



# 使用**OAuth2.0**进行身份验证和授权

## ONTAP 9

NetApp  
May 09, 2024

# 目录

使用OAuth2.0进行身份验证和授权.....	1
ONTAP OAuth2.0实施概述.....	1
概念.....	4
配置和部署.....	14

# 使用OAuth2.0进行身份验证和授权

## ONTAP OAuth2.0实施概述

从ONTAP 9.14开始、您可以选择使用开放授权(OAuth2.0)框架控制对ONTAP集群的访问。您可以使用任何ONTAP管理界面配置此功能、包括ONTAP命令行界面、System Manager和REST API。但是、只有当客户端使用REST API访问ONTAP时、才能应用OAuth2.0授权和访问控制决策。



OAuth2.0支持是在ONTAP 9.14.0中首次推出的、因此其可用性取决于您使用的ONTAP版本。请参见 "《[ONTAP 发行说明](#)》" 有关详细信息 ...

### 功能和优势

下面将介绍在ONTAP中使用OAuth2.0的主要功能和优势。

#### 支持OAuth2.0标准

OAuth2.0是行业标准授权框架。它用于使用签名访问令牌限制和控制对受保护资源的访问。使用OAuth2.0有几个好处：

- 授权配置有许多选项
- 切勿泄露客户端凭据(包括密码)
- 根据您的配置、令牌可以设置为过期
- 非常适合与REST API结合使用

#### 已通过多个常见授权服务器的测试

ONTAP实施旨在与任何符合OAuth2.0标准的授权服务器兼容。它已通过以下常见服务器或服务的测试、包括：

- auth0
- Active Directory联合身份验证服务(ADFS)
- Keycloak

#### 支持多个并发授权服务器

一个ONTAP集群最多可以定义八个授权服务器。这样、您就可以灵活地满足各种安全环境的需求。

#### 与REST角色集成

ONTAP授权决定最终取决于分配给用户或组的REST角色。这些角色或作为独立范围在访问令牌中承载、或基于本地ONTAP定义以及Active Directory或LDAP组。

#### 用于使用受发件人限制的访问令牌的选项

您可以将ONTAP和授权服务器配置为使用相互传输层安全(MTLS)、以增强客户端身份验证。它可以保证OAuth2.0访问令牌仅供最初颁发给的客户端使用。此功能支持并符合多项常见的安全建议、包括由FAPI和miter制定的安全建议。

## 实施和配置

概括地说、在开始使用时、您应该考虑OAuth2.0实施和配置的几个方面。

### ONTAP中的OAuth2.0实体

OAuth2.0授权框架定义了多个实体、这些实体可以映射到数据中心或网络中的真实或虚拟元素。下表显示了OAuth2.0实体及其对ONTAP的适应性。

OAuth2.0实体	Description
资源	提供通过内部ONTAP命令访问ONTAP资源的REST API端点。
资源所有者	默认情况下、创建或拥有受保护资源的ONTAP集群用户。
资源服务器	ONTAP集群中受保护资源的主机。
客户端	代表资源所有者或在获得资源所有者权限的情况下请求访问REST API端点的应用程序。
授权服务器	通常是一个专用服务器、负责颁发访问令牌并强制实施管理策略。

### 核心ONTAP配置

您需要将ONTAP集群配置为启用并使用OAuth2.0。这包括建立与授权服务器的连接以及定义所需的ONTAP授权配置。您可以使用任何管理界面执行此配置、其中包括：

- ONTAP 命令行界面
- System Manager
- ONTAP REST API

### 环境和支持服务

除了ONTAP定义之外、您还需要配置授权服务器。如果您使用的是组到角色映射、则还需要配置Active Directory组或LDAP等效项。

### 支持的ONTAP客户端

从ONTAP 9.14开始、REST API客户端可以使用OAuth2.0访问ONTAP。在发出REST API调用之前、您需要从授权服务器获取访问令牌。然后、客户端使用HTTP授权请求标头将此令牌作为\_p承载 令牌\_传递到ONTAP集群。根据所需的安全级别、您还可以在客户端创建和安装证书、以使用基于MTLS的受发件人限制的令牌。

## 选定术语

当您开始探索使用ONTAP部署OAuth2.0时、熟悉其中的一些术语会很有帮助。请参见 ["其他资源"](#) 有关OAuth2.0的详细信息链接。

### 访问令牌

由授权服务器发出的令牌、由OAuth2.0客户端应用程序使用它来发出访问受保护资源的请求。

### JSON Web令牌

用于格式化访问令牌的标准。JSON用于以紧凑格式表示OAuth2.0索赔、索赔分为三个主要部分。

### 受发件人限制的访问令牌

一项基于相互传输层安全(MTLS)协议的可选功能。通过在令牌中使用额外的确认声明、可确保访问令牌仅供

最初发出访问令牌的客户端使用。

## JSON Web密钥集

JWKS是ONTAP用来验证客户端提供的JWT令牌 的公共密钥集合。通常、授权服务器会通过专用URI提供这些密钥集。

## 范围

范围提供了一种限制或控制应用程序对受保护资源(如ONTAP REST API)的访问的方法。它们在访问令牌中以字符串表示。

## ONTAP REST角色

REST角色是在ONTAP 9.6中引入的、是ONTAP RBAC框架的核心部分。这些角色与ONTAP仍支持的早期传统角色不同。ONTAP中的OAuth2.0实施仅支持REST角色。

## HTTP授权标头

HTTP请求中包含的标头、用于在执行REST API调用时标识客户端和关联权限。根据身份验证和授权的执行方式、可以选择多种模式或实施方式。在向ONTAP提供OAuth2.0访问令牌时、此令牌标识为\_承载 令牌\_。

## HTTP基本身份验证

ONTAP仍支持早期的HTTP身份验证技术。纯文本凭据(用户名和密码)与冒号串联在一起、并以base64进行编码。该字符串将放置在授权请求标头中并发送到服务器。

## FAPI

OpenID Foundation的一个工作组、为金融行业提供协议、数据架构和安全建议。API最初称为财务级API。

## 斜接

一家私营非营利公司、为美国空军和美国政府提供技术和安全指导。

## 其他资源

下文提供了若干额外资源。您应查看这些站点以获取有关OAuth2.0和相关标准的更多信息。

### 协议和标准

- ["RFC 6749: 《OAuth2.0授权框架》"](#)
- ["RFC 7519: JSON Web令牌\(JWTs\)"](#)
- ["RFC 7523: 《用于OAuth2.0客户端身份验证和授权授予的JSON Web令牌\(JWT, Web令牌\)配置文件》"](#)
- ["RFC 7662: OAuth2.0令牌自省"](#)
- ["RFC报废: JWT的拥有证明密钥"](#)
- ["RFC 8705: 《OAuth2.0相互TLS客户端身份验证和受证书制约的访问令牌》"](#)

### 组织

- ["OpenID Foundation"](#)
- ["FAPI工作组"](#)
- ["斜接"](#)
- ["IANA - JWT"](#)

## 产品和服务

- ["auth0"](#)
- ["ADFS概述"](#)
- ["Keycloak"](#)

## 其他工具和实用程序

- ["验证0的Jwt"](#)
- ["OpenSSL"](#)

## NetApp文档和资源

- ["ONTAP自动化"](#) 文档。

# 概念

## 授权服务器和访问令牌

授权服务器作为OAuth2.0授权框架中的一个中央组件执行多项重要功能。

### OAuth2.0授权服务器

授权服务器主要负责创建和签名访问令牌。这些令牌包含身份和授权信息、使客户端应用程序能够有选择地访问受保护的资源。这些服务器通常彼此隔离、可通过多种不同的方式实施、包括作为独立的专用服务器或作为大型身份和访问管理产品的一部分。



有时、授权服务器可能会使用不同的术语、尤其是当OAuth2.0功能打包在更大型的身份和访问管理产品或解决方案中时。例如，术语\*身份提供程序(IDP)\*经常与\*authorization server\*互换使用。

## 管理

除了颁发访问令牌之外、授权服务器还通常通过Web用户界面提供相关管理服务。例如、您可以定义和管理：

- 用户和用户身份验证
- 范围
- 通过租户和领域实现管理隔离
- 策略实施
- 与各种外部服务的连接
- 支持其他身份协议(例如SAML)

ONTAP与符合OAuth2.0标准的授权服务器兼容。

### 定义到ONTAP

您需要为ONTAP定义一个或多个授权服务器。ONTAP可以与每台服务器安全地进行通信、以验证令牌并执行其他相关任务来支持客户端应用程序。

下面介绍了ONTAP配置的主要方面。另请参见 "[OAuth2.0部署方案](#)" 有关详细信息 ...

## 验证访问令牌的方式和位置

验证访问令牌有两个选项。

- 本地验证

ONTAP可以根据发出访问令牌的授权服务器提供的信息在本地验证访问令牌。从授权服务器检索到的信息由ONTAP进行缓存、并定期刷新。

- 远程自省

您还可以使用远程自省在授权服务器上验证令牌。自省是一种协议、允许授权方向授权服务器查询有关访问令牌的信息。它为ONTAP提供了一种从访问令牌中提取某些元数据并对令牌进行验证的方法。出于性能原因、ONTAP会缓存某些数据。

## 网络位置

ONTAP可能受防火墙保护。在这种情况下、您需要在配置中标识代理。

## 如何定义授权服务器

您可以使用任何管理界面(包括命令行界面、System Manager或REST API)为ONTAP定义授权服务器。例如、在命令行界面中、您可以使用命令 `security oauth2 client create`。

## 授权服务器的数量

一个ONTAP集群最多可以定义八个授权服务器。只要颁发者或颁发者/受众声明是唯一的、同一授权服务器就可以多次定义到同一ONTAP集群。例如、使用Keyloak时、使用不同领域时始终会出现这种情况。

## 使用OAuth2.0访问令牌

授权服务器颁发的OAuth2.0访问令牌由ONTAP进行验证、用于针对REST API客户端请求做出基于角色的访问决策。

## 获取访问令牌

您需要从为使用REST API的ONTAP集群定义的授权服务器获取访问令牌。要获取令牌、您必须直接联系授权服务器。



ONTAP不会通过问题描述访问令牌、也不会将客户端的请求重定向到授权服务器。

如何请求令牌取决于多个因素、包括：

- 授权服务器及其配置选项
- OAuth2.0授予类型
- 用于问题描述请求的客户端或软件工具

## 授予类型

`_GRANT_` 是一个定义完善的过程、包括一组网络流、用于请求和接收OAuth2.0访问令牌。根据客户端、环境和安全要求、可以使用多种不同的授予类型。下表列出了最受欢迎的补助金类型。

授予类型	Description
客户端凭据	一种仅使用凭据(如ID和共享密钥)的常见授予类型。假定客户端与资源所有者具有密切的信任关系。
Password	如果资源所有者与客户端建立了信任关系、则可以使用资源所有者密码凭据授予类型。在将旧版HTTP客户端迁移到OAuth2.0时、此功能也很有用。
授权代码	这是机密客户端的理想授予类型、并且基于基于重定向的流。它可用于获取访问令牌和刷新令牌。

## Jwt内容

OAuth2.0访问令牌格式为JWT.此内容由授权服务器根据您的配置创建。但是、令牌对客户端应用程序是不透明的。客户端没有理由检查令牌或了解其内容。

每个JWT"访问令牌都包含一组声明。这些声明描述了颁发者的特征以及基于授权服务器上管理定义的授权。下表介绍了根据标准登记的一些索赔。所有字符串都区分大小写。

款项申请	关键字	Description
颁发者	ISS	标识发出令牌的主体。款项申请处理是针对特定应用程序的。
主题	子	令牌的主题或用户。此名称的范围为全局唯一或本地唯一。
audience	澳元	令牌的目标收件人。以字符串数组的形式实施。
到期日期	有效期	令牌过期后必须拒绝的时间。

请参见 ["RFC 7519: JSON Web令牌"](#) 有关详细信息 ...

## ONTAP客户端授权选项

您可以通过多个选项自定义ONTAP客户端授权。授权决策最终取决于访问令牌中包含或派生的ONTAP REST角色。



您只能使用 ["ONTAP REST角色"](#) 为OAuth2.0配置授权时。不支持早期的ONTAP传统角色。

### 简介

ONTAP中的OAuth2.0实施灵活可靠、可为您提供保护ONTAP环境所需的选项。概括地说、用于定义ONTAP客户端授权的主要配置类别有三个。这些配置选项不能同时使用。

ONTAP会根据您的配置应用最合适的选项。请参见 ["ONTAP如何确定访问"](#) 有关ONTAP如何处理配置定义以做出访问决策的详细信息。

### OAuth2.0自包含范围

这些范围包含一个或多个自定义REST角色、每个角色封装在一个字符串中。它们与ONTAP角色定义无关。您需要在授权服务器上定义这些范围字符串。

### 本地ONTAP专用的REST角色和用户

根据您的配置、可以使用本地ONTAP标识定义来制定访问决策。选项包括：



- 单个命名的REST角色
- 将用户名与本地ONTAP用户匹配

指定角色的作用域语法为\*ONTAP角色<URL-encoded-ONTAP-role-name>。例如、如果角色为"admin"、则范围字符串将为"ONTAP角色-admin"。

### Active Directory或LDAP组

如果检查了本地ONTAP定义、但无法做出访问决定、则会使用Active Directory ("域")或LDAP ("nsswitch")组。可以通过以下两种方式之一指定组信息：

- OAuth2.0范围字符串

支持使用客户端凭据流的机密应用程序、其中没有具有组成员资格的用户。此范围应命名为\*ONTAP组-\*ONTAP <URL-encoded-ONTAP-group-name>。例如、如果组为"developing"、则范围字符串将为"ONTAP组-developing"。

- 在"组"索赔中

这适用于ADFS使用资源所有者(密码授予)流颁发的访问令牌。

### 独立的OAuth2.0范围

自包含范围是指访问令牌中包含的字符串。每个角色都是一个完整的自定义角色定义、其中包括ONTAP做出访问决策所需的一切。此范围与ONTAP本身定义的任何REST角色是分开的、并与之不同。

范围字符串的格式

在基本级别、范围表示为连续字符串、由六个冒号分隔值组成。范围字符串中使用的参数如下所述。

### ONTAP文字

范围必须以文字值开头 `ontap` 小写。此操作会将范围标识为特定于ONTAP的范围。

### 集群

此选项用于定义将哪个ONTAP集群范围设置为适用场景。这些值可以包括：

- 集群UUID

标识单个集群。

- 星号(\*)

指示适用场景all集群的范围。

您可以使用ONTAP命令行界面命令 `cluster identity show` 以显示集群的UUID。如果未指定、则范围为适用场景all集群。

Role

自身作用域中包含的REST角色的名称。ONTAP不会检查此值、也不会将其与定义给ONTAP的任何现有REST角色匹配。此名称用于日志记录。

访问级别

此值指示在范围中使用API端点时应用于客户端应用程序的访问级别。下表介绍了六个可能的值。

访问级别	Description
无	拒绝对指定端点的所有访问。
-readonly	仅允许使用GET进行读取访问。
read_create	允许读取访问以及使用POST创建新资源实例。
read_modify	允许读取访问以及使用修补程序更新现有资源的功能。
read_create_modify	允许除删除以外的所有访问。允许的操作包括GET (读取)、POST (创建)和patch (更新)。
全部	允许完全访问。

SVM

集群中SVM的名称(范围为适用场景)。使用\*\*\*值(星号)表示所有SVM。



ONTAP 9.14.1不完全支持此功能。您可以忽略SVM参数并使用星号作为占位符。查看 "《[ONTAP 发行说明](#)》" 以检查未来是否支持SVM。

REST API URI

指向一个资源或一组相关资源的完整或部分路径。字符串必须以开头 /api。如果未指定值、则会将范围限定为适用场景集群中的所有ONTAP端点。

范围示例

以下是一些独立范围的示例。

**ONTAP: : joes-Role: read\_cree\_Modify: : /API/cluster**

为分配了此角色的用户提供对的读取、创建和修改访问权限 /cluster 端点。

CLI管理工具

为了使独立范围的管理更轻松、更不容易出错、ONTAP提供了命令行界面命令 security oauth2 scope 根据输入参数生成范围字符串。

命令 security oauth2 scope 根据您的输入、有两个用例：

- CLI参数以限定字符串范围

您可以使用此版本的命令根据输入参数生成范围字符串。

- 作用域字符串到CLI参数

您可以使用此版本的命令根据输入范围字符串生成命令参数。

#### 示例

以下示例将生成一个范围字符串、其输出包含在以下命令示例后面。定义适用场景all Clusters。

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

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

### ONTAP如何确定访问

要正确设计和实施OAuth2.0、您需要了解ONTAP如何使用您的授权配置来决定客户端的访问。

#### 第1步：独立的范围

如果访问令牌包含任何自包含范围、则ONTAP会首先检查这些范围。如果没有独立范围、请转至步骤2。

如果存在一个或多个自包含范围，ONTAP将应用每个范围，直到可以明确地作出\*ALLOW或\*deny\*决定为止。如果做出明确的决定、则处理将结束。

如果ONTAP无法做出明确的访问决定、请继续执行步骤2。

#### 第2步：检查本地角色标志

ONTAP将检查标志的值 `use-local-roles-if-present`。对于定义为ONTAP的每个授权服务器、此标志的值会单独设置。

- 如果值为 `true` 继续执行步骤3。
- 如果值为 `false` 处理结束、访问被拒绝。

#### 第3步：命名ONTAP REST角色

如果访问令牌包含一个命名的REST角色、则ONTAP将使用该角色来决定访问权限。这始终会导致\*ALLOW或\*deny\*决定和处理结束。

如果没有已命名的REST角色或未找到此角色、请继续执行步骤4。

#### 第4步：本地ONTAP用户

从访问令牌中提取用户名、并尝试将其与本地ONTAP用户匹配。

如果匹配了本地ONTAP用户、则ONTAP将使用为该用户定义的角色来决定访问权限。这始终会导致\*ALLOW或\*deny\*决定和处理结束。

如果本地ONTAP用户不匹配或访问令牌中没有用户名、请继续执行步骤5。

#### 第5步：组到角色映射

从访问令牌中提取组、并尝试将其与组匹配。这些组使用Active Directory或等效的LDAP服务器进行定义。

如果存在组匹配项、ONTAP将使用为组定义的角色来决定访问权限。这始终会导致\*ALLOW或\*deny\*决定和处理结束。

如果没有组匹配项或访问令牌中没有组、则会拒绝访问并结束处理。

### OAuth2.0部署方案

在为ONTAP定义授权服务器时、可以使用多个配置选项。根据这些选项、您可以创建适合您的部署环境的授权服务器。

#### 配置参数摘要

在为ONTAP定义授权服务器时、可以使用多个配置参数。通常、所有管理界面都支持这些参数。

根据ONTAP管理界面的不同、参数名称可能略有不同。例如、在配置远程自省时、可以使用命令行界面命令参数来标识端点 `-introspection-endpoint`。但对于System Manager、等效字段为 `_Authorizationserver` 令牌自省URI\_。为了支持所有ONTAP管理界面、提供了参数的常规问题描述。根据上下文、确切的参数或字段应显而易见。

参数	Description
Name	ONTAP已知的授权服务器名称。
应用程序	ONTAP内部应用程序定义适用场景。此参数必须为*http*。
颁发者URI	FQDN、其中包含用于标识发出令牌的站点或组织的路径。
提供程序JWKS URI	包含路径和文件名的FQDN、ONTAP可从中获取用于验证访问令牌的JSON Web密钥集。
JWKS刷新闻隔	确定ONTAP刷新提供程序JWKS URI中证书信息的频率的时间间隔。该值以ISO—8601格式指定。
自省端点	包含ONTAP用于通过自省执行远程令牌验证的路径的FQDN。
客户端 ID	在授权服务器上定义的客户端名称。如果包含此值、则还需要根据接口提供关联的客户端密钥。
传出代理	这是为了在ONTAP受防火墙保护时提供对授权服务器的访问。此URI必须采用CURL格式。
使用本地角色(如果存在)	一个布尔值标志、用于确定是否使用本地ONTAP定义、包括已命名的REST角色和本地用户。
删除用户声明	ONTAP用于匹配本地用户的备用名称。使用 <code>sub</code> 字段以匹配本地用户名。

#### 部署方案

下面介绍了几种常见的部署情形。它们是根据ONTAP在本地执行令牌验证还是授权服务器在远程执行令牌验证进行组织的。每个方案都包含一个所需配置选项的列表。请参见 ["在ONTAP中部署OAuth2.0"](#) 有关配置命令的示例。



定义授权服务器后、您可以通过ONTAP管理界面显示其配置。例如、使用命令 `security oauth2 client show` 使用ONTAP命令行界面。

## 本地验证

以下部署方案基于ONTAP在本地执行令牌验证的结果。

### 无需代理即可使用自包含范围

这是仅使用OAuth2.0自包含范围的最简单部署。不使用任何本地ONTAP标识定义。您需要包含以下参数：

- Name
- 应用程序(http)
- 提供程序JWKS URI
- 颁发者URI

您还需要在授权服务器上添加范围。

### 将自包含范围与代理结合使用

此部署方案使用OAuth2.0自包含范围。不使用任何本地ONTAP标识定义。但授权服务器受防火墙保护、因此您需要配置代理。您需要包含以下参数：

- Name
- 应用程序(http)
- 提供程序JWKS URI
- 传出代理
- 颁发者URI
- audience

您还需要在授权服务器上添加范围。

### 使用本地用户角色以及代理的默认用户名映射

此部署方案使用具有默认名称映射的本地用户角色。远程用户声明使用的默认值 `sub` 因此、访问令牌中的此字段用于匹配本地用户名。用户名不得超过40个字符。授权服务器受防火墙保护、因此您还需要配置代理。您需要包含以下参数：

- Name
- 应用程序(http)
- 提供程序JWKS URI
- 使用本地角色(如果存在) (`true`)
- 传出代理
- 颁发者

您需要确保将本地用户定义为ONTAP。

### 使用本地用户角色和代理的备用用户名映射

此部署方案使用本地用户角色以及用于匹配本地ONTAP用户的备用用户名。授权服务器受防火墙保护、因此您需要配置代理。您需要包含以下参数：

- Name
- 应用程序(http)
- 提供程序JWKS URI
- 使用本地角色(如果存在) (true)
- 远程用户声明
- 传出代理
- 颁发者URI
- audience

您需要确保将本地用户定义为ONTAP。

#### 远程自省

以下部署配置基于ONTAP通过自省远程执行令牌验证。

#### 使用不带代理的独立范围

这是一个基于使用OAuth2.0独立范围的简单部署。未使用任何ONTAP标识定义。必须包含以下参数：

- Name
- 应用程序(http)
- 自省端点
- 客户端 ID
- 颁发者URI

您需要在授权服务器上定义范围以及客户端和客户端密钥。

#### 使用相互TLS进行客户端身份验证

根据您的安全需求、您可以选择配置相互TLS (MTLS)以实施强大的客户端身份验证。在OAuth2.0部署中与ONTAP结合使用时、MTLS保证访问令牌仅供最初发出访问令牌的客户端使用。

#### 采用OAuth2.0的相互TLS

传输层安全(Transport Layer Security、TLS)用于在两个应用程序(通常是客户端浏览器和Web服务器)之间建立安全通信通道。相互TLS通过客户端证书提供客户端的强标识来扩展这一功能。在ONTAP集群中与OAuth2.0结合使用时、可以通过创建和使用受发件人限制的访问令牌来扩展基本MTLS功能。

受发件人限制的访问令牌只能由最初颁发该令牌的客户端使用。为了支持此功能、请提交一份新的确认款项申请(cnf)将插入令牌中。字段包含属性 `x5t#S256` 用于保存请求访问令牌时使用的客户端证书摘要。在验证令牌时、ONTAP会验证此值。授权服务器发放的非发件人限制的访问令牌不包括额外的确认款项申请。

您需要将ONTAP配置为对每个授权服务器单独使用MTLS。例如、CLI命令 `security oauth2 client` 包括参数 `use-mutual-tls` 根据下表所示的三个值控制MTLS处理。



在每个配置中、ONTAP的结果和采取的操作取决于配置参数值以及访问令牌和客户端证书的内容。表中的参数按限制性从低到大的排列。

参数	Description
无	授权服务器已完全禁用OAuth2.0相互TLS身份验证。ONTAP不会执行MTLS客户端证书身份验证、即使令牌中存在确认声明或随TLS连接提供了客户端证书也是如此。
请求	如果客户端提供受发件人限制的访问令牌、则会强制实施OAuth2.0相互TLS身份验证。也就是说、只有在确认请求(带有属性 <code>x5t#S256</code> )。这是默认设置。
Required	对授权服务器颁发的所有访问令牌强制实施OAuth2.0相互TLS身份验证。因此、所有访问令牌都必须受发件人限制。如果访问令牌中不存在确认请求或存在无效的客户端证书、则身份验证和REST API请求将失败。

## 高级别实施流程

下面介绍了在ONTAP环境中将MTLS与OAuth2.0结合使用时所涉及的典型步骤。请参见 ["RFC 8705：《OAuth2.0相互TLS客户端身份验证和受证书制约的访问令牌》"](#) 有关详细信息：

### 第1步：创建并安装客户端证书

建立客户端身份的基础是证明了解客户端专用密钥。相应的公共密钥将放置在客户端提供的签名X.509证书中。总体而言、创建客户端证书涉及的步骤包括：

1. 生成公共密钥对和专用密钥对
2. 创建证书签名请求
3. 将CSR文件发送到知名的CA
4. CA会验证此请求并颁发签名证书

通常、您可以在本地操作系统中安装客户端证书、也可以直接使用cURL等通用实用程序来使用此证书。

### 第2步：配置ONTAP以使用MTLS

您需要将ONTAP配置为使用MTLS。此配置是针对每个授权服务器单独完成的。例如、使用命令行界面命令 `security oauth2 client` 与可选参数结合使用 `use-mutual-tls`。请参见 ["在ONTAP中部署OAuth2.0"](#) 有关详细信息 ...

### 第3步：客户端请求访问令牌

客户端需要从配置为ONTAP的授权服务器请求访问令牌。客户端应用程序必须将MTLS与步骤1中创建和安装的证书结合使用。

### 第4步：授权服务器生成访问令牌

授权服务器验证客户端请求并生成访问令牌。在此过程中、它会创建客户端证书的消息摘要、此摘要会作为确认请求包含在令牌中(字段 `cnf`)。

### 第5步：客户端应用程序将访问令牌提供给ONTAP

客户端应用程序对ONTAP集群进行REST API调用、并将访问令牌作为\*承载令牌\*包含在授权请求标头中。客户端使用的MTLS必须与请求访问令牌所用的证书相同。

### 第6步：ONTAP验证客户端和令牌。

ONTAP接收HTTP请求中的访问令牌以及用作MTLS处理一部分的客户端证书。ONTAP首先验证访问令牌中的签名。ONTAP会根据配置生成客户端证书的消息摘要，并将其与令牌中的确认声明\*cnf\*进行比较。如果这两个值匹配、则ONTAP已确认发出API请求的客户端与最初向其发出访问令牌的客户端相同。

## 配置和部署

### 准备使用ONTAP部署OAuth2.0

在ONTAP环境中配置OAuth2.0之前、您应做好部署准备。主要任务和决定摘要如下。各部分的排列通常与您应遵循的顺序一致。但是、虽然它适用于大多数部署、但您应根据需要对其进行调整以适应您的环境。您还应考虑制定正式的部署计划。



您可以根据您的环境为定义给ONTAP的授权服务器选择配置。其中包括您需要为每种部署类型指定的参数值。请参见 ["OAuth2.0部署方案"](#) 有关详细信息 ...

#### 受保护的资源和客户端应用程序

OAuth2.0是一个授权框架、用于控制对受保护资源的访问。因此、任何部署的重要第一步都是确定可用资源是什么以及哪些客户端需要访问这些资源。

#### 识别客户端应用程序

您需要确定哪些客户端在发出REST API调用时将使用OAuth2.0、以及它们需要访问哪些API端点。

#### 查看现有ONTAP REST角色和本地用户

您应查看现有ONTAP标识定义、包括REST角色和本地用户。根据您的配置OAuth2.0的方式、可以使用这些定义来决定访问。

#### 全局过渡到OAuth2.0

虽然您可以逐步实施OAuth2.0授权、但也可以通过为每个授权服务器设置一个全局标志、将所有REST API客户端立即迁移到OAuth2.0。这样、无需创建独立的范围、即可根据现有ONTAP配置做出访问决策。

#### 授权服务器

授权服务器通过颁发访问令牌并强制实施管理策略、在OAuth2.0部署中发挥着重要作用。

#### 选择并安装授权服务器

您需要选择并安装一个或多个授权服务器。熟悉身份提供程序的配置选项和过程非常重要，包括如何定义范围。

#### 确定是否需要安装授权根CA证书

ONTAP使用授权服务器的证书来验证客户端提供的签名访问令牌。为此、ONTAP需要根CA证书和任何中间证书。这些可能已随ONTAP预安装。如果不是、则需要安装它们。

#### 评估网络位置和配置

如果授权服务器受防火墙保护、则需要将ONTAP配置为使用代理服务器。

#### 客户端身份验证和授权

您需要考虑客户端身份验证和授权的几个方面。



自包含范围或本地**ONTAP**标识定义

概括地说、您可以定义在授权服务器上定义的自包含范围、也可以依赖现有的本地ONTAP身份定义(包括角色和用户)。

具有本地**ONTAP**处理的选项

如果您使用ONTAP标识定义、则必须确定要应用的定义、包括：

- 已命名REST角色
- 匹配本地用户
- Active Directory或LDAP组

本地验证或远程自省

您需要确定访问令牌是由ONTAP在本地验证、还是通过自省在授权服务器验证。此外、还需要考虑几个相关值、例如刷新间隔。

受发件人限制的访问令牌

对于需要高级别安全性的环境、您可以使用基于MTLS的发送受限访问令牌。这要求每个客户端都有一个证书。

管理界面

您可以通过任意ONTAP接口来管理OAuth2.0、其中包括：

- 命令行界面
- System Manager
- REST API

客户端如何请求访问令牌

客户端应用程序必须直接从授权服务器请求访问令牌。您需要决定如何执行此操作、包括授予类型。

## 配置 **ONTAP**

您需要执行多个ONTAP配置任务。

定义**REST**角色和本地用户

根据您的授权配置、可以使用本地ONTAP标识处理。在这种情况下、您需要查看并定义REST角色和用户定义。

核心配置

执行核心ONTAP配置需要三个主要步骤、其中包括：

- (可选)为签署授权服务器证书的CA安装根证书(以及任何中间证书)。
- 定义授权服务器。
- 为集群启用OAuth2.0处理。

## 在**ONTAP**中部署**OAuth2.0**

部署核心OAuth2.0功能主要包括三个步骤。

## 开始之前

在配置ONTAP之前、您必须为OAuth2.0部署做准备。例如、您需要评估授权服务器、包括其证书的签名方式以及是否位于防火墙之后。请参见 ["准备使用ONTAP部署OAuth2.0"](#) 有关详细信息 ...

### 第1步：安装身份验证服务器证书

ONTAP包含大量预安装的根CA证书。因此、在许多情况下、ONTAP将立即识别授权服务器的证书、而无需进行额外配置。但是、根据授权服务器证书的签名方式、您可能需要安装根CA证书和任何中间证书。

如果需要、请按照下面提供的说明安装证书。您应在集群级别安装所有必需的证书。

根据您的访问ONTAP的方式选择正确的操作步骤。

## 示例 1. 步骤

### System Manager

1. 在System Manager中、选择\*集群\*>\*设置\*。
2. 向下滚动到\*Security\*部分。
3. 单击\*Certificates\*旁边的\*→\*。
4. 在“可信证书颁发机构”选项卡下，单击“添加”。
5. 单击\*Import\*并选择证书文件。
6. 完成环境的配置参数。
7. 单击 \* 添加 \*。

### 命令行界面

1. 开始安装：

```
security certificate install -type server-ca
```

2. 查找以下控制台消息：

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

3. 使用文本编辑器打开证书文件。
4. 复制整个证书、包括以下行：

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

5. 在命令提示符后、将证书粘贴到终端中。
6. 按\*Enter\*键完成安装。
7. 使用以下方法之一确认已安装证书：

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

## 第2步：配置授权服务器

您需要为ONTAP至少定义一个授权服务器。您应根据配置和部署计划选择参数值。请查看 ["OAuth2部署方案"](#) 以确定您的配置所需的确切参数。



要修改授权服务器定义，您可以删除现有定义并创建新定义。

下面提供的示例基于第一个简单部署方案、部署方案位于 ["本地验证"](#)。自包含范围不需要代理即可使用。

根据您的访问ONTAP的方式选择正确的操作步骤。命令行界面操作步骤使用符号变量、您需要在发出命令之前替

换这些变量。

## 示例 2. 步骤

### System Manager

1. 在System Manager中、选择\*集群\*>\*设置\*。
2. 向下滚动到\*Security\*部分。
3. 单击\*OAuth2.0 authorizes\*旁边的\*+\*。
4. 选择\*更多选项\*。
5. 为您的部署提供所需的值、例如：
  - Name
  - 应用程序(http)
  - 提供程序JWKS URI
  - 颁发者URI
6. 单击 \* 添加 \*。

### 命令行界面

1. 重新创建定义：

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

例如：

```
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
```

## 第3步：启用OAuth2.0

最后一步是启用OAuth2.0。这是ONTAP集群的全局设置。



在确认ONTAP、授权服务器和任何支持服务均已正确配置之前、请勿启用OAuth2.0处理。

根据您访问ONTAP的方式选择正确的操作步骤。

### 示例 3. 步骤

#### System Manager

1. 在System Manager中、选择\*集群\*>\*设置\*。
2. 向下滚动到\*安全性部分\*。
3. 单击\*OAuth2.0 authorizes\*旁边的\*→\*。
4. 启用\*OAuth2.0授权\*。

#### 命令行界面

1. 启用OAuth2.0:

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

2. 确认已启用OAuth2.0:

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

### 使用OAuth2.0对REST API调用执行问题描述

ONTAP中的OAuth2.0实施支持REST API客户端应用程序。您可以使用cURL对简单的REST API调用进行问题描述、以便开始使用OAuth2.0。以下示例将检索ONTAP集群版本。

#### 开始之前

您必须为ONTAP集群配置并启用OAuth2.0功能。其中包括定义授权服务器。

#### 第1步：获取访问令牌

您需要获取用于REST API调用的访问令牌。令牌请求在ONTAP之外执行、确切的操作步骤取决于授权服务器及其配置。您可以通过Web浏览器、使用CURL命令或使用编程语言来请求令牌。

为了便于说明、下面提供了一个示例、说明如何使用CURL从Key斗 篷请求访问令牌。

## Keyloak示例

```
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=5iTUf9QLGxAoYaliR33v1D5A2xq09V7'
```

您应复制并保存返回的令牌。

### 第2步：对REST API调用执行问题描述操作

获得有效的访问令牌后、您可以使用带有访问令牌的cURL命令对问题描述A REST API调用进行访问。

#### 参数和变量

下表介绍了cURL示例中的两个变量。

变量	Description
\$FQDN_IP	ONTAP管理LIF的完全限定域名或IP地址。
\$access_令牌	授权服务器发出的OAuth2.0访问令牌。

您应先在Bash Shell环境中设置这些变量、然后再执行此CURL示例。例如、在Linux命令行界面中键入以下命令以设置和显示FQDN变量：

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

在本地bash shell中定义了这两个变量后、您可以复制此URL命令并将其粘贴到命令行界面中。按\*Enter\*键替换变量并问题描述命令。

#### curl 示例

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

## 版权信息

版权所有 © 2024 NetApp, Inc.。保留所有权利。中国印刷。未经版权所有者事先书面许可，本文档中受版权保护的任何部分不得以任何形式或通过任何手段（图片、电子或机械方式，包括影印、录音、录像或存储在电子检索系统中）进行复制。

从受版权保护的 NetApp 资料派生的软件受以下许可和免责声明的约束：

本软件由 NetApp 按“原样”提供，不含任何明示或暗示担保，包括但不限于适销性以及针对特定用途的适用性的隐含担保，特此声明不承担任何责任。在任何情况下，对于因使用本软件而以任何方式造成的任何直接性、间接性、偶然性、特殊性、惩罚性或后果性损失（包括但不限于购买替代商品或服务；使用、数据或利润方面的损失；或者业务中断），无论原因如何以及基于何种责任理论，无论出于合同、严格责任或侵权行为（包括疏忽或其他行为），NetApp 均不承担责任，即使已被告知存在上述损失的可能性。

NetApp 保留在不另行通知的情况下随时对本文档所述的任何产品进行更改的权利。除非 NetApp 以书面形式明确同意，否则 NetApp 不承担因使用本文档所述产品而产生的任何责任或义务。使用或购买本产品不表示获得 NetApp 的任何专利权、商标权或任何其他知识产权许可。

本手册中描述的产品可能受一项或多项美国专利、外国专利或正在申请的专利的保护。

有限权利说明：政府使用、复制或公开本文档受 DFARS 252.227-7013（2014 年 2 月）和 FAR 52.227-19（2007 年 12 月）中“技术数据权利 — 非商用”条款第 (b)(3) 条规定的限制条件的约束。

本文档中所含数据与商业产品和/或商业服务（定义见 FAR 2.101）相关，属于 NetApp, Inc. 的专有信息。根据本协议提供的所有 NetApp 技术数据和计算机软件具有商业性质，并完全由私人出资开发。美国政府对这些数据的使用权具有非排他性、全球性、受限且不可撤销的许可，该许可既不可转让，也不可再许可，但仅限在与交付数据所依据的美国政府合同有关且受合同支持的情况下使用。除本文档规定的情形外，未经 NetApp, Inc. 事先书面批准，不得使用、披露、复制、修改、操作或显示这些数据。美国政府对国防部的授权仅限于 DFARS 的第 252.227-7015(b)（2014 年 2 月）条款中明确的权利。

## 商标信息

NetApp、NetApp 标识和 <http://www.netapp.com/TM> 上所列的商标是 NetApp, Inc. 的商标。其他公司和产品名称可能是其各自所有者的商标。