



Authentication Methods

Table of contents

UFM Authentication Server

Configurations of the UFM Authentication Server

Token-Based Authentication

Proxy Authentication

Delegating Authentication to a Proxy

Azure AD Authentication

Register UFM in Azure AD Portal

Enable Azure Authentication From UFM

Azure Authentication Login Page

Kerberos Authentication

Setting Up Kerberos Server Machine

Setting Up Kerberos Client Machine

List of Figures

Figure 0. Image2022 4 28 22 56 28 Version 1 Modificationdate
1716899139247 Api V2

Figure 1. Image2021 11 26 10 42 46 Version 1 Modificationdate
1716899138567 Api V2

Figure 2. Image2021 11 26 10 42 50 Version 1 Modificationdate
1716899138057 Api V2

Figure 3. Image2021 11 26 10 43 2 Version 1 Modificationdate
1716899137490 Api V2

Figure 4. Image2023 1 23 12 18 16 Version 1 Modificationdate
1716899128693 Api V2

Figure 5. Image2023 7 26 7 16 15 Version 1 Modificationdate
1716899136897 Api V2

Figure 6. Azureauth2 Version 1 Modificationdate 1716899136150 Api V2

Figure 7. Azureauth3 Version 1 Modificationdate 1716899134940 Api V2

Figure 8. Azureauth4 Version 1 Modificationdate 1716899133917 Api V2

Figure 9. Azureauth5 Version 1 Modificationdate 1716899133420 Api V2

Figure 10. Azureauth6 Version 1 Modificationdate 1716899132713 Api
V2

Figure 11. Image2023 7 26 7 28 25 Version 1 Modificationdate
1716899131900 Api V2

Figure 12. Image2023 7 26 7 27 55 Version 1 Modificationdate
1716899130853 Api V2

Figure 13. Azureauth9 Version 1 Modificationdate 1716899130413 Api V2

Figure 14. MSFT Version 1 Modificationdate 1716899130007 Api V2

UFM User Authentication is based on standard Apache User Authentication or Internal UFM Authentication Server.

Each Web Service client application must authenticate against the UFM server to gain access to the system.

The available authentication methods supported by UFM are as follows:

Authenti- cation Method	Description	UFM Relate d Prefix	REST API Referen ce
Basic Authentication	Based on user and password, provided by the client. This method is enabled by default.	/ufmR est	Basic Authentication
Session-Based Authentication	A stateful authentication technique where sessions are used to keep track of the authenticated user. This method is enabled by default and is used by the UFM WebUI.	/ufmR estV2	Session-Based Authentication
Client-Based Authentication	Refers to an end user's device proving its own identity by providing a digital certificate that can be verified by a server in order to gain access to UFM resources	/ufmR est	Client-Based Authentication
Token-Based Authentication	Token-based authentication is a protocol which allows users to verify their identity, and in return receive a unique access token. To use UFM, the user should create a token using UFM Web UI or UFM REST API	/ufmR estV3	Token-Based Authentication
Proxy Authentication	Proxy authentication delegates the user authentication to a remote Proxy server.	/ufmR estV3	N/A
Azure AD Authentication	Microsoft Azure Authentication is a service provided by Microsoft Azure, the cloud computing platform of Microsoft. It is designed to provide secure access control and authentication for applications and services hosted on Azure.	/ufmR estV2	N/A

Authentication Method	Description	UFM Related Prefix	REST API Reference
Kerberos Authentication	Kerberos is a protocol designed to authenticate service requests between trusted hosts over an untrusted network	/ufmRestKrb	N/A

There are two optional services which can provide authentication handling of UFM.

1. Apache Web Server (used by default) - Standard Apache web server and supports the above-mentioned authentication methods.
2. UFM Authentication Server - a centralized HTTP server, is responsible for managing various authentication methods supported by UFM.

UFM Authentication Server

Configurations of the UFM Authentication Server

The UFM Authentication Server is designed to be configurable and is initially turned on by default.

Enabling the UFM Authentication Server provides a centralized service that oversees all supported authentication methods within a single service, consolidating them under a unified authentication API.

Apache utilizes the authentication server's APIs to determine a user's authentication status.

All activities of the UFM Authentication Server are logged in the `authentication_service.log` file, located at `/opt/ufm/files/log`.

To enable/disable the UFM Authentication Server, refer to [Enabling UFM Authentication Server](#).

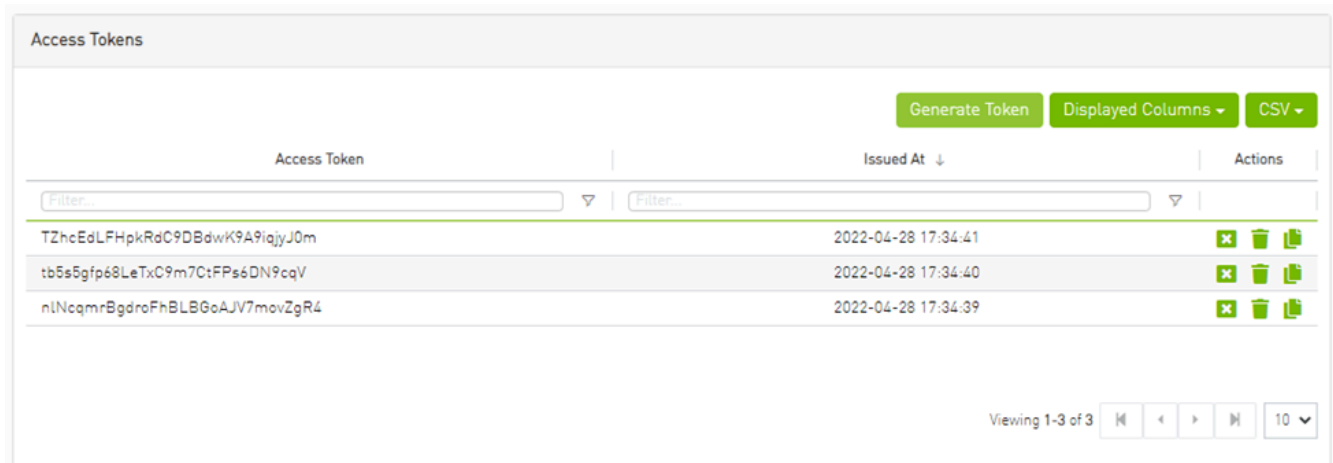
Token-Based Authentication










Token-based authentication is a protocol which allows users to verify their identity, and in return receive a unique access token. During the life of the token, users then access the UFM APIs that the token has been issued for, rather than having to re-enter credentials each time they need to use any UFM API.

Note





Under the Settings section there is a tab titled called “Access Tokens”.


The functionality of the added tab is to give the user the ability to create new tokens & manage the existing ones (list, copy, revoke, delete):



Access Token	Issued At ↓	Actions
TZhcEdLFHpkRdC9DBdwK9A9iajyJ0m	2022-04-28 17:34:41	  
tb5s5gfp68LeTx09m7C:FPs6DN9cqV	2022-04-28 17:34:40	  
nINcqmRbgdroFhBLBG0AJV7movZgR4	2022-04-28 17:34:39	  

Actions:

Name	Icon	Description
Revoke		Revoke a specific token. <div style="background-color: #ffffcc; padding: 5px;"> <p> Note The revoked token will no longer be valid.</p> </div>
Delete		Delete a specific token.
Copy		Copy specific token into the clipboard.

 **Note**

Each user is able to list and manage only the tokens that have been created by themselves. Only the users with `system_admin` role will be able to create tokens.

Proxy Authentication

Delegating Authentication to a Proxy

To allow a custom user authentication, you can configure UFM to delegate the user authentication to a remote Proxy server. The remote Proxy server is written by the user, thus, allowing flexibility on deciding how the authentication is performed.

By default, the feature is disabled. To activate the feature, configure `auth_proxy_enabled` with `true`.

Proxy should use `ufmRestV3` to send requests to UFM. The request header should contain a username and role. The available roles are `System_Admin`, `Fabric_Admin`,

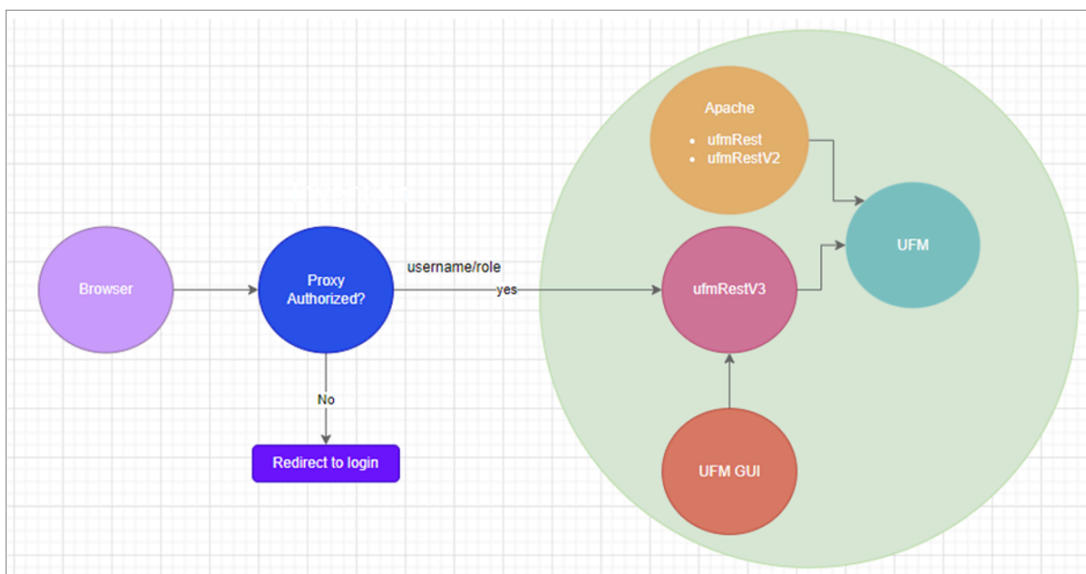
Fabric_Operator, and Monitoring_Only. If the request header is sent without a username or a role, it is rejected by the UFM.

For example:

```
[AuthProxy]
# Defaults to false, but set to true to enable this feature
auth_proxy_enabled = true
# HTTP Header name that will contain the username
auth_proxy_header_name = X_WEBAUTH_USER
# HTTP Header name that will contain the user roles. The available roles are as follows: System_Admin, Fabric_Admin, Fabric_Operator, and Monitoring_Only
auth_proxy_header_role = X_WEBAUTH_ROLE

# Set to `true` to enable auto sign up of users who do not exist in UFM DB. Defaults to `true`.
auth_proxy_auto_sign_up = true
# Limit where auth proxy requests come from by configuring a list of IP addresses.
# This can be used to prevent users spoofing the X_WEBAUTH_USER header.
# This option is required
# Example `whitelist = 192.168.1.1, 192.168.1.0/24, 2001::23, 2001::0/120`
auth_proxy_whitelist =
```

The following chart describes the flow:



Azure AD Authentication

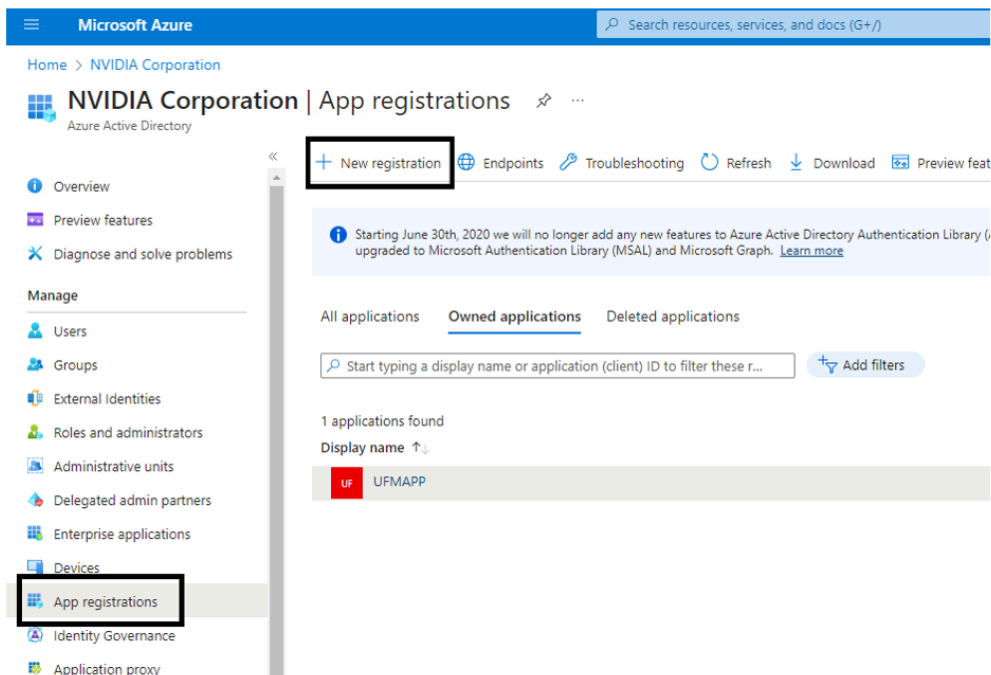
Microsoft Azure Authentication is a service provided by Microsoft Azure, the cloud computing platform of Microsoft. It is designed to provide secure access control and authentication for applications and services hosted on Azure.

UFM supports Authentication using Azure Active Directory, and to do so, you need to follow the following steps:

Register UFM in Azure AD Portal

To log in via Azure, UFM must be registered in the Azure portal using the following steps:

1. Log in to [Azure Portal](#), then click "**Azure Active Directory**" in the side menu.
2. If you have access to more than one tenant, select your account in the upper right. Set your session to the Azure AD tenant you wish to use.
3. Under "**Manage**" in the side menu, click App Registrations > New Registration.



4. Provide the application details:

1. **Name:** Enter a descriptive name.

2. **Supported account types:** Account types that are allowed to login and use the registered application.
3. **Redirect URL:** select the app type **Web**, and Add the following redirect URL `https:// /auth/login`

Home > NVIDIA Corporation | App registrations >

Register an application

*** Name**
The user-facing display name for this application (this can be changed later).

Supported account types
Who can use this application or access this API?

Accounts in this organizational directory only (NVIDIA Corporation only - Single tenant)

Accounts in any organizational directory (Any Azure AD directory - Multitenant)

Accounts in any organizational directory (Any Azure AD directory - Multitenant) and personal Microsoft accounts (e.g. Skype, Xbox)

Personal Microsoft accounts only

[Help me choose...](#)

Redirect URI (optional)
We'll return the authentication response to this URI after successfully authenticating the user. Providing this now is optional and it can be changed later, but a value is required for most authentication scenarios.

Register an app you're working on here. Integrate gallery apps and other apps from outside your organization by adding from [Enterprise applications](#).

By proceeding, you agree to the [Microsoft Platform Policies](#)

[Register](#)

Then, click **Register**. The app's **Overview** page opens.

5. Under **Manage** in the side menu, click **Certificates & Secrets** > New client secret.

Add a client secret

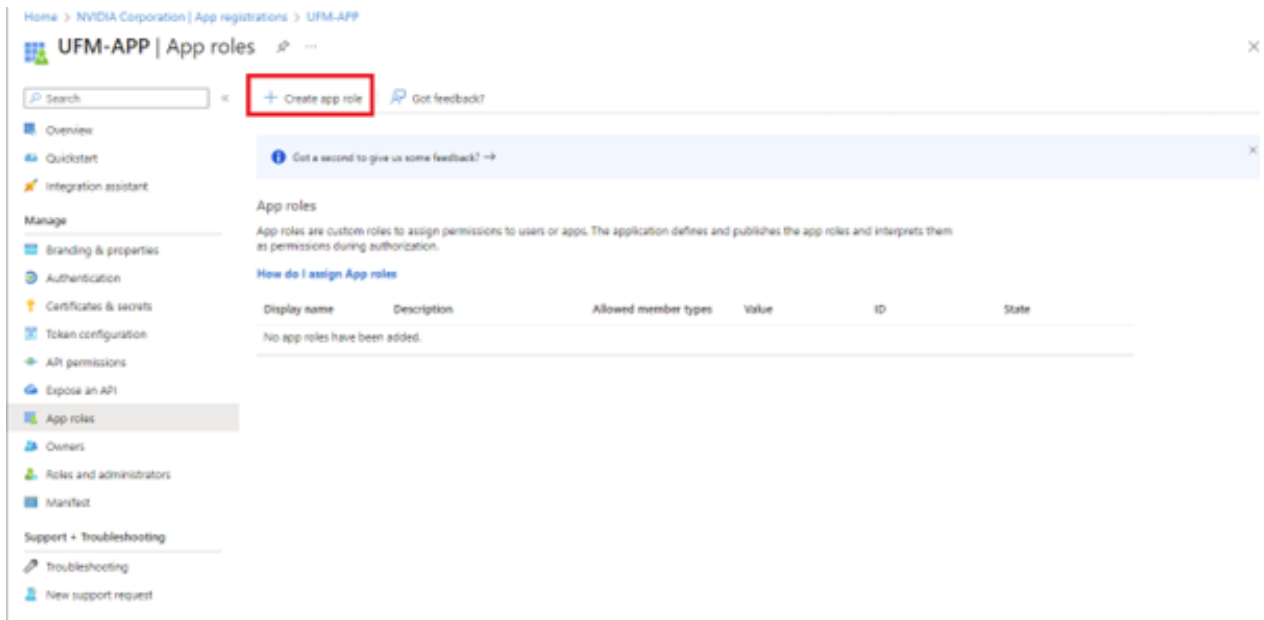


Description	<input type="text" value="UFM_APP_sec"/>
Expires	<input type="text" value="Recommended: 180 days (6 months)"/> ▼

Provide a description for the client secret and set an expiration time, then click **"Add."**

- Copy the client secret key value which will be needed to configure the UFM with Azure AD (Please note that the value of the generated secret will be hidden and will not be able to be copied/read after you leave the page.

Under **"Manage"** in the side menu, click App roles > Create app role.



- Provide the role details. Please note that the role value must be a valid UFM role; otherwise, the login will fail.

Create app role ✕

Display name * ⓘ

System_Admin ✓

Allowed member types * ⓘ

Users/Groups
 Applications
 Both (Users/Groups + Applications)

Value * ⓘ

System_Admin ✓

Description * ⓘ

System_Admin

Do you want to enable this app role? ⓘ

8. Assign the created role to the user. Follow the below steps:

App roles

App roles are custom roles to assign permissions to users or apps. The application determines permissions during authorization.

[How do I assign App roles](#) 1

Display name	Description	Allowed member ty...	Value
System_Admin	System_Admin	Users/Groups,Applicat...	Syste

Assigning app roles ✕

App roles for Users/Groups

Assign app roles with 'User' allowed member types in **Enterprise** applications or in Microsoft Graph APIs.

[Learn more about how to assign App roles for Users/Groups](#)

App roles for applications

Assign app roles with 'Applications' allowed member types in API permissions blade.

Properties


UF Name ⓘ Copy to clipboard
UFM-APP


Application ID ⓘ
0d5e6cda-9144-47a2-b685-...


Object ID ⓘ
dd2a68d5-a3e0-45e3-9c1c-...

Getting Started


3

**1. Assign users and groups**
Provide specific users and groups access to the applications
[Assign users and groups](#)

**2. Provision User Accounts**
You'll need to create user accounts in the application
[Learn more](#)

**3. Self service**
Enable users to request access to the application using their Azure AD credentials
[Get started](#)

 **Add user/group** 4  Edit assignment  Remove  Update credentials |  Columns |  Got feedback?

 The application will not appear for assigned users within My Apps. Set 'visible to users?' to yes in properties to enable this. →

Assign users and groups to app-roles for your application here. To create new app-roles for this application, use the [application registration](#).

 First 200 shown, to search all users & gro...

Display Name	Object Type	Role assigned
No application assignments found		

Home > UFM-APP | App roles > UFM-APP | Users and groups >

Add Assignment

NVIDIA Corporation

Users and groups

1 user selected.

Select a role *

System_Admin

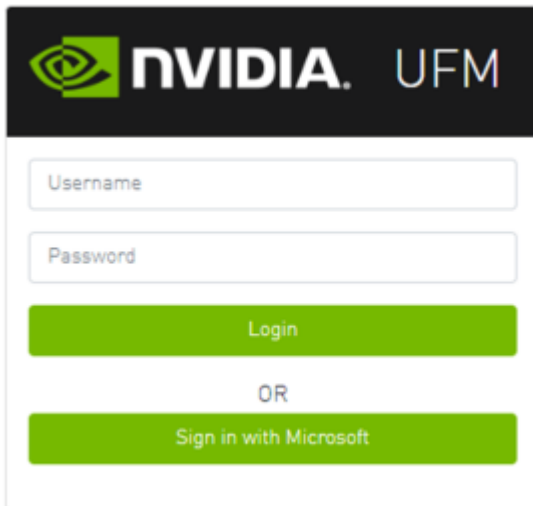
9. Click on "**Overview**" in the side menu to view the application information, such as tenant ID, client ID, and other details.

Enable Azure Authentication From UFM

Azure authentication is disabled by default. To enable it, please refer to [Enabling Azure AD Authentication](#).

Azure Authentication Login Page

After enabling and configuring Azure AD authentication, an additional button will appear on the primary UFM login page labeled 'Sign In with Microsoft,' which will lead to the main Microsoft sign-in page:

The image shows a login interface for NVIDIA UFM. At the top, there is a black header with the NVIDIA logo and the text 'NVIDIA. UFM'. Below the header, there are two input fields: 'Username' and 'Password'. Underneath these fields is a green 'Login' button. Below the 'Login' button is the text 'OR'. At the bottom, there is another green button labeled 'Sign in with Microsoft'.

Kerberos Authentication

Kerberos is a network authentication protocol designed to provide strong authentication for client-server applications by using secret-key cryptography.

The Kerberos protocol works on the basis of tickets, it helps ensure that communication between various entities in a network is secure. It uses symmetric-key cryptography, which means both the client and servers share secret keys for encrypting and decrypting communication.

To enable Kerberos Authentication, refer to [Enabling Kerberos Authentication](#).

Setting Up Kerberos Server Machine

To set up a system as a Kerberos server, perform the following:

1. Install the required packages:

```
#Redhat
sudo yum install krb5-libs krb5-server
# Ubuntu
sudo apt-get install krb5-kdc krb5-admin-server
```

2. Edit the Kerberos configuration file `'/etc/krb5.conf'` to reflect your realm, domain and other settings:


```
[libdefaults]
    default_realm = YOUR-REALM

[realms]
    YOUR-REALM = {
        kdc = your-kdc-server
        admin_server = your-admin-server
    }

[domain_realm]
    your-domain = YOUR-REALM
    your-domain = YOUR-REALM
```

3. Use the `kdb5_util` command to create the Kerberos database:

```
kdb5_util create -r YOUR-REALM -s
```

4. Add administrative principals:

```
Kadmin.local addprinc -randkey HTTP/YOUR-HOST-NAME@YOUR-REALM
```

5. Start KDC and Kadmin services:

```
sudo systemctl start krb5kdc kadmin
sudo systemctl enable krb5kdc kadmin
```

6. Generate a keytab file. The keytab file contains the secret key for a principal and is used to authenticate the service.

```
kadmin.local ktadd -k /path/to/your-keytab-file HTTP/YOUR-HOST-
NAME@YOUR-REALM
```

Replace `/path/to/your-keytab-file` with the actual path where you want to store the keytab file.

Setting Up Kerberos Client Machine

Follow the below steps to set up a system as a Kerberos client.

1. Install the required packages. When installing the UFM, the following packages will be installed as dependencies:

```
#Redhat
krb5-libs krb5-workstation mod_auth_gssapi
# Ubuntu
krb5-config krb5-user libapache2-mod-auth-gssapi
```

2. Configure the `/etc/krb5.conf` file to reflect your realm, domain, local names map and other settings:

```
kinit -k -t /path/to/your-keytab-file HTTP/YOUR-HOST-NAME@YOUR-REALM

[libdefaults]
    default_realm = YOUR-REALM

[realms]
    YOUR-REALM = {
        kdc = your-kdc-server
        admin_server = your-admin-server
        auth_to_local_names = {
            your-principle-name = your-local-user
        }
    }

[domain_realm]
    your-domain = YOUR-REALM
```

```
your-domain = YOUR-REALM
```

3. Copy the keytab file from the Kerberos server to the machine where your service runs (the client). It is important to ensure that it is kept confidential. Please ensure that the keytab file exists and that Apache has the necessary read permissions to access the keytab file; otherwise, Kerberos authentication will not function properly.
4. Obtain a Kerberos ticket-granting ticket (TGT):
5. Enable Kerberos Authentication from UFM. Kerberos authentication is disabled by default. To enable it, please refer to [Enabling Kerberos Authentication](#).
6. Test the Kerberos Authentication. You can use curl to test whether the user can authenticate to UFM REST APIs using Kerberos.

```
curl --negotiate -i -u : -k 'https://ufmc-eos01/ufmRestKrb/app/tokens'
```

© Copyright 2024, NVIDIA. PDF Generated on 06/06/2024