



ComponentSpace  
SAML for ASP.NET  
ADFS  
Relying Party  
Integration Guide

## Contents

Introduction .....	1
Enabling IdP-Initiated SSO.....	1
Adding a Relying Party .....	1
Adding a Claims Rule.....	7
Specifying the Name ID Format .....	11
Reviewing Relying Party Configuration.....	14
ADFS SAML Metadata .....	25
Service Provider Configuration .....	26
SP-Initiated SSO.....	26
IdP-Initiated SSO .....	30
SAML Logout .....	34
ADFS Authentication Methods .....	35
Windows Integrated Authentication .....	37
Browser Support .....	37
Default User Name.....	38
Troubleshooting ADFS SSO .....	39

## Introduction

This document describes integration of a service provider with Active Directory Federation Services.

The Microsoft terminology for a SAML service provider is a relying party.

## Enabling IdP-Initiated SSO

Ensure IdP-initiated SSO is enabled in ADFS using the PowerShell cmdlets `Get-AdfsProperties` and `Set-AdfsProperties`.

```
Get-AdfsProperties | Select EnableIdpInitiatedSignonpage
```

```
Set-AdfsProperties -EnableIdpInitiatedSignonPage $True
```

For more information, refer to:

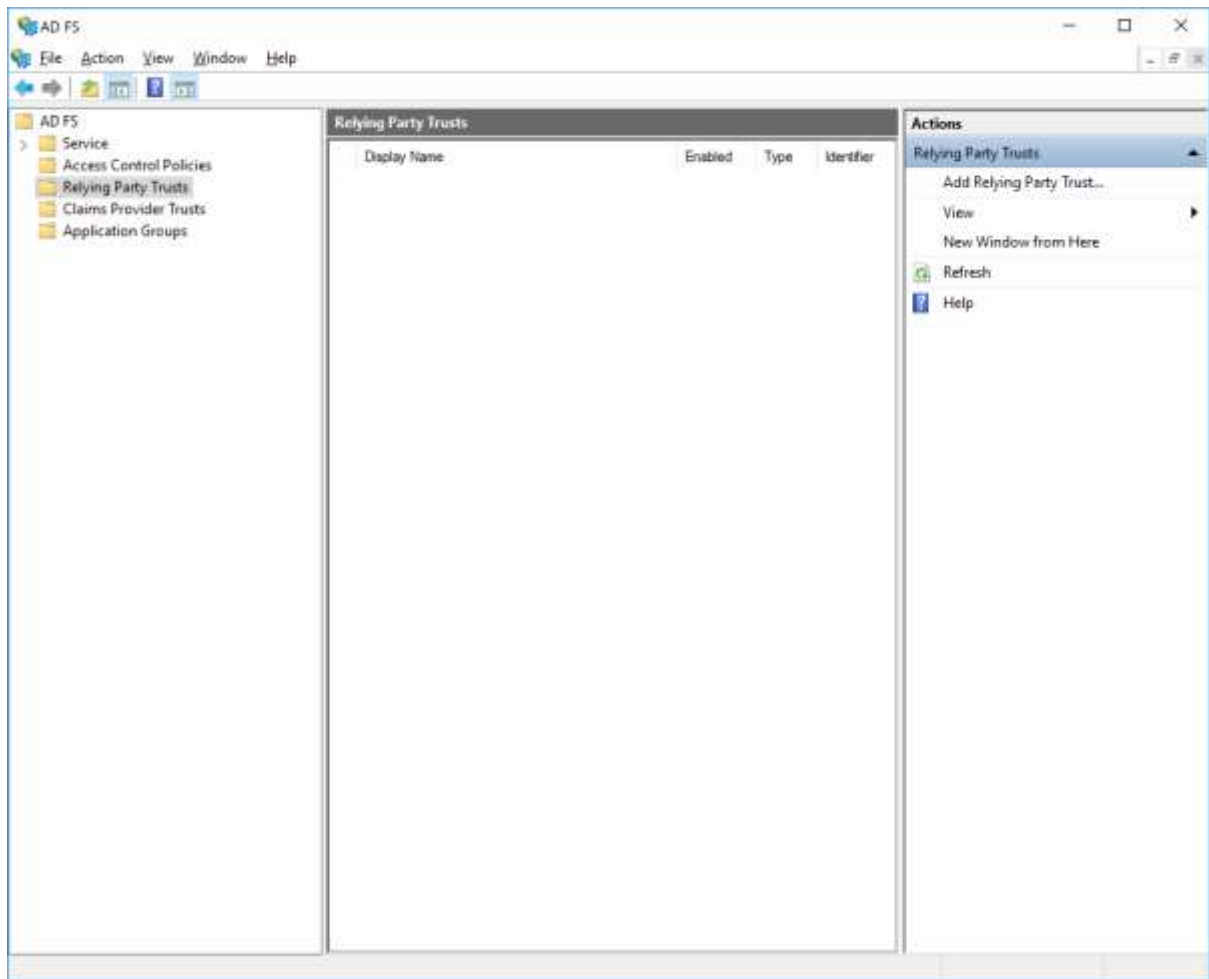
<https://blogs.technet.microsoft.com/rmilne/2017/06/20/how-to-enable-idpinitiatedsignon-page-in-ad-fs-2016/>

<https://docs.microsoft.com/en-us/powershell/module/adfs/set-adfsproperties>

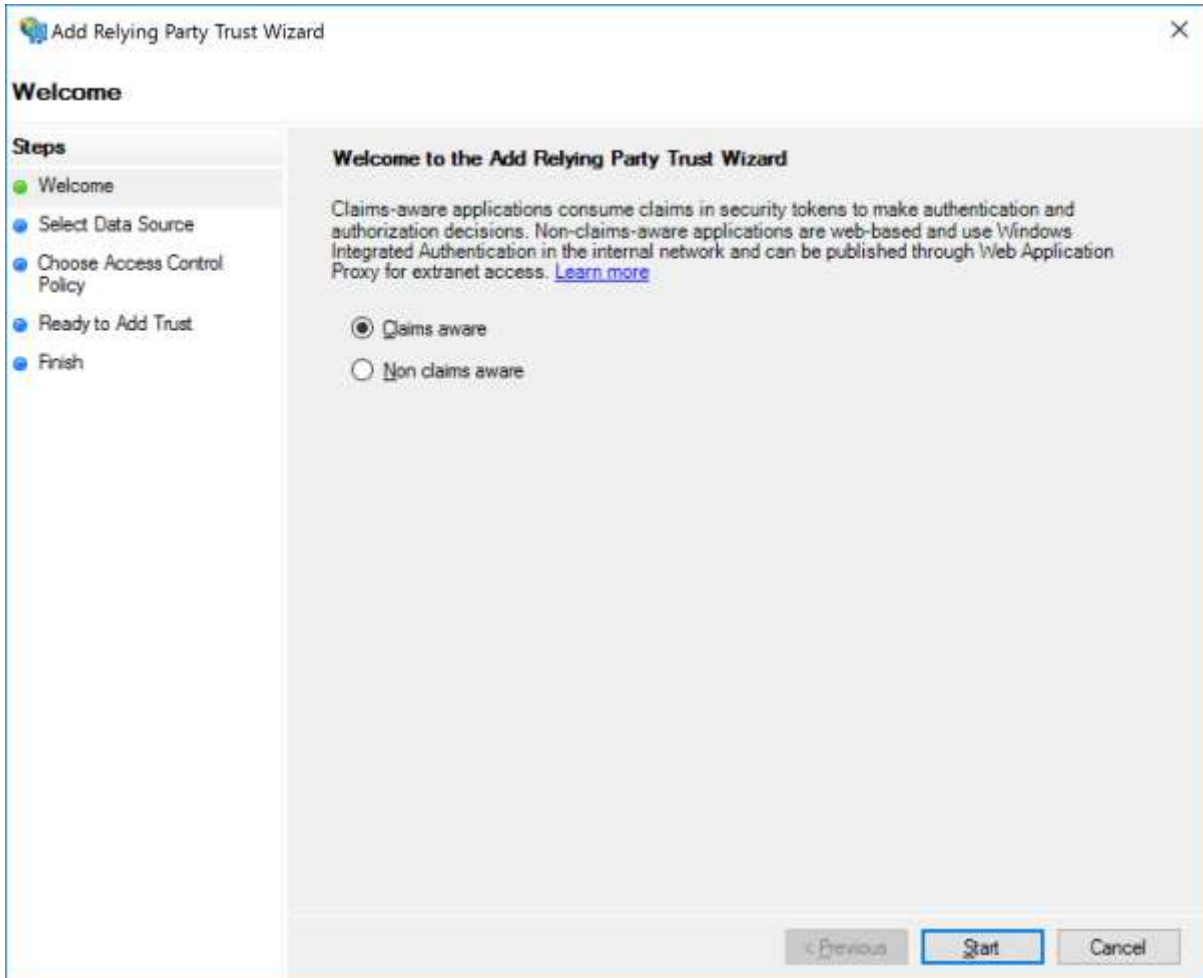
## Adding a Relying Party

Open the ADFS console and add a relying party trust.

## ComponentSpace SAML for ASP.NET ADFS Relying Party Integration Guide

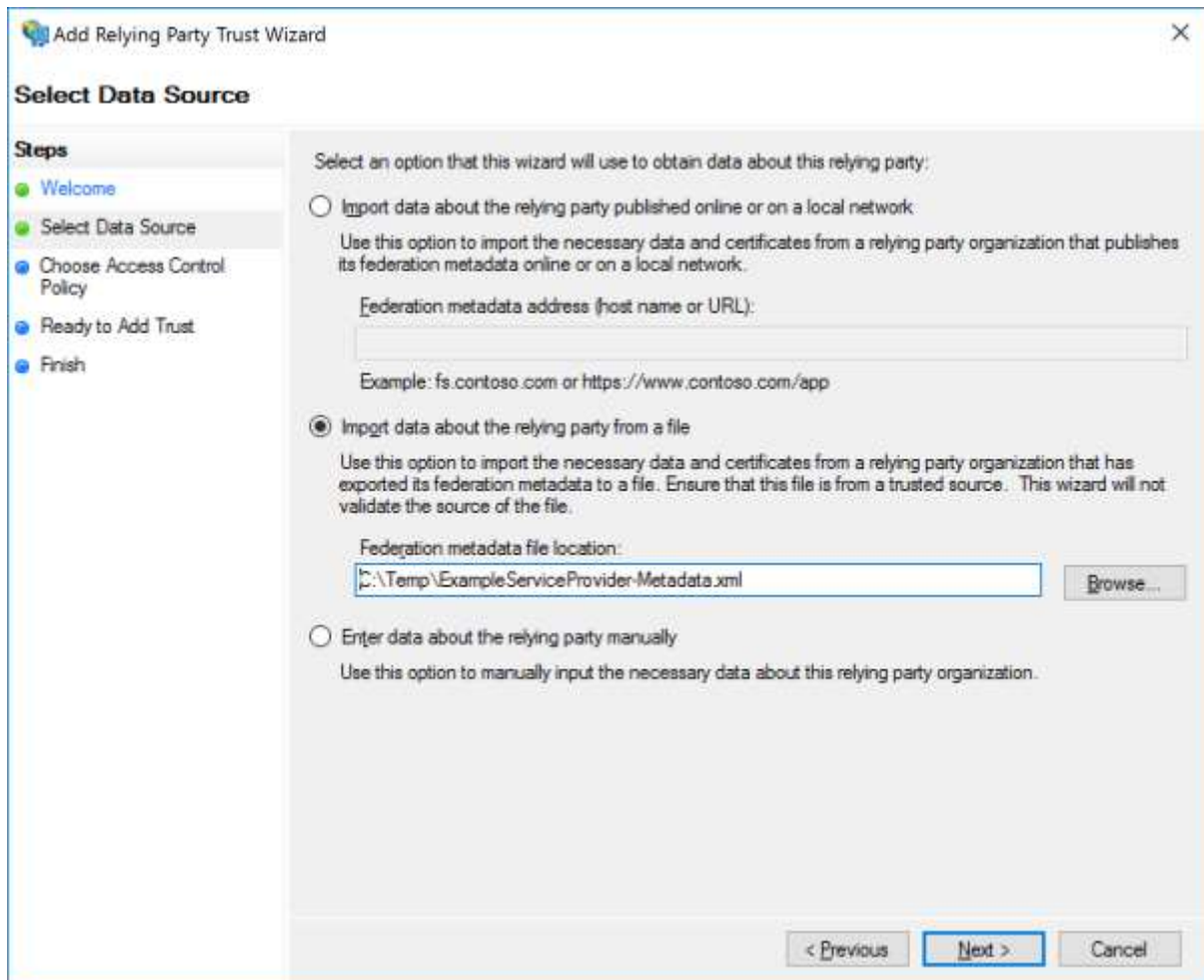


The relying party is claims aware.

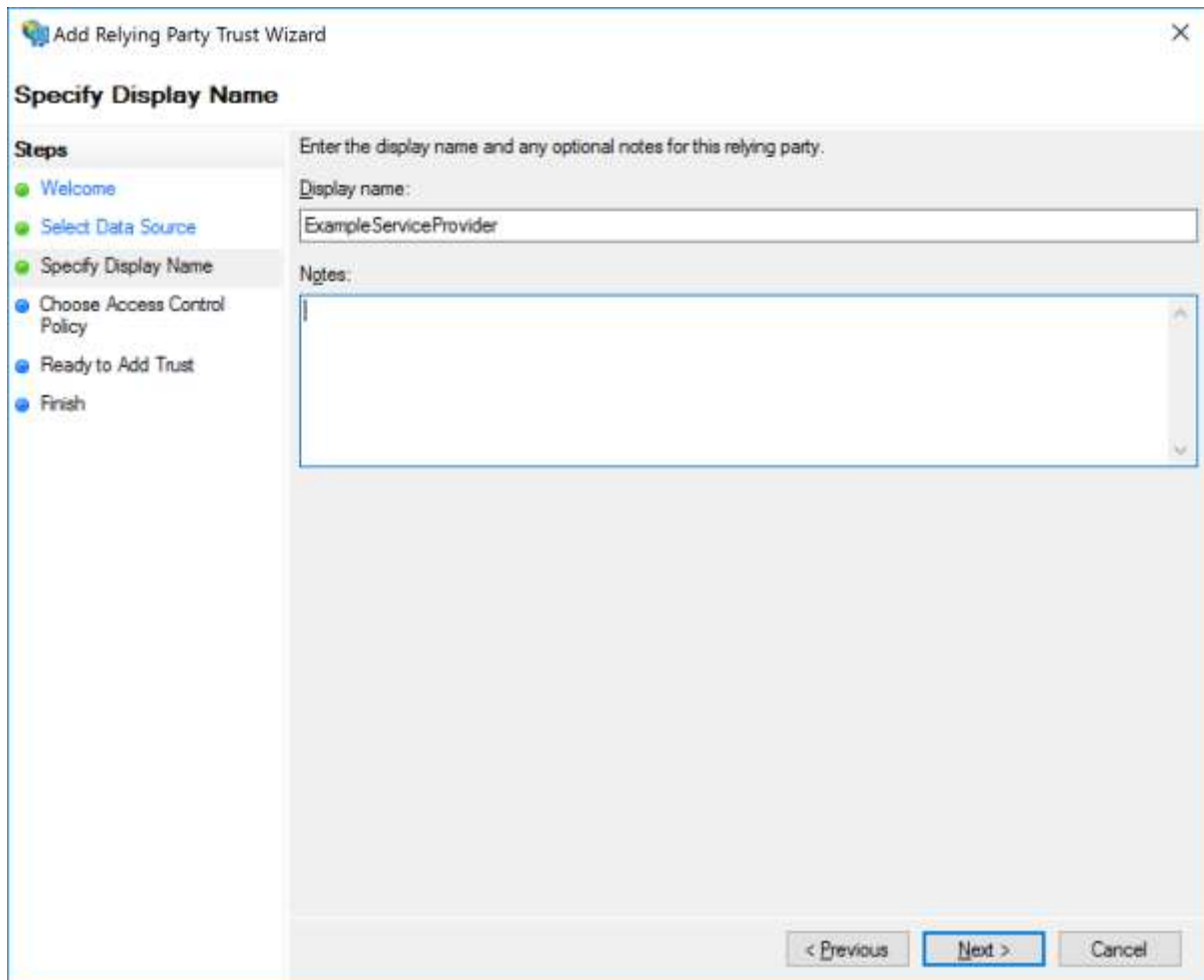


The relying party may be configured through SAML metadata or manually.

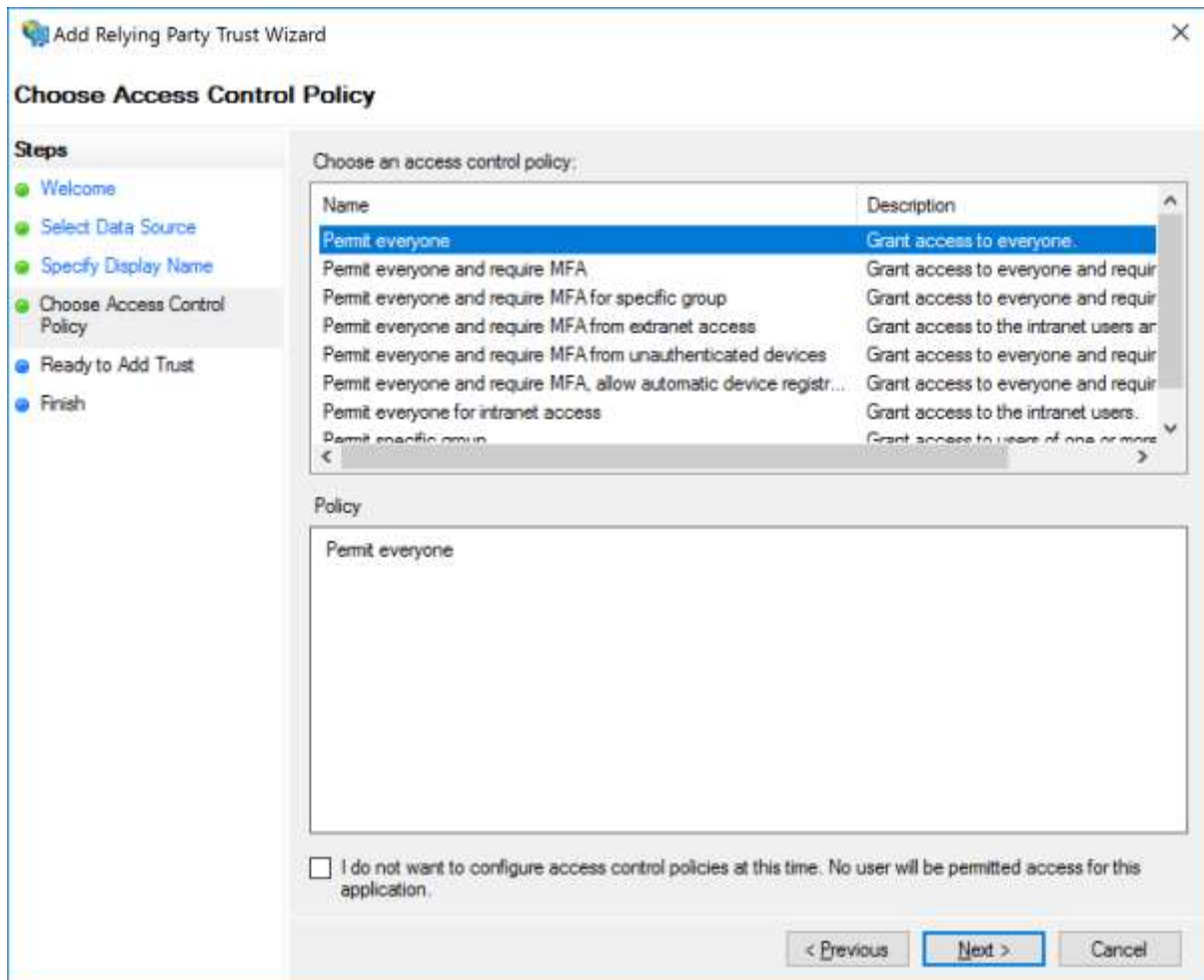
The included SAML metadata for the ExampleServiceProvider is used.



Provide a name purely for display purpose.

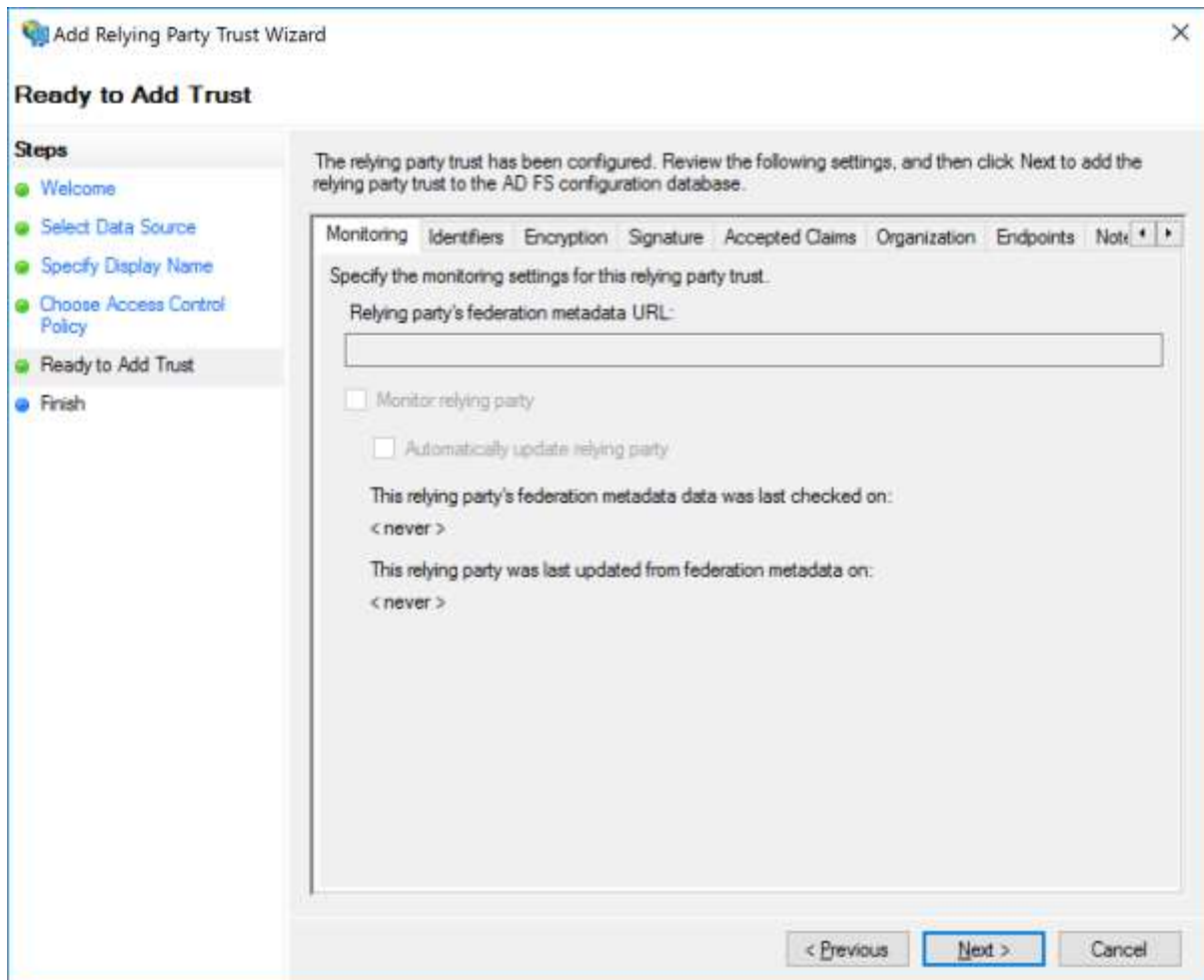


Specify the access control policy.



Review the configuration. This can be updated later if required.

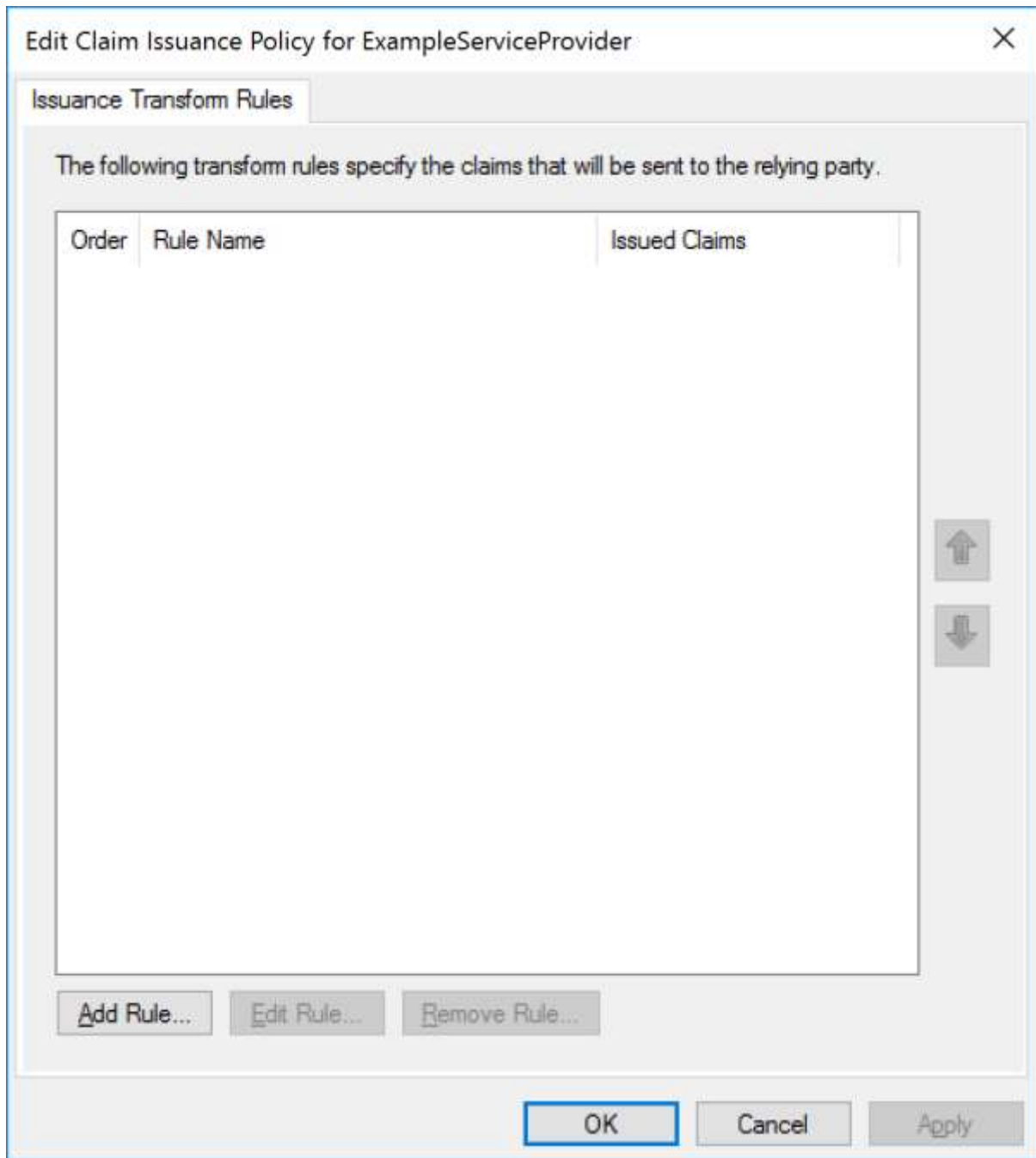




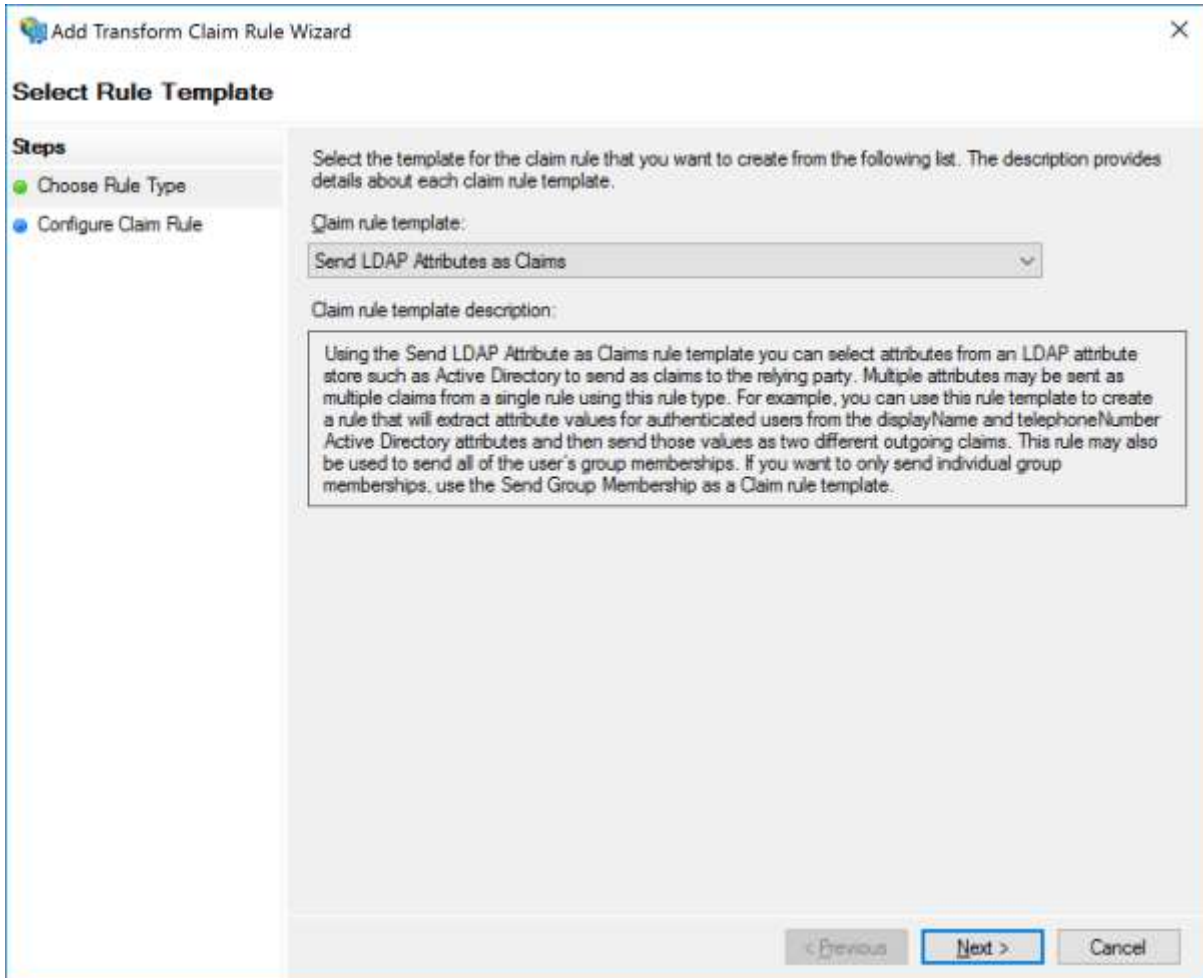
## Adding a Claims Rule

Claim rules map user information into the SAML subject name identifier and SAML attributes that are included in the SAML assertion sent to the service provider.

Add a rule.



User properties in Active Directory will be used.

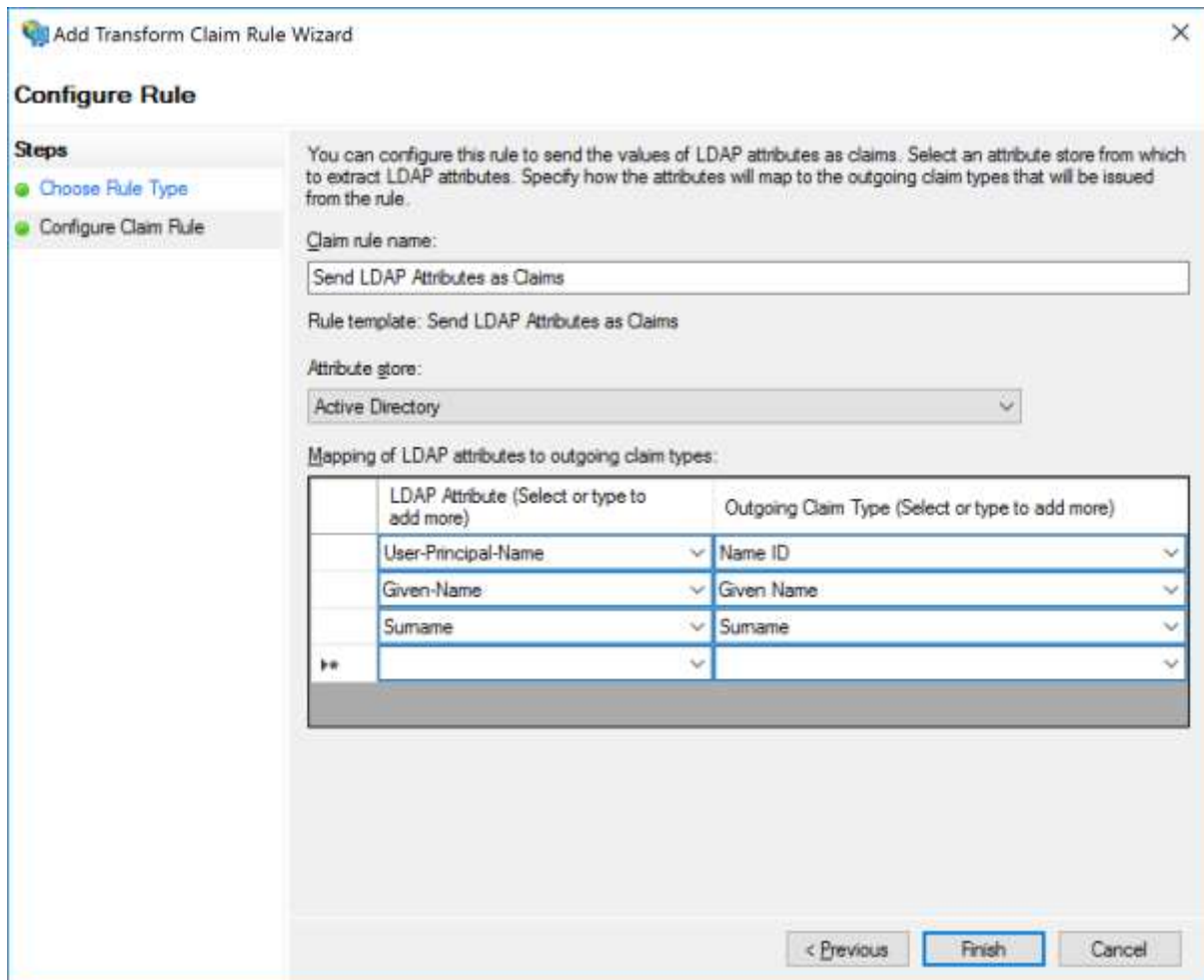


Specify the mapping.

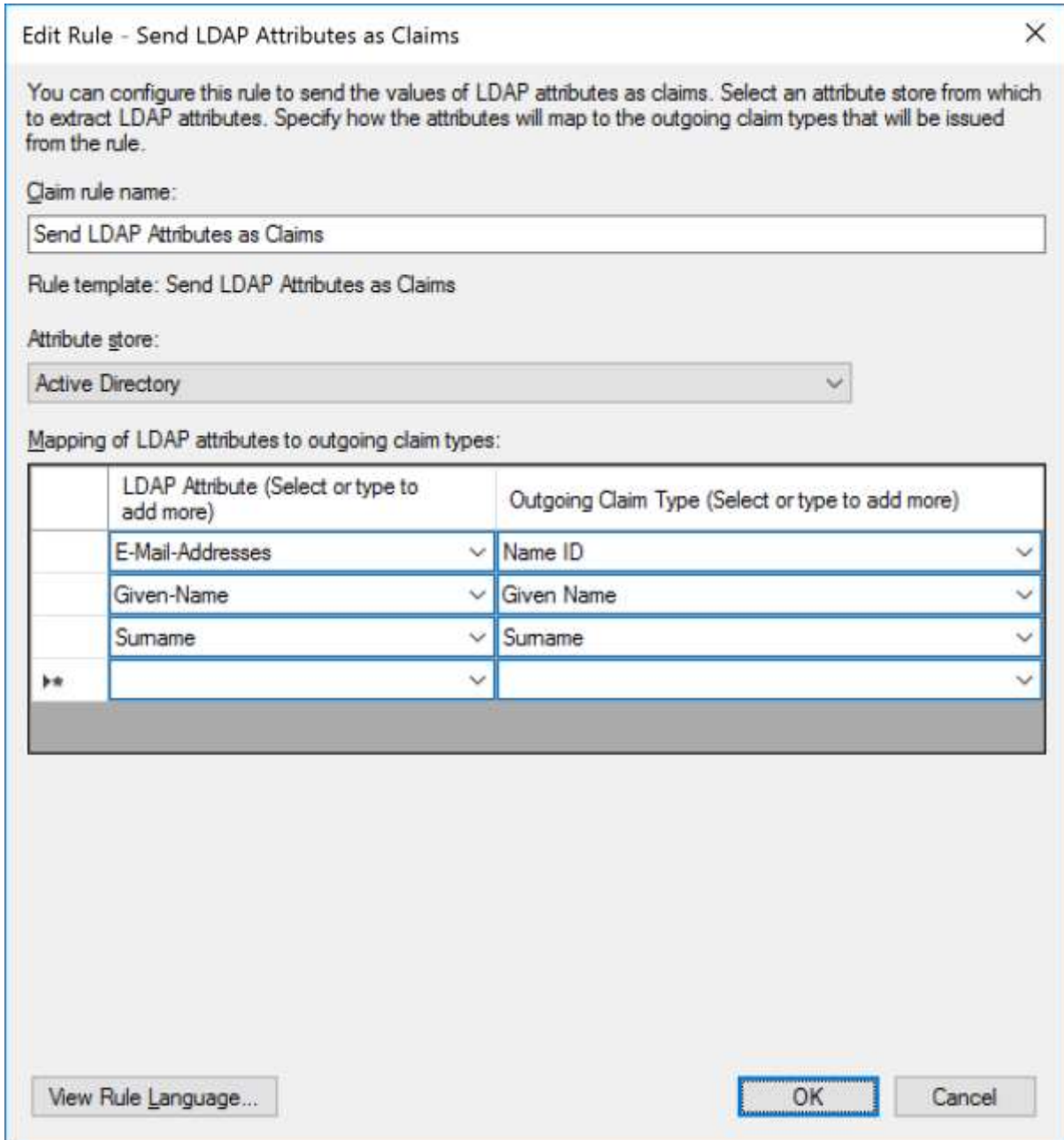
In this case the user principal name (UPN) is mapped to the SAML name identifier (Name ID).

The user's given name and surname are mapped to SAML attributes.

Note that to support SAML logout, a claims rule mapping for the SAML name identifier (Name ID) is required.



Alternatively, the user's email address may be mapped to the SAML name identifier (Name ID).



### Specifying the Name ID Format

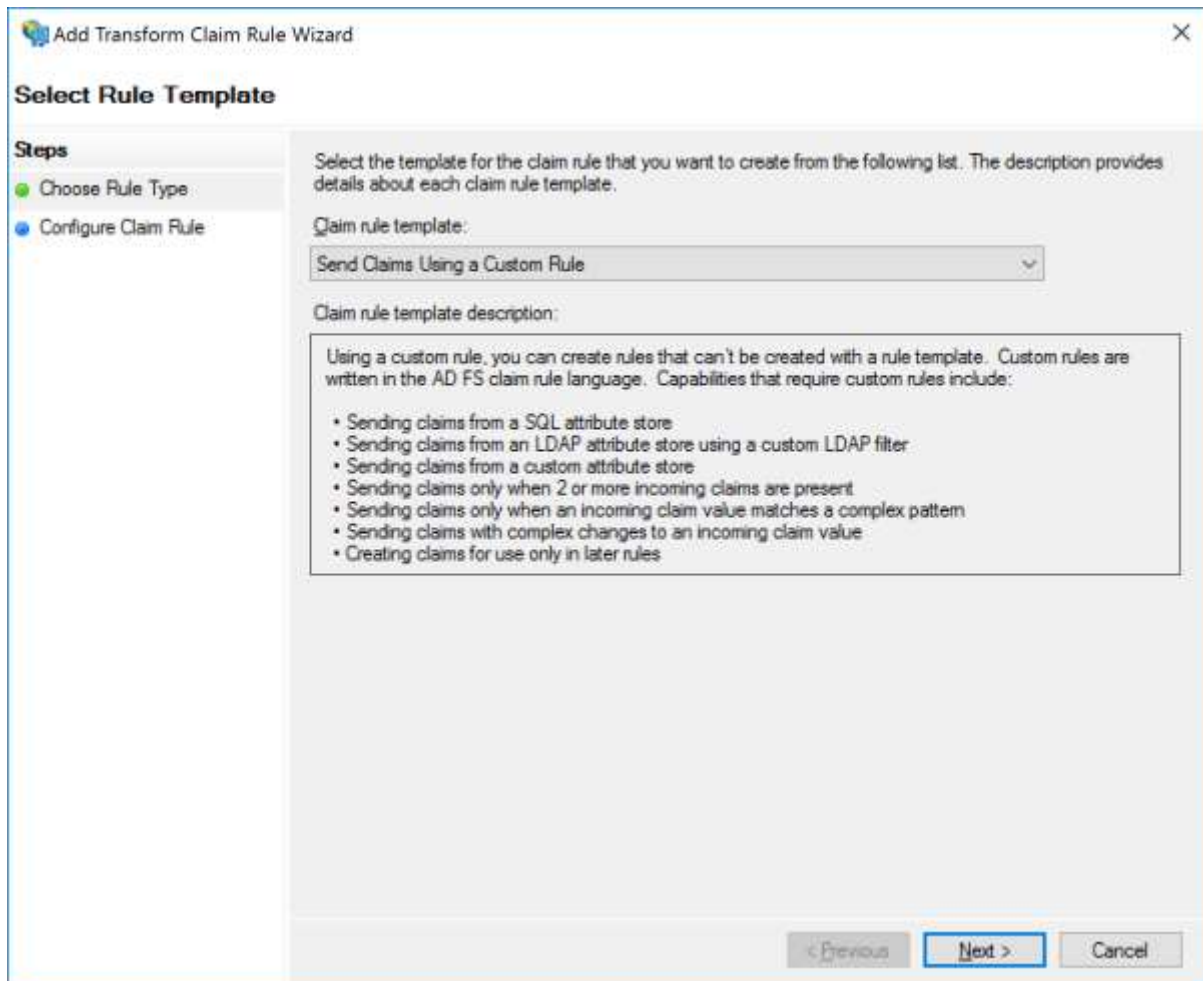
By default, no Name ID format is specified with the Name ID included in the SAML assertion.

A Name ID format may be specified if required by the service provider.

A Name ID format must be specified if the service provider specifies a Name ID policy in the SAML authn request, other than “urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified”.

For example, if the SAML authn request specifies a Name ID policy of “urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress”, the corresponding Name ID format must be returned in the SAML assertion.

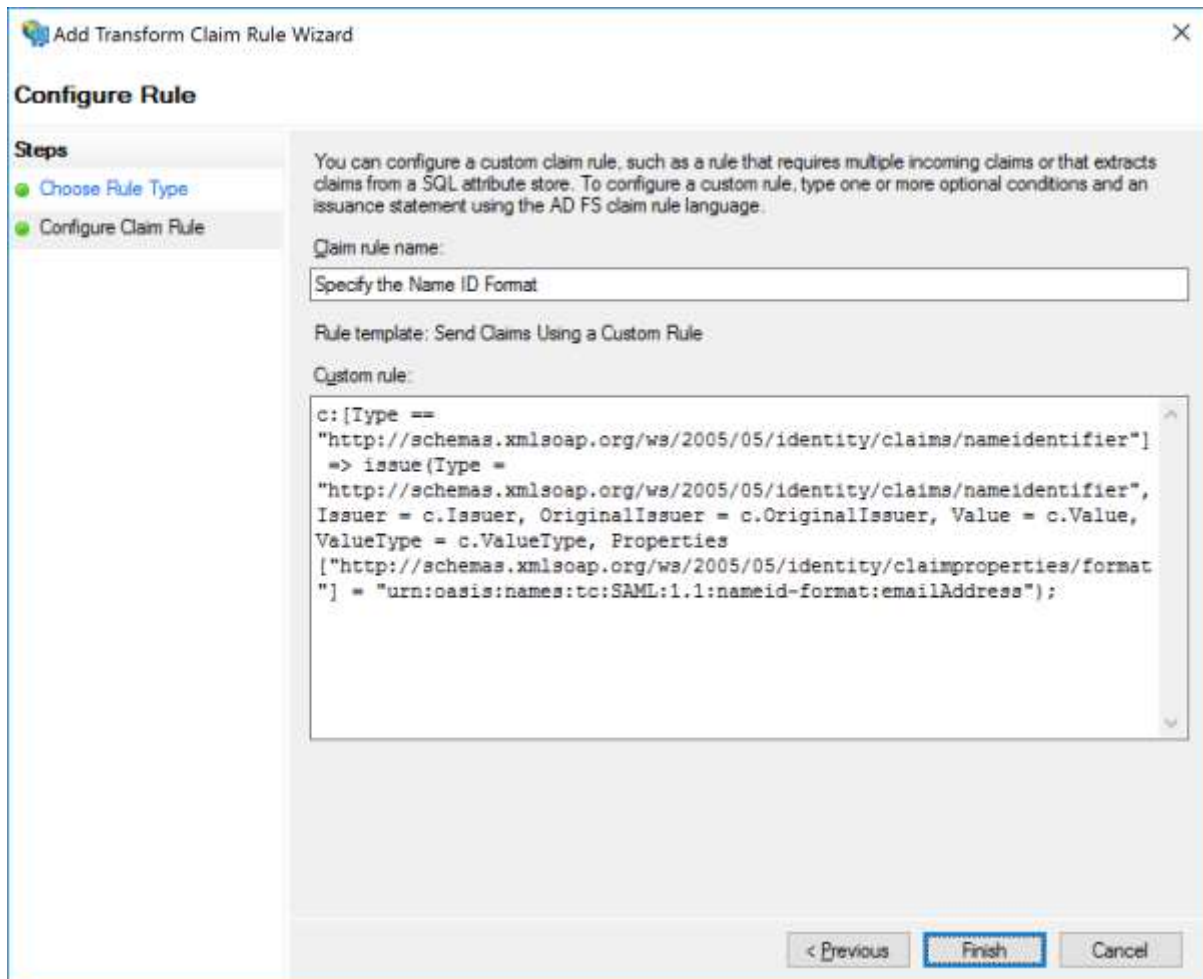
To include the Name ID format, add a custom rule.



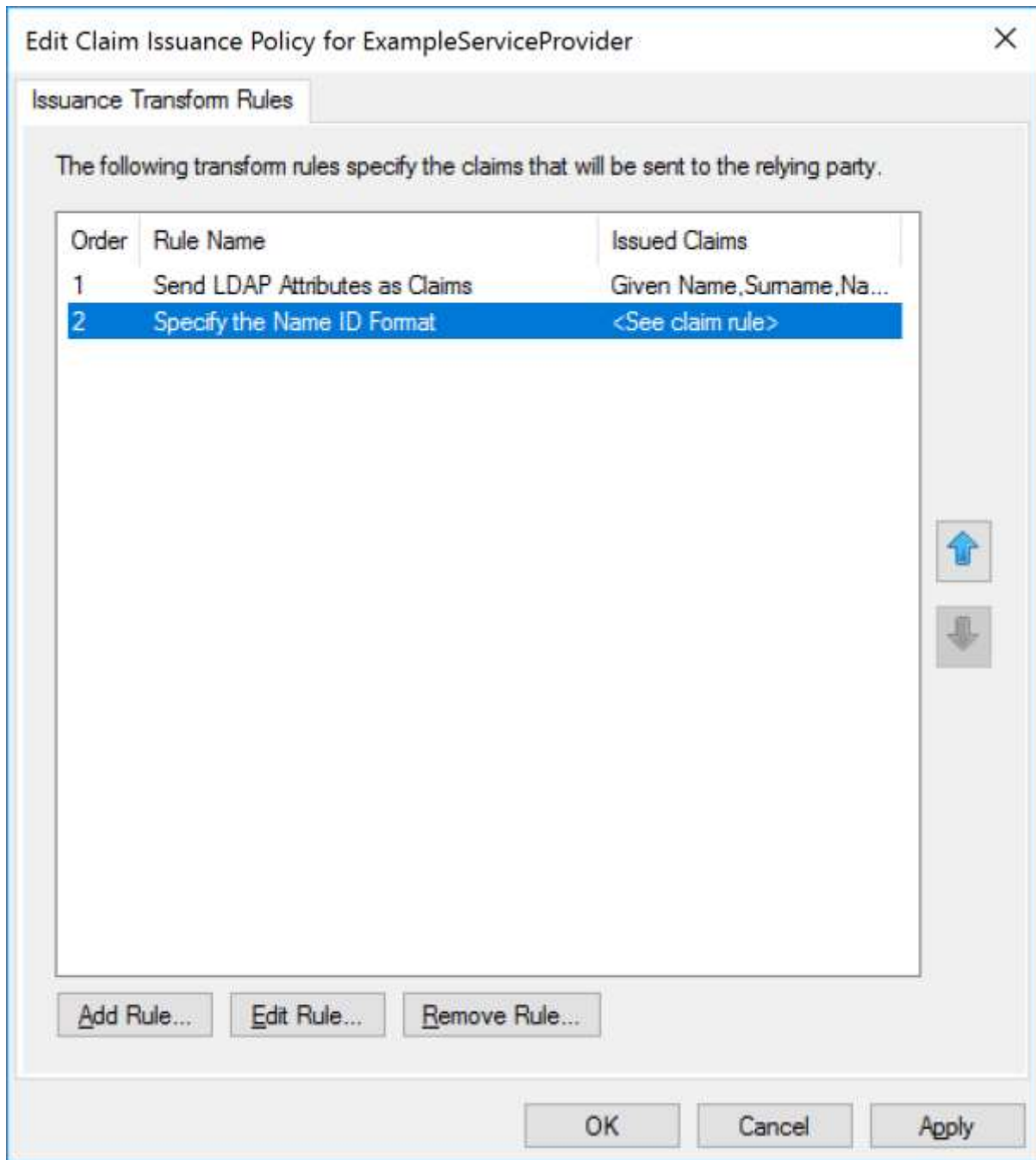
The custom rule transforms the <http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier> claim to include a <http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/format> claim property with the value urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress.

The rule is:

```
c:[Type == "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier"]
=> issue(Type = "http://schemas.xmlsoap.org/ws/2005/05/identity/claims/nameidentifier", Issuer =
c.Issuer, OriginalIssuer = c.OriginalIssuer, Value = c.Value, ValueType = c.ValueType,
Properties["http://schemas.xmlsoap.org/ws/2005/05/identity/claimproperties/format"] =
"urn:oasis:names:tc:SAML:1.1:nameid-format:emailAddress");
```



The rule order is important. The custom rule must be applied after the mapping of the LDAP attributes to outgoing claims.

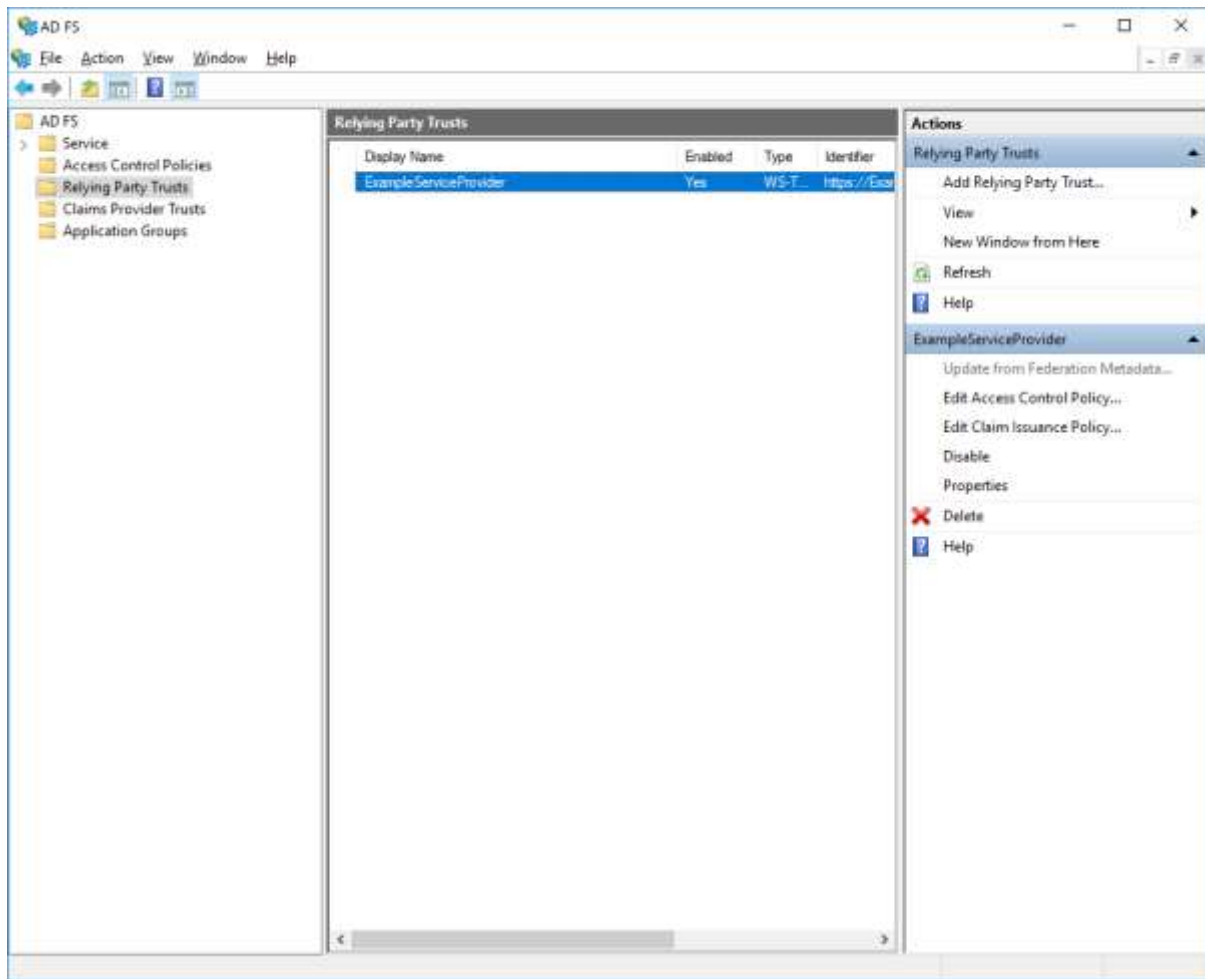


## Reviewing Relying Party Configuration

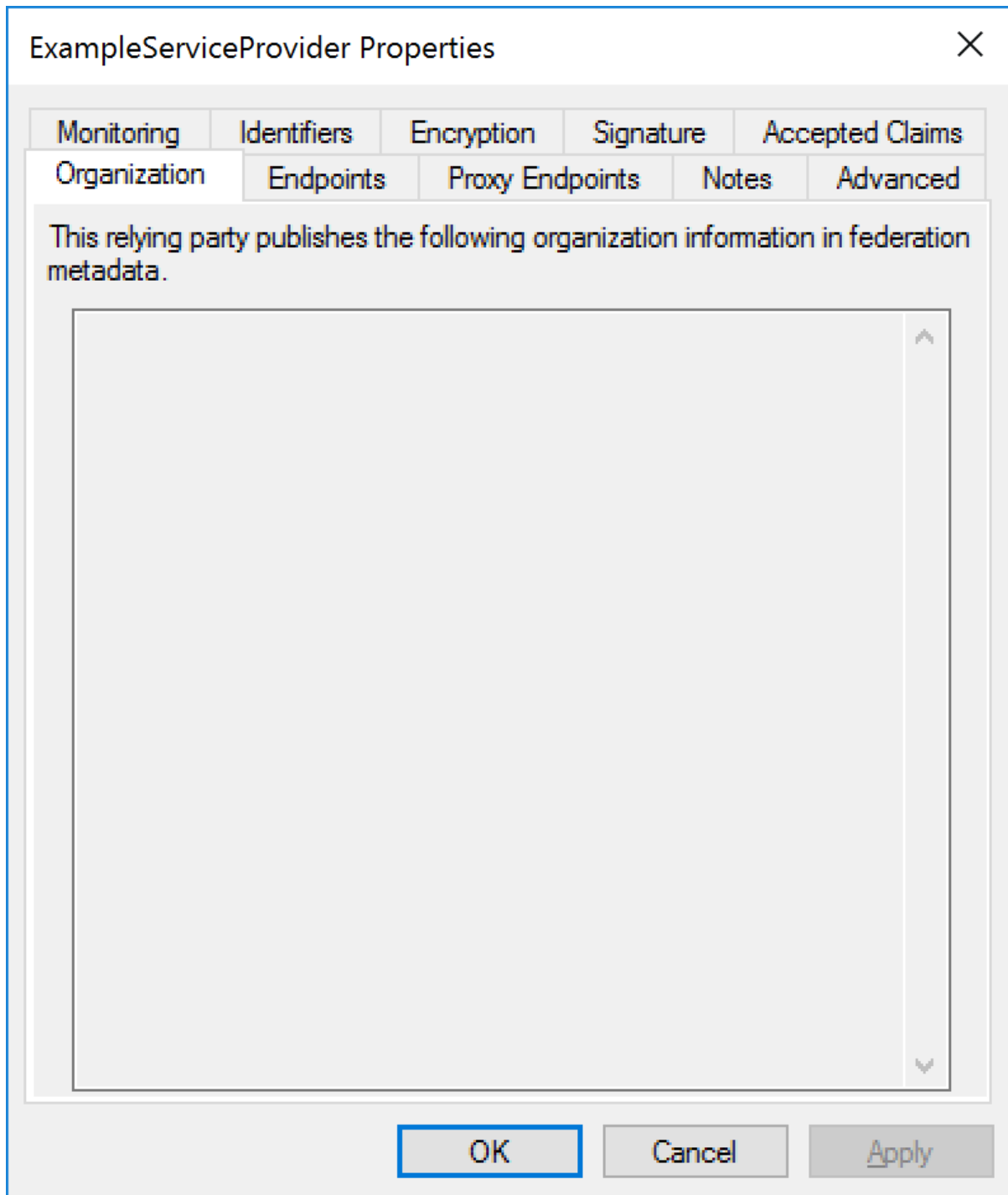
The configuration may be reviewed or modified through the relying party's property tabs.



## ComponentSpace SAML for ASP.NET ADFS Relying Party Integration Guide



The organization information from the imported SAML metadata, if any, is displayed.



The endpoints are the URLs and SAML bindings used when communicating with the service provider.

The SAML assertion consumer service receives SAML responses as part of SSO.

The SAML logout service receives logout messages as part of SAML logout.

Note that ADFS treats URLs as being case sensitive.

ExampleServiceProvider Properties ✕

Monitoring Identifiers Encryption Signature Accepted Claims  
 Organization Endpoints Proxy Endpoints Notes Advanced

Specify the endpoints to use for SAML and WS-FederationPassive protocols.

URL	Index	Binding	Default	Re
<b>SAML Assertion Consumer Endpoints</b>				
https://localhost:44360/SAM...	0	POST	Yes	
<b>SAML Logout Endpoints</b>				
https://localhost:44360/SAM...		Redirect	No	

<  >

Add SAML...

Add WS-Federation... Remove Edit...

OK Cancel Apply

Proxied endpoints aren't used in this example.

The screenshot shows a dialog box titled "ExampleServiceProvider Properties" with a close button (X) in the top right corner. The dialog has several tabs: "Monitoring", "Identifiers", "Encryption", "Signature", "Accepted Claims", "Organization", "Endpoints", "Proxy Endpoints" (which is selected), "Notes", and "Advanced".

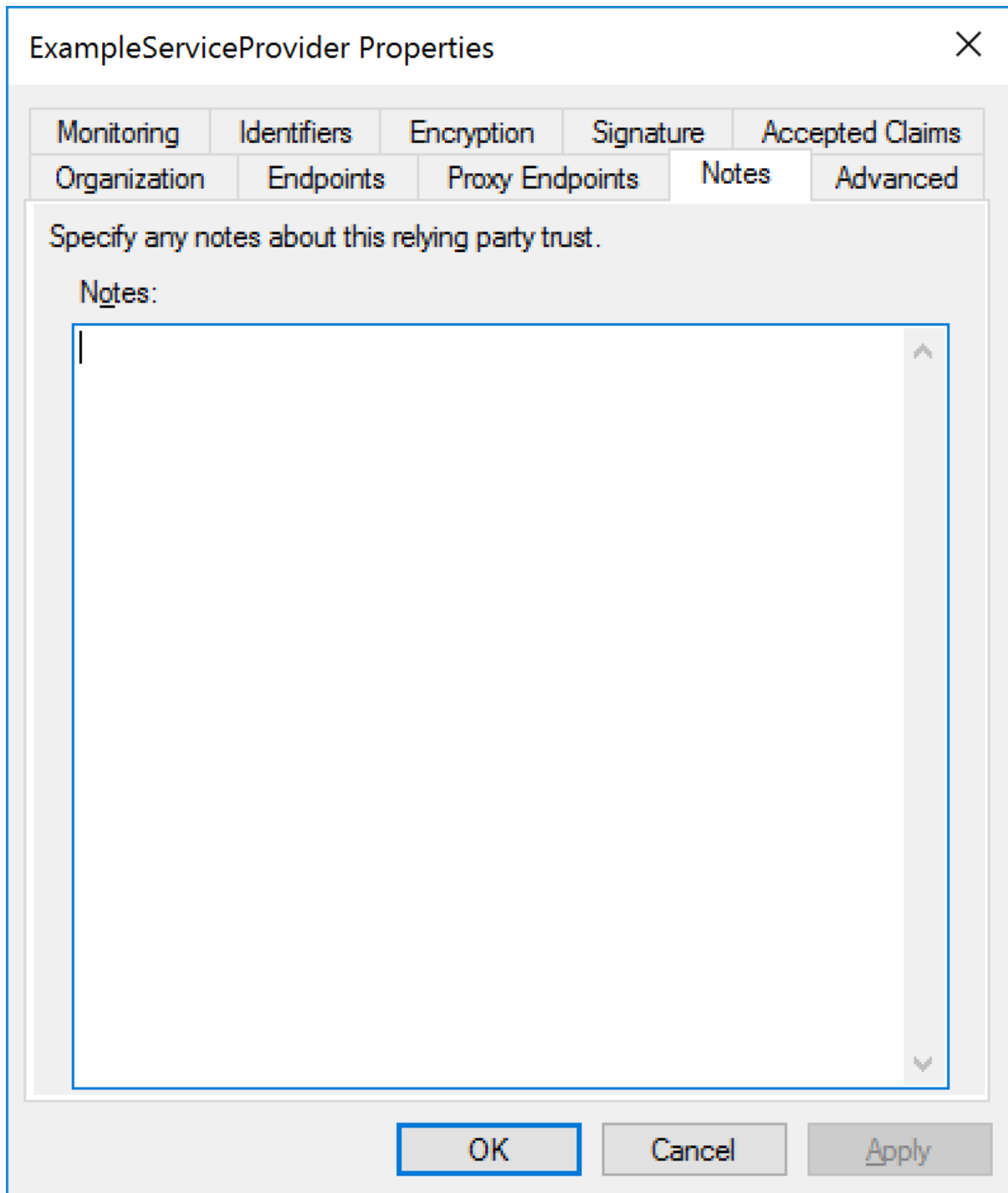
Below the tabs, the text reads: "The following shows the proxied endpoints for this relying party trust."

There is a table with one column labeled "URL". The table is currently empty.

At the bottom of the dialog, there is a checkbox with the text: "This relying party trust is published through a web application proxy for extranet access." The checkbox is currently unchecked.

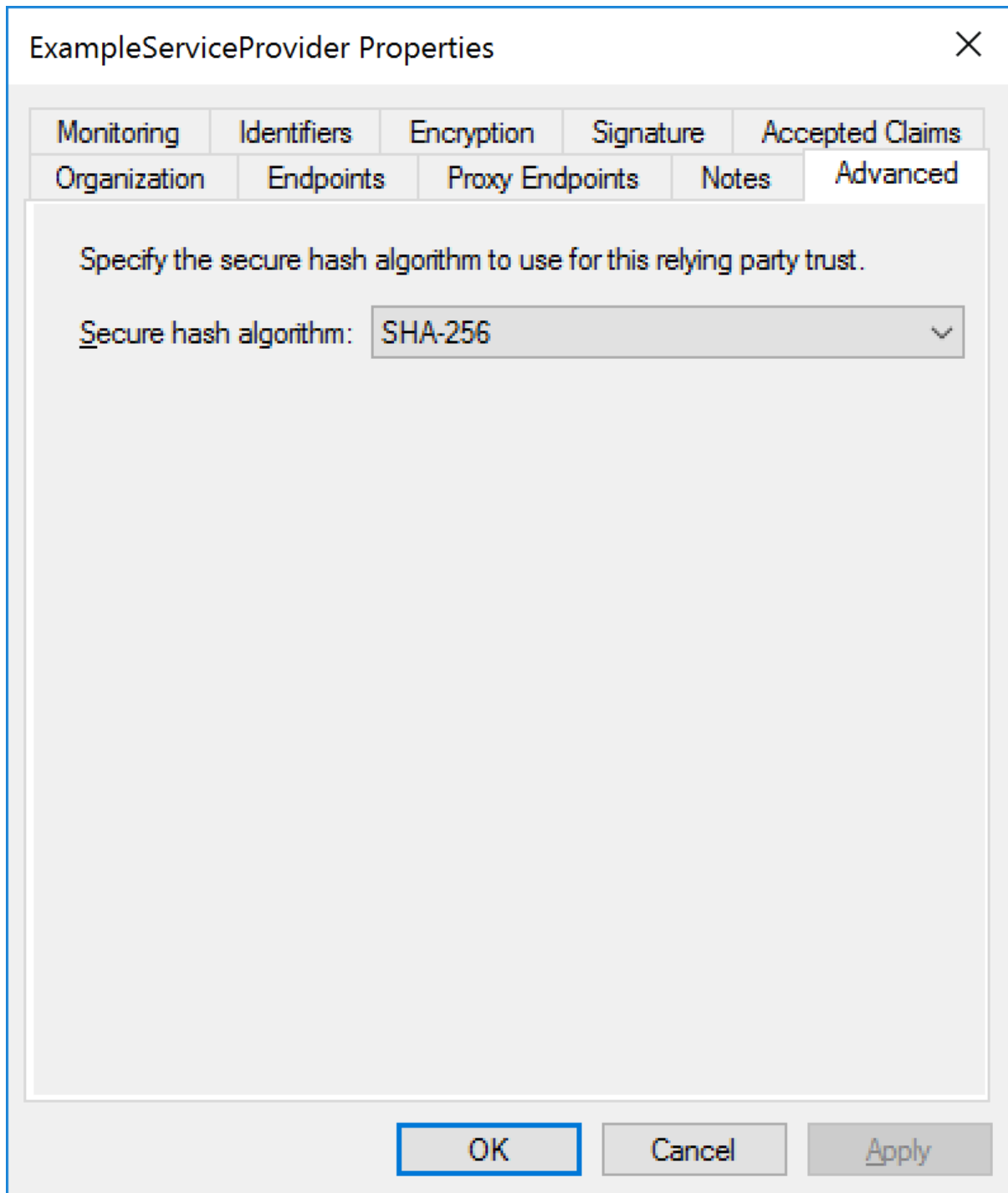
At the very bottom of the dialog are three buttons: "OK", "Cancel", and "Apply".

Notes are internal to ADFS and for documentation purposes only.



Either SHA-1 or SHA-256 may be specified as the signature algorithm.

SHA-256 is recommended.



ADFS supports monitoring a URL for SAML metadata updates.

The screenshot shows a dialog box titled "ExampleServiceProvider Properties" with a close button (X) in the top right corner. The dialog has a tabbed interface with the following tabs: Organization, Endpoints, Proxy Endpoints, Notes, Advanced, Monitoring (selected), Identifiers, Encryption, Signature, and Accepted Claims. The "Monitoring" tab is active, displaying the following content:

Specify the monitoring settings for this relying party trust.

Relying party's federation metadata URL:  
[Text input field] [Test URL]

Monitor relying party

Automatically update relying party

This relying party's federation metadata data was last checked on:  
< never >

This relying party was last updated from federation metadata on:  
< never >

At the bottom of the dialog are three buttons: OK, Cancel, and Apply.

Relying party identifiers correspond to SAML metadata entity IDs.

The relying party identifier must match exactly with the service provider's configured name.

The screenshot shows a dialog box titled 'ExampleServiceProvider Properties' with a close button (X) in the top right corner. The dialog has a tabbed interface with the following tabs: Organization, Endpoints, Proxy Endpoints, Notes, Advanced, Monitoring, Identifiers (selected), Encryption, Signature, and Accepted Claims. The main content area is titled 'Specify the display name and identifiers for this relying party trust.' It contains the following fields and controls:

- Display name:** A text box containing 'ExampleServiceProvider'.
- Relying party identifier:** An empty text box with an 'Add' button to its right. Below it is an example: 'Example: https://fs.contoso.com/adfs/services/trust'.
- Relying party identifiers:** A list box containing 'https://ExampleServiceProvider' with a 'Remove' button to its right.

At the bottom of the dialog are three buttons: 'OK' (highlighted with a blue border), 'Cancel', and 'Apply'.

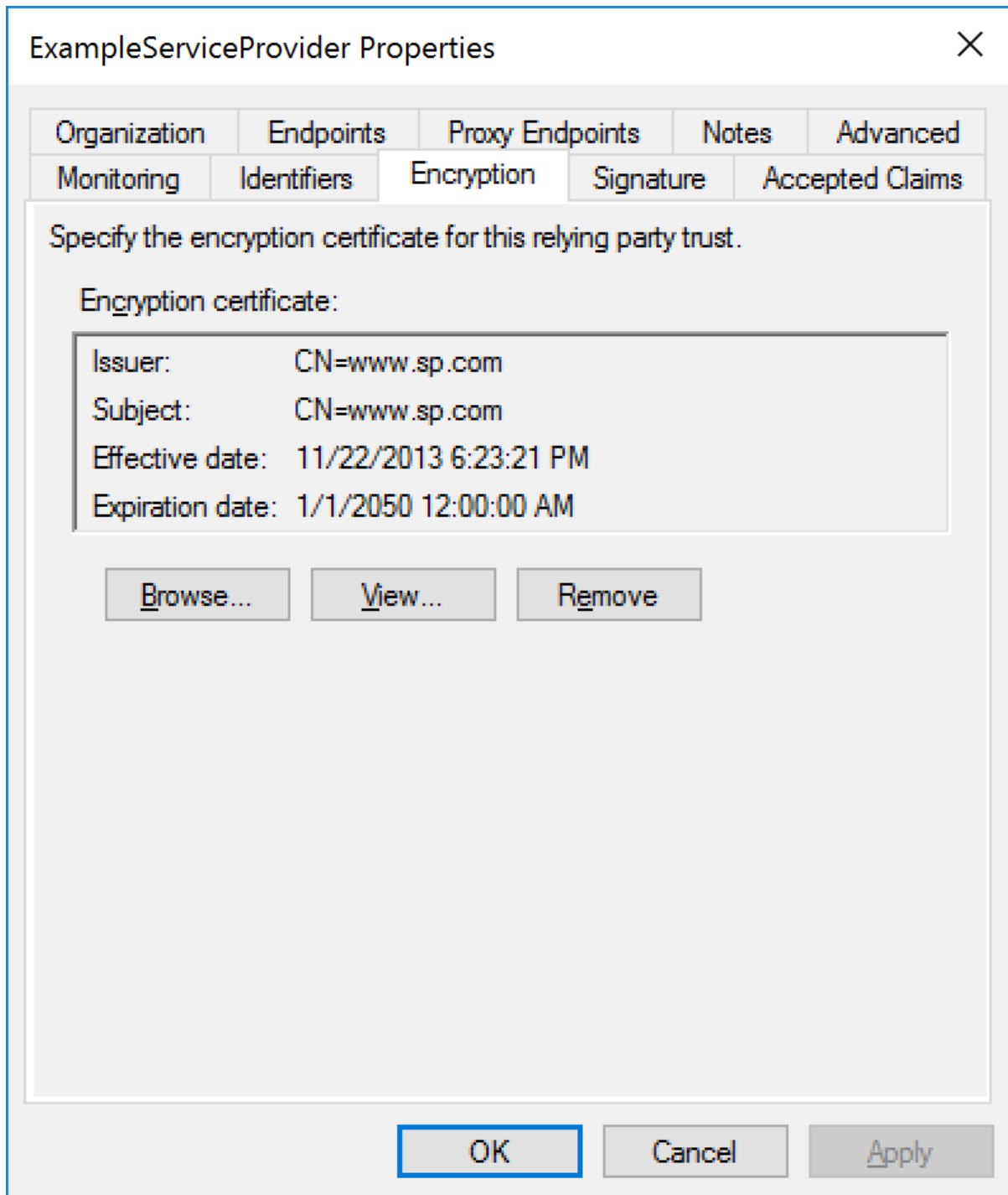
The encryption certificate is specified if the SAML assertion is to be encrypted.

If specified, it's the service provider's encryption certificate.

In many scenarios encrypting the SAML assertion isn't required as the privacy provided at the transport layer by HTTPS is sufficient.

The certificate should be removed if the SAML assertion is not to be encrypted.

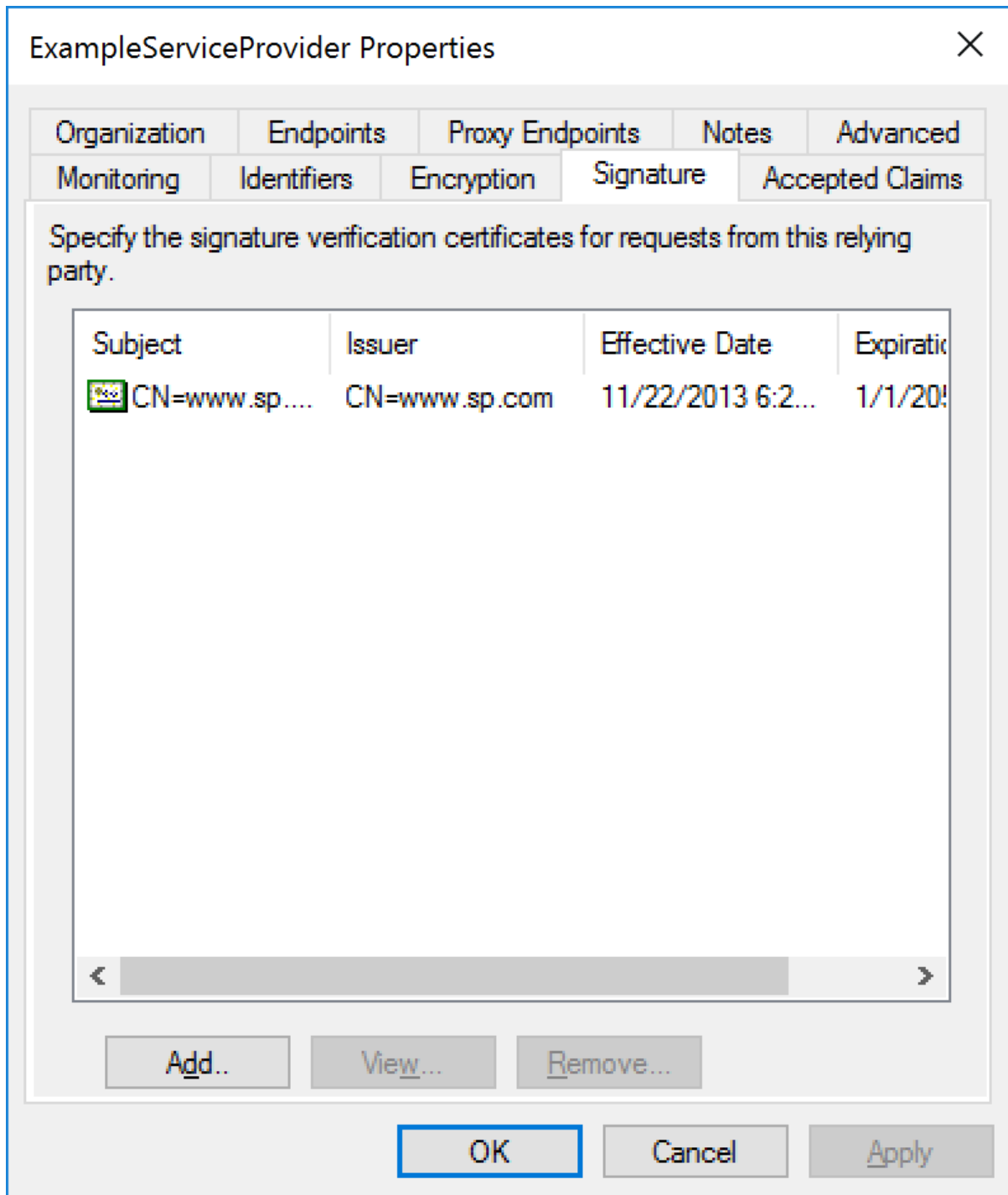




The signature certificate is specified if the signatures on SAML messages from the service provider are to be verified.

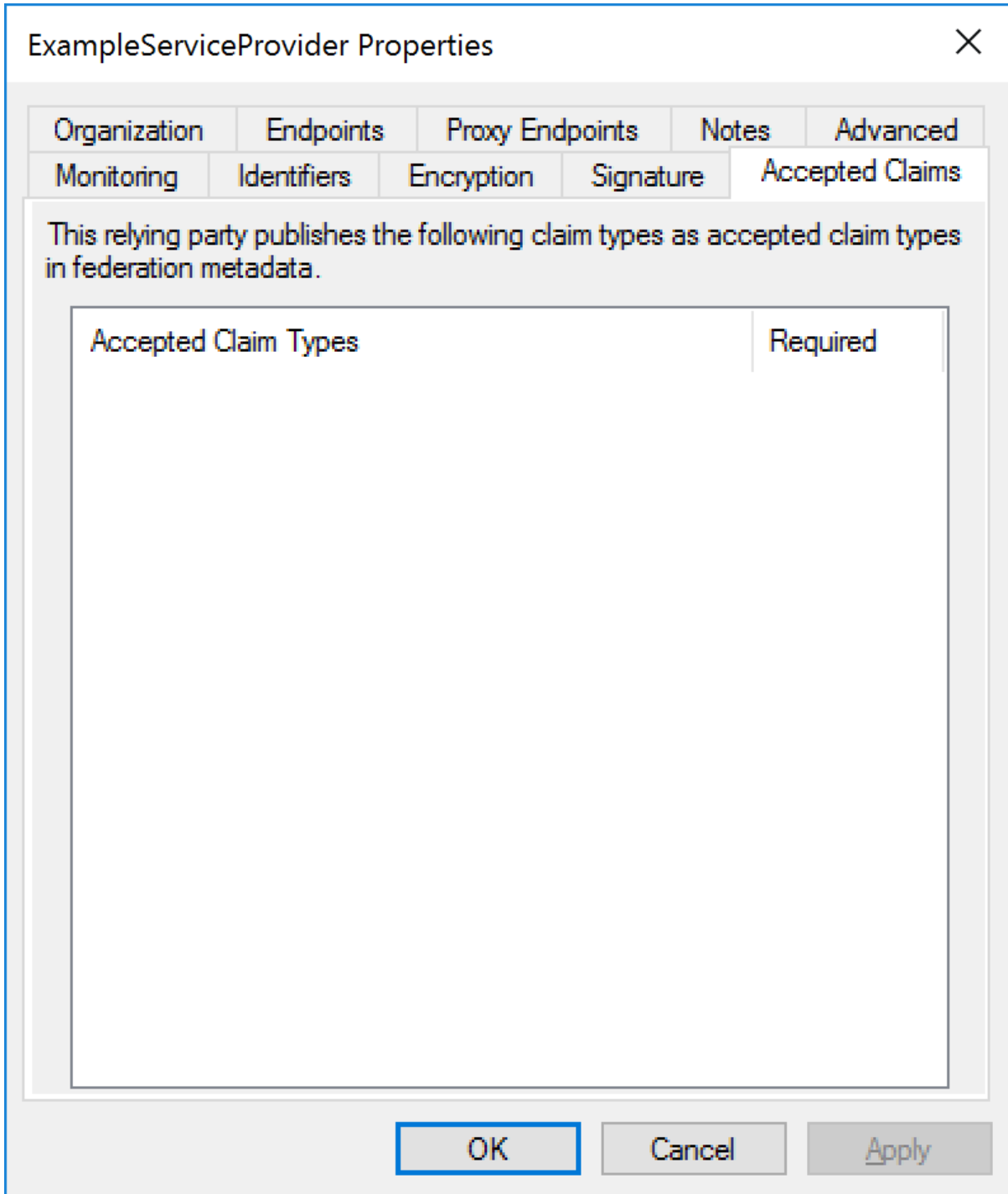
If specified, it's the service provider's signature certificate.

It's recommended that SAML messages from the service provider are signed.



The accepted claims are specified through the service provider's SAML metadata.

These are for documentation purposes and don't affect the claims sent by ADFS.



## ADFS SAML Metadata

Metadata may be downloaded from:

<https://<server-name>/FederationMetadata/2007-06/FederationMetadata.xml>

For example:

<https://ads.componenspace.com/FederationMetadata/2007-06/FederationMetadata.xml>

## Service Provider Configuration

The following partner identity provider configuration is included in the example service provider's SAML configuration.

```
<PartnerIdentityProvider
  Name="http://adfs.componentspace.com/adfs/services/trust"
  Description="ADFS"
  SignLogoutRequest="true"
  SignLogoutResponse="true"
  WantLogoutRequestSigned="true"
  WantLogoutResponseSigned="true"
  SingleSignOnServiceUrl="https://adfs.componentspace.com/adfs/ls/"
  SingleLogoutServiceUrl="https://adfs.componentspace.com/adfs/ls/"
  <PartnerCertificates>
    <Certificate FileName="Certificates\adfs.cer"/>
  </PartnerCertificates>
</PartnerIdentityProvider>
```

Some of this information was extracted from the ADFS SAML metadata.

The partner certificate file corresponds to the signing certificate included in the metadata.

ADFS doesn't require the SAML authn request to be signed although it is recommended.

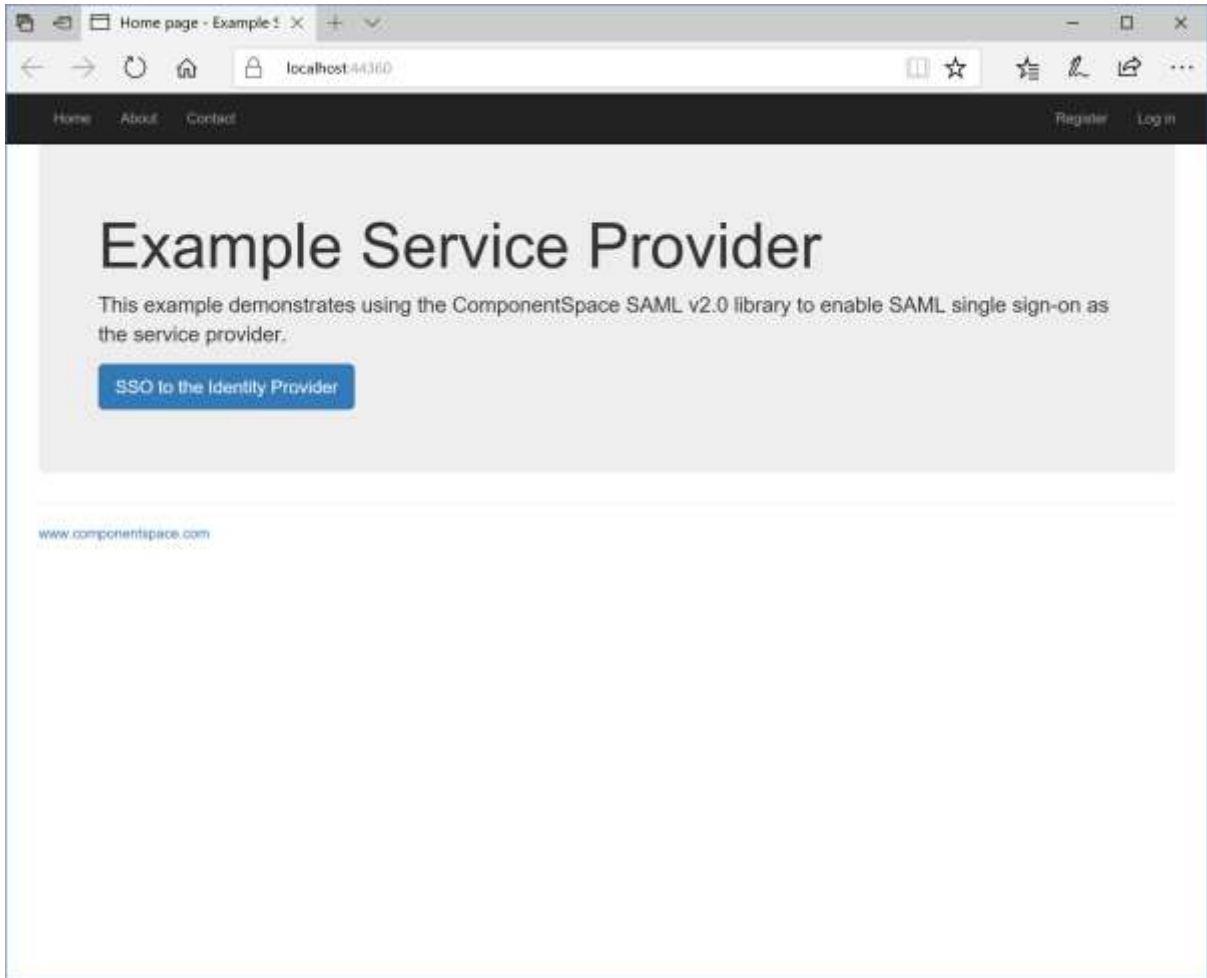
ADFS requires SAML logout messages to signed.

Ensure the PartnerName specifies the correct partner identity provider.

```
<add key="PartnerName" value="http://adfs.componentspace.com/adfs/services/trust"/>
```

## SP-Initiated SSO

Browse to the example service provider.

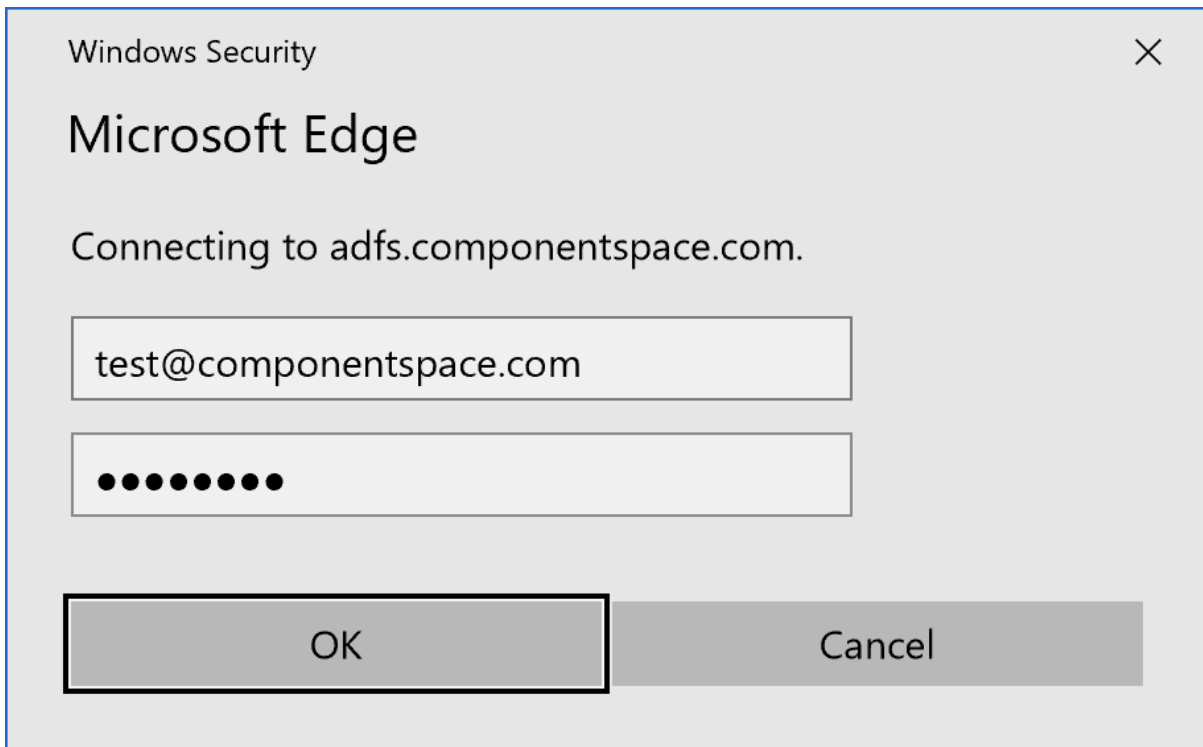


Click the button to SSO to the identity provider.

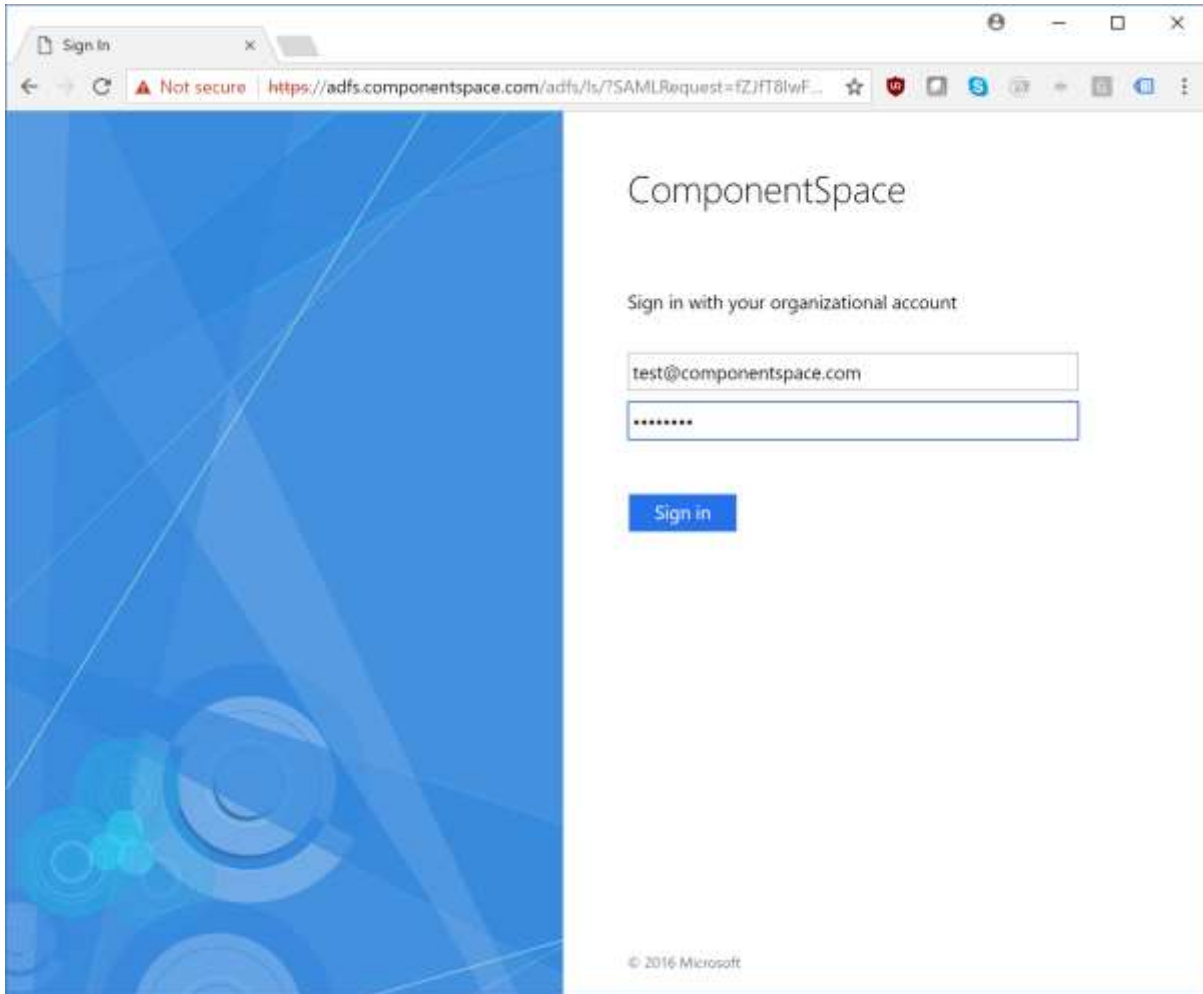
Log into ADFS.

The login method (e.g. forms authentication, Windows authentication) will be dependent on the authentication methods configured in ADFS and the browser type.

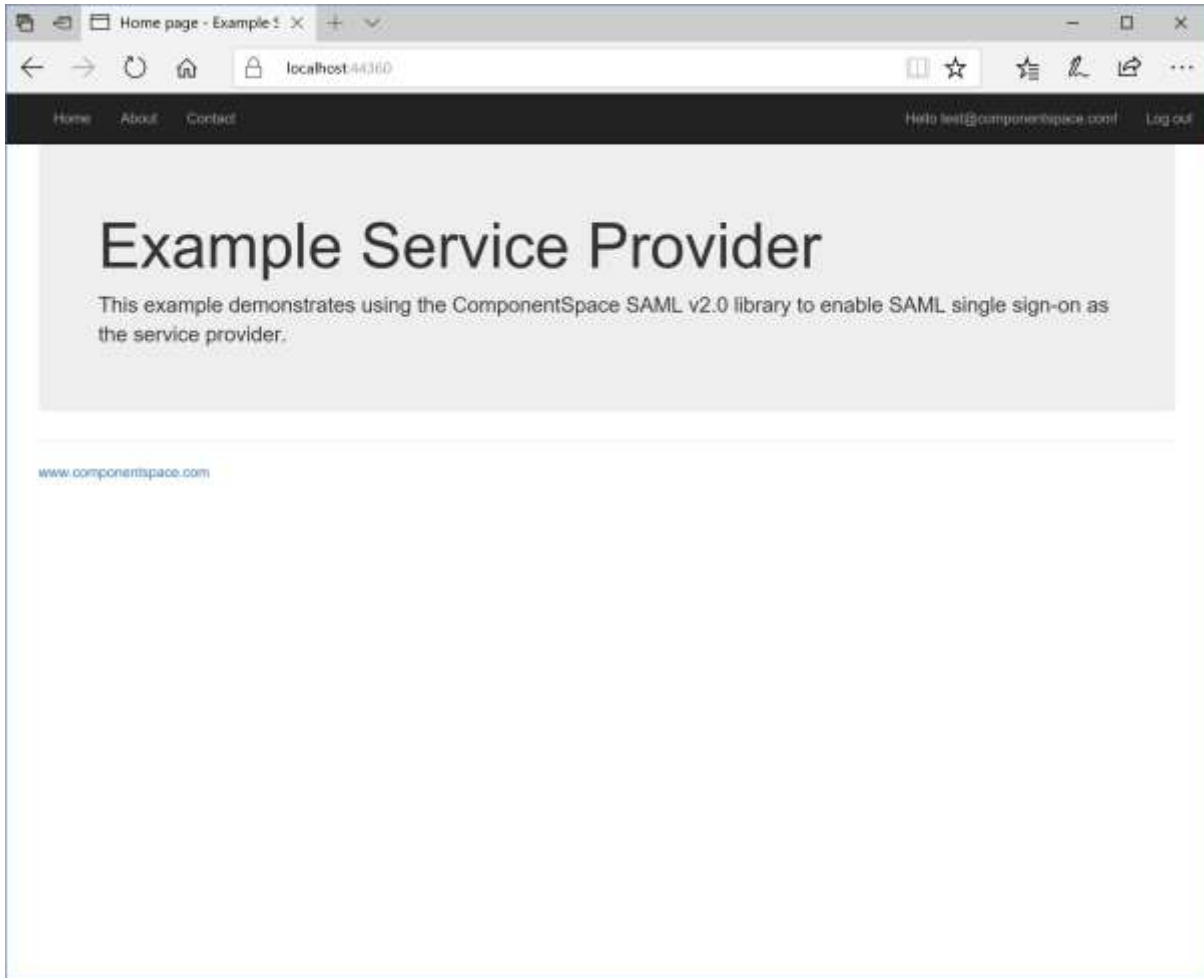
The following is the authentication prompt displayed by Microsoft Edge when Windows integrated authentication is enabled but the user is not logged into the domain.



The following is the forms authentication prompt displayed by Google Chrome.



The user is automatically logged in at the service provider.



## IdP-Initiated SSO

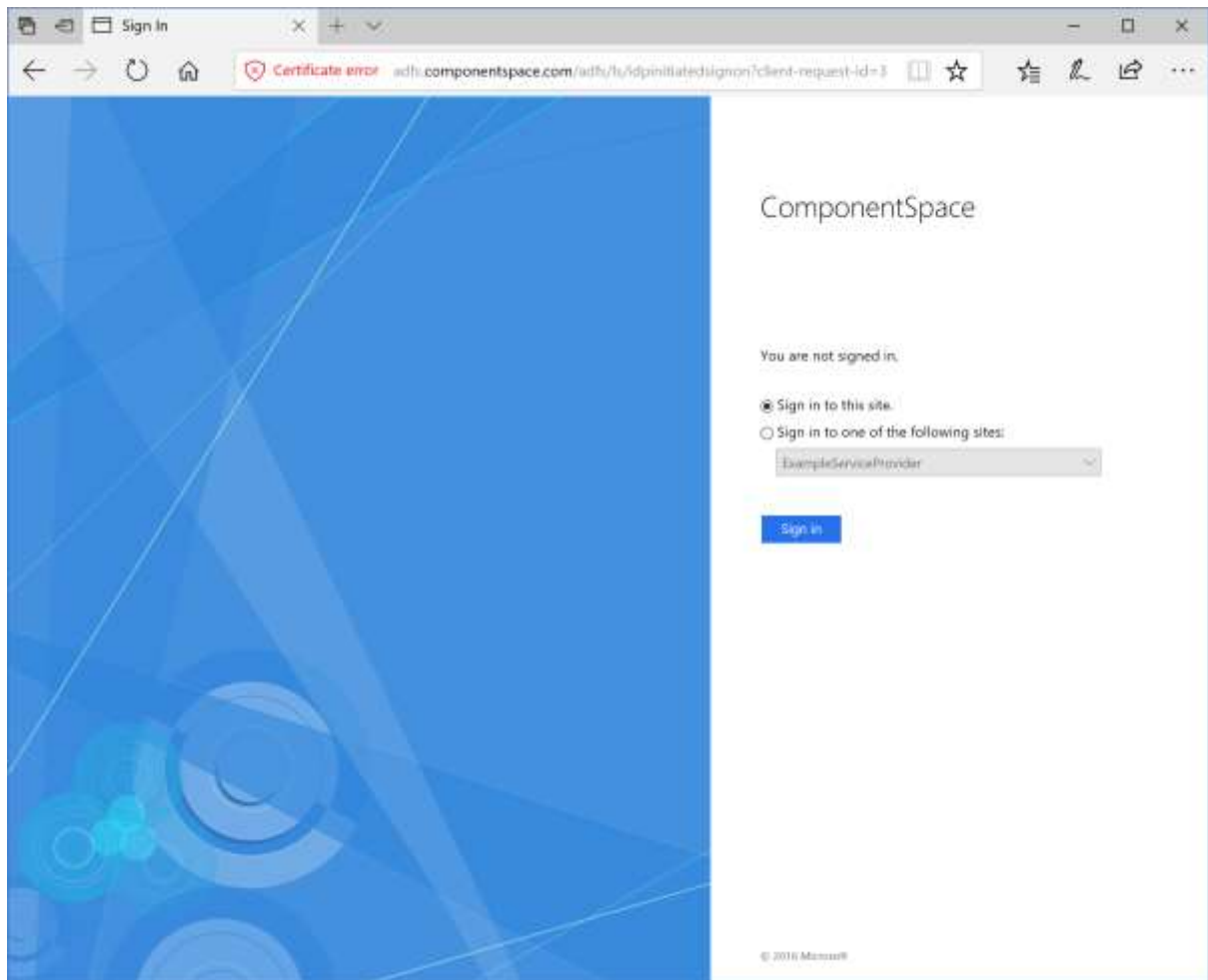
Browse to:

<https://<server-name>/adfs/ls/IdpInitiatedSignon>

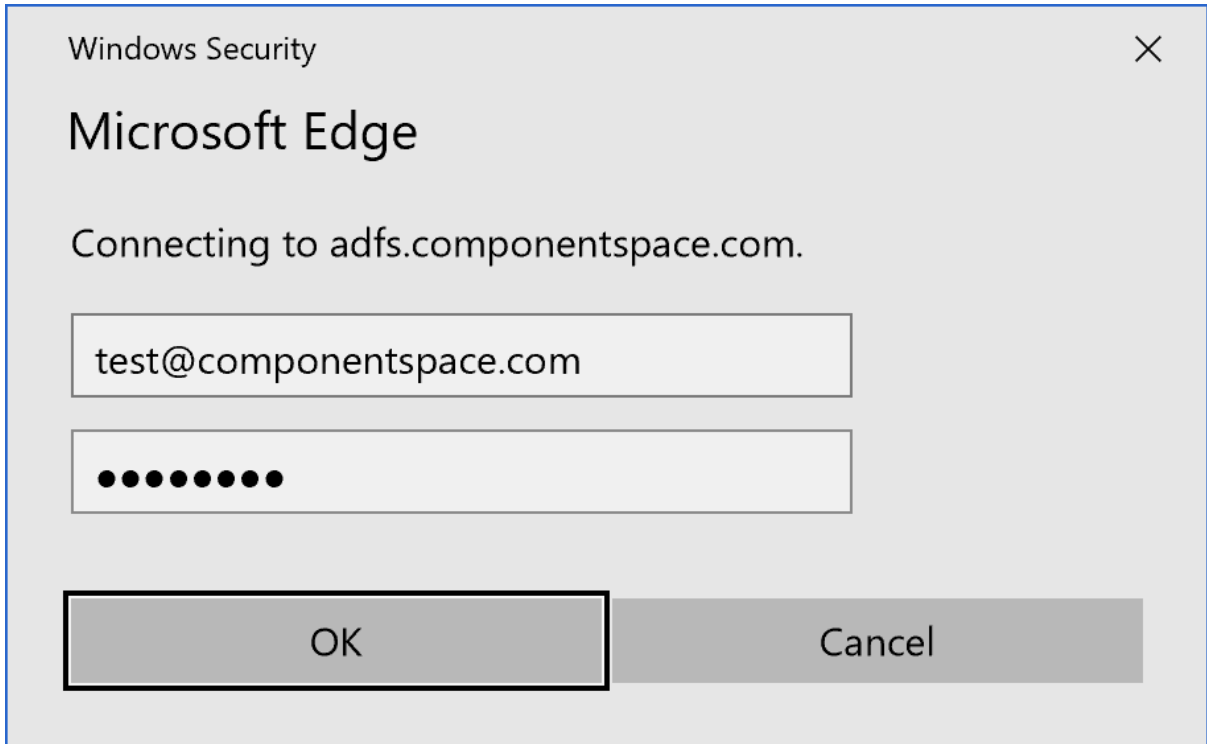
For example:

<https://adfs.componentspace.com/adfs/ls/IdpInitiatedSignon>

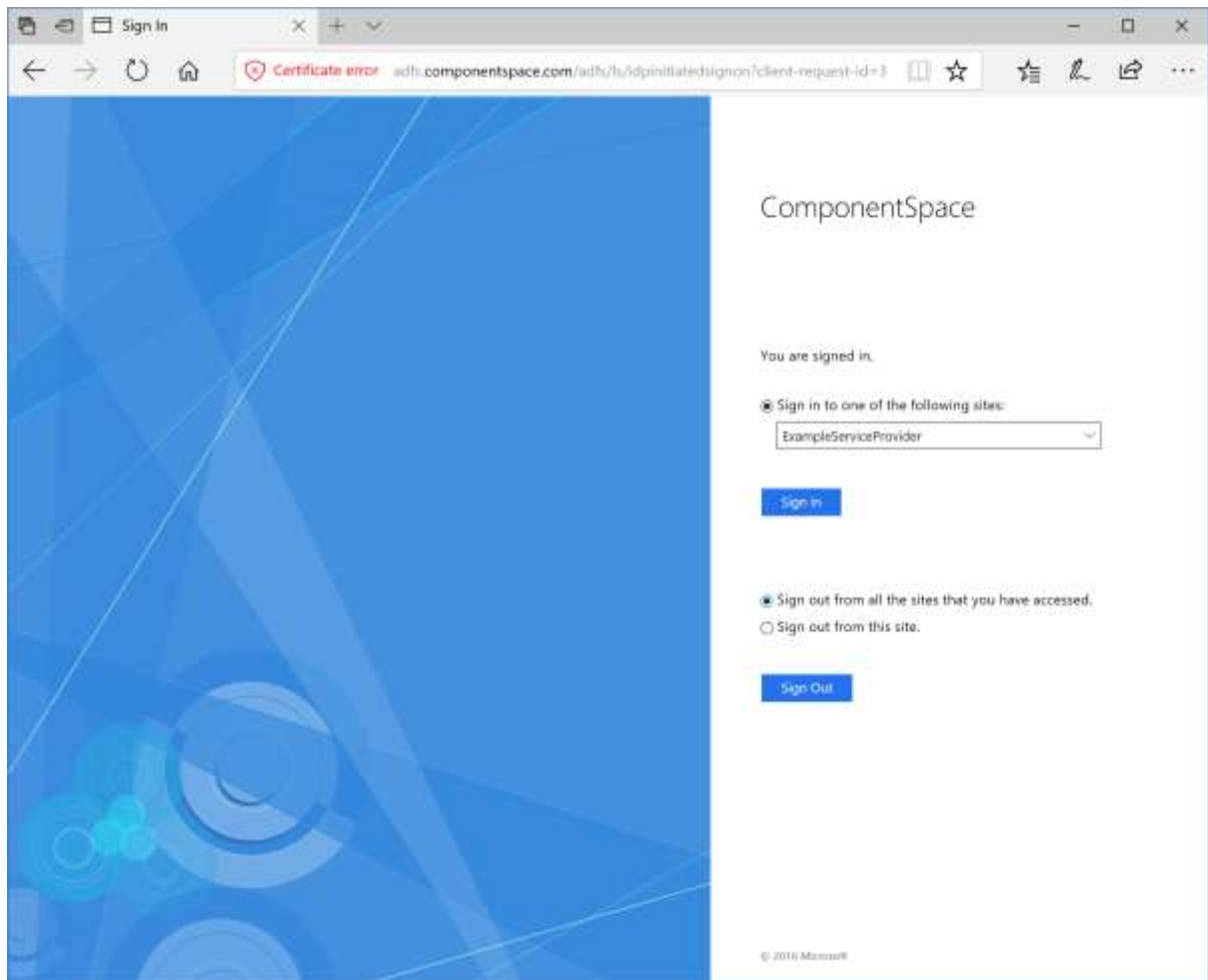




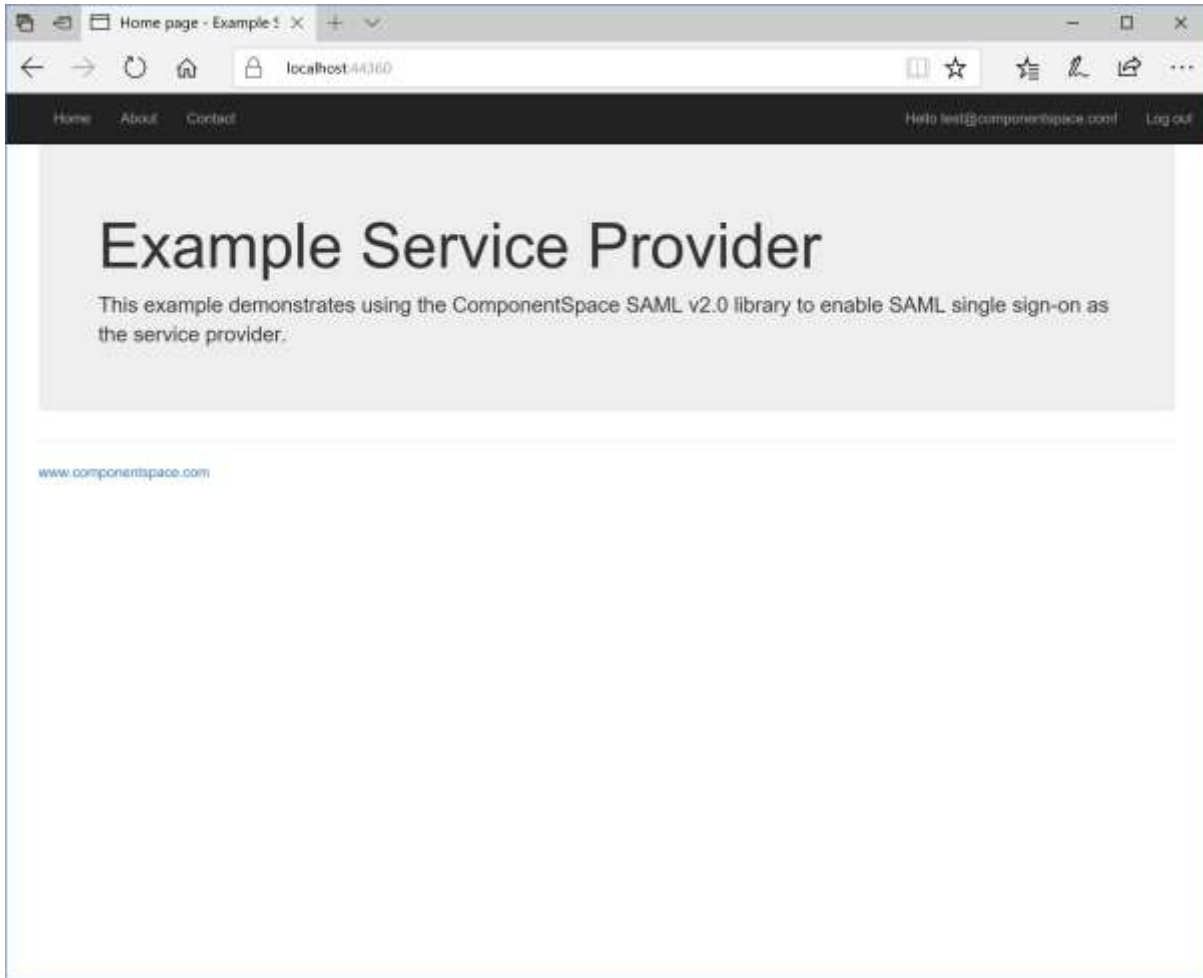
Log into ADFS.



Select the service provider and sign in.



The user is automatically logged in at the service provider.



## SAML Logout

Both SP-initiated and IdP-initiated SLO are supported.

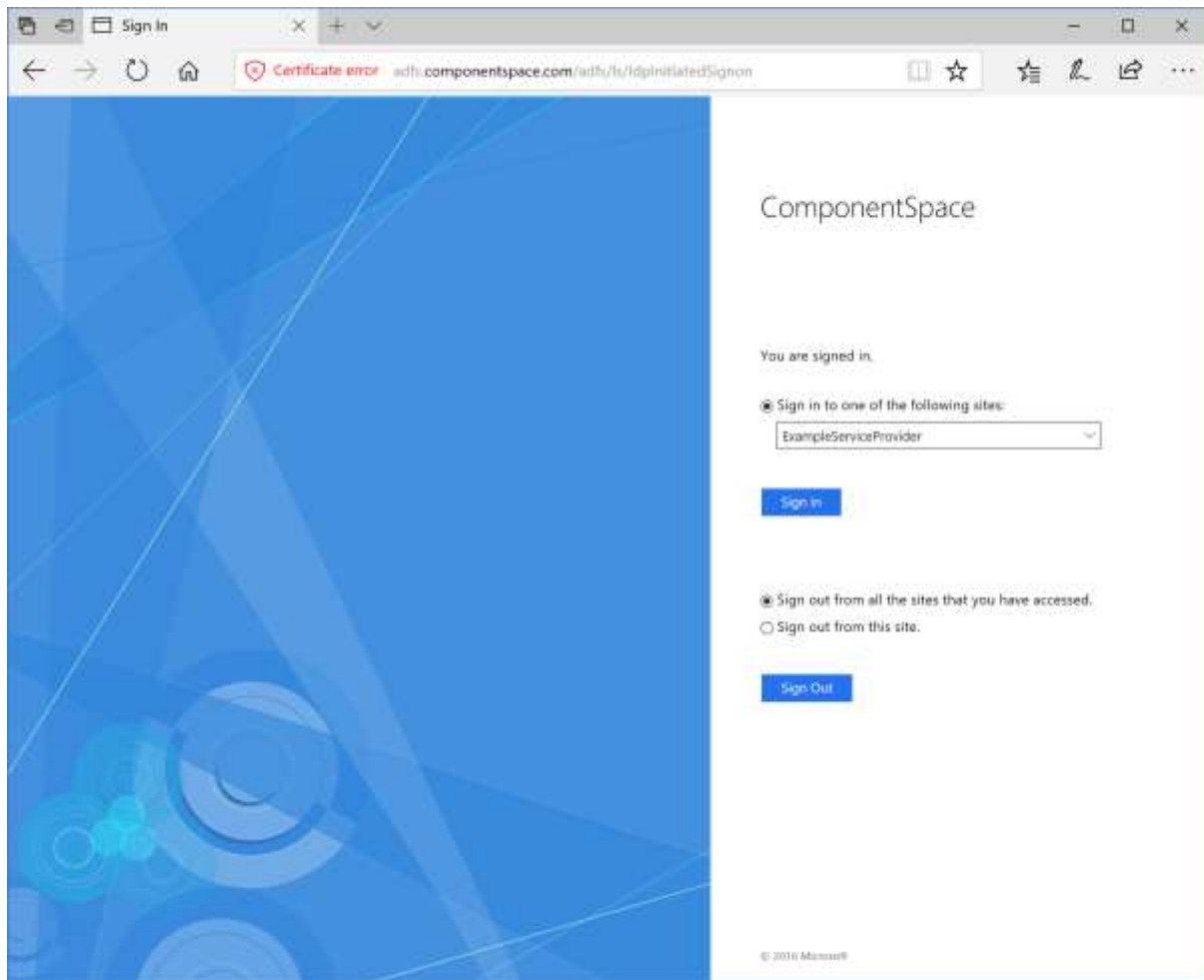
IdP-initiated SLO may be invoked from:

<https://<server-name>/adfs/ls/IdpInitiatedSignon>

For example:

<https://adfs.componentspace.com/adfs/ls/IdpInitiatedSignon>

Select to sign out from all sites.



Depending on the authentication method and the browser used, although ADFS reports logout as successful, the user may not be logged out from ADFS.

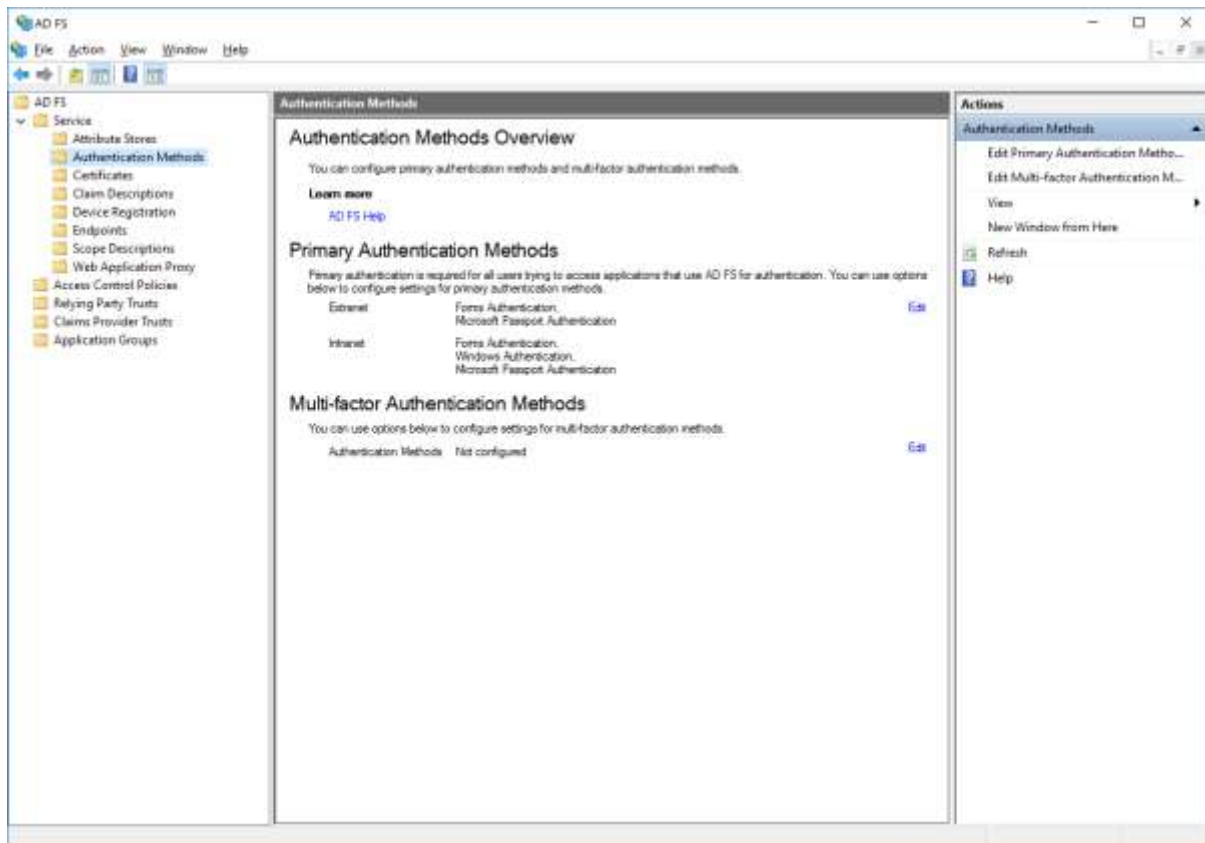
For example, with forms authentication and using Chrome, the user is logged out from ADFS.

When using Microsoft Edge, no error occurs but the user is still logged into ADFS.

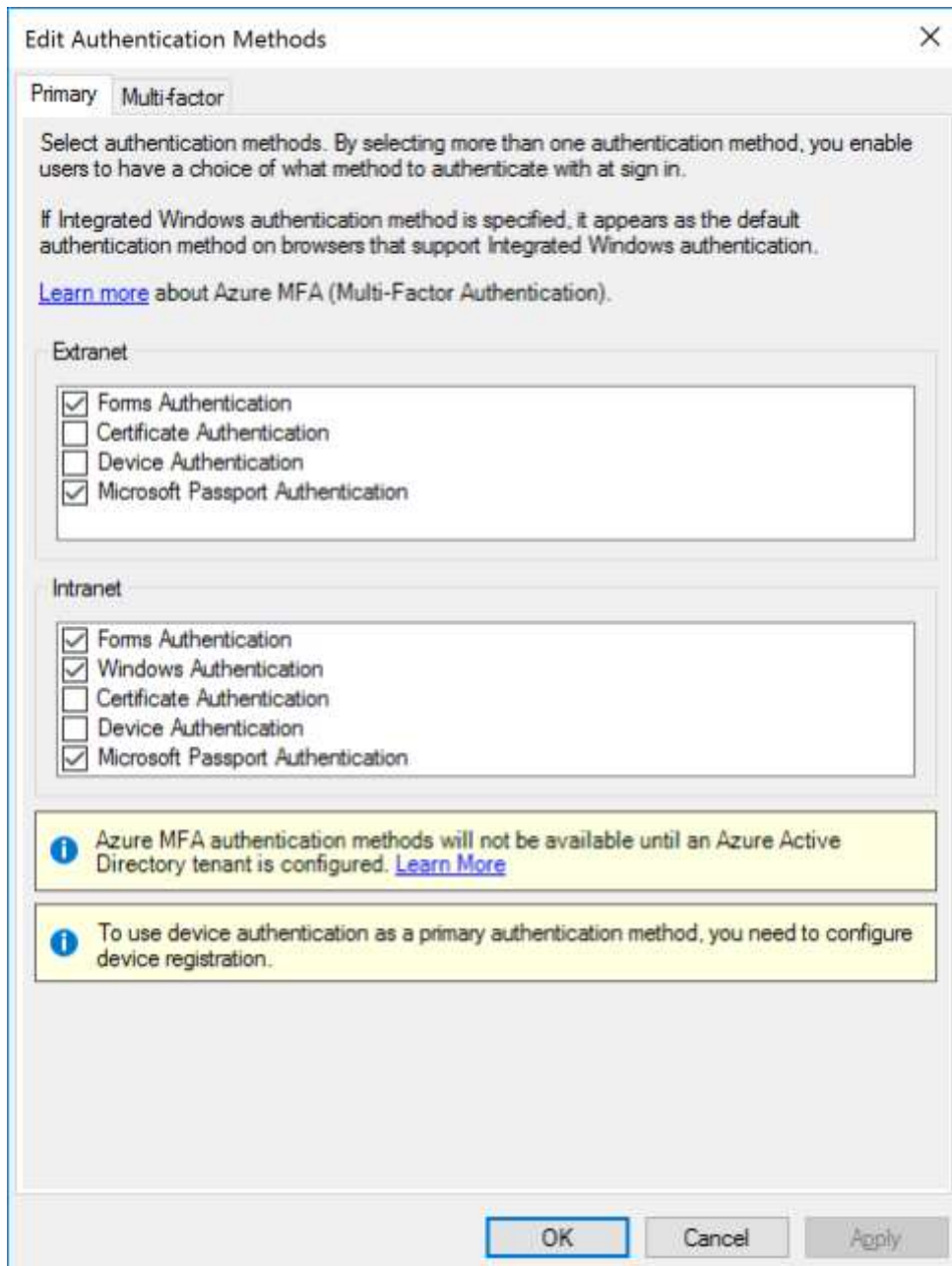
This functionality is controlled by ADFS.

## ADFS Authentication Methods

ADFS supports a number of authentication methods that may be configured based on whether the user is in the intranet or not.



The default configuration is to use Windows authentication for intranet users using a browser supporting Windows integrated authentication. Otherwise, forms authentication is used.



## Windows Integrated Authentication

For a user logged into the domain, Windows integrated authentication, means the user is not prompted to login again. The Windows user principal name is used instead.

If Windows integrated authentication is enabled but the user is not logged into the domain, ADFS returns a 401, unauthorized, error to the browser which will prompt the user for their credentials and send an authorization header along with the SAML authentication request to ADFS.

Note SAML logout will be successful but the user will remain logged into ADFS.

## Browser Support

Microsoft Edge and Internet Explorer support Windows integrated authentication by default.

Support for other browsers may be enabled using the `WIASupportedUserAgent` setting.

<https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/operations/configure-ad-fs-browser-wia>

The default settings support Internet Explorer and Microsoft Edge.

The “=~” syntax indicates a regular expression when matching the user agent.

```

Administrator: Windows PowerShell
PS C:\> Get-AdfsProperties | select -ExpandProperty WIASupportedUserAgents
MSAuthHost/1.0/In-Domain
MSIE 6.0
MSIE 7.0
MSIE 8.0
MSIE 9.0
MSIE 10.0
Trident/7.0
MSIPC
Windows Rights Management Client
MS_workFoldersClient
=~windows\s*NT.*Edge
PS C:\>
    
```

The following PowerShell command includes Chrome as a supported user agent.

Set-AdfsProperties -WIASupportedUserAgents ((Get-ADFSProperties | Select -ExpandProperty WIASupportedUserAgents) + “Chrome”)

The following command includes Firefox as a supported user agent.

Set-AdfsProperties -WIASupportedUserAgents ((Get-ADFSProperties | Select -ExpandProperty WIASupportedUserAgents) + “Firefox”)

```

Administrator: Windows PowerShell
PS C:\> Get-AdfsProperties | select -ExpandProperty WIASupportedUserAgents
MSAuthHost/1.0/In-Domain
MSIE 6.0
MSIE 7.0
MSIE 8.0
MSIE 9.0
MSIE 10.0
Trident/7.0
MSIPC
Windows Rights Management Client
MS_workFoldersClient
=~windows\s*NT.*Edge
Chrome
Firefox
PS C:\>
    
```

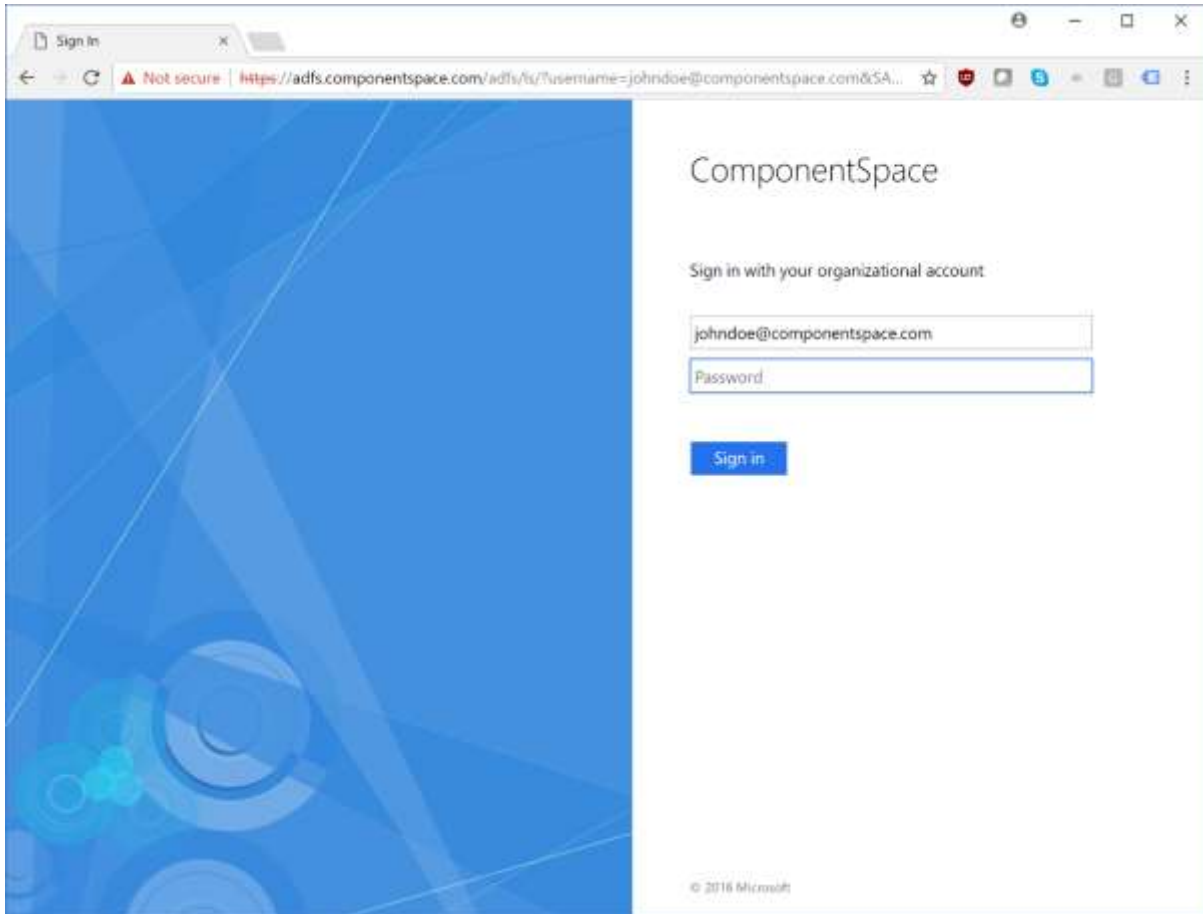
The following command removes Chrome and Firefox from the list of supported user agents.

Set-AdfsProperties -WIASupportedUserAgents (Get-ADFSProperties | Select -ExpandProperty WIASupportedUserAgents | Where-Object { \$\_ -ne "Chrome" -and \$\_ -ne "Firefox" })

## Default User Name

ADFS accepts a username query string parameter that specifies the user name to include in the login form.





The syntax is:

`https://<server-name>/adfs/ls/?username=<user-name>`

For example:

<https://adfs.componentspace.com/adfs/ls/?username=johndoe@componentspace.com>

This is useful if for some reason the user has already entered their user name at the service provider.

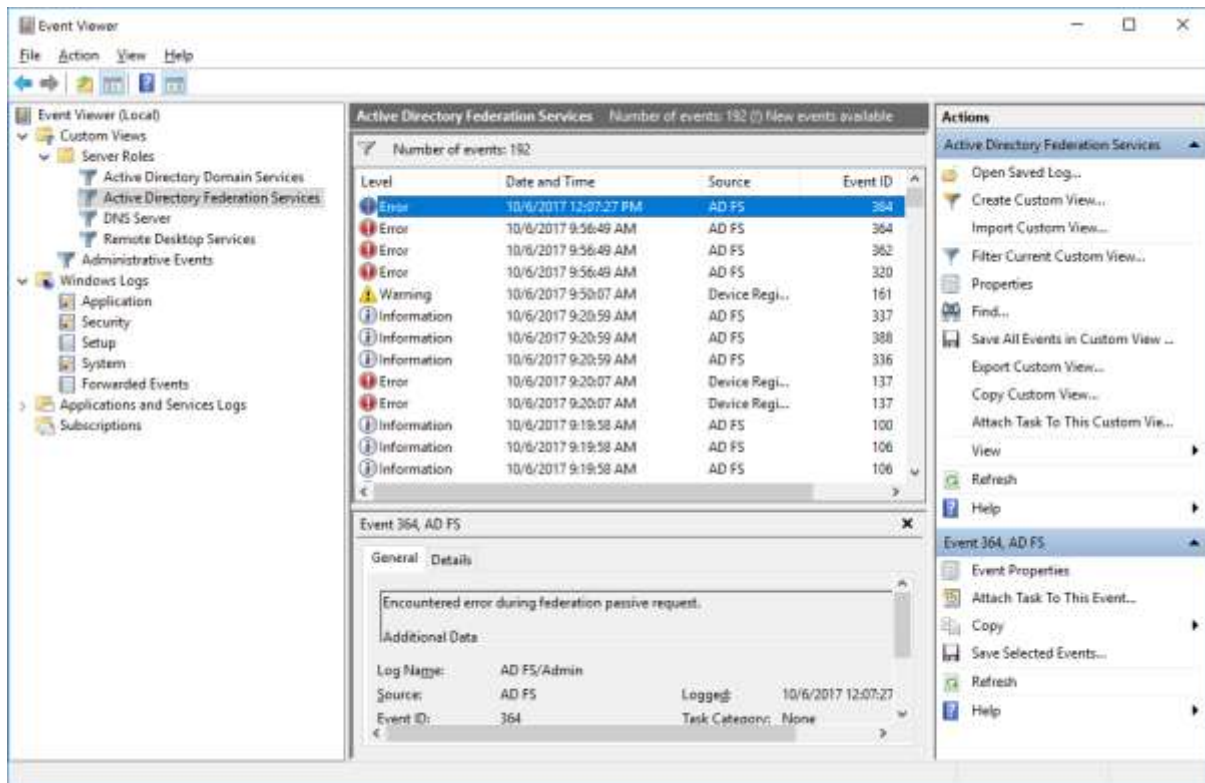
For security reasons, ADFS does not support passwords being included as a query string parameter.

The OnResolveUrl delegate may be used to update the SSO service URL with the query string parameter. Refer to the Developer Guide for details.

## Troubleshooting ADFS SSO

If an error occurs, ADFS will display a generic error message in the browser or return a generic Requester/Responder error to the service provider.

To troubleshoot configuration and other problems, refer to the ADFS event log.



For more information on troubleshooting ADFS, refer to:

<https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/troubleshooting/ad-fs-tshoot-overview>

To enable ADFS trace logging, refer to:

<https://docs.microsoft.com/en-us/windows-server/identity/ad-fs/troubleshooting/ad-fs-tshoot-logging>