 각 규칙은 적용되는 오브젝트와 StorageGRID가 오브젝트 메타데이터를 보내야 하는 대상을 지정합니다.

개체 이름의 접두어를 기준으로 개체를 필터링할 수 있습니다. 예를 들어, 접두사가 있는 개체의 메타데이터를 한 대상으로 보내고, 접두사가 있는 개체는 /videos 다른 대상으로 보낼 수 /images 있습니다.

중복되는 접두사가 있는 구성은 유효하지 않으며 제출될 때 거부됩니다. 예를 들어, 접두사가 있는 객체에 대한 하나의 규칙과 접두사가 있는 객체에 대한 두 번째 규칙이 test2 포함된 구성은 test 허용되지 않습니다.

대상은 StorageGRID 끝점의 URN을 사용하여 지정해야 합니다. 메타데이터 알림 구성이 제출될 때 끝점이 존재해야 합니다. 그렇지 않으면 요청이 400 Bad Request 실패합니다. 오류 메시지는 다음과 같습니다. *Unable to save the metadata notification (search) policy. The specified endpoint URN does not exist: URN.*

```

<MetadataNotificationConfiguration>
  <Rule>
    <ID>Rule-1</ID>
    <Status>rule-status</Status>
    <Prefix>key-prefix</Prefix>
    <Destination>
      <Urn>arn:aws:es:region:account-
ID:domain/mydomain/myindex/mytype</Urn>
    </Destination>
  </Rule>
  <Rule>
    <ID>Rule-2</ID>
    ...
  </Rule>
  ...
</MetadataNotificationConfiguration>

```

이 표에서는 메타데이터 알림 구성 XML의 요소에 대해 설명합니다.

이름	설명	필수 요소입니다
MetadataNotificationConfiguration을 참조하십시오	<p>메타데이터 알림의 개체 및 대상을 지정하는 데 사용되는 규칙의 컨테이너 태그입니다.</p> <p>하나 이상의 규칙 요소가 포함되어 있습니다.</p>	예
규칙	<p>메타데이터를 지정된 인덱스에 추가해야 하는 개체를 식별하는 규칙의 컨테이너 태그입니다.</p> <p>접두사가 겹치는 규칙은 거부됩니다.</p> <p>MetadataNotificationConfiguration 요소에 포함되어 있습니다.</p>	예
ID입니다	<p>규칙의 고유 식별자입니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	아니요
상태	<p>상태는 '활성화' 또는 '비활성화'가 될 수 있습니다.</p> <p>비활성화된 규칙에 대해 어떠한 작업도 수행되지 않습니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예

이름	설명	필수 요소입니다
접두어	<p>접두사와 일치하는 개체는 규칙의 영향을 받으며 해당 메타데이터는 지정된 대상으로 전송됩니다.</p> <p>모든 오브젝트를 일치시키려면 빈 접두사를 지정합니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
목적지	<p>규칙의 대상에 대한 컨테이너 태그입니다.</p> <p>Rule 요소에 포함되어 있습니다.</p>	예
urn	<p>객체 메타데이터가 전송되는 대상의 urn입니다. 다음 속성을 가진 StorageGRID 끝점의 URN이어야 합니다.</p> <ul style="list-style-type: none"> <li>• es 세 번째 요소여야 합니다.</li> <li>• URN은 메타데이터가 저장되는 인덱스 및 형식으로 끝나야 domain-name/myindex/mytype 합니다.</li> </ul> <p>엔드포인트는 테넌트 관리자 또는 테넌트 관리 API를 사용하여 구성됩니다. 다음과 같은 형식을 취합니다.</p> <ul style="list-style-type: none"> <li>• arn:aws:es:region- ID:domain/mydomain/myindex/mytype</li> <li>• urn:mysite:es:::mydomain/myindex/mytype</li> </ul> <p>구성 XML을 제출하기 전에 끝점을 구성해야 합니다. 그렇지 않으면 404 오류로 인해 구성이 실패합니다.</p> <p>urn은 Destination 요소에 포함됩니다.</p>	예

예를 요청하십시오

이 예제에서는 버킷에 대한 검색 통합을 활성화하는 방법을 보여 줍니다. 이 예제에서 모든 오브젝트의 오브젝트 메타데이터는 동일한 대상으로 전송됩니다.

```

PUT /test1?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host

<MetadataNotificationConfiguration>
  <Rule>
    <ID>Rule-1</ID>
    <Status>Enabled</Status>
    <Prefix></Prefix>
    <Destination>
      <Urn>urn:sgws:es:::sgws-notifications/test1/all</Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>

```

이 예에서는 접두사와 일치하는 개체의 개체 메타데이터가 /images 한 대상으로 전송되고, 접두사와 일치하는 개체의 개체 메타데이터가 /videos 두 번째 대상으로 전송됩니다.

```

PUT /graphics?x-ntap-sg-metadata-notification HTTP/1.1
Date: date
Authorization: authorization string
Host: host

<MetadataNotificationConfiguration>
  <Rule>
    <ID>Images-rule</ID>
    <Status>Enabled</Status>
    <Prefix>/images</Prefix>
    <Destination>
      <Urn>arn:aws:es:us-east-1:3333333:domain/es-
domain/graphics/imagetype</Urn>
    </Destination>
  </Rule>
  <Rule>
    <ID>Videos-rule</ID>
    <Status>Enabled</Status>
    <Prefix>/videos</Prefix>
    <Destination>
      <Urn>arn:aws:es:us-west-1:2222222:domain/es-
domain/graphics/videotype</Urn>
    </Destination>
  </Rule>
</MetadataNotificationConfiguration>

```

**JSON**이 검색 통합 서비스에 의해 생성되었습니다

버킷에 대한 검색 통합 서비스를 활성화하면 개체 메타데이터 또는 태그를 추가, 업데이트 또는 삭제할 때마다 JSON 문서가 생성되어 대상 끝점으로 전송됩니다.

이 예제는 키가 있는 개체가 라는 버킷에 `test` 생성될 때 생성될 수 있는 JSON의 예를 보여줍니다  
SGWS/Tagging.txt.test`버킷이 버전이 아니므로 `versionId 태그가 비어 있습니다.

```
{  
  "bucket": "test",  
  "key": "SGWS/Tagging.txt",  
  "versionId": "",  
  "accountId": "86928401983529626822",  
  "size": 38,  
  "md5": "3d6c7634a85436eee06d43415012855",  
  "region": "us-east-1",  
  "metadata": {  
    "age": "25"  
  },  
  "tags": {  
    "color": "yellow"  
  }  
}
```

메타데이터 알림에 포함된 개체 메타데이터입니다

이 표에는 검색 통합이 활성화된 경우 대상 끝점으로 전송되는 JSON 문서에 포함된 모든 필드가 나열됩니다.

문서 이름에는 버킷 이름, 오브젝트 이름 및 버전 ID(있는 경우)가 포함됩니다.

유형	항목 이름	설명
버킷 및 오브젝트 정보	버킷	버킷의 이름입니다
버킷 및 오브젝트 정보	키	개체 키 이름입니다
버킷 및 오브젝트 정보	버전 ID	오브젝트 버전, 버전 버킷 내 오브젝트
버킷 및 오브젝트 정보	지역	버킷 영역(예: us-east-1)
시스템 메타데이터	크기	HTTP 클라이언트에 표시되는 개체 크기(바이트)입니다
시스템 메타데이터	MD5(MD5)	개체 해시

유형	항목 이름	설명
사용자 메타데이터	메타데이터 <i>key:value</i>	객체에 대한 모든 사용자 메타데이터를 키 값 쌍으로 사용합니다
태그	태그 <i>key:value</i>	개체에 대해 정의된 모든 개체 태그를 키 값 쌍으로 사용합니다

 태그 및 사용자 메타데이터의 경우 StorageGRID는 날짜 및 숫자를 Elasticsearch에 문자열 또는 S3 이벤트 알림으로 전달합니다. 이러한 문자열을 날짜 또는 숫자로 해석하도록 Elasticsearch를 구성하려면 동적 필드 매핑 및 날짜 형식 매핑에 대한 Elasticsearch 지침을 따르십시오. 검색 통합 서비스를 구성하기 전에 인덱스에서 동적 필드 매핑을 활성화해야 합니다. 문서가 인덱싱된 후에는 인덱스에서 문서의 필드 형식을 편집할 수 없습니다.

#### 관련 정보

["테넌트 계정을 사용합니다"](#)

#### 스토리지 사용 요청 가져오기

Get Storage Usage 요청은 계정이 사용하는 총 스토리지 양과 계정과 연결된 각 버킷에 대해 알려줍니다.

계정과 해당 버킷에서 사용하는 스토리지의 양은 쿼리 매개 변수를 사용하여 수정된 ListBucket 요청에서 가져올 수 *x-ntap-sg-usage* 있습니다. 시스템에서 처리하는 PUT 및 삭제 요청과는 별도로 버킷 스토리지 사용량을 추적합니다. 특히 시스템이 과부하 상태인 경우, 사용 값이 요청 처리를 기준으로 예상 값과 일치하기 전에 약간의 지연이 있을 수 있습니다.

기본적으로 StorageGRID는 강력한 글로벌 일관성을 사용하여 사용 정보 검색을 시도합니다. 강력한 글로벌 일관성을 달성할 수 없는 경우 StorageGRID는 강력한 사이트 일관성으로 사용 정보를 검색합니다.

이 작업을 완료하려면 S3:ListAllMyBucket 권한이 있거나 계정 루트여야 합니다.

#### 요청 예

```
GET /?x-ntap-sg-usage HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

#### 응답 예

이 예에서는 두 버킷에 4개의 오브젝트와 12바이트의 데이터가 있는 계정을 보여 줍니다. 각 버킷에는 2개의 오브젝트와 6바이트의 데이터가 포함되어 있습니다.

```
HTTP/1.1 200 OK
Date: Sat, 29 Nov 2015 00:49:05 GMT
Connection: KEEP-ALIVE
Server: StorageGRID/10.2.0
x-amz-request-id: 727237123
Content-Length: 427
Content-Type: application/xml

<?xml version="1.0" encoding="UTF-8"?>
<UsageResult xmlns="http://s3.storagegrid.com/doc/2015-02-01">
<CalculationTime>2014-11-19T05:30:11.000000Z</CalculationTime>
<ObjectCount>4</ObjectCount>
<DataBytes>12</DataBytes>
<Buckets>
<Bucket>
<Name>bucket1</Name>
<ObjectCount>2</ObjectCount>
<DataBytes>6</DataBytes>
</Bucket>
<Bucket>
<Name>bucket2</Name>
<ObjectCount>2</ObjectCount>
<DataBytes>6</DataBytes>
</Bucket>
</Buckets>
</UsageResult>
```

## 버전 관리

저장된 모든 개체 버전은 응답의 및 DataBytes 값에 영향을 ObjectCount 줍니다. 표식 삭제는 합계에 추가되지 ObjectCount 않습니다.

## 관련 정보

### ["정합성"](#)

레거시 규정 준수를 위한 더 이상 사용되지 않는 버킷 요청

레거시 규정 준수를 위한 더 이상 사용되지 않는 버킷 요청

StorageGRID S3 REST API를 사용하여 레거시 규정 준수 기능을 사용하여 생성된 버킷을 관리해야 할 수 있습니다.

규정 준수 기능이 사용되지 않습니다

이전 StorageGRID 버전에서 사용할 수 있었던 StorageGRID 규정 준수 기능은 더 이상 사용되지 않으며 S3 오브젝트 잠금으로 대체되었습니다.

이전에 글로벌 규정 준수 설정을 활성화한 경우 StorageGRID 11.6에서 전역 S3 개체 잠금 설정이 활성화됩니다. Compliance를 사용하도록 설정한 상태에서 새 버킷을 더 이상 생성할 수 없지만, 필요에 따라 StorageGRID S3 REST API를 사용하여 기존의 규정을 준수하는 버킷을 관리할 수 있습니다.

- ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)
- ["ILM을 사용하여 개체를 관리합니다"](#)
- ["NetApp 기술 자료: StorageGRID 11.5에서 레거시 준수 버킷을 관리하는 방법"](#)

더 이상 사용되지 않는 규정 준수 요청:

- ["폐기됨 - 규정 준수를 위해 버킷 요청을 수정합니다"](#)

SGCompliance XML 요소는 사용되지 않습니다. 이전 버전에서는 이 StorageGRID 사용자 정의 요소를 PUT 버킷 요청의 선택적 XML 요청 본문에 포함하여 준수 버킷을 생성할 수 있었습니다.

- ["사용되지 않음 - 버킷 규정 준수"](#)

Get Bucket 준수 요청은 더 이상 사용되지 않습니다. 그러나 이 요청을 계속 사용하여 기존 레거시 준수 버킷에 현재 적용되는 준수 설정을 확인할 수 있습니다.

- ["사용되지 않음 - Put 버킷 준수"](#)

Put Bucket 준수 요청은 더 이상 사용되지 않습니다. 그러나 이 요청을 계속 사용하여 기존 레거시 준수 버킷의 준수 설정을 수정할 수 있습니다. 예를 들어, 기존 버킷을 법적 보존 상태로 놓거나 보존 기간을 늘릴 수 있습니다.

사용되지 않음: 규정 준수를 위한 **CreateBucket** 요청 수정

SGCompliance XML 요소는 사용되지 않습니다. 이전에는 이 StorageGRID 사용자 지정 요소를 CreateBucket 요청의 선택적 XML 요청 본문에 포함시켜 준수 버킷을 만들 수 있었습니다.

이전 StorageGRID 버전에서 사용할 수 있었던 StorageGRID 규정 준수 기능은 더 이상 사용되지 않으며 S3 오브젝트 잠금으로 대체되었습니다. 자세한 내용은 다음을 참조하십시오.



- ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)
- ["NetApp 기술 자료: StorageGRID 11.5에서 레거시 준수 버킷을 관리하는 방법"](#)

Compliance가 설정된 새 버킷을 더 이상 생성할 수 없습니다. 규정 준수를 위해 CreateBucket 요청 수정을 사용하여 새 준수 버킷을 생성하려고 하면 다음 오류 메시지가 반환됩니다.

The Compliance feature is deprecated.

Contact your StorageGRID administrator if you need to create new Compliant buckets.

사용되지 않음: 버킷 준수 요청 가져오기

Get Bucket 준수 요청은 더 이상 사용되지 않습니다. 그러나 이 요청을 계속 사용하여 기존

## 레거시 준수 버킷에 현재 적용되는 준수 설정을 확인할 수 있습니다.

이전 StorageGRID 버전에서 사용할 수 있었던 StorageGRID 규정 준수 기능은 더 이상 사용되지 않으며 S3 오브젝트 잠금으로 대체되었습니다. 자세한 내용은 다음을 참조하십시오.



- ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)
- ["NetApp 기술 자료: StorageGRID 11.5에서 레거시 준수 버킷을 관리하는 방법"](#)

이 작업을 완료하려면 S3:GetBucketCompliance 권한이 있거나 계정 루트여야 합니다.

### 요청 예

이 예제 요청을 사용하면 이름이 인 버킷에 대한 준수 설정을 결정할 `mybucket` 수 있습니다.

```
GET /mybucket/?x-ntap-sg-compliance HTTP/1.1
Date: date
Authorization: authorization string
Host: host
```

### 응답 예

응답 XML에 <SGCompliance> 버킷에 대해 적용되는 준수 설정이 나열됩니다. 이 예제 응답에서는 오브젝트를 그리드에 인제스트하는 시점을 시작으로 각 오브젝트를 1년(525,600분)동안 보존할 버킷의 규정 준수 설정을 보여 줍니다. 현재 이 버킷에 대한 법적 보류가 없습니다. 각 개체는 1년 후에 자동으로 삭제됩니다.

```
HTTP/1.1 200 OK
Date: date
Connection: connection
Server: StorageGRID/11.1.0
x-amz-request-id: request ID
Content-Length: length
Content-Type: application/xml

<SGCompliance>
  <RetentionPeriodMinutes>525600</RetentionPeriodMinutes>
  <LegalHold>false</LegalHold>
  <AutoDelete>true</AutoDelete>
</SGCompliance>
```

이름	설명
RetentionPeriodMinutes(주기적 지연 시간)	이 버킷에 추가된 객체의 보존 기간(분)입니다. 보존 기간은 객체가 그리드에 수집될 때 시작됩니다.

이름	설명
LegalHold	<ul style="list-style-type: none"> <li>참: 이 버킷은 현재 법적 증거 자료 보관 중입니다. 이 버킷의 오브젝트는 보존 기간이 만료된 경우에도 법적 보류가 해제될 때까지 삭제할 수 없습니다.</li> <li>거짓: 이 버킷은 현재 법적 증거 자료 보관 중이 아닙니다. 이 버킷의 오브젝트는 보존 기간이 만료되면 삭제할 수 있습니다.</li> </ul>
자동 삭제	<ul style="list-style-type: none"> <li>참: 버킷이 법적 보존 상태에 있지 않는 한, 보존 기간이 만료되면 이 버킷의 객체가 자동으로 삭제됩니다.</li> <li>False: 보존 기간이 만료되면 이 버킷의 객체가 자동으로 삭제되지 않습니다. 이러한 개체를 삭제하려면 해당 개체를 수동으로 삭제해야 합니다.</li> </ul>

## 오류 응답

버킷이 규정을 준수하도록 생성되지 않은 경우 응답에 대한 HTTP 상태 코드는 S3 오류 코드 인 XNoSuchBucketCompliance 입니다 404 Not Found.

폐기됨: 버킷 준수 요청을 넣으십시오

Put Bucket 준수 요청은 더 이상 사용되지 않습니다. 그러나 이 요청을 계속 사용하여 기존 레거시 준수 버킷의 준수 설정을 수정할 수 있습니다. 예를 들어, 기존 버킷을 법적 보존 상태로 놓거나 보존 기간을 늘릴 수 있습니다.

이전 StorageGRID 버전에서 사용할 수 있었던 StorageGRID 규정 준수 기능은 더 이상 사용되지 않으며 S3 오브젝트 잠금으로 대체되었습니다. 자세한 내용은 다음을 참조하십시오.



- ["S3 REST API를 사용하여 S3 오브젝트 잠금을 구성합니다"](#)
- ["NetApp 기술 자료: StorageGRID 11.5에서 레거시 준수 버킷을 관리하는 방법"](#)

이 작업을 완료하려면 S3:PutBucketCompliance 권한이 있거나 계정 루트 권한이 있어야 합니다.

PUT 버킷 준수 요청을 발행할 때 준수 설정의 모든 필드에 값을 지정해야 합니다.

## 요청 예

이 예제 요청은 이름이 인 버킷의 준수 설정을 mybucket 수정합니다. 이 예에서의 객체는 이제 1년이 아닌 2년 동안(1,051,200분) 보존되며, 이는 객체가 mybucket 그리드로 유입되는 시점부터 시작됩니다. 이 버킷에는 법적 구속이 없습니다. 각 개체는 2년 후에 자동으로 삭제됩니다.

```

PUT /mybucket/?x-ntap-sg-compliance HTTP/1.1
Date: date
Authorization: authorization name
Host: host
Content-Length: 152

<SGCompliance>
  <RetentionPeriodMinutes>1051200</RetentionPeriodMinutes>
  <LegalHold>false</LegalHold>
  <AutoDelete>true</AutoDelete>
</SGCompliance>

```

이름	설명
RetentionPeriodMinutes(주기적 지연 시간)	<p>이 버킷에 추가된 객체의 보존 기간(분)입니다. 보존 기간은 객체가 그리드에 수집될 때 시작됩니다.</p> <ul style="list-style-type: none"> <li>중요 * RetentionPeriodMinutes에 새 값을 지정할 때는 버킷의 현재 보존 기간과 같거나 큰 값을 지정해야 합니다. 버킷의 보존 기간이 설정된 후에는 해당 값을 줄일 수 없으며 증가만 가능합니다.</li> </ul>
LegalHold	<ul style="list-style-type: none"> <li>참: 이 버킷은 현재 법적 증거 자료 보관 중입니다. 이 버킷의 오브젝트는 보존 기간이 만료된 경우에도 법적 보류가 해제될 때까지 삭제할 수 없습니다.</li> <li>거짓: 이 버킷은 현재 법적 증거 자료 보관 중이 아닙니다. 이 버킷의 오브젝트는 보존 기간이 만료되면 삭제할 수 있습니다.</li> </ul>
자동 삭제	<ul style="list-style-type: none"> <li>참: 버킷이 법적 보존 상태에 있지 않는 한, 보존 기간이 만료되면 이 버킷의 객체가 자동으로 삭제됩니다.</li> <li>False: 보존 기간이 만료되면 이 버킷의 객체가 자동으로 삭제되지 않습니다. 이러한 개체를 삭제하려면 해당 개체를 수동으로 삭제해야 합니다.</li> </ul>

#### 규정 준수 설정에 대한 일관성

PUT 버킷 준수 요청으로 S3 버킷의 준수 설정을 업데이트하면 StorageGRID는 그리드 전체에서 버킷의 메타데이터를 업데이트하려고 시도합니다. 기본적으로 StorageGRID는 \* 강력한 글로벌 \* 일관성을 사용하여 버킷 메타데이터가 포함된 모든 데이터 센터 사이트와 모든 스토리지 노드에서 변경된 규정 준수 설정에 대해 쓰기 후 읽기 정합성을 보장합니다.

데이터 센터 사이트 또는 사이트의 여러 스토리지 노드를 사용할 수 없기 때문에 StorageGRID에서 \* 강력한 글로벌 \* 일관성을 달성할 수 없는 경우 응답에 대한 HTTP 상태 코드는입니다 503 Service Unavailable.

이 응답을 받으면 그리드 관리자에게 문의하여 필요한 스토리지 서비스를 가능한 빨리 사용할 수 있도록 해야 합니다. 그리드 관리자가 각 사이트에서 스토리지 노드를 충분히 사용할 수 없는 경우 기술 지원 부서에서 \* 강력한 사이트 \* 일관성을 적용하여 실패한 요청을 다시 시도하도록 지시할 수 있습니다.



기술 지원 부서의 지시가 있거나 이 레벨을 사용할 때 발생할 수 있는 결과를 이해하지 않는 한, Put Bucket 준수를 위해 \*Strong-site\* 일관성을 강제로 적용하지 마십시오.

일관성이 \* 강력한 사이트 \*로 감소하면 StorageGRID는 업데이트된 규정 준수 설정이 사이트 내 클라이언트 요청에 대해서만 쓰기 후 읽기 일관성을 유지할 수 있도록 보장합니다. 즉, 모든 사이트 및 스토리지 노드를 사용할 수 있을 때까지 StorageGRID 시스템에 이 버킷에 대한 여러 개의 일관되지 않은 설정이 일시적으로 있을 수 있습니다. 설정이 일치하지 않으면 예기치 않거나 원치 않는 동작이 발생할 수 있습니다. 예를 들어, 버킷을 법적 증거 자료 보관 중에 두고 일관성을 더 낮게 설정하면 일부 데이터 센터 사이트에서 버킷의 이전 규정 준수 설정(즉, 법적 증거 자료 보관)이 계속 적용될 수 있습니다. 따라서 보존 기간이 만료되면 사용자나 자동 삭제(활성화된 경우)에 의해 법적 보류라고 생각하는 개체가 삭제될 수 있습니다.

Strong-site \* 일관성을 강제로 사용하려면 다음과 같이 Put Bucket 준수 요청을 다시 발행하고 HTTP 요청 헤더를 포함시킵니다. **Consistency-Control**

```
PUT /mybucket/?x-ntap-sg-compliance HTTP/1.1
Consistency-Control: strong-site
```

## 오류 응답

- 버킷이 규정을 준수하도록 생성되지 않은 경우 응답에 대한 HTTP 상태 코드는 입니다 404 Not Found.
- 요청에서 버킷의 현재 보존 기간보다 작은 경우 `RetentionPeriodMinutes` HTTP 상태 코드는 입니다 400 Bad Request.

## 관련 정보

["사용되지 않음: 규정 준수를 위해 버킷 요청 수정을 배치합니다"](#)

## 액세스 정책 관리

### 접근 정책 사용

StorageGRID은 AWS(Amazon Web Services) 정책 언어를 사용하여 S3 테넌트가 해당 버킷 및 오브젝트 내의 버킷에 대한 액세스를 제어할 수 있도록 합니다. StorageGRID 시스템은 S3 REST API 정책 언어의 하위 집합을 구현합니다. S3 API에 대한 액세스 정책은 JSON으로 기록됩니다.

### 액세스 정책 개요

StorageGRID 세 가지 종류의 액세스 정책을 지원합니다.

- \* 버킷 정책 \* - `GetBucketPolicy`, `PutBucketPolicy` 및 `DeleteBucketPolicy` S3 API 작업 또는 테넌트 관리자 또는 테넌트 관리 API를 사용하여 관리됩니다. 버킷 정책은 버킷에 첨부되므로 버킷 소유자 계정 또는 버킷에 대한 다른 계정 및 버킷에 있는 오브젝트에 대한 사용자의 액세스를 제어하도록 구성됩니다. 버킷 정책은 하나의 버킷과 여러 그룹에만 적용됩니다.
- 테넌트 관리자 또는 테넌트 관리 API를 사용하여 구성된 \* 그룹 정책 \*입니다. 그룹 정책은 계정의 그룹에 연결되므로 해당 그룹이 해당 계정이 소유한 특정 리소스에 액세스할 수 있도록 구성됩니다. 그룹 정책은 하나의 그룹에만 적용되고 여러 버킷에 적용될 수 있습니다.
- \*세션 정책\*은 `AssumeRole` 요청의 일부로 포함됩니다. 세션 정책은 지정된 세션에만 적용되며, 그룹 및 버킷

정책에서 부여한 권한 외에도 사용자가 가진 권한을 추가로 정의합니다.



그룹, 버킷, 세션 정책 간에 우선순위 차이는 없습니다.

StorageGRID 버킷 및 그룹 정책은 아마존에서 정의한 특정 문법을 따릅니다. 각 정책 안에는 정책 문의 배열이 들어 있으며 각 문에는 다음 요소가 포함되어 있습니다.

- 정책 ID(SID)(선택 사항)
- 효과
- Principal/NotPrincipal입니다
- 리소스/NotResource입니다
- 작업/NotAction
- 조건(선택 사항)

정책 문은 이 구조를 사용하여 권한을 지정합니다. `pert <effect> <principal>이(가) <condition>이(가) 적용될 때 <Resource>에서 <Action>을(를) 수행하도록 허용/거부합니다.`

각 정책 요소는 특정 함수에 사용됩니다.

요소	설명
SID	SID 요소는 선택 사항입니다. SID는 사용자에 대한 설명으로만 제공됩니다. StorageGRID 시스템에서 저장하지만 해석되지 않습니다.
효과	Effect 요소를 사용하여 지정된 작업의 허용 여부를 설정합니다. 지원되는 작업 요소 키워드를 사용하여 버킷 또는 오브젝트에 대해 허용(또는 거부)하는 작업을 식별해야 합니다.
Principal/NotPrincipal입니다	사용자, 그룹 및 계정이 특정 리소스에 액세스하고 특정 작업을 수행하도록 허용할 수 있습니다. 요청에 S3 서명이 포함되지 않은 경우 와일드카드 문자 (*)를 보안 주체에 지정하여 익명 액세스가 허용됩니다. 기본적으로 계정 루트만 해당 계정이 소유한 리소스에 액세스할 수 있습니다.  버킷 정책에서 Principal 요소만 지정하면 됩니다. 그룹 정책의 경우 정책이 연결된 그룹이 암시적 Principal 요소입니다.
리소스/NotResource입니다	Resource 요소는 버킷 및 오브젝트를 식별합니다. ARN(Amazon Resource Name)을 사용하여 리소스를 식별하는 버킷 및 객체에 대한 권한을 허용하거나 거부할 수 있습니다.
작업/NotAction	Action 및 Effect 요소는 권한의 두 구성 요소입니다. 그룹이 리소스를 요청하면 리소스에 대한 액세스가 부여되거나 거부됩니다. 명시적으로 권한을 할당하지 않는 한 액세스가 거부되지만 명시적 DENY를 사용하여 다른 정책이 부여한 권한을 재정의할 수 있습니다.
조건	Condition 요소는 선택 요소입니다. 조건을 사용하면 식을 만들어 정책을 적용해야 하는 시기를 결정할 수 있습니다.

Action 요소에서 와일드카드 문자(\*)를 사용하여 모든 작업이나 작업의 하위 집합을 지정할 수 있습니다. 예를 들어 이 작업은 S3:GetObject, S3:PutObject 및 S3:DeleteObject 와 같은 사용 권한을 일치시킵니다.

```
s3:*Object
```

Resource 요소에서 와일드카드 문자(\ ) 및 (?)를 사용할 수 있습니다. 별표()가 0개 이상의 문자와 일치하면 물음표 (?)가 모든 단일 문자와 일치합니다.

Principal 요소에서 모든 사용자에게 권한을 부여하는 익명 액세스를 설정하는 것 외에는 와일드카드 문자는 지원되지 않습니다. 예를 들어 와일드카드(\*)를 Principal 값으로 설정합니다.

```
"Principal": "*"
```

```
"Principal": {"AWS": "*"}  
"Principal": {"AWS": "arn:aws:iam::27233906934684427525:federated-group/admin", "AWS": "arn:aws:iam::27233906934684427525:federated-group/finance"}
```

다음 예제에서는 Effect, Principal, Action 및 Resource 요소를 사용합니다. 이 예에서는 "허용"의 효과를 사용하여 Principals, 관리자 그룹 및 재무 그룹 federated-group/finance, 이름이 지정된 버킷에서 mybucket 작업을 수행할 수 있는 권한 s3>ListBucket 및 해당 버킷 내의 모든 객체에 대한 작업을 s3:GetObject 제공하는 완전한 버킷 정책 문을 보여 federated-group/admin 줍니다.

```
{  
  "Statement": [  
    {  
      "Effect": "Allow",  
      "Principal": {  
        "AWS": [  
          "arn:aws:iam::27233906934684427525:federated-group/admin",  
          "arn:aws:iam::27233906934684427525:federated-group/finance"  
        ]  
      },  
      "Action": [  
        "s3>ListBucket",  
        "s3:GetObject"  
      ],  
      "Resource": [  
        "arn:aws:s3:::mybucket",  
        "arn:aws:s3:::mybucket/*"  
      ]  
    }  
  ]  
}
```

버킷 정책은 크기 제한이 20,480바이트이고 그룹 정책은 크기 제한이 5,120바이트입니다.

## 정책의 일관성

기본적으로 그룹 정책에 대한 모든 업데이트는 최종적으로 일치합니다. 그룹 정책이 일관되면 정책 캐싱으로 인해 변경 내용이 적용되는 데 15분이 더 걸릴 수 있습니다. 기본적으로 버킷 정책에 대한 모든 업데이트는 매우 일관적입니다.

필요에 따라 버킷 정책 업데이트의 일관성 보장을 변경할 수 있습니다. 예를 들어 사이트 중단 중에 버킷 정책의 변경을 사용할 수 있도록 할 수 있습니다.

이 경우 PutBucketPolicy 요청에서 헤더를 설정하거나 Put Bucket 정합성 요청을 사용할 수 Consistency-Control 있습니다. 버킷 정책의 정합성이 보장되면 정책 캐싱으로 인해 변경 내용이 적용되는 데 8초 더 걸릴 수 있습니다.



일시적 상황을 해결하기 위해 일관성을 다른 값으로 설정한 경우 작업을 마치면 버킷 수준 설정을 원래 값으로 다시 설정해야 합니다. 그렇지 않으면 이후의 모든 버킷 요청에 수정된 설정이 사용됩니다.

## 세션 정책이란 무엇인가요?

세션 정책은 사용자가 그룹을 맙는 경우와 같이 특정 세션 동안 사용 가능한 권한을 일시적으로 제한하는 액세스 정책입니다. 세션 정책은 권한의 하위 집합만 허용할 수 있으며, 추가 권한을 부여할 수 없습니다. 그룹 자체는 더 광범위한 권한을 가질 수 있습니다.

정책 설명에 **ARN**을 사용합니다

정책 문에서 ARN은 Principal 및 Resource 요소에서 사용됩니다.

- 이 구문을 사용하여 S3 리소스 ARN을 지정합니다.

```
arn:aws:s3:::bucket-name
arn:aws:s3:::bucket-name/object_key
```

- 이 구문을 사용하여 ID 리소스 ARN(사용자 및 그룹)을 지정합니다.

```
arn:aws:iam::account_id:root
arn:aws:iam::account_id:user/user_name
arn:aws:iam::account_id:group/group_name
arn:aws:iam::account_id:federated-user/user_name
arn:aws:iam::account_id:federated-group/group_name
```

## 기타 고려 사항:

- 별표(\*)를 와일드카드로 사용하여 개체 키 안에 0개 이상의 문자를 일치시킬 수 있습니다.
- 개체 키에 지정할 수 있는 국제 문자는 JSON UTF-8 또는 JSON\u 이스케이프 시퀀스를 사용하여 인코딩해야 합니다. 퍼센트 인코딩은 지원되지 않습니다.

## "RFC 2141 URN 구문"

PutBucketPolicy 작업에 대한 HTTP 요청 본문은 charset=UTF-8로 인코딩되어야 합니다.

정책에서 리소스를 지정합니다

정책 문에서 Resource 요소를 사용하여 사용 권한이 허용되거나 거부되는 버킷 또는 개체를 지정할 수 있습니다.

- 각 정책 문에는 Resource 요소가 필요합니다. 정책에서 리소스는 요소로 표시되거나 NotResource 제외를 위해 요소로 Resource 표시됩니다.
- S3 리소스 ARN을 사용하여 리소스를 지정합니다. 예를 들면 다음과 같습니다.

```
"Resource": "arn:aws:s3:::mybucket/*"
```

- 개체 키 내에서 정책 변수를 사용할 수도 있습니다. 예를 들면 다음과 같습니다.

```
"Resource": "arn:aws:s3:::mybucket/home/${aws:username}/*"
```

- 리소스 값은 그룹 정책이 생성될 때 아직 존재하지 않는 버킷을 지정할 수 있습니다.

정책에 보안 주체를 지정합니다

Principal 요소를 사용하여 policy 문에 의해 리소스에 대한 액세스가 허용/거부된 사용자, 그룹 또는 테넌트 계정을 식별합니다.

- 버킷 정책의 각 정책 선언에는 Principal 요소가 포함되어야 합니다. 그룹 정책의 정책 설명은 그룹이 보안 주체로 인식되기 때문에 Principal 요소가 필요하지 않습니다.
- 정책에서 주체는 "Principal" 또는 "NotPrincipal" 요소로 표시됩니다.
- 계정 기반 ID는 ID 또는 ARN을 사용하여 지정해야 합니다.

```
"Principal": { "AWS": "account_id" }  
"Principal": { "AWS": "identity_arn" }
```

- 이 예에서는 계정 루트 및 계정의 모든 사용자를 포함하는 테넌트 계정 ID 27233906934684427525를 사용합니다.

```
"Principal": { "AWS": "27233906934684427525" }
```

- 계정 루트만 지정할 수 있습니다.

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:root" }
```

- 특정 페더레이션 사용자("Alex")를 지정할 수 있습니다.

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-user/Alex" }
```

- 특정 통합 그룹("관리자")을 지정할 수 있습니다.

```
"Principal": { "AWS": "arn:aws:iam::27233906934684427525:federated-group/Managers" }
```

- 익명 보안 주체를 지정할 수 있습니다.

```
"Principal": "*"
```

- 모호함을 방지하려면 사용자 이름 대신 사용자 UUID를 사용할 수 있습니다.

```
arn:aws:iam::27233906934684427525:user-uuid/de305d54-75b4-431b-adb2-eb6b9e546013
```

예를 들어 Alex가 조직을 떠나고 사용자 이름이 삭제된다고 가정해 Alex 보겠습니다. 새 Alex가 조직에 합류하여 동일한 사용자 이름이 할당된 경우 Alex 새 사용자는 원래 사용자에게 부여된 권한을 의도치 않게 상속할 수 있습니다.

- Principal 값은 버킷 정책이 생성될 때 아직 존재하지 않는 그룹/사용자 이름을 지정할 수 있습니다.

정책에서 사용 권한을 지정합니다

정책에서 Action 요소는 리소스에 대한 권한을 허용/거부하는 데 사용됩니다. 정책에서 지정할 수 있는 사용 권한 집합이 있으며, 이러한 권한은 "작업" 또는 "NotAction" 요소로 표시됩니다. 각 요소는 특정 S3 REST API 작업에 매핑됩니다.

이 표에는 버킷에 적용되는 사용 권한과 객체에 적용되는 사용 권한이 나열되어 있습니다.

- i 이제 Amazon S3는 PutBucketReplication 및 DeleteBucketReplication 작업 모두에 대해 S3:PutReplicationConfiguration 권한을 사용합니다. StorageGRID는 원래 Amazon S3 사양과 일치하는 각 작업에 대해 별도의 권한을 사용합니다.
- i 기존 값을 덮어쓰는 데 PUT을 사용할 때 삭제가 수행됩니다.

버킷에 적용되는 권한

권한	S3 REST API 작업	StorageGRID 사용자 지정
S3:생성 버킷	CreateBucket	예. <ul style="list-style-type: none"> <li>참고 *: 그룹 정책에만 사용합니다.</li> </ul>
S3:삭제 버킷	삭제 버킷	

권한	S3 REST API 작업	StorageGRID 사용자 지정
S3:DeleteBucketMetadataNotification	버킷 메타데이터 알림 구성을 삭제합니다	예
S3:삭제 BucketPolicy	DeleteBucketPolicy를 참조하십시오	
S3:DeleteReplicationConfiguration	DeleteBucketReplication 을 참조하십시오	예, 삽입 및 삭제 권한을 구분합니다
S3:GetBucketAcl	GetBucketAcl	
S3:GetBucketCompliance	버킷 규정 준수 가져오기(더 이상 사용되지 않음)	예
S3:GetBucketConsistency	버킷 일관성 확보	예
S3:GetBucketCORS	GetBucketCors 를 참조하십시오	
S3:GetEncryptionConfiguration	GetBucketEncryption을 참조하십시오	
S3:GetBucketLastAccessTime	버킷 최종 액세스 시간 가져오기	예
S3:GetBucketLocation	GetBucketLocation 을 참조하십시오	
S3:GetBucketMetadataNotification 을 참조하십시오	Bucket 메타데이터 알림 구성 가져오기	예
S3:GetBucketNotification 을 참조하십시오	GetBucketNotificationConfiguration 을 참조하십시오	
S3:GetBucketObjectLockConfiguration	GetObjectLockConfiguration 을 참조하십시오	
S3:GetBucketPolicy를 참조하십시오	GetBucketPolicy를 참조하십시오	
S3:GetBucketTagging	GetBucketTagging	
S3:GetBucketVersioning	GetBucketVersioning 을 참조하십시오	
S3:GetLifecycleConfiguration	GetBucketLifecycleConfiguration 을 참조하십시오	
S3:GetReplicationConfiguration	GetBucketReplication 을 참조하십시오	

권한	S3 REST API 작업	StorageGRID 사용자 지정
S3:ListAllMyBucket	<ul style="list-style-type: none"> <li>• ListBucket</li> <li>• 스토리지 사용량을 가져옵니다</li> </ul>	예, 스토리지 사용량 가져오기. <ul style="list-style-type: none"> <li>• 참고 *: 그룹 정책에만 사용합니다.</li> </ul>
S3:목록 버킷	<ul style="list-style-type: none"> <li>• ListObjects 를 선택합니다</li> <li>• 머리버킷</li> <li>• RestoreObject 를 선택합니다</li> </ul>	
S3:ListBucketMultipartUploads	<ul style="list-style-type: none"> <li>• ListMultipartUploads 를 참조하십시오</li> <li>• RestoreObject 를 선택합니다</li> </ul>	
S3:목록 BucketVersions	버킷 버전 가져오기	
S3: PutBucketCompliance	버킷 규정 준수(폐기됨)	예
S3: PutBucketConsistency	버킷 일관성을 유지합니다	예
S3: PutBucketCORS	<ul style="list-style-type: none"> <li>• DeleteBucketCors †</li> <li>• BucketCors의</li> </ul>	
S3:PutEncryptionConfiguration	<ul style="list-style-type: none"> <li>• DeleteBucketEncryption</li> <li>• PutBucketEncryption을 참조하십시오</li> </ul>	
S3:PutBucketLastAccessTime	버킷 최종 접근 시간	예
S3:PutBucketMetadataNotification	Put Bucket 메타데이터 알림 구성	예
S3: PutBucketNotification	PutBucketNotificationConfiguration을 참조하십시오	
S3:PutBucketObjectLockConfiguration	<ul style="list-style-type: none"> <li>• 요청 헤더가 있는 CreateBucket x-amz-bucket-object-lock-enabled: true(S3:CreateBucket 권한도 필요)</li> <li>• PutObjectLockConfiguration 을 참조하십시오</li> </ul>	
S3: PutBucketPolicy	BucketPolicy를 참조하십시오	

권한	S3 REST API 작업	StorageGRID 사용자 지정
S3: PutBucketTagging	<ul style="list-style-type: none"> <li>DeleteBucketTagging † 를 참조하십시오</li> <li>BucketTagging</li> </ul>	
S3: PutBucketVersioning	PutBucketVersioning을 참조하십시오	
S3: PutLifecycleConfiguration	<ul style="list-style-type: none"> <li>DeleteBucketLifecycle † 을 참조하십시오</li> <li>PutBucketLifecycleConfiguration을 참조하십시오</li> </ul>	
S3:PutReplicationConfiguration	PutBucketReplication을 참조하십시오	예, 삽입 및 삭제 권한을 구분합니다

#### 객체에 적용되는 권한

권한	S3 REST API 작업	StorageGRID 사용자 지정
S3:중단멀티업로드입니다	<ul style="list-style-type: none"> <li>AbortMultipartUpload 를 클릭합니다</li> <li>RestoreObject 를 선택합니다</li> </ul>	
S3:BypassGovernanceRetention	<ul style="list-style-type: none"> <li>DeleteObject 를 클릭합니다</li> <li>DeleteObjects 를 클릭합니다</li> <li>PutObjectRetention</li> </ul>	
S3:DeleteObject 를 선택합니다	<ul style="list-style-type: none"> <li>DeleteObject 를 클릭합니다</li> <li>DeleteObjects 를 클릭합니다</li> <li>RestoreObject 를 선택합니다</li> </ul>	
S3:삭제 ObjectTagging	DeleteObjectTagging 을 선택합니다	
S3:DeleteObjectVersionTagging	DeleteObjectTagging(객체의 특정 버전)	
S3:DeleteObjectVersion	DeleteObject(객체의 특정 버전)	
S3:GetObject	<ul style="list-style-type: none"> <li>GetObject 를 참조하십시오</li> <li>HeadObject 를 선택합니다</li> <li>RestoreObject 를 선택합니다</li> <li>SelectObjectContent 를 선택합니다</li> </ul>	

권한	S3 REST API 작업	StorageGRID 사용자 지정
S3:GetObjectAcl	GetObjectAcl	
S3:GetObjectLegalHold	GetObjectLegalHold 를 참조하십시오	
S3:GetObjectRetention	GetObjectRetention을 참조하십시오	
S3:GetObjectTagging	GetObjectTagging	
S3:GetObjectVersionTagging	GetObjectTagging(객체의 특정 버전)	
S3:GetObjectVersion	GetObject(객체의 특정 버전)	
S3>ListMultipartUploadParts(S3>ListMultim partUploadParts) 를	ListParts, RestoreObject 를 참조하십시오	
S3:PutObject	<ul style="list-style-type: none"> <li>PutObject 를 선택합니다</li> <li>CopyObject 를 선택합니다</li> <li>RestoreObject 를 선택합니다</li> <li>CreateMultipartUpload 를 클릭합니다</li> <li>CompleteMultipartUpload를 클릭합니다</li> <li>업로드 파트</li> <li>업로드파트 복사</li> </ul>	
S3:PutObjectLegalHold	PutObjectLegalHold를 선택합니다	
S3:PutObjectRetention	PutObjectRetention	
S3:PutObjectTagging	PutObjectTagging	
S3:PutObjectVersionTagging	PutObjectTagging(객체의 특정 버전)	
S3:PutOverwriteObject	<ul style="list-style-type: none"> <li>PutObject 를 선택합니다</li> <li>CopyObject 를 선택합니다</li> <li>PutObjectTagging</li> <li>DeleteObjectTagging 을 선택합니다</li> <li>CompleteMultipartUpload를 클릭합니다</li> </ul>	예
S3:RestoreObject	RestoreObject 를 선택합니다	

## PutOverwriteObject 권한을 사용합니다

S3:PutOverwriteObject 권한은 개체를 만들거나 업데이트하는 작업에 적용되는 사용자 지정 StorageGRID 권한입니다. 이 사용 권한의 설정에 따라 클라이언트가 개체의 데이터, 사용자 정의 메타데이터 또는 S3 오브젝트 태그 지정을 덮어쓸 수 있는지 여부가 결정됩니다.

이 권한에 사용할 수 있는 설정은 다음과 같습니다.

- \* 허용 \*: 클라이언트가 개체를 덮어쓸 수 있습니다. 기본 설정입니다.
- \* 거부 \*: 클라이언트가 개체를 덮어쓸 수 없습니다. Deny로 설정된 경우 PutOverwriteObject 권한은 다음과 같이 작동합니다.
  - 기존 객체가 같은 경로에 있는 경우:
    - 오브젝트의 데이터, 사용자 정의 메타데이터 또는 S3 오브젝트 태깅을 덮어쓸 수 없습니다.
    - 진행 중인 모든 수집 작업이 취소되고 오류가 반환됩니다.
    - S3 버전 관리를 사용하는 경우 거부 설정을 사용하면 PutObjectTagging 또는 DeleteObjectTagging 작업에서 개체 및 해당 비최신 버전에 대한 TagSet을 수정할 수 없습니다.
  - 기존 개체를 찾을 수 없으면 이 권한은 적용되지 않습니다.
- 이 권한이 없으면 Allow가 설정된 것과 효과가 같습니다.

 현재 S3 정책이 덮어쓰기를 허용하고 PutOverwriteObject 권한이 Deny로 설정된 경우 클라이언트는 객체의 데이터, 사용자 정의 메타데이터 또는 객체 태그를 덮어쓸 수 없습니다. 또한, 클라이언트 수정 방지 체크박스가 선택된 경우(구성 > 보안 설정 > 네트워크 및 개체), 해당 설정은 PutOverwriteObject 권한 설정을 재정의합니다.

정책에서 조건을 지정합니다

조건은 정책이 적용되는 시점을 정의합니다. 조건은 연산자 및 키 값 쌍으로 구성됩니다.

조건은 평가에 키 값 쌍을 사용합니다. 조건 요소에는 여러 조건이 포함될 수 있으며 각 조건에는 여러 키 값 쌍이 포함될 수 있습니다. 조건 블럭은 다음 형식을 사용합니다:

```
Condition: {
    condition_type: {
        condition_key: condition_values
    }
}
```

다음 예제에서 IPAddress 조건은 SOURCEIP 조건 키를 사용합니다.

```
"Condition": {
    "IpAddress": {
        "aws:SourceIp": "54.240.143.0/24"
        ...
    },
    ...
}
```

## 지원되는 조건 연산자

조건 연산자는 다음과 같이 분류됩니다.

- 문자열
- 숫자
- 부울
- IP 주소입니다
- Null 확인

조건 연산자	설명
StringEquals	정확한 일치(대/소문자 구분)를 기반으로 문자열 값과 키를 비교합니다.
StringNotEquals	키를 부정 일치(대/소문자 구분)를 기반으로 문자열 값과 비교합니다.
StringEqualsIgnoreCase 를 참조하십시오	정확한 일치를 기준으로 문자열 값과 키를 비교합니다(대/소문자 무시).
StringNotEqualsIgnoreCase 를 참조하십시오	Negated matching (대소문자 무시)을 기준으로 문자열 값과 키를 비교합니다.
StringLike 를 선택합니다	정확한 일치(대/소문자 구분)를 기반으로 문자열 값과 키를 비교합니다. 및? 와일드카드 문자를 포함할 수 있습니다.
StringNotLike 를 참조하십시오	키를 부정 일치(대/소문자 구분)를 기반으로 문자열 값과 비교합니다. 및? 와일드카드 문자를 포함할 수 있습니다.
NumericEquals	정확한 일치를 기준으로 키를 숫자 값과 비교합니다.
NumericNotEquals	키를 부정 일치를 기준으로 숫자 값과 비교합니다.
NumericGreaterThan	"보다 큼" 일치를 기준으로 키를 숫자 값과 비교합니다.
NumericGreaterThanOrEqual	"보다 크거나 같음" 일치를 기준으로 키를 숫자 값과 비교합니다.
NumericLessThan	"보다 작음" 일치를 기준으로 키를 숫자 값과 비교합니다.
NumericLessThanOrEqual	"보다 작거나 같음" 일치를 기준으로 키를 숫자 값과 비교합니다.
불입니다	"true 또는 false" 일치를 기준으로 키를 부울 값과 비교합니다.
IP 주소	키를 IP 주소 또는 IP 주소 범위와 비교합니다.

조건 연산자	설명
NotIpAddress 를 참조하십시오	부정 일치를 기준으로 IP 주소 또는 IP 주소 범위와 키를 비교합니다.
null입니다	현재 요청 컨텍스트에 조건 키가 있는지 확인합니다.
존재한다면	Null 조건을 제외한 모든 조건 연산자에 추가되어 해당 조건 키가 없는지 확인합니다. 조건 키가 없으면 TRUE를 반환합니다.

## 지원되는 조건 키

상태 키	작업	설명
AWS: SOURCEIP	IP 연산자	<p>요청이 전송된 IP 주소와 비교합니다. 버킷 또는 오브젝트 작업에 사용할 수 있습니다.</p> <ul style="list-style-type: none"> <li>참고: * S3 요청이 관리 노드 및 게이트웨이 노드의 로드 밸런서 서비스를 통해 전송된 경우 로드 밸런서 서비스의 IP 주소 업스트림과 비교됩니다.</li> <li>참고 *: 타사, 비투명 로드 밸런서가 사용되는 경우 이 로드 밸런서의 IP 주소와 비교합니다. 'X-Forwarded-For' 헤더의 유효성을 확인할 수 없으므로 헤더는 무시됩니다.</li> </ul>
AWS: 사용자 이름	리소스/ID입니다	요청이 전송된 보낸 사람의 사용자 이름과 비교합니다. 버킷 또는 오브젝트 작업에 사용할 수 있습니다.
S3: 구분 기호	S3:ListBucket 및 S3: ListBucketVersions 권한	는 ListObjects 또는 ListObjectVersions 요청에 지정된 구분 기호 매개 변수와 비교됩니다.

상태 키	작업	설명
S3: ExistingObjectTag/<tag-key>	S3:삭제 ObjectTagging S3:DeleteObjectVersionTagging S3:GetObject S3:GetObjectAcl S3:GetObjectTagging S3:GetObjectVersion S3:GetObjectVersionAcl S3:GetObjectVersionTagging S3:PutObjectAcl S3:PutObjectTagging S3:PutObjectVersionAcl S3:PutObjectVersionTagging	기존 개체에 특정 태그 키와 값이 있어야 합니다.
S3: 최대 키	S3>ListBucket 및 S3: ListBucketVersions 권한	는 ListObjects 또는 ListObjectVersions 요청에 지정된 max-keys 매개 변수와 비교됩니다.
s3:객체 잠금 모드	S3:PutObject	와 비교합니다 object-lock-mode PutObject, CopyObject 및 CreateMultipartUpload 요청의 요청 헤더에서 확장되었습니다.
s3:객체 잠금 모드	S3:PutObjectRetention	와 비교합니다 object-lock-mode PutObjectRetention 요청의 XML 본문에서 확장되었습니다.
S3: 오브젝트 잠금 장치 - 남은 보존 기간(일)	S3:PutObject	요청 헤더에 지정되거나 버킷 기본 보존 기간에서 계산된 유지 종료 날짜와 비교하여 x-amz-object-lock-retain-until-date 이러한 값이 다음 요청에 대해 허용되는 범위 내에 있는지 확인합니다. <ul style="list-style-type: none"> <li>PutObject 를 선택합니다</li> <li>CopyObject 를 선택합니다</li> <li>CreateMultipartUpload 를 클릭합니다</li> </ul>

상태 키	작업	설명
S3: 오브젝트 잠금 장치 - 남은 보존 기간(일)	S3:PutObjectRetention	PutObjectRetention 요청에 지정된 유지 종료 날짜와 비교하여 허용 범위 내에 있는지 확인합니다.
S3: 접두어	S3:ListBucket 및 S3: ListBucketVersions 권한	는 ListObjects 또는 ListObjectVersions 요청에 지정된 접두사 매개 변수와 비교됩니다.
S3: RequestObjectTag/<tag-key>	S3:PutObject S3:PutObjectTagging S3:PutObjectVersionTagging	개체 요청에 태그가 포함된 경우 특정 태그 키와 값이 필요합니다.
s3:x-amz-서버 측 암호화- 고객 알고리즘	S3:PutObject	와 비교합니다 sse-customer-algorithm 또는 copy-source-sse-customer-algorithm PutObject, CopyObject, CreateMultipartUpload, UploadPart, UploadPartCopy 및 CompleteMultipartUpload 요청의 요청 헤더에서 확장되었습니다.

정책에 변수를 지정합니다

정책의 변수를 사용하여 사용 가능한 정책 정보를 채울 수 있습니다. 요소의 문자열 비교와 요소에 Condition 정책 변수를 사용할 수 Resource 있습니다.

이 예제에서 변수는 \${aws:username} Resource 요소의 일부입니다.

```
"Resource": "arn:aws:s3:::bucket-name/home/${aws:username}/*"
```

이 예제에서 변수는 \${aws:username} 조건 블록의 조건 값의 일부입니다.

```
"Condition": {  
  "StringLike": {  
    "s3:prefix": "${aws:username}/*"  
    ...  
  },  
  ...  
}
```

변수	설명
\${aws:SourceIp}	SOURCEIP 키를 제공된 변수로 사용합니다.

변수	설명
<code>\$(aws:username)</code>	제공된 변수로 사용자 이름 키를 사용합니다.
<code>\$(s3:prefix)</code>	서비스별 prefix key를 제공된 variable로 사용한다.
<code>\$(s3:max-keys)</code>	서비스별 최대 키 키를 제공된 변수로 사용합니다.
<code>\$(*)</code>	특수 문자. 문자를 리터럴 * 문자로 사용합니다.
<code>\$(?)</code>	특수 문자. 문자를 리터럴? 문자로 사용합니다.
<code>\$( \$ )</code>	특수 문자. 문자를 리터럴 \$ 문자로 사용합니다.

특별한 처리가 필요한 정책을 생성합니다

때로는 정책에 따라 보안이 위험하거나 계정 루트 사용자를 잠그는 등 지속적인 작업에 위험한 사용 권한을 부여할 수 있습니다. StorageGRID S3 REST API 구현은 Amazon보다 정책 검증 중에 덜 제한적이지만 정책 평가 중에도 동일하게 엄격합니다.

정책 설명입니다	정책 유형입니다	아마존 행동	StorageGRID 동작
루트 계정에 대한 모든 권한을 스스로 거부합니다	버킷	유효하고 적용되지만 루트 사용자 계정은 모든 S3 버킷 정책 작업에 대한 권한을 보유합니다	동일합니다
사용자/그룹에 대한 모든 권한을 스스로 거부합니다	그룹	유효하고 시행되었습니다	동일합니다
외부 계정 그룹에 모든 권한을 허용합니다	버킷	주체가 잘못되었습니다	유효하지만 모든 S3 버킷 정책 작업에 대한 권한은 정책에서 허용하는 경우 405 메서드 허용되지 않음 오류를 반환합니다
외부 계정 루트 또는 사용자에게 모든 권한을 허용합니다	버킷	유효하지만 모든 S3 버킷 정책 작업에 대한 권한은 정책에서 허용하는 경우 405 메서드 허용되지 않음 오류를 반환합니다	동일합니다
모든 사용자에게 모든 작업에 대한 사용 권한을 허용합니다	버킷	유효하지만 모든 S3 버킷 정책 작업에 대한 사용 권한이 외국 계정 루트 및 사용자에 대해 405 메서드 허용 안 됨 오류를 반환합니다	동일합니다

정책 설명입니다	정책 유형입니다	아마존 행동	StorageGRID 동작
모든 작업에 대한 모든 사용자의 권한을 거부합니다	버킷	유효하고 적용되지만 루트 사용자 계정은 모든 S3 버킷 정책 작업에 대한 권한을 보유합니다	동일합니다
보안 주체는 존재하지 않는 사용자 또는 그룹입니다	버킷	주체가 잘못되었습니다	유효합니다
리소스가 존재하지 않는 S3 버킷입니다	그룹	유효합니다	동일합니다
보안 주체는 로컬 그룹입니다	버킷	주체가 잘못되었습니다	유효합니다
정책은 개체를 넣을 수 있는 비소유자 계정(익명 계정 포함) 권한을 부여합니다.	버킷	유효합니다. 객체는 생성자 계정이 소유하며 버킷 정책은 적용되지 않습니다. 생성자 계정은 개체 ACL을 사용하여 개체에 대한 액세스 권한을 부여해야 합니다.	유효합니다. 오브젝트는 버킷 소유자 계정이 소유합니다. 버킷 정책이 적용됩니다.

#### WORM(Write-Once-Read-Many) 보호

WORM(Write-Once-Read-Many) 버킷을 생성하여 데이터, 사용자 정의 오브젝트 메타데이터 및 S3 오브젝트 태깅을 보호할 수 있습니다. 새 객체를 생성하고 기존 컨텐츠를 덮어쓰거나 삭제하지 못하도록 WORM 버킷을 구성합니다. 여기에 설명된 방법 중 하나를 사용합니다.

덮어쓰기가 항상 거부되도록 하려면 다음을 수행할 수 있습니다.

- 그리드 관리자에서 구성 > 보안 > 보안 설정 > 네트워크 및 개체\*로 이동하여 \*클라이언트 수정 방지 확인란을 선택합니다.
- 다음 규칙 및 S3 정책을 적용합니다.
  - S3 정책에 PutOverwriteObject 거부 작업을 추가합니다.
  - DeleteObject 거부 작업을 S3 정책에 추가합니다.
  - S3 정책에 PutObject 허용 작업을 추가합니다.



S3 정책에서 DeleteObject를 DENY로 설정해도 "30일 후 복사본 제로" 같은 규칙이 있을 때 ILM이 개체를 삭제할 수 없습니다.



이러한 규칙과 정책이 모두 적용되더라도 동시 쓰기를 방지하지 않습니다(상황 A 참조). 순차적 완료된 덮어쓰기를 방지합니다(상황 B 참조).

- 상황 A \*: 동시 쓰기(보호 안 됨)

```
/mybucket/important.doc  
PUT#1 ---> OK  
PUT#2 -----> OK
```

- 상황 B \*: 순차적 완료된 덮어쓰기(방지됨)

```
/mybucket/important.doc  
PUT#1 -----> PUT#2 ---X (denied)
```

#### 관련 정보

- "StorageGRID ILM 규칙이 개체를 관리하는 방법"
- "버킷 정책의 예"
- "그룹 정책의 예"
- "세션 정책 예시"
- "ILM을 사용하여 개체를 관리합니다"
- "테넌트 계정을 사용합니다"

#### 세션 정책 예

다음 예를 사용하여 StorageGRID 세션 정책을 구축하세요.

예: 객체 검색을 허용하는 세션 정책 설정

이 예에서 세션의 주체는 bucket1에서만 객체를 검색할 수 있습니다. StorageGRID 관련 작업(예: StorageGRID 사용)을 제외한 다른 모든 작업은 암묵적으로 거부됩니다. "S3:PutOverwriteObject" 허가. AssumeRole API를 호출하는 동안 세션 정책은 JSON 파일로 제공될 수 있습니다.

```
{  
  "Statement": [  
    {  
      "Action": "s3:GetObject",  
      "Effect": "Allow",  
      "Resource": "arn:aws:s3:::bucket1/*"  
    }  
  ]  
}
```

#### 버킷 정책의 예

이 섹션의 예를 사용하여 버킷에 대한 StorageGRID 액세스 정책을 구축합니다.

버킷 정책은 정책이 연결된 버킷에 대한 액세스 권한을 지정합니다. 다음 툴 중 하나를 통해 S3 PutBucketPolicy API를

사용하여 버킷 정책을 구성합니다.

- "테넌트 관리자"...
- 다음 명령을 사용하는 AWS CLI(참조 "버킷 작업"):

```
> aws s3api put-bucket-policy --bucket examplebucket --policy
file://policy.json
```

예: 모든 사용자가 버킷에 읽기 전용 액세스를 허용합니다

이 예제에서는 anonymous를 비롯한 모든 사용자가 버킷의 오브젝트를 나열하고 버킷의 모든 오브젝트에 대해 GetObject 작업을 수행할 수 있습니다. 다른 모든 작업은 거부됩니다. 이 정책은 계정 루트 외에는 버킷에 쓸 수 있는 권한이 없으므로 특히 유용하지 않을 수 있습니다.

```
{
  "Statement": [
    {
      "Sid": "AllowEveryoneReadOnlyAccess",
      "Effect": "Allow",
      "Principal": "*",
      "Action": [ "s3:GetObject", "s3>ListBucket" ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}
```

예: 한 계정의 모든 사용자가 완전히 액세스할 수 있도록 허용하고 다른 계정의 모든 사용자는 버킷에 읽기 전용으로 액세스할 수 있습니다

이 예제에서는 지정된 한 계정의 모든 사용자가 버킷에 대한 전체 액세스를 허용하지만 다른 지정된 계정의 모든 사용자는 버킷을 나열하고 오브젝트 키 접두사로 시작하는 버킷의 객체에 대해 GetObject 작업을 수행할 수만 shared/ 있습니다.



StorageGRID에서 비소유자 계정(익명 계정 포함)으로 생성된 객체는 버킷 소유자 계정이 소유합니다.  
버킷 정책은 이러한 오브젝트에 적용됩니다.

```

{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "95390887230002558202"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3:GetObject",
      "Resource": "arn:aws:s3:::examplebucket/shared/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "31181711887329436680"
      },
      "Action": "s3>ListBucket",
      "Resource": "arn:aws:s3:::examplebucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "shared/*"
        }
      }
    }
  ]
}

```

예: 모든 사용자가 버킷에 대한 읽기 전용 액세스 및 지정된 그룹에 의한 전체 액세스 허용

이 예제에서는 anonymous를 포함한 모든 사용자가 버킷의 모든 오브젝트에 대해 버킷을 나열하고 GetObject 작업을 수행할 수 있지만 지정된 계정의 그룹에 속한 사용자만 Marketing 전체 액세스를 허용합니다.

```

{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/Marketing"
      },
      "Action": "s3:*",
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Principal": "*",
      "Action": ["s3>ListBucket", "s3:GetObject"],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}

```

예: 클라이언트가 **IP** 범위에 있는 경우 모든 사용자가 버킷에 대한 읽기 및 쓰기 액세스를 허용합니다

이 예제에서는 요청이 지정된 IP 범위(54.240.143.0 ~ 54.240.143.255, 54.240.143.188 제외)에서 발생한 경우 **anonymous**를 포함한 모든 사람이 버킷을 나열하고 버킷의 모든 오브젝트에 대해 오브젝트 작업을 수행할 수 있습니다. 다른 모든 작업이 거부되고 IP 범위를 벗어난 모든 요청이 거부됩니다.

```
{  
  "Statement": [  
    {  
      "Sid": "AllowEveryoneReadWriteAccessIfInSourceIpRange",  
      "Effect": "Allow",  
      "Principal": "*",  
      "Action": [ "s3:*Object", "s3>ListBucket" ],  
      "Resource":  
      ["arn:aws:s3:::examplebucket", "arn:aws:s3:::examplebucket/*"],  
      "Condition": {  
        "IpAddress": {"aws:SourceIp": "54.240.143.0/24"},  
        "NotIpAddress": {"aws:SourceIp": "54.240.143.188"}  
      }  
    }  
  ]  
}
```

예: 지정된 통합 사용자가 단독으로 버킷을 완전히 액세스할 수 있도록 허용합니다

이 예에서는 페더레이션 사용자 Alex가 버킷 및 해당 객체에 대한 전체 액세스를 examplebucket 허용합니다.  
"root"를 포함한 다른 모든 사용자는 모든 작업을 명시적으로 거부합니다. 그러나 "root"는  
PUT/GET/DeleteBucketPolicy에 대한 권한이 거부되지 않습니다.

```

{
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    },
    {
      "Effect": "Deny",
      "NotPrincipal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-user/Alex"
      },
      "Action": [
        "s3:*"
      ],
      "Resource": [
        "arn:aws:s3:::examplebucket",
        "arn:aws:s3:::examplebucket/*"
      ]
    }
  ]
}

```

#### 예: PutOverwriteObject 권한

이 예제에서 Deny PutOverwriteObject 및 DeleteObject의 효과는 아무도 개체의 데이터, 사용자 정의 메타데이터 및 S3 개체 태그를 덮어쓰거나 삭제할 수 없도록 합니다.

```

{
  "Statement": [
    {
      "Effect": "Deny",
      "Principal": "*",
      "Action": [
        "s3:PutOverwriteObject",
        "s3:DeleteObject",
        "s3:DeleteObjectVersion"
      ],
      "Resource": "arn:aws:s3:::wormbucket/*"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"
      },
      "Action": "s3>ListBucket",
      "Resource": "arn:aws:s3:::wormbucket"
    },
    {
      "Effect": "Allow",
      "Principal": {
        "AWS": "arn:aws:iam::95390887230002558202:federated-
group/SomeGroup"
      },
      "Action": "s3:*",
      "Resource": "arn:aws:s3:::wormbucket/*"
    }
  ]
}

```

## 그룹 정책의 예

이 섹션의 예제를 사용하여 그룹에 대한 StorageGRID 액세스 정책을 작성합니다.

그룹 정책은 정책이 연결된 그룹에 대한 액세스 권한을 지정합니다. `Principal`이 정책은 암시적이기 때문에 정책에 요소가 없습니다. 그룹 정책은 테넌트 관리자 또는 API를 사용하여 구성됩니다.

예: 테넌트 관리자를 사용하여 그룹 정책을 설정합니다

테넌트 관리자에서 그룹을 추가하거나 편집할 때 그룹 정책을 선택하여 이 그룹의 구성원이 가질 S3 액세스 권한을 결정할 수 있습니다. 을 ["S3 테넌트에 대한 그룹을 생성합니다"](#)참조하십시오.

- \* S3 액세스 없음 \*: 기본 옵션. 버킷 정책을 통해 액세스 권한이 부여되지 않은 한 이 그룹의 사용자는 S3 리소스에 액세스할 수 없습니다. 이 옵션을 선택하면 루트 사용자만 기본적으로 S3 리소스에 액세스할 수 있습니다.
- \* 읽기 전용 액세스 \*: 이 그룹의 사용자는 S3 리소스에 대한 읽기 전용 액세스 권한을 가집니다. 예를 들어 이 그룹의 사용자는 개체를 나열하고 개체 데이터, 메타데이터 및 태그를 읽을 수 있습니다. 이 옵션을 선택하면 읽기 전용 그룹 정책의 JSON 문자열이 텍스트 상자에 나타납니다. 이 문자열을 편집할 수 없습니다.
- \* 전체 액세스 \*: 이 그룹의 사용자는 버킷을 포함하여 S3 리소스에 대한 모든 액세스 권한을 가집니다. 이 옵션을 선택하면 전체 액세스 그룹 정책의 JSON 문자열이 텍스트 상자에 나타납니다. 이 문자열을 편집할 수 없습니다.
- \* 랜섬웨어 완화 \*: 이 샘플 정책은 이 테넌트의 모든 버킷에 적용됩니다. 이 그룹의 사용자는 일반적인 작업을 수행할 수 있지만 개체 버전 관리가 활성화된 버킷에서 개체를 영구적으로 삭제할 수는 없습니다.

모든 버킷 관리 권한이 있는 테넌트 관리자 사용자는 이 그룹 정책을 재정의할 수 있습니다. 모든 버킷 관리 권한을 신뢰할 수 있는 사용자로 제한하고 가능한 경우 MFA(Multi-Factor Authentication)를 사용합니다.

- \* 사용자 정의 \*: 그룹의 사용자에게는 텍스트 상자에 지정한 사용 권한이 부여됩니다.

예: 모든 버킷에 대한 그룹 전체 액세스 허용

이 예에서 그룹의 모든 구성원은 버킷 정책에 의해 명시적으로 거부되지 않는 한 테넌트 계정이 소유한 모든 버킷에 대해 전체 액세스가 허용됩니다.

```
{
  "Statement": [
    {
      "Action": "s3:*",
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::/*"
    }
  ]
}
```

예: 모든 버킷에 대한 그룹 읽기 전용 액세스를 허용합니다

이 예제에서 그룹의 모든 구성원은 버킷 정책에 의해 명시적으로 거부되지 않는 한 S3 리소스에 대해 읽기 전용 액세스 권한을 갖습니다. 예를 들어 이 그룹의 사용자는 개체를 나열하고 개체 데이터, 메타데이터 및 태그를 읽을 수 있습니다.

```
{  
  "Statement": [  
    {  
      "Sid": "AllowGroupReadOnlyAccess",  
      "Effect": "Allow",  
      "Action": [  
        "s3>ListAllMyBuckets",  
        "s3>ListBucket",  
        "s3>ListBucketVersions",  
        "s3>GetObject",  
        "s3>GetObjectTagging",  
        "s3>GetObjectVersion",  
        "s3>GetObjectVersionTagging"  
      ],  
      "Resource": "arn:aws:s3:::/*"  
    }  
  ]  
}
```

예: 그룹 구성원이 버킷의 "폴더"에만 모든 권한을 부여할 수 있습니다

이 예제에서 그룹의 구성원은 지정된 버킷의 특정 폴더(키 접두사)를 나열하고 액세스할 수만 있습니다. 이러한 폴더의 개인 정보를 확인할 때는 다른 그룹 정책 및 버킷 정책의 액세스 권한을 고려해야 합니다.

```

{
  "Statement": [
    {
      "Sid": "AllowListBucketOfASpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3>ListBucket",
      "Resource": "arn:aws:s3:::department-bucket",
      "Condition": {
        "StringLike": {
          "s3:prefix": "${aws:username}/*"
        }
      }
    },
    {
      "Sid": "AllowUserSpecificActionsOnlyInTheSpecificUserPrefix",
      "Effect": "Allow",
      "Action": "s3:*Object",
      "Resource": "arn:aws:s3:::department-bucket/${aws:username}/*"
    }
  ]
}

```

## 감사 로그에서 **S3** 작업을 추적했습니다

감사 메시지는 StorageGRID 서비스에서 생성되고 텍스트 로그 파일에 저장됩니다. 감사 로그에서 S3별 감사 메시지를 검토하여 버킷 및 오브젝트 작업에 대한 세부 정보를 확인할 수 있습니다.

## 감사 로그에서 버킷 작업을 추적했습니다

- CreateBucket
- 삭제 버킷
- 삭제 BucketTagging
- DeleteObjects 를 클릭합니다
- GetBucketTagging
- 머리버킷
- ListObjects 를 선택합니다
- ListObjectVersions 를 선택합니다
- 버킷 규정 준수
- BucketTagging
- PutBucketVersioning을 참조하십시오

감사 로그에서 추적된 객체 작업입니다

- CompleteMultipartUpload를 클릭합니다
- CopyObject 를 선택합니다
- DeleteObject 를 클릭합니다
- GetObject 를 참조하십시오
- HeadObject 를 선택합니다
- PutObject 를 선택합니다
- RestoreObject 를 선택합니다
- 객체 를 선택합니다
- UploadPart(ILM 규칙이 Balanced 또는 Strict 수집을 사용하는 경우)
- UploadPartCopy(ILM 규칙이 Balanced 또는 Strict 수집을 사용하는 경우)

관련 정보

- ["감사 로그 파일에 액세스합니다"](#)
- ["클라이언트가 감사 메시지를 기록합니다"](#)
- ["클라이언트가 감사 메시지를 읽습니다"](#)

## Swift REST API 사용(지원 종료)

**Swift REST API**를 사용합니다

Swift API에 대한 지원이 중단되었으며 향후 릴리즈에서 제거될 예정입니다.



이 버전의 문서 사이트에서 Swift 세부 정보가 제거되었습니다. 을 ["StorageGRID 11.8: Swift REST API 사용"](#)참조하십시오.

## 저작권 정보

Copyright © 2026 NetApp, Inc. All Rights Reserved. 미국에서 인쇄됨 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그레픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이센스와 고지사항이 적용됩니다.

본 소프트웨어는 NetApp에 의해 '있는 그대로' 제공되며 상품성 및 특정 목적에의 적합성에 대한 명시적 또는 묵시적 보증을 포함하여(이에 제한되지 않음) 어떠한 보증도 하지 않습니다. NetApp은 대체품 또는 대체 서비스의 조달, 사용 불능, 데이터 손실, 이익 손실, 영업 중단을 포함하여(이에 국한되지 않음), 이 소프트웨어의 사용으로 인해 발생하는 모든 직접 및 간접 손해, 우발적 손해, 특별 손해, 징벌적 손해, 결과적 손해의 발생에 대하여 그 발생 이유, 책임론, 계약 여부, 엄격한 책임, 불법 행위(과실 또는 그렇지 않은 경우)와 관계없이 어떠한 책임도 지지 않으며, 이와 같은 손실의 발생 가능성이 통지되었다 하더라도 마찬가지입니다.

NetApp은 본 문서에 설명된 제품을 언제든지 예고 없이 변경할 권리를 보유합니다. NetApp은 NetApp의 명시적인 서면 동의를 받은 경우를 제외하고 본 문서에 설명된 제품을 사용하여 발생하는 어떠한 문제에도 책임을 지지 않습니다. 본 제품의 사용 또는 구매의 경우 NetApp에서는 어떠한 특허권, 상표권 또는 기타 지적 재산권이 적용되는 라이센스도 제공하지 않습니다.

본 설명서에 설명된 제품은 하나 이상의 미국 특허, 해외 특허 또는 출원 중인 특허로 보호됩니다.

제한적 권리 표시: 정부에 의한 사용, 복제 또는 공개에는 DFARS 252.227-7013(2014년 2월) 및 FAR 52.227-19(2007년 12월)의 기술 데이터-비상업적 품목에 대한 권리(Rights in Technical Data -Noncommercial Items) 조항의 하위 조항 (b)(3)에 설명된 제한사항이 적용됩니다.

여기에 포함된 데이터는 상업용 제품 및/또는 상업용 서비스(FAR 2.101에 정의)에 해당하며 NetApp, Inc.의 독점 자산입니다. 본 계약에 따라 제공되는 모든 NetApp 기술 데이터 및 컴퓨터 소프트웨어는 본질적으로 상업용이며 개인 비용만으로 개발되었습니다. 미국 정부는 데이터가 제공된 미국 계약과 관련하여 해당 계약을 지원하는 데에만 데이터에 대한 전 세계적으로 비독점적이고 양도할 수 없으며 재사용이 불가능하며 취소 불가능한 라이센스를 제한적으로 가집니다. 여기에 제공된 경우를 제외하고 NetApp, Inc.의 사전 서면 승인 없이는 이 데이터를 사용, 공개, 재생산, 수정, 수행 또는 표시할 수 없습니다. 미국 국방부에 대한 정부 라이센스는 DFARS 조항 252.227-7015(b)(2014년 2월)에 명시된 권한으로 제한됩니다.

## 상표 정보

NETAPP, NETAPP 로고 및 <http://www.netapp.com/TM>에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.