NetApp

Multi-factor authentication (MFA)

SnapCenter software

NetApp
January 09, 2026

Table of Contents

Multi-factor authentication (MFA)

Manage multi-factor authentication (MFA)

You can manage Multi-factor authentication (MFA) functionality in the Active Directory Federation Service (AD FS) Server and SnapCenter Server.

Enable multi-factor authentication (MFA)

You can enable MFA functionality for SnapCenter Server using PowerShell commands.

About this task

- SnapCenter supports SSO based logins when other applications are configured in the same AD FS. In certain AD FS configurations, SnapCenter might require user authentication for security reasons depending on the AD FS session persistence.

- The information regarding the parameters that can be used with the cmdlet and their descriptions can be obtained by running `Get-Help command_name`. Alternatively, you can also see SnapCenter Software Cmdlet Reference Guide.

Before you begin

- Windows Active Directory Federation Service (AD FS) should be up and running in the respective domain.

- You should have an AD FS supported Multi-factor authentication service such as Azure MFA, Cisco Duo, and so on.

- SnapCenter and AD FS server timestamp should be the same regardless of the timezone.

- Procure and configure the authorized CA certificate for SnapCenter Server.

  CA Certificate is mandatory for the following reasons:

  ◦ Ensures that the ADFS-F5 communications do not break because the self-signed certificates are unique at the node level.

  ◦ Ensures that during upgrade, repair, or disaster recovery (DR) in a standalone or high availability configuration, the self-signed certificate does not get recreated thus avoiding MFA reconfiguration.

  ◦ Ensures IP-FQDN resolutions.

    For information on CA certificate, see Generate CA Certificate CSR file.

Steps

1. Connect to the Active Directory Federation Services (AD FS) host.

2. Download AD FS federation metadata file from "https://<host FQDN>/FederationMetadata/2007-06/FederationMetadata.xml".

3. Copy the downloaded file to SnapCenter Server to enable MFA feature.

4. Log in to SnapCenter Server as the SnapCenter Administrator user through PowerShell.

5. Using the PowerShell session, generate the SnapCenter MFA metadata file by using the *New-SmMultifactorAuthenticationMetadata -path* cmdlet.

   The path parameter specifies the path to save the MFA metadata file in the SnapCenter Server host.

6. Copy the generated file to the AD FS host to configure SnapCenter as the client entity.

7. Enable MFA for SnapCenter Server using the `Set-SmMultiFactorAuthentication` cmdlet.

8. (Optional) Check the MFA configuration status and settings by using `Get-SmMultiFactorAuthentication` cmdlet.

9. Go to the Microsoft management console (MMC) and perform the following steps:

   a. Click **File** > **Add/Remove Snapin**.

   b. In the Add or Remove Snap-ins window, select **Certificates** and then click **Add**.

   c. In the Certificates snap-in window, select the **Computer account** option, and then click **Finish**.

   d. Click **Console Root** > **Certificates – Local Computer** > **Personal** > **Certificates**.

   e. Right-click on the CA certificate bound to SnapCenter and then select **All Tasks** > **Manage Private Keys**.

   f. On the permissions wizard perform the following steps:

      i. Click **Add**.

      ii. Click **Locations** and select the concerned host (top of hierarchy).

      iii. Click **OK** in the **Locations** pop-up window.

      iv. In the object name field, enter 'IIS_IUSRS' and click **Check Names** and click **OK**.

      If the check is successful, click **OK**.

10. In the AD FS host, open AD FS management wizard and perform the following steps:

    a. Right click on **Relying Party Trusts** > **Add Relying Party Trust** > **Start**.

    b. Select the second option and browse the SnapCenter MFA Metadata file and click **Next**.

    c. Specify a display name and click **Next**.

    d. Choose an access control policy as required and click **Next**.

    e. Select the settings in the next tab to default.

    f. Click **Finish**.

    SnapCenter is now reflected as a relying party with the provided display name.

11. Select the name and perform the following steps:

    a. Click **Edit Claim Issuance Policy**.

    b. Click **Add Rule** and click **Next**.

    c. Specify a name for the claim rule.

    d. Select **Active Directory** as the attribute store.

    e. Select the attribute as **User-Principal-Name** and the outgoing claim type as **Name-ID**.

    f. Click **Finish**.

12. Run the following PowerShell commands on the ADFS server.

```
Set-AdfsRelyingPartyTrust -TargetName '<Display name of relying party >'
-SigningCertificateRevocationCheck None

Set-AdfsRelyingPartyTrust -TargetName '<Display name of relying party >'
```

```
-EncryptionCertificateRevocationCheck None
```

13. Perform the following steps to confirm that the metadata was imported successfully.

    a. Right-click the relying party trust and select **Properties**.

    b. Ensure that the Endpoints, Identifiers, and Signature fields are populated.

14. Close all the browser tabs and reopen a browser to clear the existing or active session cookies, and login again.

SnapCenter MFA functionality can also be enabled using REST APIs.

For troubleshooting information, see Simultaneous login attempts in multiple tabs shows MFA error.

Update AD FS MFA Metadata

You should update the AD FS MFA metadata in SnapCenter whenever there is any modification in the AD FS Server, such as upgrade, CA certificate renewal, DR, and so on.

Steps

1. Download AD FS federation metadata file from "https://<host FQDN>/FederationMetadata/2007-06/FederationMetadata.xml"

2. Copy the downloaded file to SnapCenter Server to update the MFA configuration.

3. Update the AD FS metadata in SnapCenter by running the following cmdlet:

```
Set-SmMultiFactorAuthentication -Path <location of ADFS MFA metadata xml file>
```

4. Close all the browser tabs and reopen a browser to clear the existing or active session cookies, and login again.

Update SnapCenter MFA metadata

You should update the SnapCenter MFA metadata in AD FS whenever there is any modification in ADFS server such as repair, CA certificate renewal, DR, and so on.

Steps

1. In the AD FS host, open AD FS management wizard and perform the following steps:

    a. Select **Relying Party Trusts**.

    b. Right click on the relying party trust that was created for SnapCenter and select **Delete**.

       The user defined name of the relying party trust will be displayed.

    c. Enable Multi-factor authentication (MFA).

       See Enable Multi-factor authentication.

2. Close all the browser tabs and reopen a browser to clear the existing or active session cookies, and login again.

Disable Multi-factor authentication (MFA)

Steps

1. Disable MFA and clean up the configuration files that were created when MFA was enabled by using the `Set-SmMultiFactorAuthentication` cmdlet.

2. Close all the browser tabs and reopen a browser to clear the existing or active session cookies, and login again.

Manage multi-factor authentication (MFA) using Rest API, PowerShell, and SCCLI

MFA login is supported from browser, REST API, PowerShell, and SCCLI. MFA is supported through an AD FS identity manager. You can enable MFA, disable MFA, and configure MFA from GUI, REST API, PowerShell, and SCCLI.

Setup AD FS as OAuth/OIDC

Configure AD FS using Windows GUI wizard

1. Navigate to **Server Manager Dashboard** > **Tools** > **ADFS Management**.

2. Navigate to **ADFS** > **Application Groups**.

   a. Right-click on **Application Groups**.

   b. Select **Add Application group** and enter **Application Name**.

   c. Select **Server Application**.

   d. Click **Next**.

3. Copy **Client Identifier**.

   This is the Client ID. .. Add Callback URL (SnapCenter Server URL) in Redirect URL. .. Click **Next**.

4. Select **Generate shared secret**.

   Copy the secret value. This is the client's secret. .. Click **Next**.

5. On the **Summary** page, click **Next**.

   a. On the **Complete** page, click **Close**.

6. Right-click on the newly added **Application Group** and select **Properties**.

7. Select **Add application** from App Properties.

8. Click **Add application**.

   Select Web API and click **Next**.

9. On the Configure Web API page, enter the SnapCenter Server URL and Client Identifier created in the previous step into the Identifier section.

   a. Click **Add**.

   b. Click **Next**.

10. On the **Choose Access Control Policy** page, select control policy based on your requirement (For example, Permit everyone and require MFA) and click **Next**.

11. On the **Configure Application Permission** page, by default openid is selected as a scope, click **Next**.

12. On the **Summary** page, click **Next**.

    On the **Complete** page, click **Close**.

13. On the **Sample Application Properties** page, click **OK**.

14. JWT token issued by an authorization server (AD FS) and intended to be consumed by the resource.

    The 'aud' or audience claim of this token must match the identifier of the resource or Web API.

15. Edit the selected WebAPI and check that Callback URL (SnapCenter Server URL) and the client identifier were added correctly.

    Configure OpenID Connect to provide a username as claims.

16. Open the **AD FS Management** tool located under the **Tools** menu at the top right of the Server Manager.

    a. Select the **Application Groups** folder from the left sidebar.

    b. Select the Web API and click **EDIT**.

    c. Go-to Issuance Transform Rules Tab

17. Click **Add Rule**.

    a. Select the **Send LDAP Attributes as Claims** in the Claim rule template dropdown.

    b. Click **Next**.

18. Enter the **Claim rule** name.

    a. Select **Active Directory** in the Attribute store dropdown.

    b. Select **User-Principal-Name** in the **LDAP Attribute** dropdown and **UPN** in the O*utgoing Claim Type* dropdown.

    c. Click **Finish**.

Create Application Group using PowerShell commands

You can create the application group, web API, and add the scope and claims using PowerShell commands. These commands are available in automated script format. For more information see <link to KB article>.

1. Create the new Application Group in AD FS by using the following comamnd.

   ```
   New-AdfsApplicationGroup -Name $ClientRoleIdentifier
   -ApplicationGroupIdentifier $ClientRoleIdentifier
   ```

   `ClientRoleIdentifier` name of your application group

   `redirectURL` valid URL for redirection after authorization

2. Create the AD FS Server Application and generate the client secret.

   ```
   Add-AdfsServerApplication -Name "$ClientRoleIdentifier - Server app"
   -ApplicationGroupIdentifier $ClientRoleIdentifier -RedirectUri $redirectURL
   -Identifier $identifier -GenerateClientSecret
   ```

3. Create the ADFS Web API application and configure the policy name it should use.

```
$identifier = (New-Guid).Guid

Add-AdfsWebApiApplication -ApplicationGroupIdentifier $ClientRoleIdentifier
-Name "App Web API"

-Identifier $identifier -AccessControlPolicyName "Permit everyone"
```

4. Get the client ID and client secret from the output of the following commands because, it is shown only one time.

```
"client_id = $identifier"

"client_secret: "$($ADFSApp.ClientSecret)
```

5. Grant the AD FS Application the allatclaims and openid permissions.

```
Grant-AdfsApplicationPermission -ClientRoleIdentifier $identifier
-ServerRoleIdentifier $identifier -ScopeNames @('openid')

$transformrule = @"

@RuleTemplate = "LdapClaims"

@RuleName = "AD User properties and Groups"

c:[Type ==
"http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname",
Issuer ==

"AD AUTHORITY"]

⇒ issue(store = "Active Directory", types =
("http://schemas.xmlsoap.org/ws/2005/05/identity/claims/upn"), query =
";userPrincipalName;{0}", param = c.Value);

"@
```

6. Write out the transform rules file.

```
$transformrule |Out-File -FilePath .\issueancetransformrules.tmp -force
-Encoding ascii $relativePath = Get-Item .\issueancetransformrules.tmp
```

7. Name the Web API Application and define its Issuance Transform Rules using an external file.

```
Set-AdfsWebApiApplication -Name "$ClientRoleIdentifier - Web API"
-TargetIdentifier

$identifier -Identifier $identifier,$redirectURL -IssuanceTransformRulesFile

$relativePath
```

Update access token expiry time

You can update the access token expiry time using the PowerShell command.

About this task

- An access token can be used only for a specific combination of user, client, and resource. Access tokens cannot be revoked and are valid until their expiry.

- By default, the expiry time of an access token is 60 minutes. This minimal expiry time is sufficient and scaled. You must provide sufficient value to avoid any ongoing business-critical jobs.

Step

To update the access token expiry time for an application group WebApi, use the following command in AD FS server.

+ `Set-AdfsWebApiApplication -TokenLifetime 3600 -TargetName "<Web API>"`

Get the bearer token from AD FS

You should fill the below-mentioned parameters in any REST client (like Postman) and it prompts you to fill in the user credentials. Additionally, you should enter the second-factor authentication (something you have & something you are) to get the bearer token.

+ The validity of the bearer token is configurable from the AD FS server per application and the default validity period is 60 minutes.

Field	Value
Grant type	Authorization Code
Callback URL	Enter your application's base URL if you do not have a callback URL.
Auth URL	[adfs-domain-name]/adfs/oauth2/authorize
Access token URL	[adfs-domain-name]/adfs/oauth2/token
Client ID	Enter the AD FS client ID
Client secret	Enter the AD FS client secret
Scope	OpenID
Client Authentication	Send as Basic AUTH Header
Resource	In the **Advance Options** tab, add the Resource field with the same value as the Callback URL, which comes as an "aud" value in the JWT token.

Configure MFA in SnapCenter Server using PowerShell, SCCLI, and REST API

You can configure MFA in SnapCenter Server using PowerShell, SCCLI, and REST API.

SnapCenter MFA CLI authentication

In PowerShell and SCCLI, the existing cmdlet (Open-SmConnection) is extended with one more field called "AccessToken" to use the bearer token to authenticate the user.

```
Open-SmConnection -Credential <PSCredential> [-SMSbaseUrl <String>] [-Port
<String>] [-RoleName <String>] [ -AccessToken <string>]
```

After the above cmdlet is executed, a session is created for the respective user to execute further SnapCenter cmdlets.

SnapCenter MFA Rest API Authentication

Use bearer token in the format *Authorization=Bearer <access token>* in REST API client (like Postman or swagger) and mention the user RoleName in the header to get a successful response from SnapCenter.

MFA Rest API Workflow

When MFA is configured with AD FS, you should authenticate using an access (bearer) token to access the SnapCenter application by any Rest API.

About this task

- You can use any REST client like Postman, Swagger UI or FireCamp.
- Get an access token and use it to authenticate subsequent requests (SnapCenter Rest API) to perform any operation.

Steps

To authenticate through AD FS MFA

1. Configure the REST client to call AD FS endpoint to get the access token.

   When you hit the button to get an access token for an application, you will be redirected to the AD FS SSO page where you must provide your AD credentials and authenticate with MFA. 1. In the AD FS SSO page, type your username or email in the Username text box.

   + Usernames must be formatted as user@domain or domain\user.

2. In the Password text box, type your password.
3. Click **Log in**.
4. From the **Sign-in Options** section, select an authentication option and authenticate (depending on your configuration).
   - Push: Approve the push notification that is sent to your phone.
   - QR Code: Use the AUTH Point mobile app to scan the QR code, then type the verification code shown in the app

- One-Time Password: Type the one-time password for your token.

5. After successful authentication, a popup will open that contains the Access, ID, and Refresh Token.

   Copy the access token and use it in the SnapCenter Rest API to perform the operation.

6. In the Rest API, you should pass the access token and role name in the header section.

7. SnapCenter validates this access token from AD FS.

   If it is a valid token, SnapCenter decodes it and gets the username.

8. Using the Username and Role Name, SnapCenter authenticates the user for an API execution.

   If the authentication succeeds, SnapCenter returns the result else an error message is displayed.

Enable or disable SnapCenter MFA functionality for Rest API, CLI, and GUI

GUI

Steps

1. Log into the SnapCenter Server as the SnapCenter Administrator.
2. Click **Settings** > **Global Settings** > **MultiFactorAuthentication(MFA) Settings**
3. Select the interface (GUI/RST API/CLI) to enable or disable the MFA login.

PowerShell interface

Steps

1. Run the PowerShell or CLI commands for enabling MFA for GUI, Rest API, PowerShell, and SCCLI.

   ```
   Set-SmMultiFactorAuthentication -IsGuiMFAEnabled -IsRestApiMFAEnabled
   -IsCliMFAEnabled -Path
   ```

   The path parameter specifies the location of the AD FS MFA metadata xml file.

   Enables MFA for SnapCenter GUI, Rest API, PowerShell, and SCCLI configured with specified AD FS metadata file path.

2. Check the MFA configuration status and settings by using the `Get-SmMultiFactorAuthentication` cmdlet.

SCCLI Interface

Steps

1. ```
   # sccli Set-SmMultiFactorAuthentication -IsGuiMFAEnabled true
   -IsRESTAPIMFAEnabled true -IsCliMFAEnabled true -Path
   "C:\ADFS_metadata\abc.xml"
   ```

2. ```
   # sccli Get-SmMultiFactorAuthentication
   ```

REST APIs

1. Run the following post API for enabling MFA for GUI, Rest API, PowerShell, and SCCLI.

Parameter	Value
Requested URL	/api/4.9/settings/multifactorauthentication
HTTP method	Post
Request Body	{ "IsGuiMFAEnabled": false, "IsRestApiMFAEnabled": true, "IsCliMFAEnabled": false, "ADFSConfigFilePath": "C:\\ADFS_metadata\\abc.xml" }
Response Body	{ "MFAConfiguration": { "IsGuiMFAEnabled": false, "ADFSConfigFilePath": "C:\\ADFS_metadata\\abc.xml", "SCConfigFilePath": null, "IsRestApiMFAEnabled": true, "IsCliMFAEnabled": false, "ADFSHostName": "win-adfs-sc49.winscedom2.com" } }

2. Check the MFA configuration status and settings by using the following API.

Parameter	Value
Requested URL	/api/4.9/settings/multifactorauthentication
HTTP method	Get
Response Body	{ "MFAConfiguration": { "IsGuiMFAEnabled": false, "ADFSConfigFilePath": "C:\\ADFS_metadata\\abc.xml", "SCConfigFilePath": null, "IsRestApiMFAEnabled": true, "IsCliMFAEnabled": false, "ADFSHostName": "win-adfs-sc49.winscedom2.com" } }