Configure a SAML 2.0 provider for portals with AD FS

Article • 02/15/2022 • 2 minutes to read • 5 contributors

In this article

Important

The steps for the configuration of Active Directory Federation Services (AD FS) might vary depending on the version of your AD FS server.

Create an AD FS relying party trust

Note

See **Configure AD FS by using PowerShell**, for information about how to perform these steps in a PowerShell script.

1. Using the AD FS Management tool, go to **Service** > **Claim Descriptions**.

   a. Select **Add Claim Description**.

   b. Specify the claim:

   - Display name: **Persistent Identifier**
   - Claim identifier: **urn:oasis:names:tc:SAML:2.0:nameid-format:persistent**
   - **Enable** check box for: Publish this claim description in federation metadata as a claim type that this federation service can accept

- **Enable** check box for: Publish this claim description in federation metadata as a claim type that this federation service can send

c. Select **OK**.

2. Using the AD FS Management tool, select **Trust Relationships** >**Relying Party Trusts**.

a. Select **Add Relying Party Trust**.

b. Welcome: Select **Start**.

c. Select Data Source: Select **Enter data about the relying party manually**, and then select **Next**.

d. Specify Display Name: Enter a name, and then select **Next**. Example: https://portal.contoso.com/

e. Choose Profile: Select **AD FS 2.0 profile**, and then select **Next**.

f. Configure Certificate: Select **Next**.

g. Configure URL: Select the **Enable support for the SAML 2.0 WebSSO protocol** check box. Relying party SAML 2.0 SSO service URL: Enter https://portal.contoso.com/signin-saml2
Note that AD FS requires that the portal run on HTTPS.

> **Note**
>
> The resulting endpoint has the following settings:
>
> - Endpoint type: **SAML Assertion Consume Endpoints**
> - Binding: **POST**
> - Index: n/a (0)
> - URL: **https://portal.contoso.com/signin-saml2**

h. Configure Identities: Enter `https://portal.contoso.com/`, select **Add**, and then select **Next**. If applicable, you can add more identities for each additional relying party portal. Users can authenticate across any or all available identities.

i. Choose Issuance Authorization Rules: Select **Permit all users to access this relying party**, and then select **Next**.

j. Ready to Add Trust: Select **Next**.

k. Select **Close**.

3. Add the **Name ID** claim to the relying party trust:

TransformWindows account name to **Name ID** claim (Transform an Incoming Claim):

- Incoming claim type: **Windows account name**
- Outgoing claim type: **Name ID**
- Outgoing name ID format: **Persistent Identifier**
- Pass through all claim values

Configure the SAML 2.0 provider

After setting up the AD FS relying party trust, you can follow the steps in Configure a SAML 2.0 provider for portals.

Identity provider–initiated sign-in

AD FS supports the identity provider–initiated single sign-on (SSO) profile of the SAML 2.0 specification . In order for the portal (service provider) to respond properly to the SAML request started by the identity provider, the RelayState parameter must be encoded properly.

The basic string value to be encoded into the SAML RelayState parameter must be in the format `ReturnUrl=/content/sub-content/`, where `/content/sub-content/` is the path to the webpage you want to go to on the portal (service provider). The path can be replaced by any valid webpage on the portal. The string value is encoded and placed into a container string of the format `RPID=<URL encoded RPID>&RelayState=<URL encoded RelayState>`. This entire string is once again encoded and added to another container of the format `<https://adfs.contoso.com/adfs/ls/idpinitiatedsignon.aspx?RelayState=<URL> encoded RPID/RelayState>`.

For example, given the service provider path `/content/sub-content/` and the relying party ID `https://portal.contoso.com/`, construct the URL with the following steps:

- Encode the value `ReturnUrl=/content/sub-content/` to get `ReturnUrl%3D%2Fcontent%2Fsub-content%2F`
- Encode the value `https://portal.contoso.com/` to get `https%3A%2F%2Fportal.contoso.com%2F`
- Encode the value `RPID=https%3A%2F%2Fportal.contoso.com%2F&RelayState=ReturnUrl%3D%2Fcontent%2Fsub-content%2F` to get `RPID%3Dhttps%253A%252F%252Fportal.contoso.com%252F%26RelayState%3DReturnUrl%253D%252Fcontent%252Fsub-content%252F`
- Prepend the AD FS identity provider–initiated SSO path to get the final URL `https://adfs.contoso.com/adfs/ls/idpinitiatedsignon.aspx?RelayState=RPID%3Dhttps%253A%252F%252Fportal.contoso.com%252F%26RelayState%3DReturnUrl%253D%252Fcontent%252Fsub-content%252F`

You can use the following PowerShell script to construct the URL. Save the script to a file named Get-IdPInitiatedUrl.ps1.

Copy

```
<#

.SYNOPSIS

Constructs an IdP-initiated SSO URL to access a portal page on the ser-
vice provider.

.PARAMETER path

The path to the portal page.

.PARAMETER rpid

The relying party identifier.

.PARAMETER adfsPath

The AD FS IdP initiated SSO page.

.EXAMPLE

PS C:\\> .\\Get-IdPInitiatedUrl.ps1 -path "/content/sub-content/" -rpid
"https://portal.contoso.com/" -adfsPath
```

```
"https://adfs.contoso.com/adfs/ls/idpinitiatedsignon.aspx"

#>

param

(

[parameter(mandatory=$true,position=0)]

$path,

[parameter(mandatory=$true,position=1)]

$rpid,

[parameter(position=2)]

$adfsPath = https://adfs.contoso.com/adfs/ls/idpinitiatedsignon.aspx

)

$state = ReturnUrl=$path

$encodedPath = [uri]::EscapeDataString($state)

$encodedRpid = [uri]::EscapeDataString($rpid)

$encodedPathRpid = [uri]::EscapeDataString("RPID=$encodedRpid&Relay-
State=$encodedPath")

$idpInitiatedUrl = {0}?RelayState={1} -f $adfsPath, $encodedPathRpid

Write-Output $idpInitiatedUrl
```

Configure AD FS by using PowerShell

The process of adding a relying party trust in AD FS can also be performed by running the following PowerShell script on the AD FS server. Save the script to a file named Add-AdxPortalRelyingPartyTrustForSaml.ps1. After running the script, continue with configuring the portal site settings.

Copy

```
<#

.SYNOPSIS
```

```
Adds a SAML 2.0 relying party trust entry for a website.

.PARAMETER domain

The domain name of the portal.

.EXAMPLE

PS C:\\> .\\Add-AdxPortalRelyingPartyTrustForSaml.ps1 -domain portal.-
contoso.com

#>

param

(

[parameter(Mandatory=$true,Position=0)]

$domain,

[parameter(Position=1)]

$callbackPath = /signin-saml2

)

$VerbosePreference = Continue

$ErrorActionPreference = Stop

Import-Module adfs

Function Add-CrmRelyingPartyTrust

{

param (

[parameter(Mandatory=$true,Position=0)]

$name

)

$identifier = https://{0}/ -f $name

$samlEndpoint = New-ADFSSamlEndpoint -Binding POST -Protocol SAMLAsser-
tionConsumer -Uri (https://{0}{1} -f $name, $callbackPath)
```

```
$identityProviderValue = Get-ADFSProperties | % { $_.Identifier.Absolu-
teUri }

$issuanceTransformRules = @'

@RuleTemplate = MapClaims

@RuleName = Transform [!INCLUDE[pn-ms-windows-short]
(../../../includes/pn-ms-windows-short.md)] Account Name to Name ID
claim

c:[Type ==
"https://schemas.microsoft.com/ws/2008/06/identity/claims/windowsac-
countname"]

=> issue(Type =
"https://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier",
Issuer = c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value,
ValueType = c.ValueType,
Properties["https://schemas.xmlsoap.org/ws/2005/05/identity/claimpropert
ies/format"] = "urn:oasis:names:tc:SAML:2.0:nameid-format:persistent");

@RuleTemplate = LdapClaims

@RuleName = Send LDAP Claims

c:[Type ==
"https://schemas.microsoft.com/ws/2008/06/identity/claims/windowsac-
countname", Issuer == "AD AUTHORITY"]

=> issue(store = "[!INCLUDE[pn-active-directory](../../../includes/pn-
active-directory.md)]", types =
("https://schemas.xmlsoap.org/ws/2005/05/identity/claims/givenname",
"https://schemas.xmlsoap.org/ws/2005/05/identity/claims/surname",
"https://schemas.xmlsoap.org/ws/2005/05/identity/claims/emailaddress"),
query = ";givenName,sn,mail;{{0}}", param = c.Value);

'@ -f $identityProviderValue

$issuanceAuthorizationRules = @'

@RuleTemplate = AllowAllAuthzRule

=> issue(Type = https://schemas.microsoft.com/authorization/claims/per-
mit, Value = true);

'@

Add-ADFSRelyingPartyTrust -Name $name -Identifier $identifier -
SamlEndpoint $samlEndpoint -IssuanceTransformRules $issuanceTransform-
Rules -IssuanceAuthorizationRules $issuanceAuthorizationRules
```

```
}

# add the 'Identity Provider' claim description if it is missing

if (-not (Get-ADFSClaimDescription | ? { $_.Name -eq Persistent Identi-
fier })) {

Add-ADFSClaimDescription -name "Persistent Identifier" -ClaimType
"urn:oasis:names:tc:SAML:2.0:nameid-format:persistent" -IsOffered:$true
-IsAccepted:$true

}

# add the portal relying party trust

Add-CrmRelyingPartyTrust $domain
```

Configure a SAML 2.0 provider

After setting up the AD FS relying party trust, you can follow the steps in Configure a SAML 2.0 provider for portals.

See also

Configure a SAML 2.0 provider for portals with Azure AD
FAQ for using SAML 2.0 in portals
Configure a SAML 2.0 provider for portals

Recommended content

Create a Relying Party Trust

Learn how to create a relying party trust manually and use federation metadata.

Create a Rule to Send Group Membership as a Claim

Learn more about: Create a Rule to Send Group Membership as a Claim

Create a Rule to Send LDAP Attributes as Claims

Learn more about: Create a Rule to Send LDAP Attributes as Claims

How URIs Are Used in AD FS

Learn more about: How URIs Are Used in AD FS

AD FS 2016 Single Sign On Settings

Learn more about: AD FS Single Sign-On Settings

AD FS Troubleshooting - Idp-Initiated Sign On

This document describes how to troubleshoot the AD FS sign on page.

Build a Custom Authentication Method for AD FS in Windows Server

This scenario describes how to build a custom authentication method for AD FS in Windows Server.

Create a Rule to Permit or Deny Users Based on an Incoming Claim

Learn more about: Create a Rule to Permit or Deny Users Based on an Incoming Claim

Show more