Using Microsoft Azure Active Directory MFA as SAML IdP with Pulse Connect Secure

Deployment Guide
Introduction

This document describes how to set up Pulse Connect Secure for SP-initiated SAML authentication using the Microsoft Azure Active Directory as the SAML IdP. It also describes the user experience with Web browser and Pulse Secure Client access methods.

Prerequisites

Ensure you have the following:

- Administrative access to the Azure Management Portal
  - Azure subscription that includes Active Directory and Multi-Factor Authentication (MFA)
- Pulse Connect Secure appliance running 8.2R1 or later.

Process Steps

The set up includes the following process steps:

- Setting up Microsoft Azure Active Directory
- Setting up Pulse Connect Secure
Setting up Microsoft Azure Active Directory

Perform the following steps to configure Azure AD:

1. Log into the Azure Management Portal.
2. In the left pane, select ACTIVE DIRECTORY.
3. Select an active directory from the active directory list, and click APPLICATIONS.
4. Click the Add button at the bottom center of the page, click ADD.
5. In the pop-up window displayed, click **Add an application my organization is developing.**

6. In the **Tell us about your application** window, enter a name for the application. This has only local significance inside Azure.
   
   **Example:** PCS-MFA
   
   Keep the default option, **WEB APPLICATION AND/OR WEB API**, and click **Next** to continue.

7. In the **App properties** window, do the following:
a. **SIGN-ON URL** – Enter a URL. As this solution is supported only for SP-Initiated SAML authentication, meaning the user will first connect to the Pulse Connect Secure service and then be redirected to Microsoft Azure. This URL is never used, but it must be provided. In this example, it is `https://pcs.adminuser1.net/Saml.sso/SAML2/POST`.

b. **APP ID URI** – This is a VERY important entry, as it must match the unique Service Provider (SP) Entity ID configured in the Pulse Connect Secure SAML Authentication Server later in this document. By default, in the Pulse Connect Secure, this is set to:
   `https://[fqdn of PCS]/dana-na/auth/saml-endpoint.cgi?p=sp1`
   where the ending `sp1` indicates it is the first SAML Service Provider.
   As there could be others already defined, you might have to go back and change this later.

   In this example, this is set to `https://pcs.adminuser1.net/dana-na/auth/saml-endpoint.cgi?p=sp2`; click the tick mark to complete.

   ✅ Note: App ID URI can be changed later. So, do not worry if you did not get this correct.

8. After the application is added, click **CONFIGURE**.

9. Scroll down to the **single sign-on** section and verify that **App ID URI** is the same as in step 7.b, that is `https://pcs.adminuser1.net/dana-na/auth/saml-endpoint.cgi?p=sp2`.

   Note, that this must match the unique Service Provider (SP) Entity ID to be configured on the PCS SAML Authentication server.
10. Change the REPLY URL to the SAML consumer URL that is used in Pulse Connect Secure. This has the format: https://[fqdn of PCS]/dana-na/auth/saml-consumer.

   In this example, it is: https://pcs.adminuser1.net/dana-na/auth/saml-consumer.cgi.

<table>
<thead>
<tr>
<th>single sign-on</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>APP ID URL</strong></td>
</tr>
<tr>
<td><strong>REPLY URL</strong></td>
</tr>
</tbody>
</table>

11. Click the Save button at the bottom of the page.

12. Click VIEW ENDPOINTS at the bottom of the screen. The App Endpoints window displays the list of endpoints. Note that the Federation Metadata Document URL is the Metadata URL for this application.

   In this example: https://login.microsoftonline.com/d2bef2fc-7f18-447b-92e2-2339e41e175d/federationmetadata/2007-06/federationmetadata.xml

   Copy and save this URL. We will return to this later in the Pulse Connect Secure configuration.

13. Close the VIEW ENDPOINTS page by clicking the tick mark.

14. Click MANAGE MANIFEST at the bottom of the screen, and select Download Manifest and save the file to your local machine.

15. Open the file in a notepad from the saved location and search for “logoutUrl”.

   ... "homepage": "https://pcs.adminuser1.net/Saml.sso/SAML2/POST",
   "identifierUris": ["https://pcs.adminuser1.net/dana-na/auth/saml-endpoint.cgi?p=sp2"],
   "keyCredentials": [],
   "knownClientApplications": [],
"logoutUrl": null,
"oauth2AllowImplicitFlow": false,
...

16. Configure the logoutUrl:

It has the format like “https:// [fqdn of PCS]/Saml.sso/SLO/Redirect”
So in this example it is:
...
"homepage": "https://pcs.adminuser1.net/Saml.sso/SAML2/POST", "identifierUris":
["https://pcs.adminuser1.net/dana-na/auth/saml-endpoint.cgi?p=sp2"],
"keyCredentials": [],
"knownClientApplications": [],
"logoutUrl": "https://pcs.adminuser1.net/Saml.sso/SLO/Redirect", "oauth2AllowImplicitFlow": false,
...

17. Save the file.
18. Click MANAGE MANIFEST at the bottom of the screen, and select Upload Manifest.

19. In the Upload Manifest window, browse and upload the file you saved.

This concludes the setup in Microsoft Azure. In next section you will configure the Pulse Connect Secure to use this SAML2.0 IdP as a SAML Authentication server.
Setting up Pulse Connect Secure

To configure Pulse Connect Secure as a SAML Service Provider (SP) to Azure as the SAML IdP, start by importing the metadata from Azure.

1. Go to **System > Configuration > SAML**.

2. Select **New Metadata Provider**.
3. Set a name for the Metadata provider.
4. In the Metadata Provider Location Configuration section:
   - If the Pulse Connect Secure has internet access from the Internal interface, the Remote option can be used with the URL copied in step 12 above.
     In this example: https://login.microsoftonline.com/d2bef2fc-7f18-447b-92e2-2339e41e175d/federationmetadata/2007-06/federationmetadata.xml
   - If the Pulse Connect Secure do not have internet access from the Internal interface, open a browser and go to the saved URL.
     In this example: https://login.microsoftonline.com/d2bef2fc-7f18-447b-92e2-2339e41e175d/federationmetadata/2007-06/federationmetadata.xml
     And save the content in a file on your local computer. Then use the Local option and upload the metadata to Pulse Connect Secure.
5. Select the **Identity Provider** (IdP) Role for the new Metadata provider, and save changes.

![Identity Provider Role Selection](image)

6. Create a SAML Authentication Server.
   a. Go to **Authorization > Auth. Servers**.
   b. Select **New: SAML Server** and click **New Server**...

![Authentication Servers](image)

c. Set a **Server Name**, locally used in Pulse Connect Secure. Select **SAML Version 2.0** and **Configuration Mode Metadata**.

d. Use the **Identity Provider Entity Id** field to pick the Metadata provider to be used.

This will use the remote or locally uploaded metadata from Azure, and automatically sets various parameters for the SAML authentication server.
User Name Template
The User Name template sets the attribute from the Azure IdP that the Pulse Connect Secure will use internally as the user name for the authenticated user in Active Users, logs etc. If this field is left blank, then Pulse Connect Secure uses the persistent **NameID** used by Microsoft, *urn:oasis:names:tc:SAML:2.0:nameid-format:persistent*. That is an anonymous string like *OP_DMpDOO_VelsV2Xim7ylaJTBJRR7pjTZeBlX7xSFw* and is not really good to use in the Pulse Connect Secure as it gives no real user visibility.

As you add a new user to Azure AD, you add few other variables like User Name, First Name, Last Name, and Display Name.

These are present in the SAML Assertion returned to PCS as SAML attributes:

Variable `userAttr.http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name = nstark@adminuser1.net`

These attributes include special characters like : and /, which need to be suppressed when used in the **User Name Template** field. This is done using the `{ }` enclosure around the attribute. 

*<userAttr.{http://schemas.xmlsoap.org/ws/2005/05/identity/claims/name}>* will set the **name** attribute `nstark@adminuser1.net` as the username.

*<userAttr.{http://schemas.microsoft.com/identity/claims/displayname}>* will set the **displayname** attribute **Ned Stark** as the username.

In this example, the attribute

*<userAttr.{http://schemas.xmlsoap.org(ws/2005/05/identity/claims/name}>*

has been used as **User Name Template** attribute.
e. Single Logout is an optional setting. If this option is selected, it prompts for a new authentication after logout. If this option is not selected and you have not closed the browser, you can reconnect without authentication.

f. Next, select the Requested Authn Context Class as Password and the Comparison Method as exact.

g. Finally, set the Metadata Validity in terms of number of days. In this example: 100 days.
7. Save the Authentication server, then open the Authentication server to validate that the **Connect Secure Entity Id:** is matching the **APP ID URI** as in step 7b in the “Setting up Microsoft Azure Active Directory” section above. If it is not matching, go back to step 8 in the “Setting up Microsoft Azure Active Directory” section and edit the entry to match the SAML Authentication Server **Connect Secure Entity Id.**

8. Once that is done, or if it did already match, configure a Realm to use this Authentication Server and you are now ready to use the service.
User Experience
This section describes accessing Pulse Connect Secure using Web browser and Pulse Secure Client.

Browser Access – Windows 10 using Edge Browser

Note: These examples use another user account: admin@adminuser1.net

When you access the Pulse Connect Secure Sign-in URL, you are automatically redirected to the Microsoft login page and authenticated with your Azure AD credentials.

As this account is set up with Multi-Factor Authentication (MFA), you must provide additional information to successfully authenticate. This verification code can be submitted via SMS, phone call, verification pushed to mobile device application that needs to be accepted, or an actual T-OTP verification code sent from the mobile application.

In this example, it is a T-OTP verification code from the Azure Authenticator Mobile Application installed on an Apple iPhone.

For more details on Microsoft Azure MFA and options, please refer to https://azure.microsoft.com/en-us/services/multi-factor-authentication/
For the prompt to provide additional verification code, use the generated T-OTP in the Mobile App.

You are then redirected back to the Pulse Connect Secure service with an authenticated session.
In the Pulse Connect Secure admin UI, this session can be seen in the **Active Users** page for the user **admin@adminuser1.net**
Pulse Secure Client – Windows 10 using Pulse Secure Client for Windows 5.2.3

In the case of using Pulse Secure Client to access the Pulse Connect Secure service, you use a connection in the Pulse Secure Client.

As this access uses SAML Authentication, a Browser window will be opened for the authentication to MS Azure. You will be prompted for authentication with your Azure AD credentials as well as the additional verification code.
Once the authentication is successful, the browser window will automatically be closed and the VPN connection is up.

The above cases are provided as a few examples as this works with macOS, Apple iOS, Android and other OS and Web browsers as well.
References