



# OAuth 2.0을 사용한 인증 및 권한 부여

## ONTAP 9

NetApp  
April 24, 2024

# 목차

- OAuth 2.0을 사용한 인증 및 권한 부여 ..... 1
  - ONTAP OAuth 2.0 구축 개요 ..... 1
  - 개념 ..... 4
  - 구성 및 배포 ..... 15

# OAuth 2.0을 사용한 인증 및 권한 부여

## ONTAP OAuth 2.0 구축 개요

ONTAP 9.14부터 OAuth 2.0(Open Authorization 2.0) 프레임워크를 사용하여 ONTAP 클러스터에 대한 액세스를 제어할 수 있습니다. 이 기능은 ONTAP CLI, System Manager, REST API를 포함한 모든 ONTAP 관리 인터페이스를 사용하여 구성할 수 있습니다. 그러나 OAuth 2.0 권한 부여 및 액세스 제어 결정은 클라이언트가 REST API를 사용하여 ONTAP에 액세스할 때만 적용할 수 있습니다.



OAuth 2.0 지원은 ONTAP 9.14.0에서 처음 도입되었으며 사용 중인 ONTAP 릴리스에 따라 가용성이 달라집니다. 를 참조하십시오 ["ONTAP 릴리즈 노트"](#) 를 참조하십시오.

### 기능 및 이점

ONTAP와 함께 OAuth 2.0을 사용할 때의 주요 기능과 이점은 다음과 같습니다.

#### OAuth 2.0 표준을 지원합니다

OAuth 2.0은 업계 표준 인증 프레임워크입니다. 서명된 액세스 토큰을 사용하여 보호된 리소스에 대한 액세스를 제한하고 제어하는 데 사용됩니다. OAuth 2.0을 사용하면 다음과 같은 여러 이점이 있습니다.

- 권한 부여 구성에 대한 다양한 옵션
- 암호를 포함한 클라이언트 자격 증명을 공개하지 마십시오
- 토큰은 구성에 따라 만료되도록 설정할 수 있습니다
- REST API와 함께 사용하는 데 가장 적합합니다

널리 사용되는 여러 인증 서버를 사용하여 테스트했습니다

ONTAP 구현은 OAuth 2.0 호환 인증 서버와 호환되도록 설계되었습니다. 다음과 같은 일반적인 서버 또는 서비스를 사용하여 테스트되었습니다.

- Auth0
- ADFS(Active Directory Federation Service)
- 키클록

#### 여러 개의 동시 인증 서버 지원

단일 ONTAP 클러스터에 대해 최대 8개의 인증 서버를 정의할 수 있습니다. 따라서 다양한 보안 환경의 요구 사항을 충족할 수 있는 유연성을 제공합니다.

#### 나머지 역할과의 통합

ONTAP 권한 부여 결정은 궁극적으로 사용자 또는 그룹에 할당된 REST 역할을 기반으로 합니다. 이러한 역할은 액세스 토큰에서 자체 포함된 범위 또는 Active Directory 또는 LDAP 그룹과 함께 로컬 ONTAP 정의를 기반으로 수행됩니다.

보낸 사람 제한 액세스 토큰을 사용하는 옵션입니다

클라이언트 인증을 강화하는 MTL(상호 전송 계층 보안)을 사용하도록 ONTAP 및 인증 서버를 구성할 수 있습니다.

OAuth 2.0 액세스 토큰은 원래 발급된 클라이언트에서만 사용됩니다. 이 기능은 FAPI 및 MITRE에서 수립한 보안 권장 사항을 포함하여 몇 가지 일반적인 보안 권장 사항을 지원하고 그에 맞게 조정됩니다.

## 구현 및 구성

OAuth 2.0 구현 및 구성의 여러 측면을 개괄적으로 살펴보면 시작할 때 고려해야 할 사항이 있습니다.

### ONTAP 내의 OAuth 2.0 엔티티

OAuth 2.0 권한 부여 프레임워크는 데이터 센터 또는 네트워크 내의 실제 요소 또는 가상 요소에 매핑할 수 있는 여러 엔티티를 정의합니다. OAuth 2.0 엔티티 및 ONTAP에 대한 적응은 아래 표에 나와 있습니다.

OAuth 2.0 엔티티	설명
리소스	내부 ONTAP 명령을 통해 ONTAP 리소스에 액세스하는 REST API 엔드포인트입니다.
리소스 소유자	보호된 리소스를 생성했거나 기본적으로 소유한 ONTAP 클러스터 사용자입니다.
리소스 서버	ONTAP 클러스터인 보호된 리소스의 호스트입니다.
클라이언트	리소스 소유자를 대신하거나 리소스 소유자의 권한이 있는 REST API 끝점에 대한 액세스를 요청하는 응용 프로그램입니다.
인증 서버	일반적으로 액세스 토큰 발급 및 관리 정책 적용을 담당하는 전용 서버입니다.

### 핵심 ONTAP 구성

OAuth 2.0을 활성화하고 사용하려면 ONTAP 클러스터를 구성해야 합니다. 여기에는 인증 서버에 대한 연결 설정 및 필요한 ONTAP 인증 구성 정의가 포함됩니다. 다음과 같은 관리 인터페이스를 사용하여 이 구성을 수행할 수 있습니다.

- ONTAP 명령줄 인터페이스입니다
- 시스템 관리자
- ONTAP REST API를 참조하십시오

### 환경 및 지원 서비스

ONTAP 정의 외에 인증 서버도 구성해야 합니다. 그룹-역할 매핑을 사용하는 경우 Active Directory 그룹 또는 이에 상응하는 LDAP도 구성해야 합니다.

### 지원되는 ONTAP 클라이언트

ONTAP 9.14부터 REST API 클라이언트는 OAuth 2.0을 사용하여 ONTAP에 액세스할 수 있습니다. REST API 호출을 실행하기 전에 인증 서버에서 액세스 토큰을 얻어야 합니다. 그런 다음 클라이언트는 HTTP 승인 요청 헤더를 사용하여 이 토큰을 `_bearer token_` 으로 ONTAP 클러스터에 전달합니다. 필요한 보안 수준에 따라 클라이언트에서 인증서를 만들고 설치하여 MTL을 기반으로 보낸 사람 제한 토큰을 사용할 수도 있습니다.

## 선택된 용어

ONTAP를 사용하여 OAuth 2.0 배포를 살펴보기 시작하면 몇 가지 용어를 익히는 것이 좋습니다. 을 참조하십시오 ["추가 리소스"](#) OAuth 2.0에 대한 자세한 내용을 보려면 링크를 클릭하십시오.

### 액세스 토큰

인증 서버에서 발급되고 OAuth 2.0 클라이언트 응용 프로그램에서 보호된 리소스에 대한 액세스를 요청하는 데 사용하는 토큰입니다.

## JSON 웹 토큰

액세스 토큰을 포맷하는 데 사용되는 표준입니다. JSON은 OAuth 2.0 클레임을 세 개의 주요 섹션으로 정렬한 작은 형식으로 표현하는 데 사용됩니다.

## 보낸 사람 제한 액세스 토큰

MTL(상호 전송 계층 보안) 프로토콜을 기반으로 하는 선택적 기능입니다. 토큰에 추가 확인 클레임을 사용하면 액세스 토큰이 원래 발급된 클라이언트에서만 액세스 토큰을 사용할 수 있습니다.

## JSON 웹 키 집합입니다

JWKS는 ONTAP에서 클라이언트가 제시한 JWT 토큰을 확인하기 위해 사용하는 공개 키 모음입니다. 키 세트는 일반적으로 전용 URI를 통해 인증 서버에서 사용할 수 있습니다.

## 범위

범위를 사용하면 ONTAP REST API와 같은 보호된 리소스에 대한 응용 프로그램의 액세스를 제한하거나 제어할 수 있습니다. 액세스 토큰은 문자열로 표시됩니다.

## ONTAP REST 역할입니다

ONTAP 9.6에 도입된 REST 역할은 ONTAP RBAC 프레임워크의 핵심 부분입니다. 이러한 역할은 ONTAP에서 여전히 지원하는 이전의 기존 역할과 다릅니다. ONTAP의 OAuth 2.0 구현은 REST 역할만 지원합니다.

## HTTP 권한 부여 헤더

REST API 호출의 일부로 클라이언트 및 관련 권한을 식별하기 위한 HTTP 요청에 포함된 헤더 인증 및 권한 부여가 수행되는 방법에 따라 몇 가지 기능 또는 구현이 가능합니다. ONTAP에 OAuth 2.0 액세스 토큰을 제시할 때 토큰은 *bearer token* 으로 식별됩니다.

## HTTP 기본 인증

초기 HTTP 인증 기법은 여전히 ONTAP에서 지원됩니다. 일반 텍스트 자격 증명(사용자 이름 및 암호)은 콜론으로 연결되고 base64로 인코딩됩니다. 이 문자열은 승인 요청 헤더에 배치되고 서버로 전송됩니다.

## 파피

금융 업계를 위한 프로토콜, 데이터 스키마 및 보안 권장 사항을 제공하는 OpenID Foundation의 작업 그룹입니다. 이 API는 원래 금융 등급 API로 알려져 있었습니다.

## 마이터

미국 공군과 미국 정부에 기술 및 보안 지침을 제공하는 비영리 민간 회사입니다.

## 추가 리소스

몇 가지 추가 리소스가 아래에 제공됩니다. OAuth 2.0 및 관련 표준에 대한 자세한 내용을 보려면 이러한 사이트를 검토해야 합니다.

## 프로토콜 및 표준

- ["RFC 6749: OAuth 2.0 인증 프레임워크"](#)
- ["RFC 7519: JSON 웹 토큰\(JWT\)"](#)
- ["RFC 7523: OAuth 2.0 클라이언트 인증 및 권한 부여에 대한 JSON 웹 토큰\(JWT\) 프로파일"](#)
- ["RFC 7662: OAuth 2.0 토큰 소개"](#)
- ["RFC 7800: JWT에 대한 소유 증명 키"](#)

- "RFC 8705: OAuth 2.0 상호 TLS 클라이언트 인증 및 인증서 바인딩된 액세스 토큰"

#### 조직

- "OpenID 파운데이션"
- "FAPI 작업 그룹"
- "마이터"
- "IANA-JWT의 약어입니다"

#### 제품 및 서비스

- "Auth0"
- "ADFS 개요"
- "키클록"

#### 추가 도구 및 유틸리티

- "Auth0에 의한 JWT"
- "OpenSSL 을 참조하십시오"

#### NetApp 설명서 및 리소스

- "ONTAP 자동화" 문서화

## 개념

### 인증 서버 및 액세스 토큰

인증 서버는 OAuth 2.0 권한 부여 프레임워크 내에서 중앙 구성 요소로 몇 가지 중요한 기능을 수행합니다.

#### OAuth 2.0 인증 서버

권한 부여 서버는 주로 액세스 토큰을 만들고 서명합니다. 이러한 토큰에는 클라이언트 응용 프로그램이 보호된 리소스에 선택적으로 액세스할 수 있도록 하는 ID 및 권한 부여 정보가 포함되어 있습니다. 서버는 일반적으로 서로 격리되며 독립 실행형 전용 서버나 대규모 ID 및 액세스 관리 제품의 일부로 구현하는 등 여러 가지 방법으로 구현할 수 있습니다.



인증 서버에는 다른 용어를 사용할 수 있습니다. 특히 OAuth 2.0 기능이 보다 큰 ID 및 액세스 관리 제품 또는 솔루션 내에 패키징되어 있는 경우 더욱 그렇습니다. 예를 들어, \* ID 공급자(IDP) \* 라는 용어는 \* 인증 서버 \* 와 같은 의미로 사용되는 경우가 많습니다.

#### 관리

권한 부여 서버는 액세스 토큰을 발급하는 것 외에도 일반적으로 웹 사용자 인터페이스를 통해 관련 관리 서비스를 제공합니다. 예를 들어 다음을 정의하고 관리할 수 있습니다.

- 사용자 및 사용자 인증
- 범위

- 테넌트 및 영역을 통한 관리 분리
- 정책 적용
- 다양한 외부 서비스에 연결
- 기타 ID 프로토콜(예: SAML) 지원

ONTAP는 OAuth 2.0 표준과 호환되는 인증 서버와 호환됩니다.

#### ONTAP로 정의

ONTAP에 하나 이상의 인증 서버를 정의해야 합니다. ONTAP는 각 서버와 안전하게 통신하여 토큰을 확인하고 클라이언트 응용 프로그램을 지원하는 기타 관련 작업을 수행합니다.

ONTAP 구성의 주요 측면은 다음과 같습니다. 도 참조하십시오 ["OAuth 2.0 배포 시나리오"](#) 를 참조하십시오.

액세스 토큰의 유효성 검사 방법 및 위치

액세스 토큰의 유효성을 검사하는 방법에는 두 가지가 있습니다.

- 로컬 검증

ONTAP는 토큰을 발급한 인증 서버에서 제공한 정보를 기반으로 액세스 토큰의 유효성을 로컬로 검사할 수 있습니다. 인증 서버에서 검색된 정보는 ONTAP에 의해 캐시되고 정기적으로 새로 고쳐집니다.

- 원격 자기 주도

또한 인증 서버에서 토큰의 유효성을 검사하기 위해 원격 검사를 사용할 수도 있습니다. introspection은 권한이 있는 사용자가 인증 서버에 액세스 토큰을 쿼리할 수 있도록 하는 프로토콜입니다. ONTAP는 액세스 토큰에서 특정 메타데이터를 추출하고 토큰의 유효성을 검사하는 방법을 제공합니다. ONTAP은 성능상의 이유로 일부 데이터를 캐싱합니다.

네트워크 위치

ONTAP가 방화벽 뒤에 있을 수 있습니다. 이 경우 프록시를 구성의 일부로 식별해야 합니다.

권한 부여 서버가 정의되는 방법

CLI, System Manager 또는 REST API를 포함한 관리 인터페이스를 사용하여 ONTAP에 권한 부여 서버를 정의할 수 있습니다. 예를 들어, CLI에서는 명령을 사용합니다 `security oauth2 client create`.

인증 서버 수입이다

단일 ONTAP 클러스터에 대해 최대 8개의 인증 서버를 정의할 수 있습니다. 발급사 또는 발급사/대상 그룹 클레임이 고유하면 동일한 인증 서버를 동일한 ONTAP 클러스터에 두 번 이상 정의할 수 있습니다. 예를 들어 Keycloak를 사용하면 다른 영역을 사용할 때 항상 이 경우가 발생합니다.

#### OAuth 2.0 액세스 토큰 사용

인증 서버에서 발급한 OAuth 2.0 액세스 토큰은 ONTAP에서 검증하고 REST API 클라이언트 요청에 대한 역할 기반 액세스 결정을 내리는 데 사용됩니다.

액세스 토큰을 가져오는 중입니다

REST API를 사용하는 ONTAP 클러스터에 정의된 인증 서버에서 액세스 토큰을 얻어야 합니다. 토큰을 얻으려면 인증

서버에 직접 연결해야 합니다.



ONTAP는 액세스 토큰을 발급하거나 클라이언트에서 인증 서버로 요청을 리디렉션하지 않습니다.

토큰을 요청하는 방법은 다음과 같은 여러 요인에 따라 달라집니다.

- 인증 서버 및 구성 옵션
- OAuth 2.0 보조금 유형
- 요청을 발급하는 데 사용되는 클라이언트 또는 소프트웨어 도구입니다

#### 허가 유형

A\_GRANT\_는 OAuth 2.0 액세스 토큰을 요청하고 수신하는 데 사용되는 네트워크 흐름 집합을 포함한 잘 정의된 프로세스입니다. 클라이언트, 환경 및 보안 요구 사항에 따라 여러 가지 다른 부여 형식을 사용할 수 있습니다. 인기 있는 보조금 유형 목록은 아래 표에 나와 있습니다.

허가 유형	설명
클라이언트 자격 증명입니다	ID 및 공유 암호 등 자격 증명만 사용하는 일반적인 부여 유형입니다. 클라이언트는 리소스 소유자와 밀접한 트러스트 관계를 갖는 것으로 간주됩니다.
암호	리소스 소유자 암호 자격 증명 부여 유형은 리소스 소유자가 클라이언트와 신뢰 관계가 설정된 경우에 사용할 수 있습니다. 레거시 HTTP 클라이언트를 OAuth 2.0으로 마이그레이션할 때도 유용합니다.
인증 코드	이는 기밀 클라이언트에 이상적인 보조금 유형이며 리디렉션 기반 흐름을 기반으로 합니다. 액세스 토큰을 가져오고 토큰을 새로 고치는 데 사용할 수 있습니다.

#### JWT 콘텐츠

OAuth 2.0 액세스 토큰은 JWT로 포맷됩니다. 콘텐츠는 사용자의 구성에 따라 인증 서버에서 만들어집니다. 그러나 토큰은 클라이언트 응용 프로그램에서 불투명합니다. 클라이언트는 토큰을 검사하거나 내용을 인식할 이유가 없습니다.

각 JWT 액세스 토큰에는 클레임 집합이 포함됩니다. 클레임은 권한 부여 서버의 관리 정의에 따라 발급자의 특성과 권한 부여를 설명합니다. 표준에 등록된 청구의 일부는 아래 표에 설명되어 있습니다. 모든 문자열은 대/소문자를 구분합니다.

청구	키워드	설명
발급사	아이에스에스주식회사	토큰을 발급한 보안 주체를 식별합니다. 신청 처리는 응용 프로그램에 따라 다릅니다.
제목	하위	토큰의 제목 또는 사용자입니다. 이름은 전역적으로 또는 로컬에서 고유하도록 범위가 지정됩니다.
대상	호주 달러	토큰을 받을 수신자입니다. 문자열 배열로 구현됩니다.
만료	만료	토큰이 만료되어 거부되어야 하는 시간입니다.

을 참조하십시오 ["RFC 7519: JSON 웹 토큰"](#) 를 참조하십시오.

## ONTAP 클라이언트 인증 옵션

ONTAP 클라이언트 인증을 사용자 지정하는 데 사용할 수 있는 몇 가지 옵션이 있습니다. 권한



부여 결정은 궁극적으로 액세스 토큰에 포함되어 있거나 액세스 토큰에서 파생된 ONTAP REST 역할을 기반으로 합니다.



만 사용할 수 있습니다 **"ONTAP REST 역할"** OAuth 2.0에 대한 권한 부여를 구성하는 경우. 이전 ONTAP의 기존 역할은 지원되지 않습니다.

## 소개

ONTAP 내의 OAuth 2.0 구현은 유연하고 강력하도록 설계되어 ONTAP 환경을 보호하는 데 필요한 옵션을 제공합니다. 상위 수준에서는 ONTAP 클라이언트 권한 부여를 정의하기 위한 세 가지 주요 구성 범주가 있습니다. 이러한 구성 옵션은 함께 사용할 수 없습니다.

ONTAP는 사용자의 구성에 따라 가장 적합한 단일 옵션을 적용합니다. 을 참조하십시오 **"ONTAP에서 액세스를 결정하는 방법"** ONTAP에서 액세스 결정을 내리기 위해 구성 정의를 처리하는 방법에 대한 자세한 내용을 참조하십시오.

### OAuth 2.0 독립형 범위

이러한 범위에는 각각 단일 문자열로 캡슐화된 하나 이상의 사용자 지정 REST 역할이 포함됩니다. ONTAP 역할 정의와는 독립적입니다. 인증 서버에서 이러한 범위 문자열을 정의해야 합니다.

#### 로컬 ONTAP 관련 REST 역할 및 사용자

구성에 따라 로컬 ONTAP ID 정의를 사용하여 액세스 결정을 내릴 수 있습니다. 옵션은 다음과 같습니다.

- 단일 이름 REST 역할입니다
- 사용자 이름과 로컬 ONTAP 사용자를 일치시킵니다

명명된 역할의 범위 구문은 \* ontap-role - \* <URL-encoded-ONTAP-role-name>입니다. 예를 들어, 역할이 "admin"인 경우 범위 문자열은 "ontap-role-admin"이 됩니다.

### Active Directory 또는 LDAP 그룹

로컬 ONTAP 정의를 검사했지만 액세스를 결정할 수 없는 경우 Active Directory("도메인") 또는 LDAP("nsswitch") 그룹이 사용됩니다. 그룹 정보는 다음 두 가지 방법 중 하나로 지정할 수 있습니다.

- OAuth 2.0 범위 문자열

그룹 멤버십을 가진 사용자가 없는 경우 클라이언트 자격 증명 흐름을 사용하여 기밀 응용 프로그램을 지원합니다. 범위의 이름은 \* ontap-group - \* <URL-encoded-ONTAP-group-name>여야 합니다. 예를 들어 그룹이 "development"인 경우 범위 문자열은 "ontap-group-development"가 됩니다.

- "그룹" 요구 사항

리소스 소유자(암호 부여) 흐름을 사용하여 ADFS에서 발급한 액세스 토큰에 사용됩니다.

### 자체 포함된 OAuth 2.0 범위

자체 포함 범위는 액세스 토큰으로 전달되는 문자열입니다. 각각은 완전한 사용자 지정 역할 정의이며 ONTAP에서 액세스 결정을 내리는 데 필요한 모든 것을 포함합니다. 범위는 ONTAP 자체 내에 정의된 모든 REST 역할과 별개입니다.

범위 문자열의 형식입니다

기본 수준에서 범위는 연속된 문자열로 표시되며 콜론으로 구분된 6개의 값으로 구성됩니다. 범위 문자열에 사용되는 매개 변수는 아래에 설명되어 있습니다.

## ONTAP 리터럴

범위는 리터럴 값으로 시작해야 합니다 `ontap` 소문자로 입력합니다. ONTAP에만 해당하는 범위를 식별합니다.

### 클러스터

범위가 적용되는 ONTAP 클러스터를 정의합니다. 값은 다음과 같습니다.

- 클러스터 UUID

단일 클러스터를 식별합니다.

- 별표(\*)

범위가 모든 클러스터에 적용됨을 나타냅니다.

ONTAP CLI 명령을 사용할 수 있습니다 `cluster identity show` 클러스터의 UUID를 표시합니다. 지정하지 않으면 범위가 모든 클러스터에 적용됩니다.

### 역할

자체 포함된 범위에 포함된 REST 역할의 이름입니다. 이 값은 ONTAP에서 검사하거나 ONTAP에 정의된 기존 REST 역할과 일치하지 않습니다. 이 이름은 로깅에 사용됩니다.

### 액세스 수준

이 값은 범위에서 API 끝점을 사용할 때 클라이언트 응용 프로그램에 적용되는 액세스 수준을 나타냅니다. 아래 표에 설명된 대로 6개의 값이 있습니다.

액세스 수준	설명
없음	지정된 끝점에 대한 모든 액세스를 거부합니다.
읽기 전용	GET를 사용하여 읽기 액세스만 허용합니다.
<code>read_create</code> 를 참조하십시오	POST를 사용하여 새 리소스 인스턴스를 만들고 읽기 액세스를 허용합니다.
<code>read_modify</code> 를 참조하십시오	패치를 사용하여 기존 리소스를 업데이트할 수 있을 뿐 아니라 읽기 액세스를 허용합니다.
<code>READ_CREATE_MODIFY</code> 을 참조하십시오	삭제를 제외한 모든 액세스를 허용합니다. 허용되는 작업에는 GET(읽기), POST(작성) 및 패치(업데이트)가 포함됩니다.
모두	전체 액세스를 허용합니다.

## SVM

클러스터 내 SVM의 이름이 범위에 적용됩니다. \* 값(별표)을 사용하여 모든 SVM을 나타냅니다.



이 기능은 ONTAP 9.14.1에서 완벽하게 지원되지 않습니다. SVM 매개 변수를 무시하고 별표를 자리 표시자로 사용할 수 있습니다. 를 검토합니다 ["ONTAP 릴리즈 노트"](#) 향후 SVM 지원 확인

## REST API URI입니다

리소스 또는 관련 리소스 집합에 대한 전체 또는 부분 경로입니다. 문자열은 로 시작해야 합니다 /api. 값을 지정하지 않으면 범위가 ONTAP 클러스터의 모든 API 끝점에 적용됩니다.

범위 예

다음은 자급식 범위의 몇 가지 예입니다.

**ONTAP: \*:joes-역할: read\_create\_modify: \*/api/cluster**

이 역할에 할당된 사용자에게 에 대한 읽기, 생성 및 수정 액세스 권한을 제공합니다 /cluster 엔드포인트.

## CLI 관리 도구

ONTAP는 자체 포함된 범위를 보다 쉽게 관리할 수 있도록 CLI 명령을 제공합니다 security oauth2 scope 입력 매개 변수를 기반으로 범위 문자열을 생성합니다.

명령을 입력합니다 security oauth2 scope 은 고객 입력에 따라 두 가지 사용 사례를 가지고 있습니다.

- 문자열 범위를 지정하는 CLI 매개 변수입니다

이 버전의 명령을 사용하여 입력 매개 변수를 기반으로 범위 문자열을 생성할 수 있습니다.

- 문자열을 CLI 매개 변수로 지정합니다

이 버전의 명령을 사용하여 입력 범위 문자열을 기반으로 명령 매개 변수를 생성할 수 있습니다.

예

다음 예제에서는 아래 명령 예제 다음에 포함된 출력으로 범위 문자열을 생성합니다. 이 정의는 모든 클러스터에 적용됩니다.

```
security oauth2 scope cli-to-scope -role joes-role -access readonly -api
/api/cluster
```

```
ontap:*:joes-role:readonly:*/api/cluster
```

## ONTAP에서 액세스를 결정하는 방법

OAuth 2.0을 올바르게 설계하고 구현하려면 ONTAP에서 클라이언트의 액세스 결정을 내리기 위해 인증 구성이 사용되는 방법을 이해해야 합니다.

### 1단계: 자체 포함 범위

액세스 토큰에 자체 포함된 범위가 포함되어 있는 경우 ONTAP에서는 해당 범위를 먼저 검사합니다. 자체 포함된 범위가 없는 경우 2단계로 이동합니다.

하나 이상의 자체 포함 범위가 있는 경우 ONTAP는 명시적 \* allow \* 또는 \* deny \* 결정을 내릴 수 있을 때까지 각

범위를 적용합니다. 명시적인 결정이 내려지면 처리가 종료됩니다.

ONTAP에서 명시적인 액세스 결정을 내릴 수 없는 경우 2단계를 계속 진행합니다.

#### 2단계: 로컬 역할 플래그를 확인합니다

ONTAP는 플래그 값을 검사합니다 `use-local-roles-if-present`. 이 플래그의 값은 ONTAP로 정의된 각 인증 서버에 대해 별도로 설정됩니다.

- 값이 `true` 이면 3단계를 계속 진행합니다.
- 값이 `false` 이면 처리가 종료되고 액세스가 거부됩니다.

#### 3단계: 이름이 지정된 **ONTAP REST** 역할입니다

액세스 토큰에 이름이 지정된 REST 역할이 포함된 경우 ONTAP는 해당 역할을 사용하여 액세스 결정을 내립니다. 이렇게 하면 항상 `* allow *` 또는 `* deny *` 결정이 되고 처리가 종료됩니다.

이름이 지정된 REST 역할이 없거나 역할을 찾을 수 없는 경우 4단계를 계속 진행하십시오.

#### 단계 4: 로컬 **ONTAP** 사용자

액세스 토큰에서 사용자 이름을 추출하여 로컬 ONTAP 사용자와 일치시키려고 시도합니다.

로컬 ONTAP 사용자가 일치하는 경우 ONTAP는 사용자에게 정의된 역할을 사용하여 액세스 결정을 내립니다. 이로 인해 항상 `* allow *` 또는 `* deny *` 결정이 내려지고 처리가 종료됩니다.

로컬 ONTAP 사용자가 일치하지 않거나 액세스 토큰에 사용자 이름이 없는 경우 5단계를 계속 진행합니다.

#### 5단계: 그룹-역할 매핑

액세스 토큰에서 그룹을 추출하고 그룹에 일치시키려고 시도합니다. 그룹은 Active Directory 또는 이에 상응하는 LDAP 서버를 사용하여 정의됩니다.

그룹 일치 항목이 있는 경우 ONTAP는 그룹에 대해 정의된 역할을 사용하여 액세스 결정을 내립니다. 이로 인해 항상 `* allow *` 또는 `* deny *` 결정이 내려지고 처리가 종료됩니다.

일치하는 그룹이 없거나 액세스 토큰에 그룹이 없으면 액세스가 거부되고 처리가 종료됩니다.

## OAuth 2.0 배포 시나리오

ONTAP에 대한 인증 서버를 정의할 때 사용할 수 있는 몇 가지 구성 옵션이 있습니다. 이러한 옵션에 따라 배포 환경에 적합한 인증 서버를 만들 수 있습니다.

#### 구성 매개 변수 요약

ONTAP에 대한 인증 서버를 정의할 때 사용할 수 있는 몇 가지 구성 매개 변수가 있습니다. 이러한 매개 변수는 일반적으로 모든 관리 인터페이스에서 지원됩니다.

매개 변수 이름은 ONTAP 관리 인터페이스에 따라 약간 다를 수 있습니다. 예를 들어, 원격 내부 조사를 구성할 때 끝점은 CLI 명령 매개 변수를 사용하여 식별됩니다 `-introspection-endpoint`. 그러나 System Manager의 경우, 해당 필드는 `_ 인증 서버 토큰 내부 URI _` 입니다. 모든 ONTAP 관리 인터페이스를 수용할 수 있도록 매개 변수에 대한 일반적인 설명이 제공됩니다. 정확한 매개 변수 또는 필드는 컨텍스트에 따라 명확해야 합니다.

매개 변수	설명
이름	ONTAP에 알려진 인증 서버의 이름입니다.
응용 프로그램	정의가 적용되는 ONTAP 내부 응용 프로그램입니다. 이 값은 * http * 여야 합니다.
발급자 URI입니다	토큰을 발급하는 사이트 또는 조직을 식별하는 경로가 있는 FQDN입니다.
공급자 JWKS URI입니다	ONTAP가 액세스 토큰의 유효성을 검사하는 데 사용되는 JSON 웹 키 세트를 가져오는 경로 및 파일 이름의 FQDN입니다.
JWKS 새로 고침 간격입니다	ONTAP가 공급자 JWKS URI에서 인증서 정보를 새로 고치는 빈도를 결정하는 시간 간격입니다. 값은 ISO-8601 형식으로 지정됩니다.
성찰의 끝점입니다	ONTAP에서 자체 조사를 통해 원격 토큰 유효성 검사를 수행하는 데 사용하는 경로가 있는 FQDN입니다.
클라이언트 ID입니다	인증 서버에 정의된 클라이언트의 이름입니다. 이 값이 포함된 경우 인터페이스를 기반으로 연결된 클라이언트 암호도 제공해야 합니다.
발신 프록시	이는 ONTAP가 방화벽 뒤에 있을 때 인증 서버에 대한 액세스를 제공하기 위한 것입니다. URI는 curl 형식이어야 합니다.
있는 경우 로컬 역할을 사용합니다	로컬 ONTAP 정의가 사용되는지 여부를 결정하는 부울 플래그(명명된 REST 역할 및 로컬 사용자 포함)
사용자 클레임을 제거합니다	ONTAP에서 로컬 사용자와 일치시키기 위해 사용하는 대체 이름입니다. 를 사용합니다 sub 로컬 사용자 이름과 일치하는 액세스 토큰의 필드입니다.

## 배포 시나리오

다음은 몇 가지 일반적인 배포 시나리오입니다. 토큰 유효성 검사는 ONTAP에서 로컬로 수행되는지 아니면 인증 서버에서 원격으로 수행되는지를 기준으로 구성됩니다. 각 시나리오에는 필요한 구성 옵션 목록이 포함되어 있습니다. 을 참조하십시오 ["ONTAP에 OAuth 2.0 배포"](#) 구성 명령의 예를 참조하십시오.



인증 서버를 정의한 후 ONTAP 관리 인터페이스를 통해 구성을 표시할 수 있습니다. 예를 들어, 명령을 사용합니다 `security oauth2 client show` ONTAP CLI 사용.

## 로컬 검증

다음 배포 시나리오는 토큰 유효성 검사를 로컬로 수행하는 ONTAP를 기반으로 합니다.

프록시 없이 자체 포함된 범위를 사용합니다

이것은 OAuth 2.0 자체 포함 범위만 사용하는 가장 간단한 배포입니다. 로컬 ONTAP ID 정의는 사용되지 않습니다. 다음 매개 변수를 포함해야 합니다.

- 이름
- 응용 프로그램(http)
- 공급자 JWKS URI입니다
- 발급자 URI입니다

또한 인증 서버에 범위를 추가해야 합니다.

프록시에 자체 포함된 범위를 사용합니다

이 배포 시나리오에서는 OAuth 2.0 자체 포함 범위를 사용합니다. 로컬 ONTAP ID 정의는 사용되지 않습니다. 하지만 인증 서버는 방화벽 뒤에 있으므로 프록시를 구성해야 합니다. 다음 매개 변수를 포함해야 합니다.

- 이름
- 응용 프로그램(http)
- 공급자 JWKS URI입니다
- 발신 프록시
- 발급자 URI입니다
- 대상

또한 인증 서버에 범위를 추가해야 합니다.

프록시에 로컬 사용자 역할 및 기본 사용자 이름 매핑을 사용합니다

이 배포 시나리오에서는 기본 이름 매핑과 함께 로컬 사용자 역할을 사용합니다. 원격 사용자 클레임은 기본값인 `sub` 액세스 토큰의 `sub` 필드는 로컬 사용자 이름과 일치시키는 데 사용됩니다. 사용자 이름은 40자 이하여야 합니다. 인증 서버는 방화벽 뒤에 있으므로 프록시를 구성해야 합니다. 다음 매개 변수를 포함해야 합니다.

- 이름
- 응용 프로그램(http)
- 공급자 JWKS URI입니다
- 있는 경우 로컬 역할을 사용합니다 (true)
- 발신 프록시
- 발급사

로컬 사용자가 ONTAP로 정의되었는지 확인해야 합니다.

프록시를 사용하여 로컬 사용자 역할 및 대체 사용자 이름 매핑을 사용합니다

이 배포 시나리오에서는 로컬 ONTAP 사용자와 일치시키는 데 사용되는 대체 사용자 이름과 함께 로컬 사용자 역할을 사용합니다. 인증 서버는 방화벽 뒤에 있으므로 프록시를 구성해야 합니다. 다음 매개 변수를 포함해야 합니다.

- 이름
- 응용 프로그램(http)
- 공급자 JWKS URI입니다
- 있는 경우 로컬 역할을 사용합니다 (true)
- 원격 사용자 클레임
- 발신 프록시
- 발급자 URI입니다
- 대상

로컬 사용자가 ONTAP로 정의되었는지 확인해야 합니다.

다음 배포 구성은 ONTAP을 기반으로 합니다. 이 구성은 자체 조사를 통해 토큰 유효성 검사를 원격으로 수행합니다.

프록시 없이 자체 포함된 범위를 사용합니다

OAuth 2.0 독립형 범위를 사용하여 간단하게 배포할 수 있습니다. ONTAP ID 정의는 사용되지 않습니다. 다음 매개 변수를 포함해야 합니다.

- 이름
- 응용 프로그램(http)
- 성찰의 끝점입니다
- 클라이언트 ID입니다
- 발급자 URI입니다

인증 서버에서 클라이언트 및 클라이언트 비밀은 물론 범위를 정의해야 합니다.

## 상호 TLS를 사용한 클라이언트 인증

보안 요구에 따라 강력한 클라이언트 인증을 구현하도록 MTL(Mutual TLS)을 선택적으로 구성할 수 있습니다. OAuth 2.0 배포의 일부로 ONTAP와 함께 사용할 경우 MTL은 액세스 토큰이 원래 발급된 클라이언트에서만 사용되도록 보장합니다.

### 상호 TLS와 OAuth 2.0

TLS(Transport Layer Security)는 두 애플리케이션(일반적으로 클라이언트 브라우저와 웹 서버)간에 보안 통신 채널을 설정하는 데 사용됩니다. 상호 TLS는 클라이언트 인증서를 통해 클라이언트를 강력하게 식별함으로써 이 기능을 확장합니다. OAuth 2.0이 있는 ONTAP 클러스터에서 사용할 경우 기본 MTL 기능은 보낸 사람 제한 액세스 토큰을 생성하고 사용하여 확장됩니다.

보낸 사람 제한 액세스 토큰은 원래 발급된 클라이언트에서만 사용할 수 있습니다. 이 기능을 지원하기 위해 새로운 확인 요청이 있습니다 (cnf)이 토큰에 삽입됩니다. 필드에 속성이 포함되어 있습니다 x5t#S256 액세스 토큰을 요청할 때 사용되는 클라이언트 인증서의 다이제스트가 들어 있습니다. 이 값은 토큰 유효성 검사의 일부로 ONTAP에서 확인합니다. 보낸 사람 제한이 없는 인증 서버에서 발급한 액세스 토큰에는 추가 확인 클레임이 포함되지 않습니다.

각 인증 서버에 대해 MTL을 별도로 사용하도록 ONTAP를 구성해야 합니다. 예를 들어, CLI 명령을 사용할 수 있습니다 security oauth2 client 매개 변수를 포함합니다 use-mutual-tls 아래 표와 같이 세 가지 값을 기반으로 MTL 처리를 제어합니다.



각 구성에서 ONTAP가 수행한 결과 및 작업은 구성 매개 변수 값과 액세스 토큰 및 클라이언트 인증서의 내용에 따라 달라집니다. 표의 매개 변수는 최소 값부터 최대 제한값까지 구성됩니다.

매개 변수	설명
없음	인증 서버에 대해 OAuth 2.0 상호 TLS 인증이 완전히 비활성화되었습니다. 토큰에 확인 클레임이 있거나 TLS 연결과 함께 클라이언트 인증서가 제공된 경우에도 ONTAP는 MTL 클라이언트 인증서 인증을 수행하지 않습니다.
요청	OAuth 2.0 상호 TLS 인증은 클라이언트가 보낸 사람 제한 액세스 토큰을 제공하는 경우 적용됩니다. 즉, MTL은 확인 요청(속성 포함)이 있는 경우에만 적용됩니다 x5t#S256 )이 액세스 토큰에 있습니다. 기본 설정입니다.

매개 변수	설명
필수 요소입니다	OAuth 2.0 상호 TLS 인증은 인증 서버에서 발급한 모든 액세스 토큰에 대해 적용됩니다. 따라서 모든 액세스 토큰은 sender-constraint여야 합니다. 액세스 토큰에 확인 클레임이 없거나 잘못된 클라이언트 인증서가 있는 경우 인증 및 REST API 요청이 실패합니다.

## 높은 수준의 구현 흐름

ONTAP 환경에서 OAuth 2.0과 함께 MTL을 사용할 때 적용되는 일반적인 단계는 다음과 같습니다. 을 참조하십시오 ["RFC 8705: OAuth 2.0 상호 TLS 클라이언트 인증 및 인증서 바인딩된 액세스 토큰"](#) 를 참조하십시오.

### 1단계: 클라이언트 인증서를 생성하고 설치합니다

클라이언트 ID 설정은 클라이언트 개인 키에 대한 지식을 입증하기 위한 것입니다. 해당 공개 키는 클라이언트에서 제공하는 서명된 X.509 인증서에 저장됩니다. 클라이언트 인증서 만들기과 관련된 단계는 다음과 같습니다.

1. 공개 및 개인 키 쌍을 생성합니다
2. 인증서 서명 요청을 만듭니다
3. CSR 파일을 잘 알려진 CA로 보냅니다
4. CA에서 요청을 확인하고 서명된 인증서를 발급합니다

일반적으로 클라이언트 인증서를 로컬 운영 체제에 설치하거나 curl과 같은 공통 유틸리티를 사용하여 직접 사용할 수 있습니다.

### 2단계: MTL을 사용하도록 ONTAP를 구성합니다

MTL을 사용하도록 ONTAP을 구성해야 합니다. 이 구성은 각 인증 서버에 대해 별도로 수행됩니다. 예를 들어, CLI를 사용하면 명령을 사용할 수 있습니다 `security oauth2 client` 선택적 매개 변수와 함께 사용됩니다 `use-mutual-tls`. 을 참조하십시오 ["ONTAP에 OAuth 2.0 배포"](#) 를 참조하십시오.

### 3단계: 클라이언트가 액세스 토큰을 요청합니다

클라이언트는 ONTAP로 구성된 인증 서버에서 액세스 토큰을 요청해야 합니다. 클라이언트 응용 프로그램은 1단계에서 생성하고 설치한 인증서가 있는 MTL을 사용해야 합니다.

### 4단계: 인증 서버가 액세스 토큰을 생성합니다

인증 서버는 클라이언트 요청을 확인하고 액세스 토큰을 생성합니다. 이 과정에서 클라이언트 인증서의 메시지 다이제스트가 생성되며, 이 다이제스트는 토큰에 확인 클레임(필드)으로 포함됩니다 `cnf`)를 클릭합니다.

### 5단계: 클라이언트 애플리케이션이 ONTAP에 액세스 토큰을 제공합니다

클라이언트 응용 프로그램은 ONTAP 클러스터에 REST API 호출을 수행하고 권한 부여 요청 헤더에 액세스 토큰을 \* 베어러 토큰 \* 으로 포함합니다. 클라이언트는 액세스 토큰을 요청하는 데 사용된 것과 동일한 인증서를 가진 MTL을 사용해야 합니다.

### 6단계: ONTAP는 클라이언트와 토큰을 확인합니다.

ONTAP는 MTL 처리의 일부로 사용되는 클라이언트 인증서뿐만 아니라 HTTP 요청으로 액세스 토큰을 받습니다. ONTAP는 먼저 액세스 토큰의 서명을 확인합니다. 구성에 따라 ONTAP는 클라이언트 인증서의 메시지 다이제스트를 생성하고 토큰의 확인 클레임 \* CNF \* 와 비교합니다. 두 값이 일치하면 ONTAP는 API 요청을 하는 클라이언트가 액세스 토큰이 원래 발급된 클라이언트와 동일하다는 것을 확인했습니다.



# 구성 및 배포

## ONTAP과 함께 OAuth 2.0을 배포할 준비를 하십시오

ONTAP 환경에서 OAuth 2.0을 구성하기 전에 배포를 준비해야 합니다. 주요 작업과 결정에 대한 요약이 아래에 나와 있습니다. 섹션의 정렬은 일반적으로 따라야 할 순서에 맞춰집니다. 그러나 대부분의 배포에는 적용되지만 필요에 따라 환경에 맞게 조정해야 합니다. 공식 배포 계획을 작성하는 것도 고려해야 합니다.



사용자 환경에 따라 ONTAP에 정의된 인증 서버에 대한 구성을 선택할 수 있습니다. 여기에는 각 배포 유형에 대해 구체화해야 하는 매개 변수 값이 포함됩니다. 을 참조하십시오 ["OAuth 2.0 배포 시나리오"](#) 를 참조하십시오.

### 보호된 리소스 및 클라이언트 응용 프로그램

OAuth 2.0은 보호된 리소스에 대한 액세스를 제어하기 위한 권한 부여 프레임워크입니다. 이 점을 감안하면 모든 배포에서 중요한 첫 단계는 사용 가능한 리소스가 무엇이고 어떤 클라이언트가 액세스할 필요가 있는지 확인하는 것입니다.

### 클라이언트 애플리케이션을 식별합니다

REST API 호출을 실행할 때 OAuth 2.0을 사용할 클라이언트와 이들이 액세스해야 하는 API 엔드포인트를 결정해야 합니다.

### 기존 ONTAP REST 역할 및 로컬 사용자를 검토합니다

REST 역할 및 로컬 사용자를 포함하여 기존 ONTAP ID 정의를 검토해야 합니다. OAuth 2.0을 구성하는 방법에 따라 이러한 정의를 액세스 결정에 사용할 수 있습니다.

### OAuth 2.0으로의 글로벌 전환

OAuth 2.0 인증을 점진적으로 구현할 수도 있지만 각 인증 서버에 대한 글로벌 플래그를 설정하여 모든 REST API 클라이언트를 OAuth 2.0으로 즉시 이동할 수도 있습니다. 따라서 자체 포함된 범위를 만들 필요 없이 기존 ONTAP 구성을 기반으로 액세스 결정을 내릴 수 있습니다.

### 인증 서버

권한 부여 서버는 액세스 토큰을 발행하고 관리 정책을 시행함으로써 OAuth 2.0 배포에서 중요한 역할을 수행합니다.

### 인증 서버를 선택하여 설치합니다

하나 이상의 인증 서버를 선택하여 설치해야 합니다. 범위를 정의하는 방법을 비롯하여 ID 공급자의 구성 옵션 및 절차를 숙지하는 것이 중요합니다.

### 인증 루트 CA 인증서를 설치해야 하는지 확인합니다

ONTAP는 인증 서버의 인증서를 사용하여 클라이언트가 제공하는 서명된 액세스 토큰의 유효성을 검사합니다. 이렇게 하려면 ONTAP에서 루트 CA 인증서와 모든 중간 인증서가 필요합니다. ONTAP와 함께 사전 설치되어 있을 수 있습니다. 그렇지 않은 경우 설치해야 합니다.

### 네트워크 위치 및 구성을 평가합니다

인증 서버가 방화벽 뒤에 있는 경우 프록시 서버를 사용하도록 ONTAP를 구성해야 합니다.

## 클라이언트 인증 및 권한 부여

클라이언트 인증 및 권한 부여에는 몇 가지 측면을 고려해야 합니다.

### 자체 포함된 범위 또는 로컬 **ONTAP ID** 정의

상위 수준에서는 권한 부여 서버에서 정의된 자체 포함 범위를 정의하거나 역할 및 사용자를 비롯한 기존 로컬 ONTAP ID 정의를 사용할 수 있습니다.

### 로컬 **ONTAP** 처리 옵션

ONTAP ID 정의를 사용하는 경우 다음을 포함하여 적용할 항목을 결정해야 합니다.

- 이름이 지정된 REST 역할입니다
- 로컬 사용자와 일치합니다
- Active Directory 또는 LDAP 그룹

### 로컬 검증 또는 원격 검사

액세스 토큰의 유효성을 ONTAP에서 로컬로 검사할지, 아니면 자체 검사를 통해 인증 서버에서 검사할지 결정해야 합니다. 또한 새로 고침 간격과 같이 고려해야 할 여러 관련 값도 있습니다.

### 보낸 사람 제한 액세스 토큰

높은 수준의 보안이 필요한 환경에서는 MTL을 기반으로 보내기 제한 액세스 토큰을 사용할 수 있습니다. 이렇게 하려면 각 클라이언트에 대한 인증서가 필요합니다.

### 관리 인터페이스

다음은 비롯한 모든 ONTAP 인터페이스를 통해 OAuth 2.0을 관리할 수 있습니다.

- 명령줄 인터페이스입니다
- 시스템 관리자
- REST API

### 클라이언트가 액세스 토큰을 요청하는 방법

클라이언트 응용 프로그램은 권한 부여 서버에서 직접 액세스 토큰을 요청해야 합니다. 허가 유형을 포함하여 이 작업을 수행하는 방법을 결정해야 합니다.

## **ONTAP**를 구성합니다

몇 가지 ONTAP 구성 작업을 수행해야 합니다.

### **REST** 역할 및 로컬 사용자를 정의합니다

인증 구성에 따라 로컬 ONTAP 식별 처리를 사용할 수 있습니다. 이 경우 REST 역할 및 사용자 정의를 검토하고 정의해야 합니다.

### 코어 구성

핵심 ONTAP 구성을 수행하는 데 필요한 주요 단계는 다음과 같습니다.

- 선택적으로 인증 서버의 인증서를 서명한 CA에 대한 루트 인증서(및 모든 중간 인증서)를 설치합니다.
- 인증 서버를 정의합니다.

- 클러스터에 대해 OAuth 2.0 처리를 활성화합니다.

## ONTAP에 OAuth 2.0 배포

핵심 OAuth 2.0 기능을 배포하려면 세 가지 기본 단계가 필요합니다.

시작하기 전에

ONTAP를 구성하기 전에 OAuth 2.0 배포를 준비해야 합니다. 예를 들어 인증서의 서명 방법 및 방화벽 뒤에 있는지 등 인증 서버를 평가해야 합니다. 을 참조하십시오 ["ONTAP와 함께 OAuth 2.0을 배포할 준비를 하십시오"](#) 를 참조하십시오.

### 1단계: 인증 서버 인증서를 설치합니다

ONTAP에는 미리 설치된 루트 CA 인증서가 다수 포함되어 있습니다. 따라서 대부분의 경우 추가 구성 없이 ONTAP에서 인증 서버의 인증서를 즉시 인식합니다. 그러나 인증 서버 인증서 서명 방법에 따라 루트 CA 인증서와 중간 인증서를 설치해야 할 수도 있습니다.

필요한 경우 아래 제공된 지침에 따라 인증서를 설치합니다. 필요한 모든 인증서를 클러스터 수준에서 설치해야 합니다.

ONTAP 액세스 방법에 따라 올바른 절차를 선택합니다.

## 예 1. 단계

### 시스템 관리자

1. System Manager에서 \* 클러스터 \* > \* 설정 \* 을 선택합니다.
2. 아래로 스크롤하여 \* 보안 \* 섹션으로 이동합니다.
3. Certificates \* 옆에 있는 \* → \* 를 클릭합니다.
4. 신뢰할 수 있는 인증 기관 \* 탭에서 \* 추가 \* 를 클릭합니다.
5. 가져오기 \* 를 클릭하고 인증서 파일을 선택합니다.
6. 사용자 환경에 대한 구성 매개 변수를 입력합니다.
7. 추가 \* 를 클릭합니다.

### CLI를 참조하십시오

1. 설치를 시작합니다.

보안 인증서설치형 server-ca

2. 다음 콘솔 메시지를 찾습니다.

```
Please enter Certificate: Press <Enter> when done
```

3. 텍스트 편집기로 인증서 파일을 엽니다.
4. 다음 행을 포함하여 전체 인증서를 복사합니다.

```
-----BEGIN CERTIFICATE-----  
  
-----END CERTIFICATE-----
```

5. 명령 프롬프트 후 터미널에 인증서를 붙여 넣습니다.
6. Enter \* 키를 눌러 설치를 완료합니다.
7. 다음 중 하나를 사용하여 인증서가 설치되었는지 확인합니다.

```
security certificate show-user-installed  
  
security certificate show
```

## 2단계: 인증 서버를 구성합니다

ONTAP에 하나 이상의 인증 서버를 정의해야 합니다. 구성 및 배포 계획에 따라 매개 변수 값을 선택해야 합니다. 검토 "OAuth2 배포 시나리오" 구성에 필요한 정확한 매개 변수를 결정합니다.



권한 부여 서버 정의를 수정하려면 기존 정의를 삭제하고 새 정의를 만듭니다.

아래에 제공된 예는 의 첫 번째 간단한 배포 시나리오를 기반으로 합니다 "로컬 검증". 독립 실행형 범위는 프록시 없이 사용됩니다.

ONTAP 액세스 방법에 따라 올바른 절차를 선택합니다. CLI 절차에서는 명령을 실행하기 전에 교체해야 하는 기호 변수를 사용합니다.

## 예 2. 단계

### 시스템 관리자

1. System Manager에서 \* 클러스터 \* > \* 설정 \* 을 선택합니다.
2. 아래로 스크롤하여 \* 보안 \* 섹션으로 이동합니다.
3. OAuth 2.0 권한 부여 \* 옆에 있는 \* + \* 를 클릭합니다.
4. 추가 옵션 \* 을 선택합니다.
5. 다음과 같이 배포에 필요한 값을 제공합니다.
  - 이름
  - 응용 프로그램(http)
  - 공급자 JWKS URI입니다
  - 발급자 URI입니다
6. 추가 \* 를 클릭합니다.

### CLI를 참조하십시오

1. 정의를 다시 만듭니다.

```
security oauth2 client create -config-name <NAME> -provider-jwks-uri  
<URI_JWKS> -application http -issuer <URI_ISSUER>
```

예를 들면 다음과 같습니다.

```
security oauth2 client create \  
-config-name auth0 \  
-provider-jwks-uri https://superzap.dev.netapp.com:8443/realms/my-  
realm/protocol/openid-connect/certs \  
-application http \  
-issuer https://superzap.dev.netapp.com:8443/realms/my-realm
```

## 3단계: OAuth 2.0을 활성화합니다

마지막 단계는 OAuth 2.0을 활성화하는 것입니다. ONTAP 클러스터에 대한 전역 설정입니다.



ONTAP, 인증 서버 및 지원 서비스가 모두 올바르게 구성되었는지 확인하기 전까지는 OAuth 2.0 처리를 활성화하지 마십시오.

ONTAP 액세스 방법에 따라 올바른 절차를 선택합니다.

### 예 3. 단계

#### 시스템 관리자

1. System Manager에서 \* 클러스터 \* > \* 설정 \* 을 선택합니다.
2. 아래로 스크롤하여 \* 보안 섹션 \* 을 찾습니다.
3. OAuth 2.0 권한 부여 \* 옆에 있는 \* → \* 를 클릭합니다.
4. OAuth 2.0 권한 부여 \* 를 활성화합니다.

#### CLI를 참조하십시오

1. OAuth 2.0 활성화:

```
security oauth2 modify -enabled true
```

2. OAuth 2.0이 활성화되어 있는지 확인합니다.

```
security oauth2 show  
Is OAuth 2.0 Enabled: true
```

### OAuth 2.0을 사용하여 REST API 호출을 실행합니다

ONTAP의 OAuth 2.0 구현은 REST API 클라이언트 애플리케이션을 지원합니다. curl을 사용하여 간단한 REST API 호출을 실행하여 OAuth 2.0을 사용할 수 있습니다. 아래 예에서는 ONTAP 클러스터 버전을 검색합니다.

#### 시작하기 전에

ONTAP 클러스터에 대해 OAuth 2.0 기능을 구성하고 사용하도록 설정해야 합니다. 여기에는 인증 서버 정의가 포함됩니다.

#### 1단계: 액세스 토큰을 획득합니다

REST API 호출에 사용할 액세스 토큰을 얻어야 합니다. 토큰 요청은 ONTAP 외부에서 수행되며 정확한 절차는 인증 서버 및 해당 구성에 따라 다릅니다. 웹 브라우저, curl 명령 또는 프로그래밍 언어를 사용하여 토큰을 요청할 수 있습니다.

설명 목적으로 curl을 사용하여 Keycloak에서 액세스 토큰을 요청하는 방법에 대한 예가 아래에 나와 있습니다.

## Keycloak 예

```
curl --request POST \  
--location  
'https://superzap.dev.netapp.com:8443/realms/peterson/protocol/openid-  
connect/token' \  
--header 'Content-Type: application/x-www-form-urlencoded' \  
--data-urlencode 'client_id=dp-client-1' \  
--data-urlencode 'grant_type=client_credentials' \  
--data-urlencode 'client_secret=5iTUf9QKLGxAoYaliR33v1D5A2xq09V7'
```

반환된 토큰을 복사하여 저장해야 합니다.

### 2단계: REST API 호출을 실행합니다

유효한 액세스 토큰이 있으면 액세스 토큰과 함께 curl 명령을 사용하여 REST API 호출을 실행할 수 있습니다.

매개 변수 및 변수

컬링 예제의 두 변수는 아래 표에 설명되어 있습니다.

변수	설명
\$FQDN_IP입니다	ONTAP 관리 LIF의 정규화된 도메인 이름 또는 IP 주소
\$access_token입니다	인증 서버에서 발급한 OAuth 2.0 액세스 토큰

curl 예제를 실행하기 전에 먼저 Bash 셸 환경에서 이러한 변수를 설정해야 합니다. 예를 들어, Linux CLI에서 다음 명령을 입력하여 FQDN 변수를 설정하고 표시합니다.

```
FQDN_IP=172.14.31.224  
echo $FQDN_IP  
172.14.31.224
```

두 변수가 모두 로컬 Bash 셸에 정의되면 curl 명령을 복사하여 CLI에 붙여 넣을 수 있습니다. Enter \* 키를 눌러 변수를 대체하고 명령을 실행합니다.

컬의 예

```
curl --request GET \  
--location "https://$FQDN_IP/api/cluster?fields=version" \  
--include \  
--header "Accept: */*" \  
--header "Authorization: Bearer $ACCESS_TOKEN"
```

## 저작권 정보

Copyright © 2024 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.