Grid Manager를 시작합니다

StorageGRID

NetApp
November 04, 2025

This PDF was generated from https://docs.netapp.com/ko-kr/storagegrid-118/admin/web-browser-requirements.html on November 04, 2025. Always check docs.netapp.com for the latest.

목차

Grid Manager를 시작합니다 . . . 1
- 웹 브라우저 요구 사항 . . . 1
- Grid Manager에 로그인합니다 . . . 1
  - 개요 . . . 1
  - 첫 번째 관리 노드에서 그리드 관리자에 로그인합니다 . . . 2
  - 다른 관리자 노드에 로그인합니다 . . . 6
- Grid Manager에서 로그아웃합니다 . . . 7
- 암호를 변경합니다 . . . 7
- StorageGRID 라이센스 정보를 봅니다 . . . 8
- StorageGRID 라이센스 정보를 업데이트합니다 . . . 9
- API를 사용합니다 . . . 9
  - Grid Management API를 사용합니다 . . . 9
  - Grid Management API 작업 . . . 12
  - Grid Management API 버전 관리 . . . 14
  - 사이트 간 요청 위조(CSRF)로부터 보호 . . . 15
  - SSO(Single Sign-On)가 활성화된 경우 API를 사용합니다 . . . 16
  - API를 사용하여 기능을 비활성화합니다 . . . 29

Grid Manager를 시작합니다

웹 브라우저 요구 사항

지원되는 웹 브라우저를 사용해야 합니다.

웹 브라우저	최소 지원 버전
Google Chrome	119
Microsoft Edge를 참조하십시오	119
Mozilla Firefox	119

브라우저 창을 권장 너비로 설정해야 합니다.

브라우저 폭	픽셀
최소	1024
최적	1280입니다

Grid Manager에 로그인합니다

지원되는 웹 브라우저의 주소 표시줄에 FQDN(정규화된 도메인 이름) 또는 관리 노드의 IP 주소를 입력하여 Grid Manager 로그인 페이지에 액세스합니다.

개요

각 StorageGRID 시스템에는 1개의 기본 관리 노드와 1차 관리자가 아닌 노드 수가 포함되어 있습니다. 관리자 노드의 그리드 관리자에 로그인하여 StorageGRID 시스템을 관리할 수 있습니다. 그러나 관리 노드는 정확히 동일하지 않습니다.

- 한 관리 노드에서 이루어진 알람 승인(레거시 시스템)은 다른 관리 노드에 복사되지 않습니다. 이러한 이유로 알람에 대해 표시되는 정보는 각 관리 노드에서 동일하지 않을 수 있습니다.
- 일부 유지 보수 절차는 기본 관리 노드에서만 수행할 수 있습니다.

HA 그룹에 연결합니다

HA(고가용성) 그룹에 관리 노드가 포함된 경우 HA 그룹의 가상 IP 주소 또는 가상 IP 주소에 매핑되는 정규화된 도메인 이름을 사용하여 연결합니다. 기본 관리 노드를 그룹의 기본 인터페이스로 선택해야 그리드 관리자에 액세스할 때 기본 관리 노드를 사용할 수 없는 경우를 제외하고 기본 관리 노드에서 액세스할 수 있습니다. 을 참조하십시오 "고가용성 그룹을 관리합니다".

SSO를 사용합니다

로그인 단계는 에 따라 약간 다릅니다 "SSO(Single Sign-On)가 구성되었습니다".

첫 번째 관리 노드에서 그리드 관리자에 로그인합니다

시작하기 전에

- 로그인 자격 증명이 있습니다.
- 을(를) 사용하고 있습니다 "지원되는 웹 브라우저".
- 쿠키는 웹 브라우저에서 활성화됩니다.
- 하나 이상의 권한이 있는 사용자 그룹에 속해 있습니다.
- Grid Manager에 대한 URL이 있습니다.

  `https://FQDN_or_Admin_Node_IP/`

  정규화된 도메인 이름, 관리자 노드의 IP 주소 또는 관리자 노드의 HA 그룹의 가상 IP 주소를 사용할 수 있습니다.

  HTTPS의 기본 포트(443)가 아닌 포트에서 Grid Manager에 액세스하려면 URL에 포트 번호를 포함시킵니다.

  `https://FQDN_or_Admin_Node_IP:port/`

  > SSO는 제한된 Grid Manager 포트에서 사용할 수 없습니다. 포트 443을 사용해야 합니다.

단계

1. 지원되는 웹 브라우저를 실행합니다.
2. 브라우저의 주소 표시줄에 Grid Manager의 URL을 입력합니다.
3. 보안 경고 메시지가 나타나면 브라우저의 설치 마법사를 사용하여 인증서를 설치합니다. 을 참조하십시오 "보안 인증서를 관리합니다".
4. Grid Manager에 로그인합니다.

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

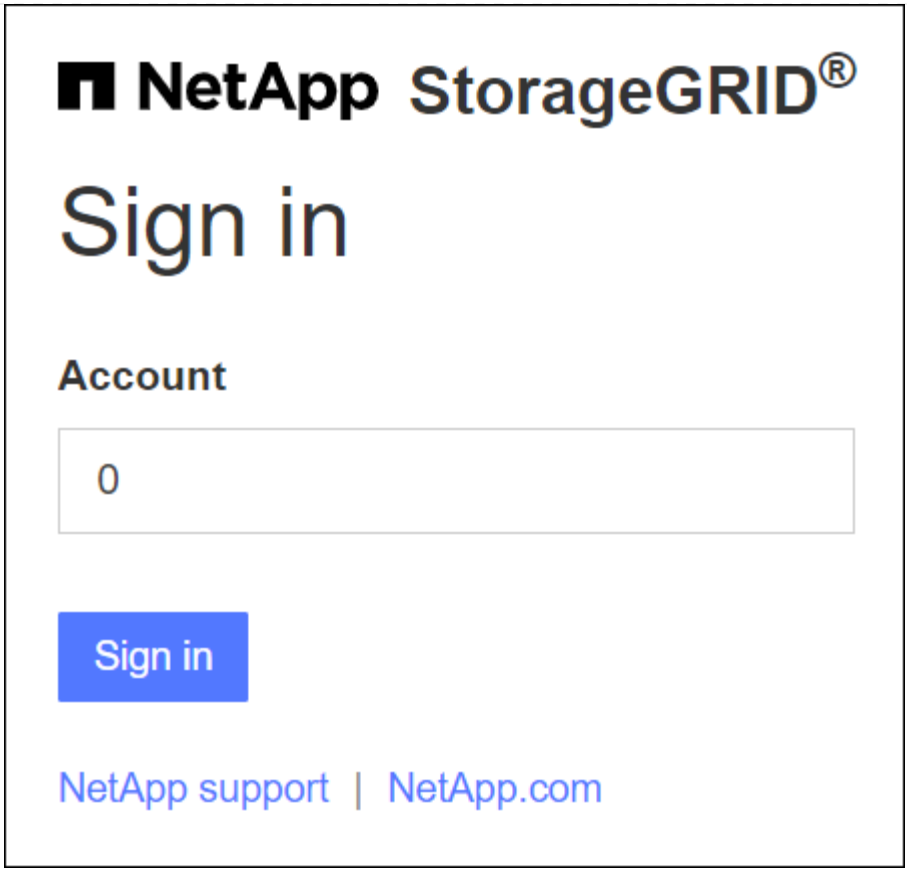
SSO를 사용하지 않습니다

a. Grid Manager의 사용자 이름과 암호를 입력합니다.

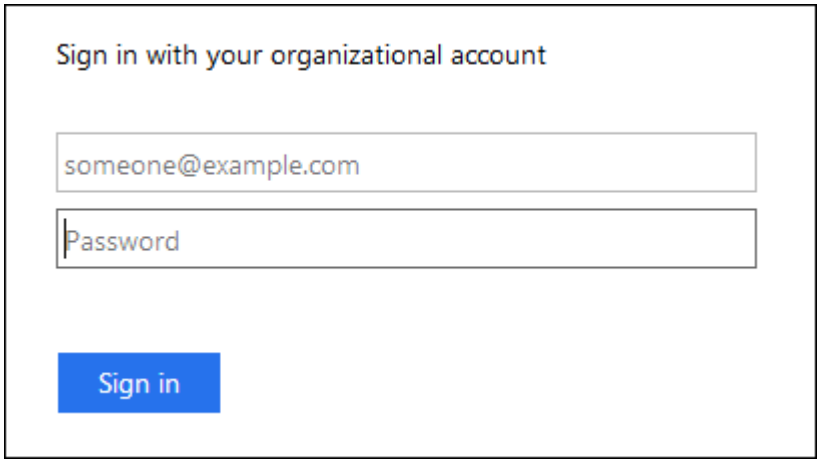
b. 로그인 * 을 선택합니다.

SSO 사용

- StorageGRID가 SSO를 사용하고 있고 이 브라우저에서 URL에 처음 액세스한 경우:
  i. 로그인 * 을 선택합니다. 계정 필드에 0을 그대로 둘 수 있습니다.



ii. 조직의 SSO 로그인 페이지에 표준 SSO 자격 증명을 입력합니다. 예를 들면 다음과 같습니다.

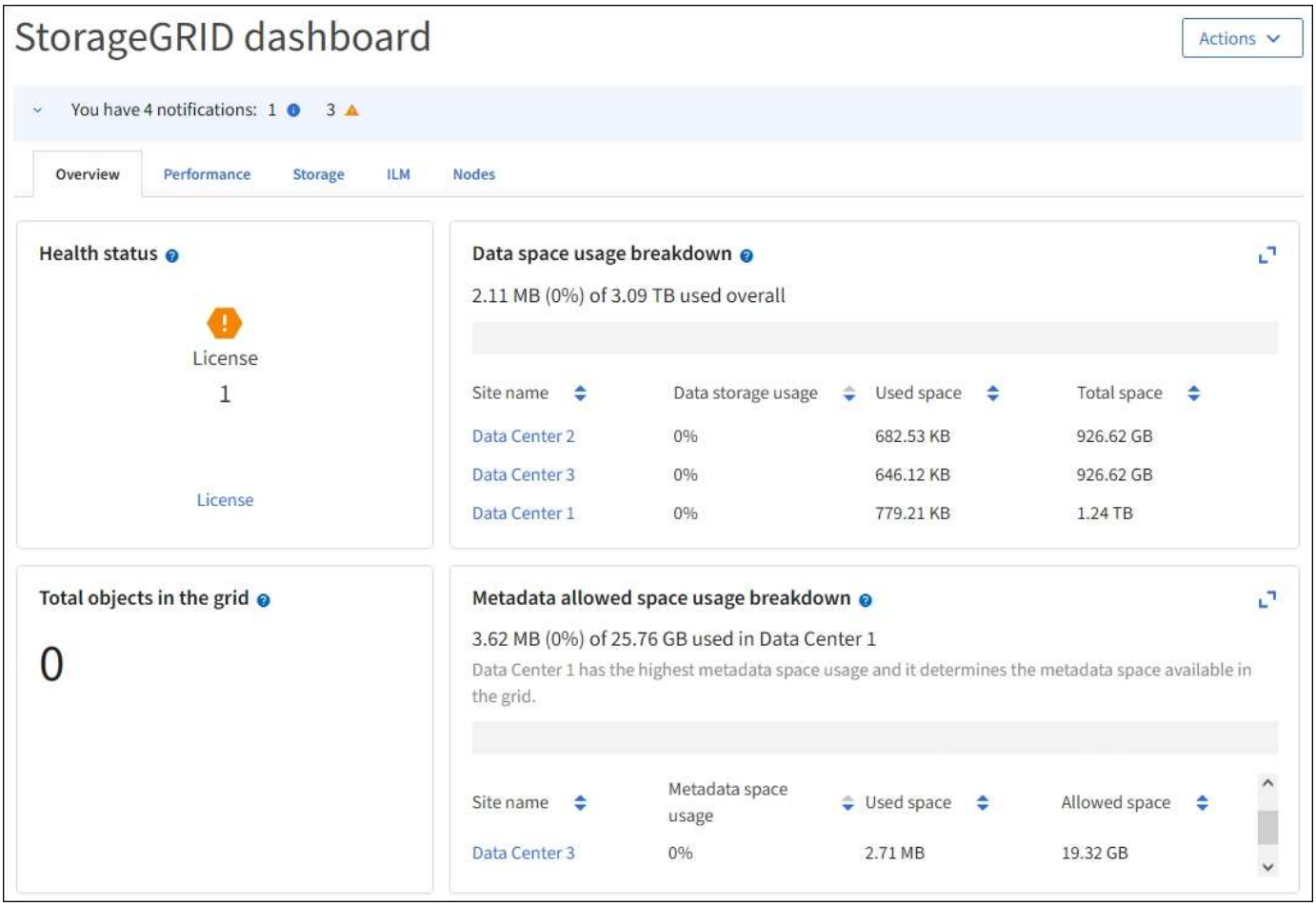


- StorageGRID가 SSO를 사용하고 있고 이전에 그리드 관리자 또는 테넌트 계정에 액세스한 경우:
  i. 최신 계정 목록에 * 0 * (Grid Manager의 계정 ID)을 입력하거나 * Grid Manager * 를 선택합니다.

ii. 로그인 * 을 선택합니다.

iii. 조직의 SSO 로그인 페이지에서 표준 SSO 자격 증명을 사용하여 로그인합니다.

로그인하면 대시보드가 포함된 그리드 관리자의 홈 페이지가 나타납니다. 제공되는 정보에 대한 자세한 내용은 을 참조하십시오 "대시보드를 보고 관리합니다".



다른 관리자 노드에 로그인합니다

다음 단계에 따라 다른 관리자 노드에 로그인합니다.

SSO를 사용하지 않습니다

단계

1. 브라우저의 주소 표시줄에 다른 관리 노드의 정규화된 도메인 이름 또는 IP 주소를 입력합니다. 필요에 따라 포트 번호를 포함시킵니다.
2. Grid Manager의 사용자 이름과 암호를 입력합니다.
3. 로그인 * 을 선택합니다.

SSO 사용

StorageGRID가 SSO를 사용하고 있고 하나의 관리 노드에 로그인한 경우 다시 로그인하지 않고도 다른 관리 노드에 액세스할 수 있습니다.

단계

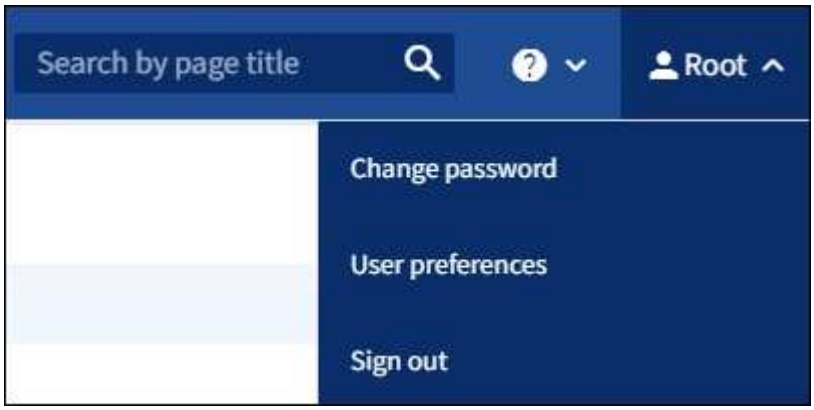
1. 브라우저의 주소 표시줄에 다른 관리 노드의 정규화된 도메인 이름 또는 IP 주소를 입력합니다.
2. SSO 세션이 만료된 경우 자격 증명을 다시 입력하십시오.

Grid Manager에서 로그아웃합니다

그리드 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. 오른쪽 위 모서리에서 사용자 이름을 선택합니다.



2. 로그아웃 * 을 선택합니다.

옵션을 선택합니다	설명
SSO가 사용되지 않습니다	관리자 노드에서 로그아웃되었습니다. 그리드 관리자 로그인 페이지가 표시됩니다. • 참고: * 둘 이상의 관리자 노드에 로그인한 경우 각 노드에서 로그아웃해야 합니다.
SSO가 활성화되었습니다	액세스 중인 모든 관리 노드에서 로그아웃되었습니다. StorageGRID 로그인 페이지가 표시됩니다. * 그리드 관리자 * 는 * 최근 계정 * 드롭다운에 기본값으로 나열되고 * 계정 ID * 필드는 0으로 표시됩니다. • 참고: * SSO가 활성화되어 있고 테넌트 관리자에도 로그인한 경우, 반드시 로그인 해야 합니다 "테넌트 계정에서 로그아웃합니다" 를 선택합니다 "SSO에서 로그아웃합니다".

암호를 변경합니다

Grid Manager의 로컬 사용자인 경우 사용자 고유의 암호를 변경할 수 있습니다.

시작하기 전에

를 사용하여 그리드 관리자에 로그인했습니다 "지원되는 웹 브라우저".

이 작업에 대해

통합 사용자로 StorageGRID에 로그인하거나 SSO(Single Sign-On)가 활성화된 경우 그리드 관리자에서 암호를 변경할 수 없습니다. 대신 Active Directory 또는 OpenLDAP와 같은 외부 ID 소스에서 암호를 변경해야 합니다.

단계

1. Grid Manager 헤더에서 *사용자 이름 * > * 암호 변경 * 을 선택합니다.
2. 현재 암호를 입력합니다.
3. 새 암호를 입력합니다.

   암호는 8자 이상 32자 이하여야 합니다. 암호는 대/소문자를 구분합니다.

4. 새 암호를 다시 입력합니다.
5. 저장 * 을 선택합니다.

StorageGRID 라이센스 정보를 봅니다

필요한 경우 그리드의 최대 스토리지 용량과 같은 StorageGRID 시스템에 대한 라이센스 정보를 볼 수 있습니다.

시작하기 전에

- 를 사용하여 그리드 관리자에 로그인했습니다 "지원되는 웹 브라우저".

이 작업에 대해

이 StorageGRID 시스템의 소프트웨어 라이센스에 문제가 있는 경우 대시보드의 상태 카드에 라이센스 상태 아이콘과 * 라이센스 * 링크가 포함됩니다. 이 숫자는 라이센스 관련 문제의 수를 나타냅니다.

단계

1. 다음 중 하나를 수행하여 라이센스 페이지에 액세스합니다.
   - 유지 관리 * > * 시스템 * > * 라이센스 * 를 선택합니다.
   - 대시보드의 상태 카드에서 라이센스 상태 아이콘 또는 * 라이센스 * 링크를 선택합니다.

     이 링크는 라이센스에 문제가 있는 경우에만 나타납니다.

2. 현재 라이센스에 대한 읽기 전용 세부 정보 보기:
    - StorageGRID 시스템 ID로, 이 StorageGRID 설치의 고유 식별 번호입니다
    - 라이센스 일련 번호입니다
    - 라이센스 유형, * 영구 * 또는 * 가입 *
    - 그리드의 라이센스가 부여된 스토리지 용량입니다
    - 지원되는 스토리지 용량입니다
    - 라이선스 종료 날짜. 영구 라이센스에 대해 * 해당 없음 * 이 나타납니다.
    - 지원 종료 날짜입니다

      이 날짜는 현재 라이센스 파일에서 읽으며 라이센스 파일을 얻은 후 지원 서비스 계약을 연장하거나 갱신한 경우 최신 날짜가 아닐 수 있습니다. 이 값을 업데이트하려면 을 참조하십시오 "StorageGRID 라이센스 정보를 업데이트합니다". Active IQ를 사용하여 실제 계약 종료 날짜를 볼 수도 있습니다.

    - 라이센스 텍스트 파일의 내용입니다

StorageGRID 라이센스 정보를 업데이트합니다

라이센스 조건이 변경될 때마다 StorageGRID 시스템의 라이센스 정보를 업데이트해야 합니다. 예를 들어 그리드에 대한 추가 스토리지 용량을 구입한 경우 라이센스 정보를 업데이트해야 합니다.

시작하기 전에

- StorageGRID 시스템에 적용할 새 라이센스 파일이 있습니다.
- 있습니다 "특정 액세스 권한".
- 프로비저닝 암호가 있습니다.

단계

1. 유지 관리 * > * 시스템 * > * 라이센스 * 를 선택합니다.
2. 라이센스 업데이트 섹션에서 * 찾아보기 * 를 선택합니다.
3. 새 라이센스 파일을 찾아 선택합니다 (`.txt`)를 클릭합니다.

   새 라이센스 파일의 유효성을 검사한 후 표시합니다.

4. 프로비저닝 암호를 입력합니다.
5. 저장 * 을 선택합니다.

API를 사용합니다

Grid Management API를 사용합니다

Grid Manager 사용자 인터페이스 대신 Grid Management REST API를 사용하여 시스템 관리 작업을 수행할 수 있습니다. 예를 들어, API를 사용하여 작업을 자동화하거나 사용자와 같은 여러

엔터티를 더 빠르게 생성할 수 있습니다.

최고 수준의 리소스

Grid Management API는 다음과 같은 최상위 리소스를 제공합니다.

- `/grid`: 액세스는 Grid Manager 사용자로 제한되며 구성된 그룹 권한을 기반으로 합니다.
- `/org`: 테넌트 계정의 로컬 또는 통합 LDAP 그룹에 속한 사용자만 액세스할 수 있습니다. 자세한 내용은 을 참조하십시오 "테넌트 계정을 사용합니다".
- `/private`: 액세스는 Grid Manager 사용자로 제한되며 구성된 그룹 권한을 기반으로 합니다. 사설 API는 사전 통보 없이 변경될 수 있습니다. StorageGRID 전용 엔드포인트도 요청의 API 버전을 무시합니다.

API 요청을 발행합니다

Grid Management API는 Swagger 오픈 소스 API 플랫폼을 사용합니다. Swagger는 개발자와 개발자가 아닌 사용자가 API를 사용하여 StorageGRID에서 실시간 작업을 수행할 수 있도록 직관적인 사용자 인터페이스를 제공합니다.

Swagger 사용자 인터페이스는 각 API 작동에 대한 전체 세부 정보와 문서를 제공합니다.

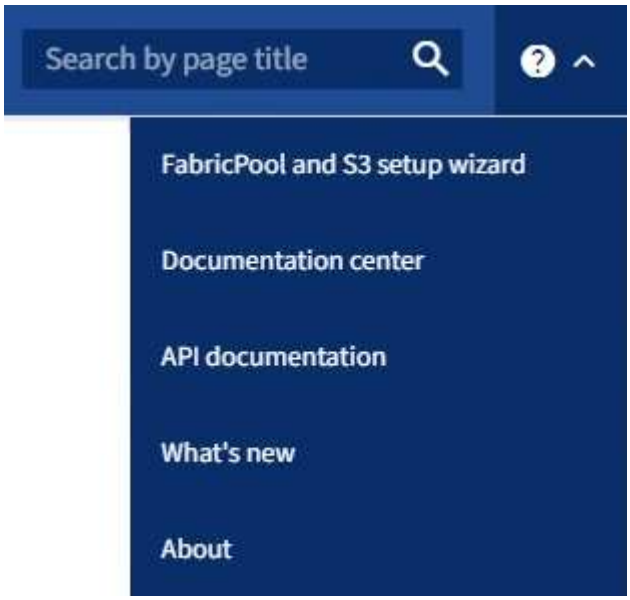
시작하기 전에

- 를 사용하여 그리드 관리자에 로그인했습니다 "지원되는 웹 브라우저".
- 있습니다 "특정 액세스 권한".

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

단계

1. Grid Manager 헤더에서 도움말 아이콘을 선택하고 * API documentation * 을 선택합니다.



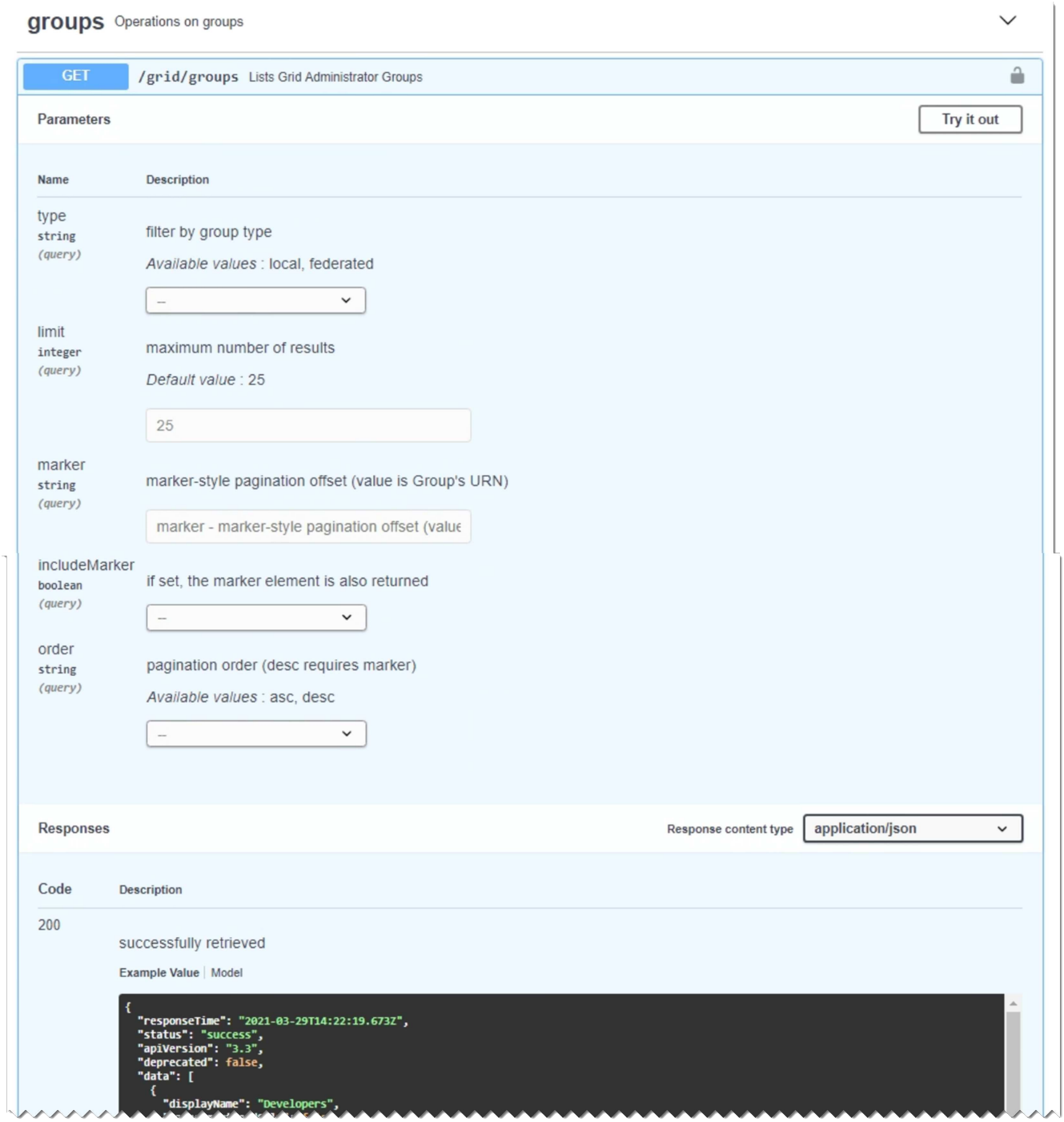
2. 전용 API로 작업을 수행하려면 StorageGRID 관리 API 페이지에서 * 전용 API 설명서 * 로 이동 * 을 선택합니다.

사설 API는 사전 통보 없이 변경될 수 있습니다. StorageGRID 전용 엔드포인트도 요청의 API 버전을 무시합니다.

3. 원하는 작업을 선택합니다.

   API 작업을 확장하면 가져오기, 가져오기, 업데이트 및 삭제와 같은 사용 가능한 HTTP 작업을 볼 수 있습니다.

4. 끝점 URL, 필수 또는 선택적 매개 변수 목록, 요청 본문(필요한 경우) 예제 및 가능한 응답을 비롯한 요청 세부 정보를 보려면 HTTP 작업을 선택합니다.



5. 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.
6. 예제 요청 본문을 수정해야 하는지 확인합니다. 이 경우 * Model * 을 선택하여 각 필드의 요구 사항을 확인할 수

있습니다.

7. 체험하기 * 를 선택합니다.
8. 필요한 매개 변수를 제공하거나 요청 본문을 필요에 따라 수정합니다.
9. Execute * 를 선택합니다.
10. 응답 코드를 검토하여 요청이 성공했는지 확인합니다.

Grid Management API 작업

Grid Management API는 사용 가능한 작업을 다음 섹션으로 구성합니다.

> 이 목록에는 공용 API에서 사용할 수 있는 작업만 포함됩니다.

- * ACCOUNT *: 새 계정 생성 및 지정된 계정의 스토리지 사용량 검색을 포함하여 스토리지 테넌트 계정을 관리하는 작업입니다.
- * ALARMS *: 현재 경보(레거시 시스템)를 나열하고, 현재 알림과 노드 연결 상태 요약을 포함하여 그리드의 상태에 대한 정보를 반환하는 작업.
- * alert-history *: 해결된 알림의 작업.
- 알림 메시지 수신자 *: 경고 알림 수신자(이메일)에 대한 작업.
- * alert-rules *: 경고 규칙에 대한 작업.
- * alert-silences *: 경고 작동 중.
- * 경고 *: 경고 작업.
- * 감사 *: 감사 구성을 나열하고 업데이트하는 작업.
- * auth *: 사용자 세션 인증을 수행하기 위한 작업.

  Grid Management API는 Bearer Token Authentication Scheme을 지원한다. 로그인하려면 인증 요청의 JSON 본문에 사용자 이름 및 암호를 입력합니다(즉, `POST /api/v3/authorize`)를 클릭합니다. 사용자가 성공적으로 인증되면 보안 토큰이 반환됩니다. 이 토큰은 후속 API 요청 헤더("Authorization:Bearer_token_")에 제공되어야 합니다. 토큰은 16시간 후에 만료됩니다.

  > StorageGRID 시스템에 대해 Single Sign-On이 활성화된 경우 인증을 위해 다른 단계를 수행해야 합니다. "SSO(Single Sign-On)가 활성화된 경우 API에 로그인 인증"을 참조하십시오.

  인증 보안 강화에 대한 자세한 내용은 "사이트 간 요청 위조로부터 보호"를 참조하십시오.

- * 클라이언트-인증서 *: 외부 모니터링 도구를 사용하여 StorageGRID에 안전하게 액세스할 수 있도록 클라이언트 인증서를 구성하는 작업
- * config *: 그리드 관리 API 제품 릴리스 및 버전과 관련된 작업. 제품 릴리스 버전과 해당 릴리스에서 지원하는 Grid Management API의 주요 버전을 나열할 수 있으며 더 이상 사용되지 않는 API 버전을 사용하지 않도록 설정할 수 있습니다.
- * 비활성화됨 - 기능 *: 비활성화된 기능을 보기 위한 작업.
- * DNS-서버 *: 구성된 외부 DNS 서버를 나열하고 변경하는 작업.
- * 드라이브 세부 정보 *: 특정 스토리지 어플라이언스 모델을 위한 드라이브 작업.

- * endpoint-domain-names *: S3 끝점 도메인 이름을 나열하고 변경하는 작업.
- * 삭제 코딩 * : 삭제 코딩 프로필에 대한 작업.
- * 확장 *: 확장 작업(절차 수준).
- * 확장 노드 *: 확장 시 작업(노드 레벨).
- * 확장 사이트 *: 확장 시 운영(사이트 레벨)
- * GRID-NETWORKS *: 그리드 네트워크 목록을 나열하고 변경하는 작업.
- * GRID-Passwords *: 그리드 암호 관리 작업.
- * 그룹 *: 로컬 그리드 관리자 그룹을 관리하고 외부 LDAP 서버에서 통합 그리드 관리자 그룹을 검색하는 작업.
- * identity-source *: 외부 ID 소스를 구성하고 통합 그룹 및 사용자 정보를 수동으로 동기화하는 작업
- * ILM *: 정보 수명 주기 관리(ILM)의 운영
- **In-progress-procedures**: 현재 진행 중인 유지보수 절차를 검색합니다.
- 라이센스 *: StorageGRID 라이센스를 검색하고 업데이트하는 작업.
- **logs**: 로그 파일을 수집하고 다운로드하는 작업입니다
- * 메트릭 *: 일정 기간 동안 단일 시점 및 범위 메트릭 쿼리의 인스턴스 메트릭 쿼리를 비롯한 StorageGRID 메트릭의 작업 Grid Management API는 Prometheus 시스템 모니터링 도구를 백엔드 데이터 소스로 사용합니다. Prometheus 쿼리 구성에 대한 자세한 내용은 Prometheus 웹 사이트를 참조하십시오.

> 다음을 포함한 메트릭 `private` 해당 이름은 내부용으로만 사용해야 합니다. 이러한 메트릭은 사전 통지 없이 StorageGRID 릴리스 간에 변경될 수 있습니다.

- * 노드 세부 정보 *: 노드 세부 정보에 대한 작업.
- * 노드 상태 *: 노드 상태에 대한 작업
- * 노드-스토리지-상태 *: 노드 스토리지 상태의 작업.
- * NTP-서버 *: 외부 NTP(Network Time Protocol) 서버를 나열하거나 업데이트하는 작업.
- * 오브젝트 *: 오브젝트 및 오브젝트 메타데이터의 작동
- * 복구 *: 복구 절차를 위한 작업.
- * recovery-package *: 복구 패키지를 다운로드하기 위한 작업.
- * 지역 *: 영역을 보고 만드는 작업.
- * S3 오브젝트 잠금 *: 글로벌 S3 오브젝트 잠금 설정 시 작업.
- * server-certificate *: Grid Manager 서버 인증서를 보고 업데이트하는 작업.
- * SNMP *: 현재 SNMP 구성에 대한 작업.
- * 스토리지 - 워터마크 *: 스토리지 노드 워터마크입니다.
- * traffic-classes *: 트래픽 분류 정책을 위한 운영.
- * 신뢰할 수 없는 클라이언트-네트워크 *: 신뢰할 수 없는 클라이언트 네트워크 구성에서의 작업.
- * 사용자 *: 그리드 관리자 사용자를 보고 관리하는 작업.

Grid Management API 버전 관리

Grid Management 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. 에 대한 Swagger API 설명서의 * config * 섹션을 참조하십시오 "Grid Management API를 참조하십시오" 를 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

를 사용합니다 `GET /versions` 지원되는 API 주 버전 목록을 반환하기 위한 API 요청입니다. 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

path 매개 변수를 사용하여 API 버전을 지정할 수 있습니다 (`/api/v4`) 또는 머리글 (`Api-Version: 4`)를 클릭합니다. 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.

기능을 활성화하려면 를 설정합니다 `csrfToken` 매개 변수 대상 `true` 인증 중. 기본값은 입니다 `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"
```

참이면 A입니다 `GridCsrfToken` 쿠키는 Grid Manager 및 에 대한 로그인의 임의 값으로 설정됩니다 `AccountCsrfToken` 쿠키는 테넌트 관리자에 대한 로그인에 대한 임의 값으로 설정됩니다.

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

- 를 클릭합니다 `X-Csrf-Token` CSRF 토큰 쿠키의 값으로 설정된 헤더.
- 폼 인코딩된 바디를 수용하는 끝점의 경우: A `csrfToken` 폼 인코딩된 요청 본문 매개 변수입니다.

추가 예제 및 세부 정보는 온라인 API 설명서를 참조하십시오.

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

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

SSO(Single Sign-On)가 활성화된 경우 API 사용(Active Directory)

있는 경우 "SSO(Single Sign-On) 구성 및 활성화" Active Directory를 SSO 공급자로 사용하는 경우, 그리드 관리 API 또는 테넌트 관리 API에 유효한 인증 토큰을 얻기 위해 일련의 API 요청을 실행해야 합니다.

SSO(Single Sign-On)가 활성화된 경우 API에 로그인합니다

Active Directory를 SSO ID 공급자로 사용하는 경우 다음 지침이 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 작업에 대해

인증 토큰을 얻으려면 다음 예 중 하나를 사용할 수 있습니다.

- 를 클릭합니다 `storagegrid-ssoauth.py` StorageGRID 설치 파일 디렉터리에 있는 Python 스크립트 (`./rpms` Red Hat Enterprise Linux의 경우 `./debs` Ubuntu 또는 Debian, 및 의 경우 `./vsphere` VMware의 경우).
- curl 요청의 워크플로 예

  컬을 너무 느리게 수행하면 컬링 작업 시간이 초과될 수 있습니다. 다음과 같은 오류가 표시될 수 있습니다. `A valid SubjectConfirmation was not found on this Response.`

  > 예제 curl 워크플로는 다른 사용자가 암호를 볼 수 없도록 보호하지 않습니다.

URL 인코딩 문제가 있는 경우 다음 오류가 표시될 수 있습니다. `Unsupported SAML version.`

단계

1. 인증 토큰을 얻으려면 다음 방법 중 하나를 선택합니다.

- 를 사용합니다 `storagegrid-ssoauth.py` Python 스크립트. 2단계로 이동합니다.
- curl 요청을 사용합니다. 3단계로 이동합니다.

2. 을(를) 사용하려는 경우 `storagegrid-ssoauth.py` 스크립트에서 Python 인터프리터로 스크립트를 전달하고 스크립트를 실행합니다.

   프롬프트가 표시되면 다음 인수에 대한 값을 입력합니다.

   - SSO 방법 ADFS 또는 ADFS를 입력합니다.
   - SSO 사용자 이름입니다
   - StorageGRID가 설치된 도메인입니다
   - StorageGRID의 주소입니다
   - 테넌트 관리 API에 액세스하려는 경우 테넌트 계정 ID입니다.

```
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 토큰은 출력에 제공됩니다. 이제 SSO가 사용되지 않는 경우 API를 사용하는 방법과 유사하게 다른 요청에 토큰을 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'
```

   > Grid Management API에 액세스하려면 0 As를 사용합니다 `TENANTACCOUNTID`.

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

      이 예는 에 대한 서명된 인증 URL에 대한 POST 요청을 보여 줍니다 `TENANTACCOUNTID`. 결과가 로 전달됩니다 `python -m json.tool` JSON 인코딩을 제거합니다.

```
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 예제의 응답에는 URL로 인코딩된 서명된 URL이 포함되어 있지만 추가 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. 를 저장합니다 `SAMLRequest` 후속 명령에 사용할 응답에 따라.

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

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

한 가지 옵션은 이전 응답의 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 ID:

```
<form method="post" id="loginForm" autocomplete="off"
novalidate="novalidate" onKeyPress="if (event && event.keyCode == 13)
Login.submitLoginRequest();" action="/adfs/ls/?
SAMLRequest=fZHRToMwFIZfhb...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 응답에서 양식 작업으로 자격 증명을 보냅니다.

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

AD FS는 헤더에 추가 정보가 포함된 302 리디렉션을 반환합니다.

> SSO 시스템에 대해 MFA(다중 요소 인증)가 활성화된 경우 양식 게시물에는 두 번째 암호 또는 다른 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. 를 저장합니다 `MSISAuth` 응답에서 받은 쿠키입니다.

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

h. 인증 POST에서 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 로그아웃 사용을 위한 AD FS 세션 정보가 포함되며 응답 본문에는 숨겨진 양식 필드에 SALMLResponse가 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
XFxVWx3bkl1MnFuUSUzZCUzZCYmJiYmXzE3MjAyZTA5LThmMDgtNDRkZC04Yzg5LTQ3ND
UxYzA3ZjkzYw==; path=/adfs; HttpOnly; Secure
Set-Cookie: MSISAuthenticated=MTEvNy8yMDE4IDQ6MzI6NTkgUE0=;
path=/adfs; HttpOnly; Secure
Set-Cookie: MSISLoopDetectionCookie=MjAxOC0xMS0wNzoxNjozMjo1OVpcMQ==;
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="PHNhbWxwOlJlc3BvbnN...1scDpSZXNwb25zZT4=" /><input
type="hidden" name="RelayState" value="12345" />
```

i. 를 저장합니다 `SAMLResponse` 숨겨진 필드에서:

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

j. 저장된 를 사용합니다 `SAMLResponse`, StorageGRID을 만듭니다`/api/saml-response` StorageGRID 인증 토큰 생성 요청.

용 `RelayState`, 테넌트 계정 ID를 사용하거나 그리드 관리 API에 로그인하려면 0을 사용합니다.

```
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 포함됩니다.

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

a. 응답에 인증 토큰을 다른 이름으로 저장합니다 `MYTOKEN`.

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

이제 를 사용할 수 있습니다 `MYTOKEN` 다른 요청에서는 SSO가 사용되지 않는 경우 API를 사용하는 방법과 유사합니다.

SSO(Single Sign-On)가 활성화된 경우 API에서 로그아웃합니다

SSO(Single Sign-On)가 활성화된 경우 그리드 관리 API 또는 테넌트 관리 API에서 로그아웃하기 위해 일련의 API 요청을 실행해야 합니다. Active Directory를 SSO ID 공급자로 사용하는 경우 다음 지침이 적용됩니다

이 작업에 대해

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

단계

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

```
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%3
D",
    "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. 로그아웃 URL에 요청을 보내 SLO를 트리거하고 StorageGRID로 다시 리디렉션합니다.

```
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: MSISSignoutProtocol=U2FtbA==; expires=Tue, 20 Nov 2018
22:35:03 GMT; path=/adfs; HttpOnly; Secure
```

4. StorageGRID bearer token을 삭제한다.

StorageGRID 베어러 토큰을 삭제하는 것은 SSO를 사용하지 않는 것과 동일한 방식으로 작동합니다. 'cookie "sso=true"가 제공되지 않으면 사용자는 SSO 상태에 영향을 주지 않고 StorageGRID에서 로그아웃됩니다.

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

A `204 No Content` 응답 - 사용자가 로그아웃되었음을 나타냅니다.

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

SSO(Single Sign-On)가 활성화된 경우 API 사용(Azure)

있는 경우 "SSO(Single Sign-On) 구성 및 활성화" Azure를 SSO 공급자로 사용하는 경우, 두 개의 예제 스크립트를 사용하여 Grid Management API 또는 Tenant Management API에 유효한 인증 토큰을 얻을 수 있습니다.

Azure Single Sign-On이 활성화된 경우 API에 로그인합니다

Azure를 SSO ID 공급자로 사용하는 경우 다음 지침이 적용됩니다

시작하기 전에

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

이 작업에 대해

인증 토큰을 얻으려면 다음 예제 스크립트를 사용할 수 있습니다.

- 를 클릭합니다 `storagegrid-ssoauth-azure.py` Python 스크립트
- 를 클릭합니다 `storagegrid-ssoauth-azure.js` Node.js 스크립트

두 스크립트 모두 StorageGRID 설치 파일 디렉터리에 있습니다 (`./rpms` Red Hat Enterprise Linux의 경우 `./debs` Ubuntu 또는 Debian, 및 의 경우 `./vsphere` VMware의 경우).

Azure와의 API 통합 기능을 직접 작성하려면 을 참조하십시오 `storagegrid-ssoauth-azure.py` 스크립트. Python 스크립트는 StorageGRID에 직접 두 개의 요청을 하고(먼저 SAMLRequest를 받고 나중에 인증 토큰을 얻기 위해) Node.js 스크립트를 호출하여 Azure와 상호 작용하여 SSO 작업을 수행합니다.

SSO 작업은 일련의 API 요청을 사용하여 실행할 수 있지만, 그렇게 하는 것은 간단하지 않습니다. Puppeteer Node.js 모듈은 Azure SSO 인터페이스를 스크레핑하는 데 사용됩니다.

URL 인코딩 문제가 있는 경우 다음 오류가 표시될 수 있습니다. `Unsupported SAML version.`

단계

1. 다음과 같이 필요한 종속성을 설치합니다.
   a. Node.js를 설치합니다(참조) "https://nodejs.org/en/download/")를 클릭합니다.
   b. 필요한 Node.js 모듈(puppeteer 및 jsdom)을 설치합니다.

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

2. Python 스크립트를 Python 인터프리터로 전달하여 스크립트를 실행합니다.

   그런 다음 Python 스크립트는 해당 Node.js 스크립트를 호출하여 Azure SSO 상호 작용을 수행합니다.

3. 프롬프트가 표시되면 다음 인수에 대한 값을 입력하거나 매개 변수를 사용하여 전달합니다.
   - Azure에 로그인하는 데 사용되는 SSO 이메일 주소입니다
   - StorageGRID의 주소입니다
   - 테넌트 관리 API에 액세스하려는 경우 테넌트 계정 ID입니다
4. 메시지가 표시되면 암호를 입력하고 요청 시 Azure에 MFA 권한을 제공할 준비를 합니다.

```
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 MFA가 Microsoft Authenticator를 사용하여 수행된 것으로 가정합니다. 다른 형태의 MFA를 지원하도록 스크립트를 수정해야 할 수도 있습니다(예: 텍스트 메시지에 수신된 코드 입력).

StorageGRID 인증 토큰은 출력에 제공됩니다. 이제 SSO가 사용되지 않는 경우 API를 사용하는 방법과 유사하게 다른 요청에 토큰을 사용할 수 있습니다.

SSO(Single Sign-On)가 활성화된 경우 API 사용(PingFederate)

있는 경우 "SSO(Single Sign-On) 구성 및 활성화" 그리고 PingFederate를 SSO 공급자로 사용하는 경우 일련의 API 요청을 발급하여 Grid Management API 또는 Tenant Management API에 유효한 인증 토큰을 얻어야 합니다.

SSO(Single Sign-On)가 활성화된 경우 API에 로그인합니다

이 지침은 PingFederate를 SSO ID 공급자로 사용하는 경우 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 작업에 대해

인증 토큰을 얻으려면 다음 예 중 하나를 사용할 수 있습니다.

- 를 클릭합니다 `storagegrid-ssoauth.py` StorageGRID 설치 파일 디렉터리에 있는 Python 스크립트 (`./rpms` Red Hat Enterprise Linux의 경우 `./debs` Ubuntu 또는 Debian, 및 의 경우 `./vsphere` VMware의 경우).
- curl 요청의 워크플로 예

  컬을 너무 느리게 수행하면 컬링 작업 시간이 초과될 수 있습니다. 다음과 같은 오류가 표시될 수 있습니다. `A valid SubjectConfirmation was not found on this Response.`

  > 예제 curl 워크플로는 다른 사용자가 암호를 볼 수 없도록 보호하지 않습니다.

  URL 인코딩 문제가 있는 경우 다음 오류가 표시될 수 있습니다. `Unsupported SAML version.`

단계

1. 인증 토큰을 얻으려면 다음 방법 중 하나를 선택합니다.
   - 를 사용합니다 `storagegrid-ssoauth.py` Python 스크립트. 2단계로 이동합니다.
   - curl 요청을 사용합니다. 3단계로 이동합니다.
2. 을(를) 사용하려는 경우 `storagegrid-ssoauth.py` 스크립트에서 Python 인터프리터로 스크립트를 전달하고 스크립트를 실행합니다.

   프롬프트가 표시되면 다음 인수에 대한 값을 입력합니다.

   - SSO 방법 "pingfederate"(PINGFEDERATE, Pingfederate 등)의 모든 변형을 입력할 수 있습니다.

- SSO 사용자 이름입니다
- StorageGRID가 설치된 도메인입니다. 이 필드는 PingFederate에 사용되지 않습니다. 빈 칸으로 두거나 원하는 값을 입력할 수 있습니다.
- StorageGRID의 주소입니다
- 테넌트 관리 API에 액세스하려는 경우 테넌트 계정 ID입니다.

```
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 토큰은 출력에 제공됩니다. 이제 SSO가 사용되지 않는 경우 API를 사용하는 방법과 유사하게 다른 요청에 토큰을 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'
```

> Grid Management API에 액세스하려면 0 As를 사용합니다 `TENANTACCOUNTID`.

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

   이 예제에서는 TENANTACCOUNTID에 대한 서명된 인증 URL에 대한 POST 요청을 보여 줍니다. 결과는 python-m json.tool에 전달되어 JSON 인코딩을 제거합니다.

```
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 예제의 응답에는 URL로 인코딩된 서명된 URL이 포함되어 있지만 추가 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. 를 저장합니다 `SAMLRequest` 후속 명령에 사용할 응답에 따라.

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

d. 응답과 쿠키를 내보내고 응답을 에코합니다.

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

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

e. 'pf.adapterId' 값을 내보내고 응답을 에코합니다.

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

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

f. 'href' 값을 내보내고(후행 슬래시/ 제거) 응답을 에코합니다.

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

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

g. '조치' 값 내보내기:

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

h. 자격 증명과 함께 쿠키 보내기:

```
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. 를 저장합니다 `SAMLResponse` 숨겨진 필드에서:

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

j. 저장된 를 사용합니다 `SAMLResponse`, StorageGRID을 만듭니다`/api/saml-response` StorageGRID 인증 토큰 생성 요청.

용 `RelayState`, 테넌트 계정 ID를 사용하거나 그리드 관리 API에 로그인하려면 0을 사용합니다.

```
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 포함됩니다.

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

a. 응답에 인증 토큰을 다른 이름으로 저장합니다 `MYTOKEN`.

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

이제 를 사용할 수 있습니다 `MYTOKEN` 다른 요청에서는 SSO가 사용되지 않는 경우 API를 사용하는 방법과 유사합니다.

SSO(Single Sign-On)가 활성화된 경우 API에서 로그아웃합니다

SSO(Single Sign-On)가 활성화된 경우 그리드 관리 API 또는 테넌트 관리 API에서 로그아웃하기 위해 일련의 API 요청을 실행해야 합니다. 이 지침은 PingFederate를 SSO ID 공급자로 사용하는 경우 적용됩니다

이 작업에 대해

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

단계

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

```
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/idp/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/idp/SLO.saml2?SAMLRequest=fZDNboMwEIRfhZ...HcQ%3D%3D'
```

3. 로그아웃 URL에 요청을 보내 SLO를 트리거하고 StorageGRID로 다시 리디렉션합니다.

```
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 bearer token을 삭제한다.

   StorageGRID 베어러 토큰을 삭제하는 것은 SSO를 사용하지 않는 것과 동일한 방식으로 작동합니다. 'cookie "sso=true"가 제공되지 않으면 사용자는 SSO 상태에 영향을 주지 않고 StorageGRID에서 로그아웃됩니다.

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

A `204 No Content` 응답 - 사용자가 로그아웃되었음을 나타냅니다.

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

API를 사용하여 기능을 비활성화합니다

그리드 관리 API를 사용하여 StorageGRID 시스템의 특정 기능을 완전히 비활성화할 수 있습니다. 기능이 비활성화되면 해당 기능과 관련된 작업을 수행할 수 있는 권한을 아무도 할당할 수 없습니다.

이 작업에 대해

비활성화된 기능 시스템을 사용하면 StorageGRID 시스템의 특정 기능에 액세스하지 못하게 할 수 있습니다. 루트 사용자 또는 * 루트 액세스 * 권한이 있는 관리자 그룹에 속한 사용자가 해당 기능을 사용할 수 없도록 하는 유일한 방법은 기능을 비활성화하는 것입니다.

이 기능이 어떻게 유용한지 이해하려면 다음 시나리오를 고려해 보십시오.

_Company A는 테넌트 계정을 생성하여 StorageGRID 시스템의 스토리지 용량을 임대하는 서비스 공급자입니다. 회사 A는 임차자의 객체 보안을 보호하기 위해 계정이 배포된 후 자신의 직원이 테넌트 계정에 액세스할 수 없도록 하려고 합니다. _

_회사 A는 그리드 관리 API에서 기능 비활성화 시스템을 사용하여 이 목표를 달성할 수 있습니다. 그리드 관리자(UI 및 API 모두)에서 * 테넌트 루트 암호 변경 * 기능을 완전히 비활성화함으로써 회사 A는 루트 사용자 및 * 루트 액세스 * 권한이 있는 그룹에 속하는 사용자를 포함하여 관리자 사용자가 테넌트 계정의 루트 사용자에 대한 암호를 변경할 수 없도록 할 수 있습니다. _

단계

1. Grid Management API에 대한 Swagger 문서에 액세스합니다. 을 참조하십시오 "Grid Management API를 사용합니다".
2. 기능 비활성화 끝점을 찾습니다.
3. 테넌트 루트 암호 변경 등의 기능을 비활성화하려면 다음과 같이 API로 본문을 보냅니다.

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

   요청이 완료되면 테넌트 루트 암호 변경 기능이 비활성화됩니다. 테넌트 루트 암호 변경 * 관리 권한이 더 이상 사용자 인터페이스에 표시되지 않으며 테넌트의 루트 암호를 변경하려고 시도하는 모든 API 요청이 "403 사용 권한 없음"과 함께 실패합니다.

비활성화된 피처를 다시 활성화합니다

기본적으로 그리드 관리 API를 사용하여 비활성화된 기능을 다시 활성화할 수 있습니다. 그러나 비활성화된 피처가 다시 활성화되지 않도록 하려면 * activateFeatures * 기능 자체를 비활성화할 수 있습니다.

> activateFeatures * 기능을 다시 활성화할 수 없습니다. 이 기능을 비활성화하려는 경우 비활성화된 다른 모든 기능을 다시 활성화할 수 있는 기능이 영구적으로 손실됩니다. 손실된 기능을 복원하려면 기술 지원 부서에 문의해야 합니다.

단계

1. Grid Management API에 대한 Swagger 문서에 액세스합니다.
2. 기능 비활성화 끝점을 찾습니다.
3. 모든 기능을 다시 활성화하려면 다음과 같이 API로 본문을 보내십시오.

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

   이 요청이 완료되면 테넌트 루트 암호 변경 기능을 포함한 모든 기능이 다시 활성화됩니다. 이제 사용자 인터페이스에 * 테넌트 루트 암호 변경 * 관리 권한이 표시되며, 사용자에게 * 루트 액세스 * 또는 * 테넌트 루트 암호 변경 * 관리 권한이 있는 경우 테넌트의 루트 암호를 변경하려고 시도하는 모든 API 요청이 성공합니다.

> 이전 예에서는 _ALL_DEACTED 피처가 재활성화됩니다. 비활성화된 상태로 유지되어야 하는 다른 기능이 비활성화된 경우, PUT 요청에 명시적으로 지정해야 합니다. 예를 들어 테넌트 루트 암호 변경 기능을 다시 활성화하고 경보 승인 기능을 계속 비활성화하려면 다음 PUT 요청을 보내십시오.
>
> ```
> { "grid": { "alarmAcknowledgment": true } }
> ```

저작권 정보

Copyright © 2025 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 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/TM에 나열된 마크는 NetApp, Inc.의 상표입니다. 기타 회사 및 제품 이름은 해당 소유자의 상표일 수 있습니다.