워크플로우
ONTAP Automation

NetApp
May 24, 2024

목차

워크플로우

워크플로우 사용을 준비하십시오

라이브 ONTAP 배포와 함께 사용하기 전에 워크플로의 구조와 형식을 잘 알고 있어야 합니다.

ONTAP 릴리스에서 사용하려는 워크플로에서 모든 API 호출을 지원하는지 확인해야 합니다. 을 참조하십시오 "API 참조입니다" 를 참조하십시오.

소개

Workflow _ 은(는) 특정 관리 작업 또는 목표를 달성하는 데 필요한 하나 이상의 단계 시퀀스입니다. ONTAP 워크플로에는 각 작업을 수행하는 데 필요한 핵심 단계와 매개 변수가 포함되어 있습니다. 이 서비스는 ONTAP 자동화 환경을 사용자 지정하기 위한 시작점을 제공합니다.

단계 유형

ONTAP 워크플로의 각 단계는 다음 유형 중 하나입니다.

- REST API 호출(curl 및 JSON 예 등의 세부 정보 포함)
- 다른 ONTAP 워크플로우를 수행하거나 호출합니다
- 기타 관련 작업(구성 결정 등)

REST API 호출

대부분의 워크플로 단계는 REST API 호출입니다. 이 단계에서는 curl 예제 및 기타 정보가 포함된 일반적인 형식을 사용합니다. 를 참조하십시오 "API 참조입니다" REST API 호출에 대한 자세한 내용은

단일 단계 워크플로우

워크플로는 한 단계만 포함할 수 있습니다. 이러한 _단일 단계 워크플로_는 여러 단계가 포함된 워크플로와 약간 다르게 형식이 지정됩니다. 예를 들어, 명시적인 단계 이름이 제거됩니다. 작업 또는 작업은 워크플로 제목에 따라 명확해야 합니다.

입력 변수

워크플로우는 최대한 일반화되도록 설계되어 모든 ONTAP 환경에서 사용할 수 있습니다. 이 점을 고려하여 REST API 호출은 curl 예제 및 기타 입력에 변수를 사용합니다. 그런 다음 REST API 호출을 다양한 ONTAP 환경에 쉽게 조정할 수 있습니다.

기본 **URL** 형식입니다

curl 또는 프로그래밍 언어를 통해 ONTAP REST API에 직접 액세스할 수 있습니다. 이 경우 기본 URL이 ONTAP 온라인 설명서 또는 System Manager에 액세스할 때 사용하는 URL과 다릅니다.

API에 직접 액세스할 때 도메인 또는 IP 주소에 * API * 를 추가해야 합니다. 예를 들면 다음과 같습니다.

```
https://ontap.demo-example.com/api
```

을 참조하십시오 "ONTAP REST API 액세스 방법" 를 참조하십시오.

공통 입력 매개변수

대부분의 REST API 호출에서 일반적으로 사용되는 몇 가지 입력 매개 변수가 있습니다. 이러한 매개 변수는 일반적으로 개별 워크플로에 설명되어 있지 않습니다. 매개 변수에 대해 잘 알고 있어야 합니다. 을 참조하십시오 "API 요청을 제어하는 입력 변수입니다" 를 참조하십시오.

특정 REST API 호출에 추가 매개 변수가 필요한 경우 각 워크플로에 대한 curl 예제에 대한 추가 입력 매개 변수 * 섹션에 해당 매개 변수가 포함됩니다.

변수 형식

워크플로우 예제에 사용되는 ID 값과 기타 변수는 불투명하며 ONTAP 클러스터마다 다를 수 있습니다. 예제의 가독성을 높이기 위해 실제 값은 사용되지 않습니다. 변수가 대신 사용됩니다. 일관된 형식과 예약된 이름을 기반으로 하는 이 접근 방식은 다음과 같은 이점을 제공합니다.

- curl 및 JSON 샘플은 읽기 쉽고 이해하기 쉽습니다.
- 모든 키워드가 동일한 형식을 사용하므로 빠르게 식별할 수 있습니다.
- 값을 복사하여 재사용할 수 없으므로 보안 노출이 없습니다.

변수 형식은 Bash 셸 환경에서 사용하도록 지정됩니다. 각 변수는 달러 기호로 시작하고 필요에 따라 큰따옴표로 묶습니다. 그러면 Bash에서 인식할 수 있습니다. 대문자는 이름에 일관되게 사용됩니다.

다음은 일반적인 변수 키워드 중 일부입니다. 이 목록은 완전하지 않으며 필요에 따라 추가 변수가 사용됩니다. 그들의 의미는 상황에 따라 분명해야 합니다.

키워드	유형	설명
$FQDN_IP입니다	URL	ONTAP 관리 LIF의 정규화된 도메인 이름 또는 IP 주소
$클러스터_ID	경로	API 작업이 실행되는 ONTAP 클러스터를 식별하는 UUIDv4 값입니다.
$BASIC_AUTH입니다	머리글	HTTP 기본 인증에 사용되는 자격 증명 문자열입니다.

JSON 입력 예

POST 또는 패치를 사용하는 경우와 같은 일부 REST API 호출에는 요청 본문에 JSON 입력이 필요합니다. JSON 입력 예는 명확한 구별을 위해 컬링 예와 별도로 제시됩니다. JSON 입력 예제를 아래 설명된 기술 중 하나와 함께 사용할 수 있습니다.

로컬 파일에 저장

JSON 입력 예제를 파일로 복사하여 로컬에 저장할 수 있습니다. curl 명령은 를 사용하는 파일을 참조합니다 `--data` 로 파일 이름을 나타내는 값을 가진 매개 변수입니다 `@` 접두어.

컬링 예제 다음에 터미널에 붙여 넣습니다

먼저 컬을 복사하여 터미널 쉘에 붙여 넣어야 합니다. 그런 다음 예제를 편집하여 를 완전히 제거합니다 `--data` 끝에 매개 변수를 지정하고 로 바꿉니다 `--data-raw` 매개 변수. 마지막으로, 업데이트된 매개 변수가 있는 curl 명령 다음에 오도록 JSON 예제에 복사하여 붙여 넣습니다. 하나의 따옴표를 사용하여 JSON 입력 예제를 래핑해야 합니다.

인증 옵션

REST API에 사용할 수 있는 기본 인증 기술은 HTTP 기본 인증입니다. ONTAP 9.14부터 토큰 기반 인증 및 권한 부여와 함께 OAuth 2.0(Open Authorization) 프레임워크를 사용할 수도 있습니다.

HTTP 기본 인증

기본 인증을 사용할 때는 각 HTTP 요청에 사용자 자격 증명을 포함해야 합니다. 자격 증명을 보내는 방법에는 두 가지가 있습니다.

HTTP 요청 헤더를 생성합니다

수동으로 권한 부여 헤더를 생성하고 HTTP 요청에 포함할 수 있습니다. 이는 CLI에서 curl 명령을 사용하거나 자동화 코드와 함께 프로그래밍 언어를 사용할 때 수행할 수 있습니다. 개괄적인 단계는 다음과 같습니다.

1. 사용자 및 암호 값을 콜론으로 연결합니다.

   ```
   admin:david123
   ```

2. 전체 문자열을 base64로 변환:

   ```
   YWRtaW46ZGF2aWQxMjM=
   ```

3. 요청 헤더를 작성합니다.

   ```
   Authorization: Basic YWRtaW46ZGF2aWQxMjM=
   ```

워크플로 컬링 예제에는 사용하기 전에 업데이트해야 하는 * $BASIC_AUTH * 변수가 있는 이 헤더가 포함됩니다.

curl 매개 변수를 사용합니다

curl을 사용할 때 다른 옵션은 권한 부여 헤더를 제거하고 대신 curl * user * 매개 변수를 사용하는 것입니다. 예를 들면 다음과 같습니다.

```
--user username:password
```

사용자 환경에 적합한 자격 증명을 대체해야 합니다. 자격 증명은 base64로 인코딩되지 않습니다. 이 매개 변수를 사용하여 curl 명령을 실행하면 문자열이 인코딩되고 Authorization 헤더가 생성됩니다.

OAuth 2.0 을 참조하십시오

OAuth 2.0을 사용하는 경우 외부 인증 서버에서 액세스 토큰을 요청하고 각 HTTP 요청에 포함시켜야 합니다. 기본적인 상위 단계는 아래에 설명되어 있습니다. 도 참조하십시오 "ONTAP OAuth 2.0 구축 개요" OAuth 2.0에 대한 자세한 내용 및 ONTAP와 함께 사용하는 방법

ONTAP 환경을 준비합니다

REST API를 사용하여 ONTAP에 액세스하기 전에 ONTAP 환경을 준비하고 구성해야 합니다. 상위 수준에서는 다음과 같은 단계가 포함됩니다.

- ONTAP로 보호되는 리소스 및 클라이언트 식별
- 기존 ONTAP REST 역할 및 사용자 정의를 검토합니다
- 인증 서버를 설치하고 구성합니다

- 클라이언트 권한 부여 정의를 설계하고 구성합니다
- ONTAP를 구성하고 OAuth 2.0을 활성화합니다

액세스 토큰을 요청합니다

ONTAP 및 인증 서버가 정의되고 활성화되어 있으면 OAuth 2.0 토큰을 사용하여 REST API 호출을 수행할 수 있습니다. 첫 번째 단계는 인증 서버에서 액세스 토큰을 요청하는 것입니다. 이 작업은 서버에 기반한 여러 가지 다른 기술 중 하나를 사용하여 ONTAP 외부에서 수행됩니다. ONTAP는 액세스 토큰을 발급하거나 리디렉션을 수행하지 않습니다.

HTTP 요청 헤더를 생성합니다

액세스 토큰을 얻은 후 권한 부여 헤더를 생성하고 HTTP 요청에 포함할 수 있습니다. curl 또는 프로그래밍 언어를 사용하여 REST API에 액세스하든 관계없이 모든 클라이언트 요청에 헤더를 포함해야 합니다. 다음과 같이 헤더를 생성할 수 있습니다.

```
Authorization: Bearer eyJhbGciOiJSUzI1NiIsInR5cCIgOiAiSld …
```

Bash와 함께 예제 사용

워크플로 컬 예제를 직접 사용하는 경우 해당 변수에 포함된 변수를 환경에 적합한 값으로 업데이트해야 합니다. 예제를 수동으로 편집하거나 아래 설명된 대로 Bash 셸을 사용하여 대신 사용할 수 있습니다.

Bash를 사용하면 curl 명령마다 한 번 설정하는 대신 셸 세션에서 한 번 변수 값을 설정할 수 있다는 이점이 있습니다.

단계

1. Linux 또는 유사한 운영 체제와 함께 제공되는 Bash 셸을 엽니다.
2. 실행할 컬링 예제에 포함된 변수 값을 설정합니다. 예를 들면 다음과 같습니다.

   ```
   CLUSTER_ID=ce559b75-4145-11ee-b51a-005056aee9fb
   ```

3. 워크플로 페이지에서 컬링 예제를 복사하여 셸 터미널에 붙여 넣습니다.
4. Enter * 키를 누르면 다음 작업이 수행됩니다.
   a. 설정한 변수 값으로 대체합니다
   b. curl 명령을 실행합니다

클러스터

클러스터 구성을 가져옵니다

특정 필드를 포함하여 ONTAP 클러스터에 대한 구성을 검색할 수 있습니다. 이 작업은 클러스터의 상태를 평가하거나 구성을 업데이트하기 전에 수행할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/클러스터

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
필드	쿼리	아니요	반환할 값을 선택합니다. 예를 들면 다음과 같습니다 `contact` 및 `version`.

curl 예: 클러스터 연락처 정보를 검색합니다

이 예제에서는 단일 필드를 검색하는 방법을 보여 줍니다. 전체 클러스터 개체 및 구성을 가져오려면 를 제거해야 합니다 `fields` 쿼리 매개 변수입니다.

```
curl --request GET \
--location "https://$FQDN_IP/api/cluster?fields=contact" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예

```
{
  "contact": "support@company-demo.com"
}
```

클러스터 연락처를 업데이트합니다

클러스터의 연락처 정보를 업데이트할 수 있습니다. 요청이 비동기적으로 처리되므로 연결된 백그라운드 작업이 성공적으로 완료되었는지 여부도 확인해야 합니다.

1단계: 클러스터 연락처 정보를 업데이트합니다

API 호출을 실행하여 클러스터 연락처 정보를 업데이트할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
패치	/api/클러스터

처리 유형

비동기식

컬의 예

```
curl --request PATCH \
--location "https://$FQDN_IP/api/cluster" \
--include \
--header "Content-Type: application/json" \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "contact": "support@company-demo.com"
}
```

JSON 출력 예

작업 객체가 반환됩니다. 다음 단계에서 사용할 작업 식별자를 저장해야 합니다.

```
{ "job": {
    "uuid": "d877f5bb-3aa7-11e9-b6c6-005056a78c89",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/d877f5bb-3aa7-11e9-b6c6-005056a78c89"
      }
    }
  }
}
```

2단계**:** 작업 상태를 검색합니다

워크플로우를 수행합니다 "작업 인스턴스를 가져옵니다" 를 확인합니다 `state` 값은 입니다 `success`.

3단계**:** 클러스터 연락처 정보 확인

워크플로우를 수행합니다 "클러스터 구성을 가져옵니다". 를 설정해야 합니다 `fields` 에 쿼리 매개 변수 `contact`.

작업 인스턴스를 가져옵니다

특정 ONTAP 작업의 인스턴스를 검색할 수 있습니다. 일반적으로 이 작업을 수행하여 작업 및 관련 작업이 성공적으로 완료되었는지 확인할 수 있습니다.

일반적으로 비동기 요청을 실행한 후 제공되는 작업 개체의 UUID가 필요합니다. 검토도 합니다 "작업 개체를 사용한 비동기 처리" ONTAP 내부 작업으로 작업하기 전에

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/cluster/job/{uuid}

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$JOB_ID입니다	경로	예	요청되는 작업을 식별하는 데 필요합니다.

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/cluster/jobs/$JOB_ID" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예

상태 값과 기타 필드는 반환된 작업 오브젝트에 포함됩니다. 이 예의 작업은 ONTAP 클러스터를 업데이트하는 과정에서 실행되었습니다.

```
{
  "uuid": "d877f5bb-3aa7-11e9-b6c6-005056a78c89",
  "description": "PATCH /api/cluster",
  "state": "success",
  "message": "success",
  "code": 0,
  "_links": {
    "self": {
      "href": "/api/cluster/jobs/d877f5bb-3aa7-11e9-b6c6-005056a78c89"
    }
  }
}
```

NAS

파일 보안 권한

파일 보안 및 감사 정책 관리 준비

ONTAP 클러스터 내에서 SVM을 통해 사용할 수 있는 파일에 대한 권한 및 감사 정책을 관리할 수 있습니다.

개요

ONTAP에서는 SACL(시스템 액세스 제어 목록) 및 DACL(임의 액세스 제어 목록)을 사용하여 파일 개체에 사용 권한을 할당합니다. ONTAP 9.9.1부터 REST API는 SACL 및 DACL 권한 관리를 지원합니다. API를 사용하여 파일 보안 권한 관리를 자동화할 수 있습니다. 대부분의 경우 여러 CLI 명령 또는 ONTAPI(ZAPI) 호출 대신 단일 REST API 호출을 사용할 수 있습니다.

> 9.9.1 이전 버전의 ONTAP 릴리스에서는 CLI 통과 기능을 사용하여 SACL 및 DACL 권한 관리를 자동화할 수 있습니다. 을 참조하십시오 "마이그레이션 고려 사항" 및 "전용 CLI를 사용하면 ONTAP REST API를 통해 패스스루를 수행할 수 있습니다" 를 참조하십시오.

REST API를 사용하여 ONTAP 파일 보안 서비스를 관리하는 방법을 보여주는 몇 가지 예제 워크플로우를 사용할 수 있습니다. 워크플로우를 사용하고 REST API 호출을 전송하기 전에 를 검토하십시오 "워크플로우 사용을 준비하십시오".

Python을 사용하는 경우 스크립트도 참조하십시오 "file_security_permissions.py" 일부 파일 보안 작업을 자동화하는 방법에 대한 예를 참조하십시오.

ONTAP REST API와 **ONTAP CLI** 명령 비교

대부분의 작업에서 ONTAP REST API를 사용할 경우 동일한 ONTAP CLI 명령 또는 ONTAPI(ZAPI) 호출보다 더 적은 수의 호출이 필요합니다. 아래 표에는 API 호출 목록과 각 작업에 필요한 CLI 명령 목록이 나와 있습니다.

ONTAP REST API를 참조하십시오	**ONTAP CLI**를 참조하십시오
'get/protocols/file-security/Effective-permissions/'	'vserver security file-directory show-Effective-permissions'를 선택합니다
'POST/PROTOCOLS/FILE-SECURITY/permissions/'	1. 'vserver security file-directory NTFS create' 2. 'vserver security file-directory NTFS DACL add' 3. 'vserver security file-directory NTFS SACL add' 4. 'vserver security file-directory policy create'를 참조하십시오 5. 'vserver security file-directory policy task add' 6. 'vserver security file-directory apply'
'패치/프로토콜/파일-보안/권한/'	'vserver security file-directory NTFS modify'를 참조하십시오
삭제/프로토콜/파일-보안/권한/'	1. 'vserver security file-directory NTFS DACL remove' 2. 'vserver security file-directory NTFS SACL remove'

관련 정보

- "파일 권한을 보여 주는 Python 스크립트"
- "ONTAP REST API를 통해 파일 보안 권한 관리 간소화"
- "전용 CLI를 사용하면 ONTAP REST API를 통해 패스스루를 수행할 수 있습니다"

파일에 대한 유효 사용 권한을 얻습니다

특정 파일 또는 폴더에 대한 현재 유효 권한을 검색할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/protocols/file-security/effective-permissions/{svm.uuid}/{path}

처리 유형

동기식이다

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	파일이 포함된 SVM의 UUID입니다.
$FILE_PATH입니다	경로	예	파일 또는 폴더의 경로입니다.

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/protocols/file-security/effective-
permissions/$SVM_ID/$FILE_PATH" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "svm": {
    "uuid": "cf5f271a-1beb-11ea-8fad-005056bb645e",
    "name": "vs1"
  },
  "user": "administrator",
  "type": "windows",
  "path": "/",
  "share": {
    "path": "/"
  },
  "file_permission": [
    "read",
    "write",
    "append",
    "read_ea",
    "write_ea",
    "execute",
    "delete_child",
    "read_attributes",
    "write_attributes",
    "delete",
    "read_control",
    "write_dac",
    "write_owner",
    "synchronize",
    "system_security"
  ],
  "share_permission": [
    "read",
    "read_ea",
    "execute",
    "read_attributes",
    "read_control",
    "synchronize"
  ]
}
```


파일에 대한 감사 정보를 가져옵니다

특정 파일 또는 폴더에 대한 감사 정보를 검색할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/protocols/file-security/permissions/{svm.uuid}/{path}

처리 유형

동기식이다

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	파일이 포함된 SVM의 UUID입니다.
$FILE_PATH입니다	경로	예	파일 또는 폴더의 경로입니다.

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예

```
{
  "svm": {
    "uuid": "9479099d-5b9f-11eb-9c4e-0050568e8682",
    "name": "vs1"
  },
  "path": "/parent",
  "owner": "BUILTIN\\Administrators",
  "group": "BUILTIN\\Administrators",
  "control_flags": "0x8014",
  "acls": [
    {
      "user": "BUILTIN\\Administrators",
      "access": "access_allow",
      "apply_to": {
        "files": true,
        "sub_folders": true,
        "this_folder": true
      },
      "advanced_rights": {
```

```
        "append_data": true,
        "delete": true,
        "delete_child": true,
        "execute_file": true,
        "full_control": true,
        "read_attr": true,
        "read_data": true,
        "read_ea": true,
        "read_perm": true,
        "write_attr": true,
        "write_data": true,
        "write_ea": true,
        "write_owner": true,
        "synchronize": true,
        "write_perm": true
      },
      "access_control": "file_directory"
    },
    {
      "user": "BUILTIN\\Users",
      "access": "access_allow",
      "apply_to": {
        "files": true,
        "sub_folders": true,
        "this_folder": true
      },
      "advanced_rights": {
        "append_data": true,
        "delete": true,
        "delete_child": true,
        "execute_file": true,
        "full_control": true,
        "read_attr": true,
        "read_data": true,
        "read_ea": true,
        "read_perm": true,
        "write_attr": true,
        "write_data": true,
        "write_ea": true,
        "write_owner": true,
        "synchronize": true,
        "write_perm": true
      },
      "access_control": "file_directory"
    }
  ],
```

```
  "inode": 64,
  "security_style": "mixed",
  "effective_style": "ntfs",
  "dos_attributes": "10",
  "text_dos_attr": "----D---",
  "user_id": "0",
  "group_id": "0",
  "mode_bits": 777,
  "text_mode_bits": "rwxrwxrwx"
}
```

파일에 새 사용 권한을 적용합니다

특정 파일이나 폴더에 새 보안 설명자를 적용할 수 있습니다.

1단계**:** 새 사용 권한을 적용합니다

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/api/protocols/file-security/permissions/{svm.uuid}/{path}

처리 유형

비동기식

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	파일이 포함된 SVM의 UUID입니다.
$FILE_PATH입니다	경로	예	파일 또는 폴더의 경로입니다.

컬의 예

```
curl --request POST --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"acls\": [ { \"access\": \"access_allow\", \"advanced_rights\": {
\"append_data\": true, \"delete\": true, \"delete_child\": true,
\"execute_file\": true, \"full_control\": true, \"read_attr\": true,
\"read_data\": true, \"read_ea\": true, \"read_perm\": true,
\"write_attr\": true, \"write_data\": true, \"write_ea\": true,
\"write_owner\": true, \"write_perm\": true }, \"apply_to\": { \"files\":
true, \"sub_folders\": true, \"this_folder\": true }, \"user\":
\"administrator\" } ], \"control_flags\": \"32788\", \"group\": \"S-1-5-
21-2233347455-2266964949-1780268902-69700\", \"ignore_paths\": [
\"/parent/child2\" ], \"owner\": \"S-1-5-21-2233347455-2266964949-
1780268902-69304\", \"propagation_mode\": \"propagate\"}'
```

JSON 출력 예

```
{
  "job": {
    "uuid": "3015c294-5bbc-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/3015c294-5bbc-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

2단계: 작업 상태를 검색합니다

워크플로우를 수행합니다 "작업 인스턴스를 가져옵니다" 를 확인합니다 `state` 값은 입니다 `success`.

보안 설명자 정보를 업데이트합니다

기본 소유자, 그룹 또는 컨트롤 플래그를 포함하여 특정 파일 또는 폴더로 특정 보안 설명자를 업데이트할 수 있습니다.

1단계: 보안 설명자를 업데이트합니다

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
패치	/api/protocols/file-security/permissions/{svm.uuid}/{path}

처리 유형

비동기식

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	파일이 포함된 SVM의 UUID입니다.
$FILE_PATH입니다	경로	예	파일 또는 폴더의 경로입니다.

컬의 예

```
curl --request POST --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"control_flags\": \"32788\", \"group\": \"everyone\", \"owner\":
\"user1\"}'
```

JSON 출력 예

```
{
  "job": {
    "uuid": "6f89e612-5bbd-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/6f89e612-5bbd-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

2단계**:** 작업 상태를 검색합니다

워크플로우를 수행합니다 "작업 인스턴스를 가져옵니다" 를 확인합니다 `state` 값은 입니다 `success`.

액세스 제어 항목을 삭제합니다

특정 파일 또는 폴더에서 기존 ACE(액세스 제어 항목)를 삭제할 수 있습니다. 변경 내용이 자식 개체에 전파됩니다.

1단계: **ACE**를 삭제합니다

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
삭제	/api/protocols/file-security/permissions/{svm.uuid}/{path}

처리 유형

비동기식

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	파일이 포함된 SVM의 UUID입니다.
$FILE_PATH입니다	경로	예	파일 또는 폴더의 경로입니다.

컬의 예

```
curl --request DELETE --location "https://$FQDN_IP/api/protocols/file-
security/permissions/$SVM_ID/$FILE_PATH?return_timeout=0" --include
--header "Accept */*" --header "Authorization: Basic $BASIC_AUTH" --data
'{ \"access\": \"access_allow\", \"apply_to\": { \"files\": true,
\"sub_folders\": true, \"this_folder\": true }, \"ignore_paths\": [
\"/parent/child2\" ], \"propagation_mode\": \"propagate\"}'
```

JSON 출력 예

```
{
  "job": {
    "uuid": "3015c294-5bbc-11eb-9c4e-0050568e8682",
    "_links": {
      "self": {
        "href": "/api/cluster/jobs/3015c294-5bbc-11eb-9c4e-0050568e8682"
      }
    }
  }
}
```

2단계: 작업 상태를 검색합니다

워크플로우를 수행합니다 "작업 인스턴스를 가져옵니다" 를 확인합니다 `state` 값은 입니다 `success`.

네트워킹

IP 인터페이스를 나열합니다

클러스터 및 SVM에 할당된 IP LIF를 검색할 수 있습니다. 이렇게 하면 네트워크 구성을 확인하거나 다른 LIF를 추가하려고 할 때 사용할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/network/ip/interfaces 를 참조하십시오

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
필드	쿼리	아니요	관련 구성 값의 제한된 목록을 반환합니다.

curl 예: 기본 구성 값으로 모든 **LIF**를 반환합니다

```
curl --request GET \
--location "https://$FQDN_IP/api/network/ip/interfaces" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

curl 예: **4**가지 특정 구성 값을 가진 모든 **LIF**를 반환합니다

```
curl --request GET \
--location
"https://$FQDN_IP/api/network/ip/interfaces?fields=name,scope,svm.name,ip.
address" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "records": [
    {
      "uuid": "5ded9e38-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_mgmt1",
      "ip": {
        "address": "172.29.151.116"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/5ded9e38-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "bb03c162-999e-11ee-acad-005056ae6bd8",
      "name": "cluster_mgmt",
      "ip": {
        "address": "172.29.186.156"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/bb03c162-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c5ffbd03-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data1",
      "ip": {
        "address": "172.29.186.150"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c5ffbd03-999e-11ee-acad-
```

```
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c6612abe-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data2",
      "ip": {
        "address": "172.29.186.151"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c6612abe-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c6b21b94-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data3",
      "ip": {
        "address": "172.29.186.152"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c6b21b94-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c7025322-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data4",
      "ip": {
        "address": "172.29.186.153"
      },
      "scope": "svm",
      "svm": {
```

```
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c7025322-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c752cc66-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data5",
      "ip": {
        "address": "172.29.186.154"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c752cc66-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "c7a03719-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data6",
      "ip": {
        "address": "172.29.186.155"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/c7a03719-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "ccd4c59c-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data4_inet6",
```

```
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac5"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/ccd4c59c-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "d9144c30-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data6_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac7"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d9144c30-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "d961c13b-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data1_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac2"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d961c13b-999e-11ee-acad-
005056ae6bd8"
        }
```

```
      }
    },
    {
      "uuid": "d9ac8d6a-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data5_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac6"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d9ac8d6a-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "d9fce1a3-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data2_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac3"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/d9fce1a3-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "da4995a0-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_data3_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac4"
      },
      "scope": "svm",
      "svm": {
        "name": "vs0"
      },
```

```
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/da4995a0-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "da9e7afd-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_cluster_mgmt_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:300f::ac8"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/da9e7afd-999e-11ee-acad-
005056ae6bd8"
        }
      }
    },
    {
      "uuid": "e6db58b4-999e-11ee-acad-005056ae6bd8",
      "name": "sti214-vsim-sr027o_mgmt1_inet6",
      "ip": {
        "address": "fd20:8b1e:b255:3008::1a0"
      },
      "scope": "cluster",
      "_links": {
        "self": {
          "href": "/api/network/ip/interfaces/e6db58b4-999e-11ee-acad-
005056ae6bd8"
        }
      }
    }
  ],
  "num_records": 16,
  "_links": {
    "self": {
      "href":
"/api/network/ip/interfaces?fields=name,scope,svm.name,ip.address"
    }
  }
}
```

보안

계정

계정을 나열합니다

계정 목록을 검색할 수 있습니다. 보안 환경을 평가하거나 새 계정을 생성하기 전에 이 작업을 수행할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/security/accounts

처리 유형

동기식이다

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/security/accounts" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "records": [
    {
      "owner": {
        "uuid": "642573a8-9d14-11ee-9330-005056aed3de",
        "name": "vs0",
        "_links": {
          "self": {
            "href": "/api/svm/svms/642573a8-9d14-11ee-9330-
005056aed3de"
          }
        }
      },
      "name": "vsadmin",
      "_links": {
        "self": {
          "href": "/api/security/accounts/642573a8-9d14-11ee-9330-
005056aed3de/vsadmin"
        }
      }
    },
    {
      "owner": {
        "uuid": "fdb6fe29-9d13-11ee-9330-005056aed3de",
        "name": "sti214nscluster-1"
      },
      "name": "admin",
      "_links": {
        "self": {
          "href": "/api/security/accounts/fdb6fe29-9d13-11ee-9330-
005056aed3de/admin"
        }
      }
    },
    {
      "owner": {
        "uuid": "fdb6fe29-9d13-11ee-9330-005056aed3de",
        "name": "sti214nscluster-1"
      },
      "name": "autosupport",
      "_links": {
        "self": {
          "href": "/api/security/accounts/fdb6fe29-9d13-11ee-9330-
```

```
005056aed3de/autosupport"
        }
      }
    }
  ],
  "num_records": 3,
  "_links": {
    "self": {
      "href": "/api/security/accounts"
    }
  }
}
```

인증서 및 키

설치된 인증서를 나열합니다

ONTAP 클러스터에 설치된 인증서를 나열할 수 있습니다. 이렇게 하면 특정 인증서를 사용할 수 있는지 확인하거나 특정 인증서의 ID를 가져올 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/security/certificates 를 참조하십시오

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
Max_records(최대 레코드	쿼리	아니요	반환할 레코드 수를 지정합니다.

curl 예: 인증서 **3**개를 반환합니다

```
curl --request GET \
--location "https://$FQDN_IP/api/security/certificates?max_records=3" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "records": [
    {
      "uuid": "dad822c2-573c-11ee-a310-005056aecc29",
      "name": "vs0_17866DB5C933E2EA",
      "_links": {
        "self": {
          "href": "/api/security/certificates/dad822c2-573c-11ee-a310-
005056aecc29"
        }
      }
    },
    {
      "uuid": "7d8e5570-573c-11ee-a310-005056aecc29",
      "name": "BuypassClass3RootCA",
      "_links": {
        "self": {
          "href": "/api/security/certificates/7d8e5570-573c-11ee-a310-
005056aecc29"
        }
      }
    },
    {
      "uuid": "7dbb2191-573c-11ee-a310-005056aecc29",
      "name": "EntrustRootCertificationAuthority",
      "_links": {
        "self": {
          "href": "/api/security/certificates/7dbb2191-573c-11ee-a310-
005056aecc29"
        }
      }
    }
  ],
  "num_records": 3,
  "_links": {
    "self": {
      "href": "/api/security/certificates?max_records=3"
    },
    "next": {
      "href": "/api/security/certificates?start.svm_id=sti214nscluster-
1&start.uuid=7dbb2191-573c-11ee-a310-005056aecc29&max_records=3"
    }
  }
}
```

ONTAP 클러스터에 서명된 X.509 인증서를 설치할 수 있습니다. 강력한 인증이 필요한 ONTAP 기능이나 프로토콜을 구성할 때 이 작업을 수행할 수 있습니다.

시작하기 전에

설치할 인증서가 있어야 합니다. 또한 필요에 따라 중간 인증서가 설치되어 있는지 확인해야 합니다.

> 아래에 포함된 JSON 입력 예제를 사용하기 전에 를 업데이트해야 합니다 `public_certificate` 환경에 대한 인증서를 사용하여 얻을 수 있습니다.

1단계: 인증서를 설치합니다

API 호출을 실행하여 인증서를 설치할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/api/security/certificates 를 참조하십시오

curl 예: 클러스터 수준에서 루트 CA 인증서를 설치합니다

```
curl --request POST \
--location "https://$FQDN_IP/api/security/certificates" \
--include \
--header "Content-Type: application/json" \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예


```
{
    "type": "server_ca",
    "public_certificate":
    "-----BEGIN CERTIFICATE-----
MIID0TCCArkCFGYdznvTVvaY1VZPNfy4yCCyPph6MA0GCSqGSIb3DQEBCwUAMIGk
MQswCQYDVQQGEwJVUzELMAkGA1UECAwCTkMxDDAKBgNVBAcMA1JUUDEWMBQGA1UE
CgwNT05UQVAgRXhhbXBsZTETMBEGA1UECwwKT05UQVAgOS4xNDEcMBoGA1UEAwwT
Ki5vbnRhcC1leGFtcGxlLmNvbTEvMC0GCSqGSIb3DQEJARYgZGF2aWQucGV0ZXJz
b25Ab250YXAtZXhhbXBsZS5jb20wHhcNMjMxMDA1MTUyOTE4WhcNMjQxMDA0MTUy
OTE4WjCBpDELMAkGA1UEBhMCVVMxCzAJBgNVBAgMAk5DMQwwCgYDVQQHDANSVFAx
FjAUBgNVBAoMDU9OVEFQIEV4YW1wbGUxEzARBgNVBAsMCk9OVEFQIDkuMTQxHDAa
BgNVBAMMEyoub250YXAtZXhhbXBsZS5jb20xLzAtBgkqhkiG9w0BCQEWIGRhdmlk
LnBldGVyc29uQG9udGFwLWV4YW1wbGUuY29tMIIBIjANBgkqhkiG9w0BAQEFAAOC
AQ8AMIIBCgKCAQEAxQgy8mhb1Jhkf0D/MBodpzgW0aSp2jGbWJ+Zv2G8BXkp1762
dPHRkv1hnx9JvwkK4Dba05GiCiD5t3gjH/jUQMSFb+VwDbVmubVFnXjkm/4Q7sea
tMtA/ZpQdZbQFZ5RKtdWz7dzzPYEl2x8Q1Jc8Kh7NxERNMtgupGWZZn7mfXKYr4O
N/+vgahIhDibS8YK5rf1w6bfmrik9E2D+PEab9DX/1DL5RX4tZ1H2OkyN2UxoBR6
Fq7l6n1Hi/5yR0Oi1xStN6sO7EPoGak+KSlK41q+EcIKRo0bP4mEQp8WMjJuiTkb
5MmeYoIpWEUgJK7S0M6Tp/3bTh2CST3AWxiNxQIDAQABMA0GCSqGSIb3DQEBCwUA
A4IBAQABfBqOuROmYxdfrj93OyIiRoDcoMzvo8cHGNUsuhnlBDnL2O3qhWEs97s0
mIy6zFMGnyNYa0t4i1cFsGDKP/JuljmYHjvv+2lHWnxHjTo7AOQCnXmQH5swoDbf
o1Vjqz8Oxz+PRJ+PA3dF5/8zqaAR6QreAN/iFR++6nUq1sbbM7w03tthBVMgo/h1
E9I2jVOZsqMFujm2CYfMs4XkZtrYmN6nZA8JcUpDjIWcAVbQYurMnna9r42oS3GB
WB/FE9n+P+FfJyHJ93KGcCXbH5RF2pi3wLlHilbvVuCjLRrhJ8U20I5mZoiXvABc
IpYuBcuKXLwAarhDEacXttVjC+Bq
-----END CERTIFICATE-----"
}
```


2단계**:** 인증서가 설치되었는지 확인합니다

워크플로우를 수행합니다 "설치된 인증서를 나열합니다" 인증서를 사용할 수 있는지 확인합니다.

RBAC

RBAC를 사용하도록 준비합니다

ONTAP RBAC 기능은 환경에 따라 여러 가지 방법으로 사용할 수 있습니다. 이 섹션에서는 몇 가지 일반적인 시나리오를 워크플로로 설명합니다. 각각의 경우 특정 보안 및 관리 목표에 중점을 둡니다.

역할을 생성하고 ONTAP 사용자 계정에 역할을 할당하기 전에 아래에 제시된 주요 보안 요구 사항 및 옵션을 검토하여 준비해야 합니다. 또한 에서 일반적인 워크플로 개념을 검토해야 합니다 "워크플로우 사용을 준비하십시오".

어떤 **ONTAP** 릴리스를 사용하고 있습니까**?**

ONTAP 릴리스는 사용 가능한 REST 엔드포인트 및 RBAC 기능을 결정합니다.

보호된 리소스 및 범위를 식별합니다

보호할 리소스 또는 명령과 범위(클러스터 또는 SVM)를 식별해야 합니다.

사용자는 어떤 액세스 권한을 가져야 합니까**?**

리소스 및 범위를 파악한 후에는 부여할 액세스 수준을 결정해야 합니다.

사용자는 **ONTAP**에 어떻게 액세스합니까**?**

사용자는 REST API 또는 CLI 또는 둘 다를 통해 ONTAP에 액세스할 수 있습니다.

기본 제공 역할 중 하나가 충분합니까**,** 아니면 사용자 지정 역할이 필요합니까**?**

기존 기본 제공 역할을 사용하는 것이 더 편리하지만 필요한 경우 새 사용자 지정 역할을 만들 수 있습니다.

어떤 유형의 역할이 필요합니까**?**

보안 요구 사항 및 ONTAP 액세스를 기반으로 REST 또는 기존 역할을 생성할지 여부를 선택해야 합니다.

역할을 생성합니다

SVM 볼륨 작업으로 액세스 제한

SVM 내에서 스토리지 볼륨 관리를 제한하는 역할을 정의할 수 있습니다.

이 워크플로 정보

먼저 복제를 제외한 모든 주요 볼륨 관리 기능에 액세스할 수 있도록 기존 역할이 생성됩니다. 역할은 다음과 같은 특성으로 정의됩니다.

- 가져오기, 생성, 수정 및 삭제를 포함한 모든 CRUD 볼륨 작업을 수행할 수 있습니다
- 볼륨 클론을 생성할 수 없습니다

그런 다음 필요에 따라 역할을 업데이트할 수 있습니다. 이 워크플로우에서는 두 번째 단계에서 사용자가 볼륨 클론을 생성할 수 있도록 역할이 변경됩니다.

1단계**:** 역할을 만듭니다

API 호출을 실행하여 RBAC 역할을 생성할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/API/보안/역할

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "name": "role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      { "path": "volume create", "access": "all" },
      { "path": "volume delete", "access": "all" }
    ]
}
```

2단계**:** 역할을 업데이트합니다

API 호출을 실행하여 기존 역할을 업데이트할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/API/보안/역할

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	역할 정의가 포함된 SVM의 UUID입니다.
$ROLE_NAME입니다	경로	예	업데이트할 SVM 내에서 역할의 이름입니다.

컬의 예

```
curl --request POST \
--location
"https://$FQDN_IP/api/security/roles/$SVM_ID/$ROLE_NAME/priveleges" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "path": "volume clone",
  "access": "all"
}
```

데이터 보호 관리 활성화

사용자에게 제한된 데이터 보호 기능을 제공할 수 있습니다.

이 워크플로 정보

생성된 전통적인 역할은 다음과 같은 특성을 가지고 정의됩니다.

- SnapMirror 관계를 업데이트할 뿐만 아니라 스냅샷을 생성 및 삭제할 수 있습니다
- 볼륨 또는 SVM과 같은 상위 레벨의 객체를 생성하거나 수정할 수 없습니다

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/API/보안/역할

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "name": "role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "volume snapshot create", "access": "all"},
      {"path": "volume snapshot delete", "access": "all"},
      {"path": "volume show", "access": "readonly"},
      {"path": "vserver show", "access": "readonly"},
      {"path": "snapmirror show", "access": "readonly"},
      {"path": "snapmirror update", "access": "all"}
  ]
}
```

ONTAP 보고서 생성을 허용합니다

REST 역할을 생성하여 사용자에게 ONTAP 보고서를 생성할 수 있는 기능을 제공할 수 있습니다.

이 워크플로 정보

생성된 역할은 다음과 같은 특성으로 정의됩니다.

- 용량 및 성능(예: 볼륨, qtree, LUN, 애그리게이트, 노드, SnapMirror 관계 포함)
- 더 높은 수준의 오브젝트(예: 볼륨 또는 SVM)를 생성하거나 수정할 수 없음

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/API/보안/역할

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "name": "rest_role1",
  "owner": {
    "name": "cluster-1",
    "uuid": "852d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "/api/storage/volumes", "access": "readonly"},
      {"path": "/api/storage/qtrees", "access": "readonly"},
      {"path": "/api/storage/luns", "access": "readonly"},
      {"path": "/api/storage/aggregates", "access": "readonly"},
      {"path": "/api/cluster/nodes", "access": "readonly"},
      {"path": "/api/snapmirror/relationships", "access": "readonly"},
      {"path": "/api/svm/svms", "access": "readonly"}
  ]
}
```

역할이 있는 사용자를 생성합니다

이 워크플로를 사용하여 연결된 REST 역할을 가진 사용자를 만들 수 있습니다.

이 워크플로 정보

이 워크플로에는 사용자 지정 REST 역할을 만들고 새 사용자 계정과 연결하는 데 필요한 일반적인 단계가 포함되어 있습니다. 사용자와 역할 모두 SVM 범위를 가지고 있으며 특정 데이터 SVM과 연관됩니다. 일부 단계는 선택 사항이거나 환경에 따라 변경해야 할 수 있습니다.

1단계**:** 클러스터의 데이터 **SVM**을 나열합니다

다음 REST API 호출을 수행하여 클러스터에 SVM을 표시합니다. 각 SVM의 UUID 및 이름은 출력에 제공됩니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/svm/sSVM

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/svm/svms?order_by=name" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

작업을 마친 후

새 사용자 및 역할을 생성할 목록에서 원하는 SVM을 선택합니다.

2단계: SVM에 정의된 사용자를 나열합니다

다음 REST API 호출을 수행하여 선택한 SVM에 정의된 사용자를 나열할 수 있습니다. 소유자 매개 변수를 통해 SVM을 식별할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/security/accounts

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/security/accounts?owner.name=dmp" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

작업을 마친 후

SVM에 이미 정의된 사용자를 기준으로 새 사용자의 이름을 선택합니다.

3단계: SVM에 정의된 REST 역할 나열

선택한 SVM에 정의된 역할을 나열하려면 다음 REST API 호출을 수행합니다. 소유자 매개 변수를 통해 SVM을 식별할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/API/보안/역할

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/security/roles?owner.name=dmp" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

작업을 마친 후

SVM에 이미 정의된 역할에 따라 새 역할에 맞는 고유한 이름을 선택하십시오.

4단계: 사용자 지정 REST 역할을 만듭니다

SVM에서 맞춤형 REST API를 생성하여 다음 REST API 호출을 수행합니다. 이 역할에는 처음에 모든 액세스가 거부되도록 기본 액세스 권한을 * 없음 * 으로 설정하는 하나의 권한만 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/API/보안/역할

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/security/roles" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "name": "dprole1",
  "owner": {
    "name": "dmp",
    "uuid": "752d96be-f17c-11ec-9d19-005056bbad91"
  },
  "privileges": [
      {"path": "/api", "access": "none"},
  ]
}
```

작업을 마친 후

필요에 따라 3단계를 다시 수행하여 새 역할을 표시합니다. ONTAP CLI에서 역할을 표시할 수도 있습니다.

5단계: 권한을 추가하여 역할을 업데이트합니다

필요에 따라 권한을 추가하여 역할을 수정하려면 다음 REST API 호출을 수행합니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/api/security/roles/{owner.uuid}/{name}/권한

컬링 예제에 대한 추가 입력 매개 변수

모든 REST API 호출에 공통되는 매개 변수 외에 다음 매개 변수가 이 단계의 cURL 예에도 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
$SVM_ID입니다	경로	예	역할 정의가 포함된 SVM의 UUID입니다.
$ROLE_NAME입니다	경로	예	업데이트할 SVM 내의 역할 이름입니다.

컬의 예

```
curl --request POST \
--location
"https://$FQDN_IP/api/security/roles/$SVM_ID/$ROLE_NAME/privileges" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "path": "/api/storage/volumes",
  "access": "readonly"
}
```

작업을 마친 후

필요에 따라 3단계를 다시 수행하여 새 역할을 표시합니다. ONTAP CLI에서 역할을 표시할 수도 있습니다.

6단계**:** 사용자 생성

사용자 계정 생성을 위해 다음 REST API 호출을 수행합니다. 위에서 생성한 * dprole1 * 역할은 새 사용자와 연결됩니다.

역할 없이 사용자를 만들 수 있습니다. 이 경우 사용자에게 기본 역할(둘 중 하나)이 할당됩니다 `admin` 또는 `vsadmin`) 사용자가 클러스터 또는 SVM 범위로 정의되었는지 여부에 따라 결정됩니다. 다른 역할을 할당하도록 사용자를 수정해야 합니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/api/security/accounts

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/security/accounts" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "owner": {"uuid":"daf84055-248f-11ed-a23d-005056ac4fe6"},
  "name": "david",
  "applications": [
      {"application":"ssh",
       "authentication_methods":["password"],
       "second_authentication_method":"none"}
  ],
  "role":"dprole1",
  "password":"netapp123"
}
```

작업을 마친 후

새 사용자의 자격 증명을 사용하여 SVM 관리 인터페이스에 로그인할 수 있습니다.

스토리지

애그리게이트를 나열합니다

클러스터에서 애그리게이트 목록을 검색할 수 있습니다. 이를 통해 활용도와 성능을 평가할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/스토리지/디스크

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
node.name	쿼리	아니요	각 애그리게이트가 연결된 노드를 식별하는 데 사용될 수 있습니다.

curl 예: 기본 구성 값으로 모든 애그리게이트를 반환합니다

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/aggregates" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

curl 예: 특정 구성 값으로 모든 애그리게이트를 반환합니다

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/aggregates?fields=node.name" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예

```
{
  "records": [
    {
      "uuid": "760d8137-fc59-47da-906a-cc28db0a1c1b",
      "name": "sti214_vsim_sr027o_aggr1",
      "node": {
        "name": "sti214-vsim-sr027o"
      },
      "_links": {
        "self": {
          "href": "/api/storage/aggregates/760d8137-fc59-47da-906a-
cc28db0a1c1b"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/storage/aggregates?fields=node.name"
    }
  }
}
```

디스크를 나열합니다

클러스터에서 디스크 목록을 검색할 수 있습니다. 이렇게 하면 애그리게이트를 생성할 때 사용할 스페어를 하나 이상 찾을 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/스토리지/디스크

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
상태	쿼리	아니요	새 애그리게이트에 사용할 수 있는 스페어 디스크를 식별하는 데 사용할 수 있습니다.

curl 예: 모든 디스크를 반환합니다

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/disks" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

curl 예: 스페어 디스크를 반환합니다

```
curl --request GET \
--location "https://$FQDN_IP/api/storage/disks?state=spare" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "records": [
    {
      "name": "NET-1.20",
      "state": "spare",
      "_links": {
        "self": {
          "href": "/api/storage/disks/NET-1.20"
        }
      }
    },
    {
      "name": "NET-1.12",
      "state": "spare",
      "_links": {
        "self": {
          "href": "/api/storage/disks/NET-1.12"
        }
      }
    },
    {
      "name": "NET-1.7",
      "state": "spare",
      "_links": {
        "self": {
          "href": "/api/storage/disks/NET-1.7"
        }
      }
    }
  ],
  "num_records": 3,
  "_links": {
    "self": {
      "href": "/api/storage/disks?state=spare"
    }
  }
}
```


지원

있습니다

EMS 지원 서비스 관리 준비

ONTAP 클러스터에 대한 EMS(Event Management System) 처리를 설정하고 필요에 따라 EMS 메시지를 조회할 수 있다.

개요

ONTAP EMS 서비스 사용 방법을 보여 주는 몇 가지 예제 워크플로가 있습니다. 워크플로우를 사용하고 REST API 호출을 전송하기 전에 를 검토하십시오 "워크플로우 사용을 준비하십시오".

파이썬을 사용한다면, 성경도 참조하십시오 "events.py" 일부 EMS 관련 작업을 자동화하는 방법에 대한 예를 참조하십시오.

ONTAP REST API와 **ONTAP CLI** 명령 비교

대부분의 작업에서 ONTAP REST API를 사용할 경우 동일한 ONTAP CLI 명령보다 더 적은 수의 호출을 필요로 합니다. 아래 표에는 API 호출 목록과 각 작업에 필요한 CLI 명령 목록이 나와 있습니다.

ONTAP REST API를 참조하십시오	**ONTAP CLI**를 참조하십시오
GET/SUPPORT/EMS	이벤트 구성 쇼
POST/SUPPORT/EMS/목적지	1. 이벤트 알림 메시지 목적지 작성 2. 이벤트 알림 작성
'Get/support/EMS/events	이벤트 로그 쇼
'POST/support/EMS/filters	1. '이벤트 필터 생성 - 필터 이름 <filtername>' 2. '이벤트 필터 규칙 add-filter-name <filtername>'

관련 정보

- "EMS를 설명하는 Python 스크립트"
- "ONTAP REST API: 심각도가 높은 이벤트 자동 알림"

EMS 로그 이벤트 나열

모든 이벤트 알림 메시지 또는 특정 특성을 가진 메시지만 검색할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/support/ems/events 를 참조하십시오

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
필드	쿼리	아니요	응답에 포함할 특정 필드를 요청하는 데 사용됩니다.
Max_records(최대 레코드	쿼리	아니요	단일 요청에서 반환되는 레코드 수를 제한하는 데 사용할 수 있습니다.
로그_메시지	쿼리	아니요	특정 텍스트 값을 검색하고 일치하는 메시지만 반환하는 데 사용됩니다.
message.severity	쿼리	아니요	반환된 메시지를 과 같은 특정 심각도의 메시지로 제한합니다 `alert`.

curl 예: 최신 메시지와 이름 값을 반환합니다

```
curl --request GET \
--location
"https://$FQDN_IP/api/support/ems/events?fields=message.name&max_records=1
" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

curl 예: 특정 텍스트와 심각도가 포함된 메시지를 반환합니다

```
curl --request GET \
--location
"https://$FQDN_IP/api/support/ems/events?log_message=*disk*&message.severi
ty=alert" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "records": [
    {
      "node": {
        "name": "malha-vsim1",
        "uuid": "da4f9e62-9de3-11ec-976a-005056b369de",
        "_links": {
          "self": {
            "href": "/api/cluster/nodes/da4f9e62-9de3-11ec-976a-
005056b369de"
          }
        }
      },
      "index": 4602,
      "time": "2022-03-18T06:37:46-04:00",
      "message": {
        "severity": "alert",
        "name": "raid.autoPart.disabled"
      },
      "log_message": "raid.autoPart.disabled: Disk auto-partitioning is
disabled on this system: the system needs a minimum of 4 usable internal
hard disks.",
      "_links": {
        "self": {
          "href": "/api/support/ems/events/malha-vsim1/4602"
        }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href":
"/api/support/ems/events?log_message=*disk*&message.severity=alert&max_rec
ords=1"
    },
    "next": {
      "href": "/api/support/ems/events?start.keytime=2022-03-
18T06%3A37%3A46-04%3A00&start.node.name=malha-
vsim1&start.index=4602&log_message=*disk*&message.severity=alert"
    }
  }
}
```

EMS 구성을 가져옵니다

ONTAP 클러스터에 대한 현재 EMS 구성을 조회할 수 있다. 구성을 업데이트하거나 새 EMS 알림을 생성하기 전에 이 작업을 수행할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/support/EMS

처리 유형

동기식이다

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/support/ems" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예

```
{
  "proxy_url": "https://proxyserver.mycompany.com",
  "proxy_user": "proxy_user",
  "mail_server": "mail@mycompany.com",
  "_links": {
    "self": {
      "href": "/api/resourcelink"
    }
  },
  "pubsub_enabled": "1",
  "mail_from": "administrator@mycompany.com"
}
```

EMS Notification을 생성한다

다음 워크플로를 사용하여 선택한 이벤트 메시지를 수신할 새 EMS 알림 대상을 만들 수 있습니다.

1단계: 시스템 전체 이메일 설정을 구성합니다

다음 API 호출을 실행하여 시스템 전체 이메일 설정을 구성할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
패치	/api/support/EMS

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
메일_보낸 사람	쿼리	예	를 설정합니다 `from` 알림 전자 메일 메시지의 필드입니다.
메일_서버	쿼리	예	대상 SMTP 메일 서버를 구성합니다.

컬의 예

```
curl --request PATCH \
--location
"https://$FQDN_IP/api/support/ems?mail_from=administrator@mycompany.com&ma
il_server=mail@mycompany.com" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

2단계**:** 메시지 필터 정의

API 호출을 실행하여 메시지와 일치하는 필터 규칙을 정의할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/api/support/ems/filters 를 참조하십시오

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
필터	바디	예	필터 구성에 대한 값을 포함합니다.

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/support/ems/filters" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "name": "test-filter",
  "rules.type": ["include"],
  "rules.message_criteria.severities": ["emergency"]
}
```

3단계**:** 메시지 대상을 만듭니다

API 호출을 실행하여 메시지 대상을 생성할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
게시	/api/support/ems/destinations 를 참조하십시오

처리 유형

동기식이다

Curl 예제의 추가 입력 매개변수

모든 REST API 호출에서 일반적으로 사용되는 매개 변수 외에도 이 단계의 curl 예제에도 다음 매개 변수가 사용됩니다.

매개 변수	유형	필수 요소입니다	설명
대상 구성	바디	예	이벤트 대상에 대한 값을 포함합니다.

컬의 예

```
curl --request POST \
--location "https://$FQDN_IP/api/support/ems/destinations" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH" \
--data @JSONinput
```

JSON 입력 예

```
{
  "name": "test-destination",
  "type": "email",
  "destination": "administrator@mycompany.com",
  "filters.name": ["important-events"]
}
```

SVM

SVM을 나열합니다

ONTAP 클러스터 내에 정의된 SVM(스토리지 가상 머신)을 표시할 수 있습니다. 이 작업은 특정 SVM의 식별자를 찾거나 새로운 SVM을 생성하기 전에 이름이 고유한지 확인할 때 수행할 수 있습니다.

HTTP 메서드 및 끝점입니다

이 REST API 호출은 다음과 같은 메소드와 엔드포인트를 사용합니다.

HTTP 메소드	경로
가져오기	/api/svm/sSVM

컬의 예

```
curl --request GET \
--location "https://$FQDN_IP/api/svm/svms" \
--include \
--header "Accept: */*" \
--header "Authorization: Basic $BASIC_AUTH"
```

JSON 출력 예


```
{
  "records": [
    {
      "uuid": "71bd74f8-40dc-11ee-b51a-005056aee9fa",
      "name": "vs0",
      "_links": {
          "self": {
            "href": "/api/svm/svms/71bd74f8-40dc-11ee-b51a-005056aee9fa"
          }
      }
    }
  ],
  "num_records": 1,
  "_links": {
    "self": {
      "href": "/api/svm/svms"
    }
  }
}
```