



API 사용

StorageGRID software

NetApp
July 23, 2026

목차

API 사용	1
StorageGRID 그리드 관리 API를 사용하십시오	1
최상위 리소스	1
API 요청 발행	1
StorageGRID의 Grid Management API 작업	4
StorageGRID의 Grid Management API 버전 관리	5
현재 릴리스에서 지원되는 API 버전 확인	6
요청에 사용할 API 버전 지정	7
StorageGRID에서 CSRF(교차 사이트 요청 위조) 방지	7
싱글 사인온(SSO)이 활성화된 경우 API를 사용합니다.	8
SSO(Single Sign-On)가 활성화된 경우(Active Directory) StorageGRID API를 사용하십시오	8
싱글 사인온이 활성화된 경우(Entra ID) StorageGRID API를 사용하십시오	14
SSO(Single Sign-On)가 활성화된 경우(PingFederate) StorageGRID API를 사용합니다	16
API를 사용하여 StorageGRID 기능을 비활성화합니다	21
비활성화된 기능 다시 활성화	22

API 사용

StorageGRID 그리드 관리 API를 사용하십시오

Grid Manager 사용자 인터페이스 대신 Grid Management REST API를 사용하여 시스템 관리 작업을 수행할 수 있습니다. 예를 들어, API를 사용하여 작업을 자동화하거나 사용자 등의 여러 엔티티를 더 빠르게 생성할 수 있습니다.

최상위 리소스

그리드 관리 API는 다음과 같은 최상위 리소스를 제공합니다.

- `/grid`: 액세스는 Grid Manager 사용자로 제한되며 구성된 그룹 권한을 기반으로 합니다.
- `/org`: 액세스는 테넌트 계정에 대한 로컬 또는 페더레이션 LDAP 그룹에 속한 사용자로 제한됩니다. 자세한 내용은 ["테넌트 계정 사용"](#)을 참조하십시오.
- `/private`: 액세스는 Grid Manager 사용자로 제한되며 구성된 그룹 권한을 기반으로 합니다. 비공개 API는 예고 없이 변경될 수 있습니다. StorageGRID 비공개 엔드포인트는 요청의 API 버전도 무시합니다.

API 요청 발행

Grid Management API는 오픈 소스 API 플랫폼인 Swagger를 사용합니다. Swagger는 개발자와 비개발자 모두 API를 통해 StorageGRID에서 실시간 작업을 수행할 수 있도록 직관적인 사용자 인터페이스를 제공합니다.

Swagger 사용자 인터페이스는 각 API 작업에 대한 완전한 세부 정보와 문서를 제공합니다.

시작하기 전에

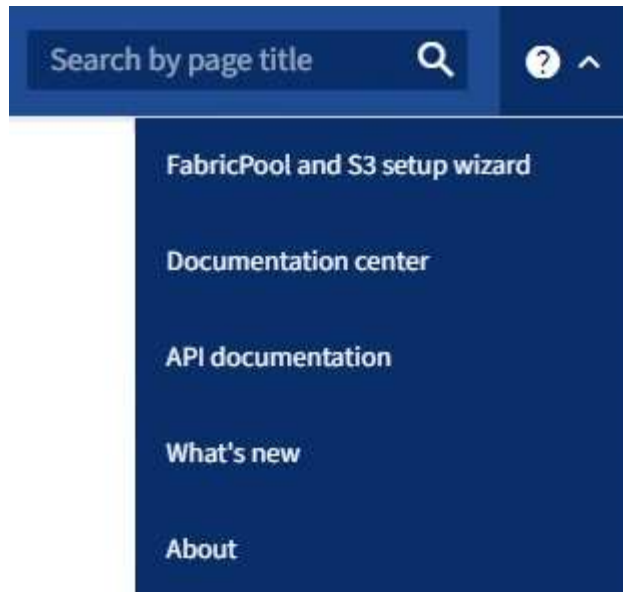
- 현재 Grid Manager에 ["지원되는 웹 브라우저"](#)을(를) 사용하여 로그인되어 있습니다.
- ["특정 액세스 권한"](#)이(가) 있습니다.



API 문서 웹페이지를 사용하여 수행하는 모든 API 작업은 실시간 작업입니다. 실수로 구성 데이터 또는 기타 데이터를 생성, 업데이트 또는 삭제하지 않도록 주의하십시오.

단계

1. 그리드 관리자 헤더에서 도움말 아이콘을 선택한 다음 ***API 문서***를 선택합니다.



2. 비공개 API를 사용하여 작업을 수행하려면 StorageGRID 관리 API 페이지에서 *비공개 API 문서로 이동*을 선택하십시오.

비공개 API는 예고 없이 변경될 수 있습니다. StorageGRID 비공개 엔드포인트는 요청의 API 버전을 무시합니다.

3. 원하는 작업을 선택하십시오.

API 작업을 확장하면 GET, PUT, UPDATE, DELETE와 같은 사용 가능한 HTTP 작업을 확인할 수 있습니다.

4. HTTP 작업을 선택하면 엔드포인트 URL, 필수 또는 선택적 매개변수 목록, 요청 본문 예시(필요한 경우) 및 가능한 응답을 포함한 요청 세부 정보를 볼 수 있습니다.

GET
/grid/groups
Lists Grid Administrator Groups

Parameters
Try it out

Name	Description
type string (query)	filter by group type Available values : local, federated --
limit integer (query)	maximum number of results Default value : 25 25
marker string (query)	marker-style pagination offset (value is Group's URN) marker - marker-style pagination offset (value
includeMarker boolean (query)	if set, the marker element is also returned --
order string (query)	pagination order (desc requires marker) Available values : asc, desc --

Responses
Response content type application/json

Code	Description
200	successfully retrieved Example Value Model <pre>{ "responseTime": "2021-03-29T14:22:19.673Z", "status": "success", "apiVersion": "3.3", "deprecated": false, "data": [{ "displayName": "Developers",</pre>

- 요청에 그룹 또는 사용자 ID와 같은 추가 매개변수가 필요한지 확인합니다. 그런 다음 이러한 값을 가져옵니다. 필요한 정보를 얻기 위해 다른 API 요청을 먼저 수행해야 할 수도 있습니다.
- 예제 요청 본문을 수정해야 하는지 판단하세요. 수정해야 하는 경우, *모델*을 선택하여 각 필드에 대한 요구 사항을 확인할 수 있습니다.
- *직접 사용해보기*를 선택합니다.
- 필요한 매개변수를 제공하거나 필요에 따라 요청 본문을 수정하십시오.
- *실행*을 선택합니다.
- 요청이 성공했는지 여부를 확인하려면 응답 코드를 검토하십시오.

StorageGRID의 Grid Management API 작업

그리드 관리 API는 사용 가능한 작업을 다음 섹션으로 구성합니다.



이 목록에는 공개 API에서 사용할 수 있는 작업만 포함되어 있습니다.

- 계정: 스토리지 테넌트 계정을 관리하는 작업으로, 새 계정 생성 및 특정 계정의 스토리지 사용량 조회 등이 포함됩니다.
- **alert-history**: 해결된 경고에 대한 작업 내역입니다.
- **alert-receivers**: 알림 수신자(이메일)에 대한 작업입니다.
- **alert-rules**: 경고 규칙에 대한 작업.
- **alert-silences**: 경고 해제 관련 작업.
- **alerts**: 알림에 대한 작업.
- 감사: 감사 구성을 나열하고 업데이트하는 작업입니다.
- **auth**: 사용자 세션 인증을 수행하는 작업입니다.

그리드 관리 API는 베어러 토큰 인증 방식을 지원합니다. 로그인하려면 인증 요청의 JSON 본문에 사용자 이름과 비밀번호를 입력해야 합니다(예: `POST /api/v3/authorize`). 사용자 인증에 성공하면 보안 토큰이 반환됩니다. 이 토큰은 이후 API 요청의 헤더에 반드시 포함해야 합니다("Authorization: Bearer *token*"). 토큰은 16시간 후에 만료됩니다.



StorageGRID 시스템에서 싱글 사인온(SSO)이 활성화된 경우 인증을 위해 다른 단계를 수행해야 합니다. "싱글 사인온이 활성화된 경우 API 인증"을 참조하십시오.

인증 보안 개선에 대한 자세한 내용은 "교차 사이트 요청 위조 방지"를 참조하십시오.

- **client-certificates**: 외부 모니터링 도구를 사용하여 StorageGRID에 안전하게 액세스할 수 있도록 클라이언트 인증서를 구성하는 작업입니다.
- **config**: 제품 릴리스 및 그리드 관리 API 버전과 관련된 작업입니다. 제품 릴리스 버전과 해당 릴리스에서 지원하는 그리드 관리 API의 주요 버전을 나열할 수 있으며, 더 이상 사용되지 않는 API 버전을 비활성화할 수 있습니다.
- **deactivated-features**: 비활성화되었을 수 있는 기능을 확인하기 위한 작업입니다.
- **dns-servers**: 구성된 외부 DNS 서버를 나열하고 변경하는 작업입니다.
- **drive-details**: 특정 스토리지 어플라이언스 모델의 드라이브에 대한 작업입니다.
- **endpoint-domain-names**: S3 엔드포인트 도메인 이름을 나열하고 변경하는 작업입니다.
- **erasure-coding**: 이레이저 코딩 프로파일에 대한 작업.
- 확장: 확장에 대한 작업(절차 수준).
- **expansion-nodes**: 확장에 대한 작업(노드 수준).
- **expansion-sites**: 확장(사이트 수준)에 대한 작업.
- **grid-networks**: 그리드 네트워크 목록을 표시하고 변경하는 작업입니다.
- **grid-passwords**: 그리드 암호 관리를 위한 작업입니다.

- 그룹: 로컬 그리드 관리자 그룹을 관리하고 외부 LDAP 서버에서 페더레이션된 그리드 관리자 그룹을 검색하는 작업입니다.
- **identity-source**: 외부 ID 소스를 구성하고 페더레이션된 그룹 및 사용자 정보를 수동으로 동기화하는 작업입니다.
- **ilm**: 정보 라이프사이클 관리(ILM) 관련 운영.
- **in-progress-procedures**: 현재 진행 중인 유지 관리 절차를 검색합니다.
- 라이선스: StorageGRID 라이선스를 검색하고 업데이트하는 작업입니다.
- 로그: 로그 파일을 수집하고 다운로드하는 작업.v
- 메트릭: StorageGRID 메트릭에 대한 작업으로, 특정 시점의 메트릭 쿼리와 특정 기간 동안의 메트릭 쿼리를 포함합니다. Grid Management API는 백엔드 데이터 소스로 Prometheus 시스템 모니터링 도구를 사용합니다. Prometheus 쿼리 구성에 대한 자세한 내용은 Prometheus 웹 사이트를 참조하십시오.



이름에 *private*이(가) 포함된 메트릭은 내부 용도로만 사용됩니다. 이러한 메트릭은 StorageGRID 릴리스 간에 예고 없이 변경될 수 있습니다.

- **node-details**: 노드 세부 정보에 대한 작업입니다.
- **node-health**: 노드 상태에 대한 작업.
- **node-storage-state**: 노드 스토리지 상태에 대한 작업.
- **ntp-servers**: 외부 네트워크 시간 프로토콜(NTP) 서버를 나열하거나 업데이트하는 작업입니다.
- 객체: 객체 및 객체 메타데이터에 대한 작업.
- 복구: 복구 절차를 위한 작업.
- **recovery-package**: 복구 패키지를 다운로드하는 작업입니다.
- **regions**: 지역을 보고 생성하는 작업입니다.
- **s3-object-lock**: 전역 S3 객체 잠금 설정에 대한 작업입니다.
- **server-certificate**: Grid Manager 서버 인증서를 보고 업데이트하는 작업입니다.
- **snmp**: 현재 SNMP 구성에 대한 작업입니다.
- **storage-watermarks**: 스토리지 노드 워터마크.
- **traffic-classes**: 트래픽 분류 정책에 대한 작업입니다.
- **untrusted-client-network**: 신뢰할 수 없는 클라이언트 네트워크 구성에 대한 작업입니다.
- 사용자: Grid Manager 사용자를 보고 관리하는 작업입니다.

StorageGRID의 Grid Management API 버전 관리

그리드 관리 API는 버전 관리를 사용하여 서비스 중단 없는 업그레이드를 지원합니다.

예를 들어, 이 요청 URL은 API 버전 4를 지정합니다.

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

API의 주 버전은 이전 버전과 호환되지 않는 변경 사항이 있을 때 업데이트됩니다. 반대로 API의 부 버전은 이전 버전과 호환되는 변경 사항이 있을 때 업데이트됩니다. 호환되는 변경 사항에는 새로운 엔드포인트 또는 새로운 속성 추가가

포함됩니다.

다음 예시는 변경 유형에 따라 API 버전이 어떻게 증가하는지 보여줍니다.

API 변경 유형	이전 버전	새 버전
이전 버전과 호환됩니다	2.1	2.2
이전 버전과 호환되지 않습니다	2.1	3.0

StorageGRID 소프트웨어를 처음 설치할 때는 최신 버전의 API만 활성화됩니다. 하지만 StorageGRID의 새 기능 릴리스로 업그레이드할 때는 적어도 하나의 StorageGRID 기능 릴리스 동안 이전 버전의 API를 계속 사용할 수 있습니다.



지원되는 버전을 구성할 수 있습니다. "[그리드 관리 API](#)"에 대한 Swagger API 문서의 **config** 섹션을 참조하십시오. 모든 API 클라이언트를 최신 버전으로 업데이트한 후에는 이전 버전에 대한 지원을 비활성화해야 합니다.

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

- 응답 헤더는 "Deprecated: true"입니다.
- JSON 응답 본문에 "deprecated": true가 포함되어 있습니다.
- nms.log 파일에 더 이상 사용되지 않음을 알리는 경고 메시지가 추가됩니다. 예를 들면 다음과 같습니다.

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

현재 릴리스에서 지원되는 API 버전 확인

지원되는 API 주요 버전 목록을 반환하려면 GET /versions API 요청을 사용하십시오. 이 요청은 Swagger API 문서의 **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 사용할 API 버전 지정

API 버전은 경로 매개변수 (/api/v4) 또는 헤더 (Api-Version: 4)를 사용하여 지정할 수 있습니다. 두 값을 모두 제공하는 경우 헤더 값이 경로 값을 재정의합니다.

```
curl https://[IP-Address]/api/v4/grid/accounts

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

StorageGRID에서 CSRF(교차 사이트 요청 위조) 방지

쿠키를 사용하는 인증 방식을 강화하기 위해 CSRF 토큰을 사용하면 StorageGRID에 대한 교차 사이트 요청 위조(CSRF) 공격을 방지하는 데 도움이 됩니다. Grid Manager와 Tenant Manager는 이 보안 기능을 자동으로 활성화하며, 다른 API 클라이언트는 로그인 시 활성화 여부를 선택할 수 있습니다.

공격자는 HTTP 폼 POST와 같은 방식으로 다른 사이트에 요청을 보내 로그인한 사용자의 쿠키를 이용해 특정 요청을 발생시킬 수 있습니다.

StorageGRID는 CSRF 토큰을 사용하여 CSRF 공격으로부터 보호합니다. 이 기능이 활성화되면 특정 쿠키의 내용이 특정 헤더 또는 특정 POST 본문 매개변수의 내용과 일치해야 합니다.

이 기능을 활성화하려면 인증 중에 csrfToken 매개변수를 `true`로 설정하십시오. 기본값은 `false`입니다.

```
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"
```

이 값이 true이면 GridCsrfToken 쿠키는 Grid Manager 로그인 시 임의의 값으로 설정되고, AccountCsrfToken 쿠키는 Tenant Manager 로그인 시 임의의 값으로 설정됩니다.

쿠키가 존재하는 경우, 시스템 상태를 변경할 수 있는 모든 요청(POST, PUT, PATCH, DELETE)에는 다음 중 하나가 반드시 포함되어야 합니다.

- The X-Csrf-Token 헤더이며, 헤더의 값은 CSRF 토큰 쿠키의 값으로 설정됩니다.
- 폼 인코딩된 본문을 허용하는 엔드포인트의 경우: A csrfToken 폼 인코딩된 요청 본문 매개변수입니다.

추가 예제 및 자세한 내용은 온라인 API 문서를 참조하십시오.



CSRF 토큰 쿠키가 설정된 요청은 CSRF 공격에 대한 추가적인 보호 조치로 JSON 요청 본문을 예상하는 모든 요청에 대해 "Content-Type: application/json" 헤더를 강제 적용합니다.

싱글 사인온(SSO)이 활성화된 경우 API를 사용합니다.

SSO(Single Sign-On)가 활성화된 경우(Active Directory) StorageGRID API를 사용하십시오

"싱글 사인온(SSO)을 구성하고 활성화했습니다."가 있고 Active Directory를 SSO 공급자로 사용하는 경우, 그리드 관리 API 또는 테넌트 관리 API에 유효한 인증 토큰을 얻기 위해 일련의 API 요청을 실행해야 합니다.

싱글 사인온이 활성화된 경우 API에 로그인합니다.

이 지침은 Active Directory를 SSO ID 공급자로 사용하는 경우에 적용됩니다.

시작하기 전에

- StorageGRID 사용자 그룹에 속한 페더레이션 사용자의 SSO 사용자 이름과 암호를 알고 있습니다.
- 테넌트 관리 API에 액세스하려면 테넌트 계정 ID를 알고 있어야 합니다.

이 작업 정보

인증 토큰을 얻으려면 다음 예시 중 하나를 사용할 수 있습니다.

- 해당 `storagegrid-ssoauth.py` Python 스크립트는 StorageGRID 설치 파일 디렉터리(`./rpms RHEL`, `./debs Ubuntu` 또는 `Debian`, `./vsphere VMware`)에 있습니다.
- curl 요청의 예시 워크플로입니다.

curl 워크플로를 너무 느리게 실행하면 시간 초과 오류가 발생할 수 있습니다. 다음과 같은 오류 메시지가 표시될 수 있습니다. `A valid SubjectConfirmation was not found on this Response.`



예시로 제시된 curl 워크플로는 다른 사용자가 비밀번호를 볼 수 없도록 보호하지 않습니다.

URL 인코딩 문제가 있는 경우 다음과 같은 오류 메시지가 표시될 수 있습니다. `Unsupported SAML version.`

단계

1. 인증 토큰을 얻으려면 다음 방법 중 하나를 선택하십시오.
 - Use the `storagegrid-ssoauth.py` Python 스크립트를 사용합니다. 2단계로 이동합니다.
 - curl 요청을 사용합니다. 3단계로 이동합니다.
2. `storagegrid-ssoauth.py` 스크립트를 사용하려면 Python 인터프리터에 스크립트를 전달하고 실행하십시오.

메시지가 나타나면 다음 인수에 대한 값을 입력하십시오.

- SSO 방식입니다. ADFS 또는 `adfs`를 입력하세요.
- SSO 사용자 이름
- StorageGRID가 설치된 도메인
- StorageGRID의 주소
- 테넌트 관리 API에 액세스하려면 테넌트 계정 ID가 필요합니다.

```
python3 storagegrid-ssoauth.py
sso_method: adfs
saml_user: my-sso-username
saml_domain: my-domain
sg_address: storagegrid.example.com
tenant_account_id: 12345
Enter the user's SAML password:
*****

*****
StorageGRID Auth Token: 56eb07bf-21f6-40b7-afob-5c6cacfb25e7
```

StorageGRID 인증 토큰이 출력 결과에 제공됩니다. 이제 SSO를 사용하지 않을 때 API를 사용하는 방법과 유사하게 이 토큰을 다른 요청에 사용할 수 있습니다.

3. curl 요청을 사용하려면 다음 절차를 따르십시오.

a. 로그인에 필요한 변수를 선언합니다.

```
export SAMLUSER='my-sso-username'
export SAMLPASSWORD='my-password'
export SAMLDOMAIN='my-domain'
export TENANTACCOUNTID='12345'
export STORAGEGRID_ADDRESS='storagegrid.example.com'
export AD_FS_ADDRESS='adfs.example.com'
```



그리드 관리 API에 액세스하려면 0을 `TENANTACCOUNTID`로 사용하십시오.

b. 서명된 인증 URL을 받으려면 `/api/v3/authorize-saml`에 POST 요청을 보내고 응답에서 추가 JSON 인코딩을 제거하십시오.

이 예시는 `TENANTACCOUNTID`에 대한 서명된 인증 URL의 POST 요청을 보여줍니다. 결과는 JSON 인코딩을 제거하기 위해 `python -m json.tool`에 전달됩니다.

```
curl -X POST "https://$STORAGEGRID_ADDRESS/api/v3/authorize-saml" \
-H "accept: application/json" -H "Content-Type: application/json" \
--data "{\"accountId\": \"$TENANTACCOUNTID\"}" | python -m
json.tool
```

이 예시의 응답에는 URL 인코딩된 서명된 URL이 포함되지만, 추가적인 JSON 인코딩 계층은 포함되지 않습니다.

```
{
  "apiVersion": "3.0",
  "data":
  "https://adfs.example.com/adfs/ls/?SAMLRequest=fZHLbsIwEEV%2FJTuv7...
sSl%2BfQ33cvfwA%3D&RelayState=12345",
  "responseTime": "2018-11-06T16:30:23.355Z",
  "status": "success"
}
```

- c. 응답에서 `SAMLRequest`을(를) 저장하여 이후 명령에서 사용하십시오.

```
export SAMLREQUEST='fZHLbsIwEEV%2FJTuv7...sSl%2BfQ33cvfwA%3D'
```

- d. AD FS에서 클라이언트 요청 ID가 포함된 전체 URL을 가져옵니다.

한 가지 방법은 이전 응답에서 받은 URL을 사용하여 로그인 양식을 요청하는 것입니다.

```
curl "https://$AD_FS_ADDRESS/adfs/ls/?SAMLRequest=
$SAMLREQUEST&RelayState=$TENANTACCOUNTID" | grep 'form method="post"
id="loginForm"'
```

응답에는 클라이언트 요청 ID가 포함되어 있습니다.

```
<form method="post" id="loginForm" autocomplete="off"
novalidate="novalidate" onKeyPress="if (event && event.keyCode == 13)
Login.submitLoginRequest();" action="/adfs/ls/?
SAMLRequest=fZHRTToMwFIZfhh...UJikvo77sXPw%3D%3D&RelayState=12345&clie
nt-request-id=00000000-0000-0000-ee02-0080000000de" >
```

- e. 응답에서 클라이언트 요청 ID를 저장합니다.

```
export SAMLREQUESTID='00000000-0000-0000-ee02-0080000000de'
```

- f. 이전 응답의 양식 작업으로 자격 증명을 전송합니다.

```
curl -X POST "https://$AD_FS_ADDRESS
/adfs/ls/?SAMLRequest=$SAMLREQUEST&RelayState=$TENANTACCOUNTID&client
-request-id=$SAMLREQUESTID" \
--data "UserName=$SAMLUSER@$SAMLDOMAIN&Password=
$SAMPLPASSWORD&AuthMethod=FormsAuthentication" --include
```

AD FS는 헤더에 추가 정보를 포함하여 302 리디렉션을 반환합니다.



SSO 시스템에 다단계 인증(MFA)이 활성화된 경우, 폼 제출 시 두 번째 암호 또는 기타 자격 증명도 포함됩니다.

```
HTTP/1.1 302 Found
Content-Length: 0
Content-Type: text/html; charset=utf-8
Location:
https://adfs.example.com/adfs/ls/?SAMLRequest=fZHRTOMwFIZfhb...UJikvo
77sXPw%3D%3D&RelayState=12345&client-request-id=00000000-0000-0000-
ee02-0080000000de
Set-Cookie: MSISAuth=AAEAADAvsHpXk6ApV...pmP0aEiNtJvWY=; path=/adfs;
HttpOnly; Secure
Date: Tue, 06 Nov 2018 16:55:05 GMT
```

g. 응답에서 MSISAuth 쿠키를 저장합니다.

```
export MSISAuth='AAEAADAvsHpXk6ApV...pmP0aEiNtJvWY='
```

h. 인증 POST 요청에서 받은 쿠키를 사용하여 지정된 위치로 GET 요청을 보냅니다.

```
curl "https://$AD_FS_ADDRESS/adfs/ls/?SAMLRequest=
$SAMLREQUEST&RelayState=$TENANTACCOUNTID&client-request-
id=$SAMLREQUESTID" \
--cookie "MSISAuth=$MSISAuth" --include
```

응답 헤더에는 나중에 로그아웃할 때 사용할 AD FS 세션 정보가 포함되고, 응답 본문에는 숨겨진 폼 필드에 SAMLResponse가 포함됩니다.

```

HTTP/1.1 200 OK
Cache-Control: no-cache,no-store
Pragma: no-cache
Content-Length: 5665
Content-Type: text/html; charset=utf-8
Expires: -1
Server: Microsoft-HTTPAPI/2.0
P3P: ADFS doesn't have P3P policy, please contact your site's admin
for more details
Set-Cookie:
SamlSession=a3dpbnRlcnMtUHJpbWFyeS1BZG1pbi0xNzgmRmFsc2Umcng4NnJDZmFKV
XFxVWx3bk1lMnFuUSUzZCUzZCYmJiYmXzE3MjAyZTA5LTNmMDgtNDRkZC04Yzg5LTQ3ND
UxYzA3ZjkzYw==; path=/adfs; HttpOnly; Secure
Set-Cookie: MSISAuthenticated=MTEvNy8yMDE4IDQ6MzI6NTkgUE0=;
path=/adfs; HttpOnly; Secure
Set-Cookie: MSISLoopDetectionCookie=MjAxOC0xMS0wNzoxNjoxMjoxOVpcMQ==;
path=/adfs; HttpOnly; Secure
Date: Wed, 07 Nov 2018 16:32:59 GMT

<form method="POST" name="hiddenform"
action="https://storagegrid.example.com:443/api/saml-response">
  <input type="hidden" name="SAMLResponse"
value="PHNhbWxwOlJlc3Bvb3N...1scDpSZXNwb25zZT4=" /><input
type="hidden" name="RelayState" value="12345" />

```

- i. 숨겨진 필드에서 `SAMLResponse`을(를) 저장합니다.

```
export SAMLResponse='PHNhbWxwOlJlc3Bvb3N...1scDpSZXNwb25zZT4='
```

- j. 저장된 `SAMLResponse`을 사용하여 StorageGRID/api/saml-response 요청을 통해 StorageGRID 인증 토큰을 생성합니다.

`RelayState`의 경우, 테넌트 계정 ID를 사용하거나, 그리드 관리 API에 로그인하려면 0을 사용하십시오.

```

curl -X POST "https://$STORAGEGRID_ADDRESS:443/api/saml-response" \
-H "accept: application/json" \
--data-urlencode "SAMLResponse=$SAMLResponse" \
--data-urlencode "RelayState=$TENANTACCOUNTID" \
| python -m json.tool

```

응답에는 인증 토큰이 포함되어 있습니다.

```
{
  "apiVersion": "3.0",
  "data": "56eb07bf-21f6-40b7-af0b-5c6cacfb25e7",
  "responseTime": "2018-11-07T21:32:53.486Z",
  "status": "success"
}
```

a. 응답에 포함된 인증 토큰을 'MYTOKEN'으로 저장합니다.

```
export MYTOKEN="56eb07bf-21f6-40b7-af0b-5c6cacfb25e7"
```

이제 'MYTOKEN'를 SSO를 사용하지 않을 때 API를 사용하는 방식과 유사하게 다른 요청에도 사용할 수 있습니다.

싱글 사인온(SSO)이 활성화된 경우 **API**에서 로그아웃

싱글 사인온(SSO)이 활성화된 경우, 그리드 관리 API 또는 테넌트 관리 API에서 로그아웃하려면 일련의 API 요청을 실행해야 합니다. 이 지침은 Active Directory를 SSO ID 공급자로 사용하는 경우에 적용됩니다.

이 작업 정보

필요한 경우 조직의 단일 로그아웃 페이지에서 로그아웃하여 StorageGRID API에서 로그아웃할 수 있습니다. 또는 유효한 StorageGRID 베어러 토큰이 필요한 StorageGRID에서 단일 로그아웃(SLO)을 트리거할 수도 있습니다.

단계

1. 서명된 로그아웃 요청을 생성하려면 SLO API에 'cookie "sso=true"'를 전달하세요.

```
curl -k -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \
-H "accept: application/json" \
-H "Authorization: Bearer $MYTOKEN" \
--cookie "sso=true" \
| python -m json.tool
```

로그아웃 URL이 반환됩니다.

```
{
  "apiVersion": "3.0",
  "data":
"https://adfs.example.com/adfs/ls/?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D",
  "responseTime": "2018-11-20T22:20:30.839Z",
  "status": "success"
}
```

2. 로그아웃 URL을 저장합니다.

```
export LOGOUT_REQUEST
='https://adfs.example.com/adfs/ls/?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D'
```

3. SLO를 트리거하고 StorageGRID로 다시 리디렉션하려면 로그아웃 URL에 요청을 보내십시오.

```
curl --include "$LOGOUT_REQUEST"
```

302 응답이 반환됩니다. 리디렉션 위치는 API 전용 로그아웃에는 적용되지 않습니다.

```
HTTP/1.1 302 Found
Location: https://$STORAGEGRID_ADDRESS:443/api/saml-logout?SAMLResponse=fVLLasMwEPwVo7ss%...%23rsa-sha256
Set-Cookie: MSISignoutProtocol=U2FtbA==; expires=Tue, 20 Nov 2018 22:35:03 GMT; path=/adfs; HttpOnly; Secure
```

4. StorageGRID 베어러 토큰을 삭제합니다.

StorageGRID 베어러 토큰 삭제는 SSO를 사용하지 않을 때와 동일한 방식으로 작동합니다. 쿠키 "sso=true"가 제공되지 않으면 SSO 상태에 영향을 주지 않고 사용자가 StorageGRID에서 로그아웃됩니다.

```
curl -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \
-H "accept: application/json" \
-H "Authorization: Bearer $MYTOKEN" \
--include
```

A 204 No Content 응답은 사용자가 로그아웃되었음을 나타냅니다.

```
HTTP/1.1 204 No Content
```

싱글 사인온이 활성화된 경우(Entra ID) StorageGRID API를 사용하십시오

"싱글 사인온(SSO)을 구성하고 활성화했습니다."가 있고 Entra ID를 SSO 공급자로 사용하는 경우, 두 가지 예제 스크립트를 사용하여 Grid Management API 또는 테넌트 관리 API에 유효한 인증 토큰을 얻을 수 있습니다.

Entra ID 싱글 사인온이 활성화된 경우 API에 로그인

이 지침은 Entra ID를 SSO ID 공급자로 사용하는 경우에 적용됩니다.

시작하기 전에

- StorageGRID 사용자 그룹에 속한 페더레이션 사용자의 SSO 이메일 주소와 비밀번호를 알고 있습니다.
- 테넌트 관리 API에 액세스하려면 테넌트 계정 ID를 알고 있어야 합니다.

이 작업 정보

인증 토큰을 얻으려면 다음 예제 스크립트를 사용할 수 있습니다.

- storagegrid-ssoauth-azure.py Python 스크립트
- The storagegrid-ssoauth-azure.js Node.js 스크립트

두 스크립트 모두 StorageGRID 설치 파일 디렉터리(`./rpms`RHEL, `./debs`Ubuntu 또는 Debian, `./vsphere`VMware`)에 있습니다.

Entra ID와의 API 통합을 직접 작성하려면 `storagegrid-ssoauth-azure.py` 스크립트를 참조하세요. 이 Python 스크립트는 StorageGRID에 직접 두 번의 요청(첫 번째는 SAMLRequest를 가져오기 위한 요청, 두 번째는 인증 토큰을 가져오기 위한 요청)을 보내고, Node.js 스크립트를 호출하여 Entra ID와 상호 작용하며 SSO 작업을 수행합니다.

SSO 작업은 일련의 API 요청을 통해 실행할 수 있지만, 그 과정이 간단하지는 않습니다. Puppeteer Node.js 모듈을 사용하여 Entra ID SSO 인터페이스를 스크래핑합니다.

URL 인코딩 문제가 있는 경우 다음과 같은 오류 메시지가 표시될 수 있습니다. `Unsupported SAML version.`

단계

1. 다음과 같이 필요한 종속성을 설치하십시오.

- a. Node.js를 설치합니다(참조 "<https://nodejs.org/en/download/>").
- b. 필요한 Node.js 모듈(puppeteer 및 jsdom)을 설치합니다.

```
npm install -g <module>
```

2. 파이썬 스크립트를 파이썬 인터프리터에 전달하여 스크립트를 실행합니다.

그러면 파이썬 스크립트는 해당 Node.js 스크립트를 호출하여 Entra ID SSO 상호 작용을 수행합니다.

3. 메시지가 표시되면 다음 인수에 대한 값을 입력하거나 매개변수를 사용하여 전달하십시오.

- Entra ID에 로그인하는 데 사용되는 SSO 이메일 주소
- StorageGRID의 주소
- 테넌트 관리 API에 액세스하려는 경우 테넌트 계정 ID

4. 메시지가 나타나면 비밀번호를 입력하고, 요청 시 Entra ID에 대한 다단계 인증(MFA)을 제공할 준비를 하십시오.

```
c:\Users\user\Documents\azure_sso>py storagegrid-azure-ssoauth.py --sso-email-address user@my-domain.com
--sg-address storagegrid.examp.e.com --tenant-account-id 0
Enter the user's SSO password:
*****

Watch for and approve a 2FA authorization request
*****
StorageGRID Auth Token: {'responseTime': '2021-10-04T21:30:48.807Z', 'status': 'success', 'apiVersion':
'3.4', 'data': '4807d93e-a3df-48f2-9680-906cd255979e'}
```



이 스크립트는 Microsoft Authenticator를 사용한 MFA를 가정합니다. 다른 형태의 MFA(예: 문자 메시지로 받은 코드 입력)를 지원하도록 스크립트를 수정해야 할 수도 있습니다.

StorageGRID 인증 토큰이 출력 결과에 제공됩니다. 이제 SSO를 사용하지 않을 때 API를 사용하는 방법과 유사하게 이 토큰을 다른 요청에 사용할 수 있습니다.

SSO(Single Sign-On)가 활성화된 경우(PingFederate) StorageGRID API를 사용합니다

"싱글 사인온(SSO)을 구성하고 활성화했습니다."가 있고 PingFederate를 SSO 제공업체로 사용하는 경우, 그리드 관리 API 또는 테넌트 관리 API에 유효한 인증 토큰을 얻기 위해 일련의 API 요청을 실행해야 합니다.

싱글 사인온이 활성화된 경우 **API**에 로그인합니다.

이 지침은 SSO ID 공급자로 PingFederate를 사용하는 경우에 적용됩니다.

시작하기 전에

- StorageGRID 사용자 그룹에 속한 페더레이션 사용자의 SSO 사용자 이름과 암호를 알고 있습니다.
- 테넌트 관리 API에 액세스하려면 테넌트 계정 ID를 알고 있어야 합니다.

이 작업 정보

인증 토큰을 얻으려면 다음 예시 중 하나를 사용할 수 있습니다.

- 해당 `storagegrid-ssoauth.py` Python 스크립트는 StorageGRID 설치 파일 디렉터리(`./rpms RHEL`, `./debs Ubuntu` 또는 `Debian`, `./vsphere VMware`)에 있습니다.
- `curl` 요청의 예시 워크플로입니다.

`curl` 워크플로를 너무 느리게 실행하면 시간 초과 오류가 발생할 수 있습니다. 다음과 같은 오류 메시지가 표시될 수 있습니다. `A valid SubjectConfirmation was not found on this Response.`



예시로 제시된 `curl` 워크플로는 다른 사용자가 비밀번호를 볼 수 없도록 보호하지 않습니다.

URL 인코딩 문제가 있는 경우 다음과 같은 오류 메시지가 표시될 수 있습니다. `Unsupported SAML version.`

단계

1. 인증 토큰을 얻으려면 다음 방법 중 하나를 선택하십시오.
 - Use the `storagegrid-ssoauth.py` Python 스크립트를 사용합니다. 2단계로 이동합니다.
 - `curl` 요청을 사용합니다. 3단계로 이동합니다.
2. `storagegrid-ssoauth.py` 스크립트를 사용하려면 Python 인터프리터에 스크립트를 전달하고 실행하십시오.

메시지가 나타나면 다음 인수에 대한 값을 입력하십시오.

- SSO 방식입니다. "pingfederate"의 어떤 변형이든 입력할 수 있습니다(PINGFEDERATE, pingfederate 등).
- SSO 사용자 이름

- StorageGRID가 설치된 도메인입니다. 이 필드는 PingFederate에 사용되지 않습니다. 비워두거나 임의의 값을 입력할 수 있습니다.
- StorageGRID의 주소
- 테넌트 관리 API에 액세스하려면 테넌트 계정 ID가 필요합니다.

```
python3 storagegrid-ssoauth.py
sso_method: pingfederate
saml_user: my-sso-username
saml_domain:
sg_address: storagegrid.example.com
tenant_account_id: 12345
Enter the user's SAML password:
*****

*****
StorageGRID Auth Token: 56eb07bf-21f6-40b7-afob-5c6cacfb25e7
```

StorageGRID 인증 토큰이 출력 결과에 제공됩니다. 이제 SSO를 사용하지 않을 때 API를 사용하는 방법과 유사하게 이 토큰을 다른 요청에 사용할 수 있습니다.

3. curl 요청을 사용하려면 다음 절차를 따르십시오.

a. 로그인에 필요한 변수를 선언합니다.

```
export SAMLUSER='my-sso-username'
export SAMLPASSWORD='my-password'
export TENANTACCOUNTID='12345'
export STORAGEGRID_ADDRESS='storagegrid.example.com'
```



그리드 관리 API에 액세스하려면 0을 'TENANTACCOUNTID'로 사용하십시오.

b. 서명된 인증 URL을 받으려면 '/api/v3/authorize-saml'에 POST 요청을 보내고 응답에서 추가 JSON 인코딩을 제거하십시오.

이 예시는 TENANTACCOUNTID에 대한 서명된 인증 URL을 요청하는 POST 요청을 보여줍니다. 결과는 python -m json.tool에 전달되어 JSON 인코딩이 제거됩니다.

```
curl -X POST "https://$STORAGEGRID_ADDRESS/api/v3/authorize-saml" \
-H "accept: application/json" -H "Content-Type: application/json" \
--data "{\"accountId\": \"$TENANTACCOUNTID\"}" | python -m
json.tool
```

이 예시의 응답에는 URL 인코딩된 서명된 URL이 포함되지만, 추가적인 JSON 인코딩 계층은 포함되지 않습니다.

```
{
  "apiVersion": "3.0",
  "data": "https://my-pf-baseurl/idp/SSO.saml2?...",
  "responseTime": "2018-11-06T16:30:23.355Z",
  "status": "success"
}
```

- c. 응답에서 'SAMLRequest'을(를) 저장하여 이후 명령에서 사용하십시오.

```
export SAMLREQUEST="https://my-pf-baseurl/idp/SSO.saml2?..."
```

- d. 응답과 쿠키를 내보내고 응답을 출력합니다.

```
RESPONSE=$(curl -c - "$SAMLREQUEST")
```

```
echo "$RESPONSE" | grep 'input type="hidden" name="pf.adapterId"
id="pf.adapterId"'
```

- e. 'pf.adapterId' 값을 내보내고 응답을 출력합니다.

```
export ADAPTER='myAdapter'
```

```
echo "$RESPONSE" | grep 'base'
```

- f. 'href' 값을 내보내고(끝에 있는 슬래시(/)를 제거), 응답을 출력합니다.

```
export BASEURL='https://my-pf-baseurl'
```

```
echo "$RESPONSE" | grep 'form method="POST"'
```

- g. 'action' 값을 내보냅니다.

```
export SSOPING='/idp/.../resumeSAML20/idp/SSO.ping'
```

- h. 쿠키를 자격 증명과 함께 전송:

```
curl -b <(echo "$RESPONSE") -X POST "$BASEURL$SSOPING" \
--data "pf.username=$SAMLUSER&pf.pass=
$SAMLPASSWORD&pf.ok=clicked&pf.cancel=&pf.adapterId=$ADAPTER"
--include
```

- i. 숨겨진 필드에서 `SAMLResponse`을(를) 저장합니다.

```
export SAMLResponse='PHNhbWxwOlJlc3BvbnN...1scDpSZXNwb25zZT4='
```

- j. 저장된 `SAMLResponse`을 사용하여 StorageGRID/api/saml-response 요청을 통해 StorageGRID 인증 토큰을 생성합니다.

`RelayState`의 경우, 테넌트 계정 ID를 사용하거나, 그리드 관리 API에 로그인하려면 0을 사용하십시오.

```
curl -X POST "https://$STORAGEGRID_ADDRESS:443/api/saml-response" \
-H "accept: application/json" \
--data-urlencode "SAMLResponse=$SAMLResponse" \
--data-urlencode "RelayState=$TENANTACCOUNTID" \
| python -m json.tool
```

응답에는 인증 토큰이 포함되어 있습니다.

```
{
  "apiVersion": "3.0",
  "data": "56eb07bf-21f6-40b7-af0b-5c6cacfb25e7",
  "responseTime": "2018-11-07T21:32:53.486Z",
  "status": "success"
}
```

- a. 응답에 포함된 인증 토큰을 `MYTOKEN`으로 저장합니다.

```
export MYTOKEN="56eb07bf-21f6-40b7-af0b-5c6cacfb25e7"
```

이제 `MYTOKEN`를 SSO를 사용하지 않을 때 API를 사용하는 방식과 유사하게 다른 요청에도 사용할 수 있습니다.

싱글 사인온(SSO)이 활성화된 경우 API에서 로그아웃

싱글 사인온(SSO)이 활성화된 경우, 그리드 관리 API 또는 테넌트 관리 API에서 로그아웃하려면 일련의 API 요청을

실행해야 합니다. 이 지침은 PingFederate를 SSO ID 공급자로 사용하는 경우에 적용됩니다.

이 작업 정보

필요한 경우 조직의 단일 로그아웃 페이지에서 로그아웃하여 StorageGRID API에서 로그아웃할 수 있습니다. 또는 유효한 StorageGRID 베어러 토큰이 필요한 StorageGRID에서 단일 로그아웃(SLO)을 트리거할 수도 있습니다.

단계

1. 서명된 로그아웃 요청을 생성하려면 SLO API에 `cookie "sso=true"`를 전달하세요.

```
curl -k -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \
-H "accept: application/json" \
-H "Authorization: Bearer $MYTOKEN" \
--cookie "sso=true" \
| python -m json.tool
```

로그아웃 URL이 반환됩니다.

```
{
  "apiVersion": "3.0",
  "data": "https://my-ping-
url/ldap/SLO.saml2?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D",
  "responseTime": "2021-10-12T22:20:30.839Z",
  "status": "success"
}
```

2. 로그아웃 URL을 저장합니다.

```
export LOGOUT_REQUEST='https://my-ping-
url/ldap/SLO.saml2?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D'
```

3. SLO를 트리거하고 StorageGRID로 다시 리디렉션하려면 로그아웃 URL에 요청을 보내십시오.

```
curl --include "$LOGOUT_REQUEST"
```

302 응답이 반환됩니다. 리디렉션 위치는 API 전용 로그아웃에는 적용되지 않습니다.

```
HTTP/1.1 302 Found
Location: https://$STORAGEGRID_ADDRESS:443/api/saml-
logout?SAMLResponse=fVLLasMwEPwVo7ss%...%23rsa-sha256
Set-Cookie: PF=QoKs...SgCC; Path=/; Secure; HttpOnly; SameSite=None
```

4. StorageGRID 베어러 토큰을 삭제합니다.

StorageGRID 베어러 토큰 삭제는 SSO를 사용하지 않을 때와 동일한 방식으로 작동합니다. 쿠키 ``sso=true``가 제공되지 않으면 SSO 상태에 영향을 주지 않고 사용자가 StorageGRID에서 로그아웃됩니다.

```
curl -X DELETE "https://$STORAGEGRID_ADDRESS/api/v3/authorize" \
-H "accept: application/json" \
-H "Authorization: Bearer $MYTOKEN" \
--include
```

A 204 No Content 응답은 사용자가 로그아웃되었음을 나타냅니다.

```
HTTP/1.1 204 No Content
```

API를 사용하여 StorageGRID 기능을 비활성화합니다

StorageGRID 시스템에서 특정 기능을 완전히 비활성화하려면 그리드 관리 API를 사용할 수 있습니다. 기능이 비활성화되면 해당 기능과 관련된 작업을 수행할 수 있는 권한을 누구에게도 부여할 수 없습니다.

이 작업 정보

비활성화된 기능 시스템을 사용하면 StorageGRID 시스템의 특정 기능에 대한 액세스를 차단할 수 있습니다. 기능을 비활성화하는 것은 루트 액세스 권한이 있는 관리자 그룹에 속한 루트 사용자 또는 사용자가 해당 기능을 사용하지 못하도록 하는 유일한 방법입니다.

이 기능이 어떻게 유용할 수 있는지 이해하려면 다음 시나리오를 고려하십시오.

A사는 테넌트 계정을 생성하여 StorageGRID 시스템의 스토리지 용량을 임대하는 서비스 제공업체입니다. A사는 임대 고객의 오브젝트 보안을 보호하기 위해 테넌트 계정 배포 후에는 자사 직원이 어떠한 테넌트 계정에도 액세스할 수 없도록 보장하고자 합니다.

A사는 그리드 관리 API의 기능 비활성화 시스템을 사용하여 이 목표를 달성할 수 있습니다. 그리드 관리자(UI 및 API 모두)에서 테넌트 루트 암호 변경 기능을 완전히 비활성화함으로써, A사는 루트 사용자 및 루트 액세스 권한이 있는 그룹에 속한 사용자를 포함한 관리자 사용자가 어떤 테넌트 계정의 루트 사용자 암호도 변경할 수 없도록 합니다.

단계

1. Grid Management API에 대한 Swagger 문서에 액세스합니다. ["그리드 관리 API 사용"](#)을 참조하십시오.
2. 기능 비활성화 엔드포인트를 찾으십시오.
3. 테넌트 루트 암호 변경과 같은 기능을 비활성화하려면 다음과 같은 본문을 API에 전송하십시오.

```
{ "grid": { "changeTenantRootPassword": true} }
```

요청이 완료되면 테넌트 루트 암호 변경 기능이 비활성화됩니다. 테넌트 루트 암호 변경 관리 권한은 더 이상 사용자 인터페이스에 표시되지 않으며, 테넌트의 루트 암호를 변경하려는 API 요청은 "403 Forbidden" 오류와 함께 실패합니다.

비활성화된 기능 다시 활성화

기본적으로 그리드 관리 API를 사용하여 비활성화된 기능을 다시 활성화할 수 있습니다. 하지만 비활성화된 기능이 다시 활성화되지 않도록 하려면 **activateFeatures** 기능 자체를 비활성화하면 됩니다.



activateFeatures 기능은 다시 활성화할 수 없습니다. 이 기능을 비활성화하면 비활성화된 다른 모든 기능을 영구적으로 다시 활성화할 수 없게 됩니다. 손실된 기능을 복구하려면 기술 지원팀에 문의해야 합니다.

단계

1. 그리드 관리 API에 대한 Swagger 문서를 참조하십시오.
2. 기능 비활성화 엔드포인트를 찾으십시오.
3. 모든 기능을 다시 활성화하려면 다음과 같은 본문을 API로 전송하세요.

```
{ "grid": null }
```

이 요청이 완료되면 테넌트 루트 암호 변경 기능을 포함한 모든 기능이 다시 활성화됩니다. 테넌트 루트 암호 변경 관리 권한이 이제 사용자 인터페이스에 표시되며, 사용자가 루트 액세스 또는 테넌트 루트 암호 변경 관리 권한을 가지고 있는 경우 테넌트의 루트 암호를 변경하려는 모든 API 요청이 성공적으로 처리됩니다.



위 예시는 비활성화된 모든 기능을 다시 활성화합니다. 비활성화된 상태로 유지되어야 하는 다른 기능이 있는 경우, PUT 요청에 해당 기능을 명시적으로 지정해야 합니다. 예를 들어, 테넌트 루트 암호 변경 기능을 다시 활성화하고 storageAdmin 관리 권한은 계속 비활성화하려면 다음과 같은 PUT 요청을 보내십시오.

```
{ "grid": {"storageAdmin": true} }
```

저작권 정보

Copyright © 2026 NetApp, Inc. All Rights Reserved. 미국에서 인쇄된 본 문서의 어떠한 부분도 저작권 소유자의 사전 서면 승인 없이는 어떠한 형식이나 수단(복사, 녹음, 녹화 또는 전자 검색 시스템에 저장하는 것을 비롯한 그래픽, 전자적 또는 기계적 방법)으로도 복제될 수 없습니다.

NetApp이 저작권을 가진 자료에 있는 소프트웨어에는 아래의 라이선스와 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 라이선스도 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 라이선스를 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 라이선스는 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.