Multi-Factor Authentication
Using Advanced Authentication

April 2019

Access Manager is a comprehensive access management solution that provides secure access to enterprise and web applications. Using traditional one-factor authentication, such as providing username and password, to access a resource can have many vulnerabilities. Access Manager supports multi-factor authentication to provide secure access from any device with minimal administration. You can integrate NetIQ Advanced Authentication with Access Manager to use multi-factor authentication. Advanced Authentication delivers various authentication mechanisms that enable identity assurance and proofing apart from traditional username and password based authentication. You can authenticate on diverse platforms by using various authenticators such as Fingerprint, OTP, and Smartphone.

For more information about Access Manager, see Access Manager Overview.


- Section 1, “Prerequisites,” on page 1
- Section 2, “Implementation Approaches,” on page 2
- Section 3, “Enabling Multi-factor Authentication Using Advanced Authentication,” on page 2
- Section 4, “Migrating from Plug-in-based to OAuth-based Integration,” on page 8
- Section 5, “Legal Notice,” on page 9

1 Prerequisites

☐ Access Manager is installed and configured.

See NetIQ Access Manager 4.5 Installation and Upgrade Guide.

☐ Advanced Authentication server 5.6 or later is installed and configured.


An Access Manager administrator account is available.
An Advanced Authentication administrator account is available.

2 Implementation Approaches

You can integrate Advanced Authentication with Access Manager by using any one of the following approaches:

- **Plug-in-based approach**: The Advanced Authentication functionality is embedded in Access Manager.
- **OAuth-based approach**: This is available in Access Manager 4.4 and later versions. This approach uses the OAuth claims-based authentication mechanism for secure and trusted communication. Any new methods introduced in the Advanced Authentication server become dynamically available in Access Manager without making any modification in the product.

The following table lists the differences between Plug-in-based and OAuth-based approaches:

<table>
<thead>
<tr>
<th>Plug-in-based</th>
<th>OAuth-based</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uses Advanced Authentication Rest API.</td>
<td>Uses OAuth protocol.</td>
</tr>
<tr>
<td>Requires configuring each method separately.</td>
<td>Requires configuring only the Advanced Authentication Generic class. You can configure all Advanced Authentication methods using this class.</td>
</tr>
<tr>
<td>Any new method, which is added in Advanced Authentication after integration, is not available in Access Manager. You may need to upgrade Access Manager to a higher version to use that new method.</td>
<td>If any new method is introduced in the Advanced Authentication server, it is available in Access Manager automatically without any upgrade.</td>
</tr>
<tr>
<td>Supports brand customization.</td>
<td>Advanced Authentication 6.0 and later versions support branding customization.</td>
</tr>
</tbody>
</table>

It is recommended to integrate Advanced Authentication with Access Manager by using the OAuth-based approach.

3 Enabling Multi-factor Authentication Using Advanced Authentication

Enabling multi-factor authentication consists of the following tasks:

1. **Integrating Advanced Authentication with Access Manager**
2. **Configuring Multi-Factor Authentication**
3. **End Users Enrollment in the Advanced Authentication Self-Service Portal**
4. **Verifying the Integration**
3.1 Integrating Advanced Authentication with Access Manager

To integrate both products, you must first configure the Advanced Authentication server and then configure Advanced Authentication server details in Access Manager.

Configure the Advanced Authentication Server

1. Log in to Advanced Authentication as an administrator.
2. Verify that the NAM event is available in Events.

**NOTE:** The NAM event is created by default when you install Advanced Authentication. There might be a rare scenario where the NAM event does not get created by default. Re-installing Advanced Authentication will resolve the issue.

3. Set up a central user store that both Advanced Authentication and Access Manager will use while authenticating a user. You can add a new repository in Advanced Authentication server or configure details of an existing Access Manager user store. If you add a new repository in Advanced Authentication, configure the same repository when you Configure the Advanced Authentication Server Details in Access Manager.

For more information about how to add a repository, see Adding a Repository (https://www.netiq.com/documentation/advanced-authentication-62/server-administrator-guide/data/adding_repository.html).

4. Configure methods.

An Advanced Authentication method verifies the identity of a user who tries to access resources. You can configure the methods depending on your requirement. For example, in an Email OTP method, you can specify the values of different parameters, such as OTP period, OTP format, subject, and error message.


5. Create a chain.

A chain is a combination of methods. A user needs to execute and succeed all methods of a chain to be authenticated. While creating a chain, add the methods in the order of priority of execution. In Roles and Groups, assign the chain to the user group that is configured in the repository. For example, specify XYZ\Allowed RODC Password Replication Group, where XYZ is the name of the repository.


6. (Required only for the OAuth-based approach) Configure an event.

Advanced Authentication provides authentication events for Access Manager. An event leverages the Advanced Authentication functionalities for Access Manager. Access Manager triggers the respective authentication event when a user tries to access it.

Perform the following steps to configure an event:

6a. Click Events > Add.
6b. Specify a name for the event.
6c. Select OAuth2 from Event type.
6d. Select the required chains.

**NOTE:** You need Client ID and Client secret while configuring the Advanced Authentication server in Access Manager. You cannot view Client secret later, therefore you must make a note of this value.
Specify HTTPS://<identity server-url>:<port>/nidp/oauth/nam/callback in Redirect URIs.

For example, if the Identity Server URL is HTTPS://domain.example.com:8443/nidp, where domain.example.com is the domain name and 8443 is the port, specify HTTPS://domain.example.com:8443/nidp/oauth/nam/callback.

NOTE: For the Plug-in based methods, you do not need to create the OAuth 2.0 event, because a default NAM event is created when you install Advanced Authentication. Access manager uses the NAM event if you integrate using the Plug-in based approach and OAuth 2.0 event when you integrate using OAuth-based approach.

7 (Required only for the Plug-in-based approach) Assign the created chain to the NAM event in the Advanced Authentication server.

Configure the Advanced Authentication Server Details in Access Manager

1 Click Devices > Identity Servers > Shared Settings > Advanced Authentication.

2 Specify the domain name or IP address of the Advanced Authentication server and port in Server Domain.

NOTE: When using the Plug-in-based methods, skip to Step 5 on page 5.

3 (Required only for OAuth-based approach) Select Integrate using OAuth under OAuth Event Configuration.

4 (Required only for OAuth-based approach) Specify the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Name</td>
<td>Specify an event name. This event name must be identical to the event name specified in the Advanced Authentication administration portal.</td>
</tr>
<tr>
<td>Client ID</td>
<td>Specify the client ID that was generated while creating the OAuth 2.0 event in the Advanced Authentication administration portal.</td>
</tr>
<tr>
<td>Client Secret</td>
<td>Specify the client secret that was generated while creating the OAuth 2.0 event in the Advanced Authentication administration portal.</td>
</tr>
</tbody>
</table>

Access Manager uses the endpoint links to retrieve token and user details from the Advanced Authentication server. These are default endpoint links. If the values of the URIs change because of modification of the Advanced Authentication authorization server, then you can change the values here.

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Authorization URL</td>
<td>Access Manager uses this URL to retrieve the authorization code from the Advanced Authentication server.</td>
</tr>
<tr>
<td>Token URL</td>
<td>Access Manager uses this URL to exchange the authorization code with the access token.</td>
</tr>
<tr>
<td>User Info URL</td>
<td>Access Manager sends the access token to this URL to get the user details from the Advanced Authentication server.</td>
</tr>
</tbody>
</table>

The fields under Integration URLs are auto-populated after you specify the server domain address.
5 Click Apply.

6 Verify that the config.xml file is available in each Identity Server node at the following location:
   Linux: /etc/aaplugin/
   Windows: C:/Program Files/Novell/aaplugin

**NOTE:** On Windows, if you see an error in the log indicating that the configuration file is not found at C:\Program Files (x86)\Novell\aaplugin, perform the following steps:

1. Create a directory named Novell in C:\Program Files (x86).
2. Copy the aaplugin directory and its content from C:\Program Files\Novell\ to C:\Program Files (x86)\Novell\ and try authentication again.

7 Verify that the endpoint has been created in the Advanced Authentication server. Go to the Advanced Authentication administration portal and verify that the hostname or domain name of the Identity Server cluster is displayed as the endpoint under Endpoints.

8 In Access Manager, go to Dashboard and click Certificates > Trusted Roots to verify if the Advanced Authentication server certificate is available.

   If the certificate is not available, then perform the following steps to import the certificate:

   8a Click Certificates > Trusted Roots > Auto-Import From Server.
   8b Specify the server IP/DNS, port, and certificate name.
   8c Click OK.

9 Configure the same user store or repository that you added in the Advanced Authentication server. See Step 3 on page 3.

   9a Click Devices > Identity Servers > Servers > Edit > Local > User Stores > New.
   9b Specify the details and click Finish.
   9c Update Identity Server.

Skip this step if you have configured an existing Access Manager user store in the Advanced Authentication server.
3.2 Configuring Multi-Factor Authentication

Access Manager performs the first factor authentication when you protect a resource or an application using Access Manager. You can use Advanced Authentication to perform the second or third factor authentication.

The following procedure illustrates the steps for configuring multi-factor authentication when the integration is done using the OAuth-based approach:

1 Create a class.
   1a Click Devices > Identity Servers > Edit > Local > Classes.
   1b Click New, then specify the following details:
      - **Display name:** Specify a name for the class
      - **Java class:** Select Advanced Authentication Generic Class. The Java class path is configured automatically
   1c Click Next > Finish.

2 Create a method for this class.
   2a Click Devices > Identity Server > Edit > Local > Classes > New.
   2b Select a chain in Advanced Authentication Chains. If you do not specify any chain, the user is prompted to select the preferred chain for authentication.

   **NOTE:** If no chain is listed in Advanced Authentication Chains, create a chain in the Advanced Authentication server. If a chain is available in the Advanced Authentication server, but the chain is not listed in Advanced Authentication Chains, then assign the chain to the configured Access Manager OAuth event in the Advanced Authentication administration portal.

You can create multiple methods using the Advanced Authentication Generic Class. You do not need to create a new class every time you create a new method. You just need to add the new chain to the event in Advanced Authentication Administration portal, as mentioned in the Step 6d on page 3. Then while creating the method, select the chain in Advanced Authentication Chains.

3 Create a contract for the method.
   3a Click Devices > Identity Servers > Edit > Local > Contracts > New.
   3b In URI, specify a value that uniquely identifies the contract from all other contracts. This value is used to identify this contract for external providers and is a unique path value that you create. For example, specify /nam/AAgenericcontract or /mycompany/name/password/form.
   3c In Methods, first add an Access Manager’s authentication method (for example, Secure Name/Password - Form) and then the Advanced Authentication method that you created in the preceding step.

   **NOTE:** You can use more than one Advanced Authentication methods.

   3d Click Apply > OK.
   3e Update Identity Server.

3.3 End Users Enrollment in the Advanced Authentication Self-Service Portal

To perform authentication with Advanced Authentication, end users must enroll all methods of an authentication chain that they can use for authentication. A method or an authenticator is a set of encrypted data that contains user's authentication information. Users can use authenticators to log in to different resources.

Users must perform the following steps to enroll authenticators:

1. Access the Advanced Authentication Self-Service portal.
   - URL of the portal when it is not protected by Access Gateway: https://<Advanced Authentication hostname or IP address>/account
   - URL of the portal when Access Gateway protects Identity Server and Advanced Authentication: https://<Access Gateway hostname>/account

2. Select a method from Add Authenticator to enroll.
   - For example, to enroll Email OTP method, select Email OTP, specify your email ID, and click Save.
   - Email OTP is displayed in the Enrolled Authenticators section.

3. Verify that the enrolled authenticator is working.
   - a. Click Email OTP > Test.
   - b. Specify the OTP and click Next.

   Authenticator enrollment is successful when you receive a confirmation message.

Users can enroll multiple authenticators using the preceding procedure.

3.4 Verifying the Integration

To verify that the integration is successful, create a dummy user account and enroll one or more authenticators.

For information about how an end user enrolls to authenticators, see Section 3.3, “End Users Enrollment in the Advanced Authentication Self-Service Portal,” on page 7.

Use this user account to access a protected resource by executing the contract created in Access Manager.

Verifying the Plug-in-based Integration

Perform the following steps in Access Manager:

1. Create an Advanced Authentication class. You can use a Dynamic class or any other class except the Generic class.

2. Create a method and include the class created in the previous step, add a repository, and add the Advanced Authentication Enrollment URL property.
   - Specify the URL of Advanced Authentication portal for authenticator enrollments.
   - For example:
     - URL of the portal when it is not protected by Access Gateway: https://<Advanced Authentication hostname or IP address>/account
     - URL of the portal when Access Gateway protects Identity Server and Advanced Authentication: https://<Access Gateway hostname>/account
3 Create a contract. First add an Access Manager-specific method that supports LDAP credential-based authentication, such Secure Name/Password - Form and Name/Password - Basic and then add the Advanced Authentication method that you created in the previous step.

4 Using the dummy user’s account, access Identity Server or a protected resource to which this contract has been assigned and execute this contract. ([https://<identity server-url>:<port>/nidp](https://<identity server-url>:<port>/nidp))

   The user must be able to authenticate to each method: first to Access Manager’s method and then to the Advanced Authentication’s method.

   If authentication succeeds, the integration is successful.

### Verifying the OAuth-based Integration

Perform the following steps in Access Manager:

1 Create a class using Advanced Authentication Generic class.

2 Create a method with this class and select the required chain in Advanced Authentication Chains.

3 Create a contract. First add an Access Manager-specific method that supports LDAP credential-based authentication, such Secure Name/Password - Form and Name/Password - Basic and then add the Advanced Authentication method that you created in the previous step.

4 Using the dummy user’s account, access Identity Server or a protected resource to which this contract has been assigned and execute this contract. ([https://<identity server-url>:<port>/nidp](https://<identity server-url>:<port>/nidp))

   Specify the user name and password for first factor authentication. And then Identity Server redirects the login request to Advanced Authentication OSP for chain execution.

   On the OSP page, you can select the chain that you want to authenticate with. If you have selected a chain while configuring the method, then you will be prompted with the same chain on the OSP page.

   If authentication succeeds on the OSP page and you are redirected to Identity Server or protected resource, the integration is successful.

### 4 Migrating from Plug-in-based to OAuth-based Integration

You can migrate from the Plug-in-based approach to the OAuth-based approach by performing the following steps:

1 Log in to the Advanced Authentication administration portal as an administrator.

2 Configure an OAuth 2.0 event.

   1. Click Events > Add.

   2. Specify a name for the event.

   3. Select OAuth2 in Event type.

   4. Select the required chains.

   **NOTE:** You need Client ID and Client secret while configuring the Advanced Authentication server in Access Manager. You cannot view the Client secret later, therefore you must make a note of this value.

   For example, if the Identity Server URL is `https://domain.example.com:8443/nidp`, where `domain.example.com` is the domain name and `8443` is the port, specify `https://domain.example.com:8443/nidp/oauth/nam/callback`.

3 Log in to Access Manager Administration Console and perform the following steps:
   3a Click Devices > Identity Servers > Shared Settings > Advanced Authentication.
   3b Select Integrate using OAuth under OAuth Event Configuration.
   3c Specify the following details:

<table>
<thead>
<tr>
<th>Field</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event Name</td>
<td>Specify an event name. This event name must be identical to the event name specified in the Advanced Authentication administration portal.</td>
</tr>
<tr>
<td>Client ID</td>
<td>Specify the client ID that was generated while creating the OAuth 2.0 event in the Advanced Authentication administration portal.</td>
</tr>
<tr>
<td>Client Secret</td>
<td>Specify the client secret that was generated while creating the OAuth 2.0 event in the Advanced Authentication administration portal.</td>
</tr>
</tbody>
</table>

3d Click Apply.

4 Verify the integration. See “Verifying the OAuth-based Integration” on page 8.


5 Legal Notice

For information about legal notices, trademarks, disclaimers, warranties, export and other use restrictions, U.S. Government rights, patent policy, and FIPS compliance, see [https://www.netiq.com/company/legal/](https://www.netiq.com/company/legal/).

Copyright © 2019 NetIQ Corporation. All Rights Reserved.