개념
ONTAP 9

NetApp
April 24, 2024

목차

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 패키지되어 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 수입니다

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.

허가 유형	설명
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.
read_create 를 참조하십시오	POST를 사용하여 새 리소스 인스턴스를 만들고 읽기 액세스를 허용합니다.
read_modify 를 참조하십시오	패치를 사용하여 기존 리소스를 업데이트할 수 있을 뿐 아니라 읽기 액세스를 허용합니다.
READ_CREATE_MODIFY 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 이면 `true` 3단계를 계속 진행합니다.
- 값이 이면 `false` 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)

매개 변수	설명
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 을 사용합니다 `sub` 액세스 토큰의 이 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.

원격 자기 주도

다음 배포 구성은 ONTAP를 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.
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 만들기와 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**: MTL**을 사용하도록 **ONTAP**를 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 **ONTAP**에 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**: ONTAP**는 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.