SafeNet Authentication Client

Integration Guide

Using SafeNet Authentication Client CBA for Exchange 2016 with ADFS

gemalto
security to be free

Document Number: 007-013762-001, Rev. A

Release Date: May 2017

Contents

Third-Party Software Acknowledgement

This document is intended to help users of Gemalto products when working with third-party software, such as Exchange 2016 with ADFS.

Material from third-party software is being used solely for the purpose of making instructions clear. Screen images and content obtained from third-party software will be acknowledged as such.

Description

Remote access poses both a security and a compliance challenge to IT organizations. The ability to positively identify users (often remote users) requesting access to resources is a critical consideration in achieving a secure remote access solution. Deploying remote access solution without strong authentication is like putting your sensitive data in a vault (the datacenter), and leaving the key (user password) under the door mat.

A robust user authentication solution is required to screen access and provide proof-positive assurance that only authorized users are allowed access.

PKI is and effective strong authentication solution to the functional, security, and compliance requirements.

SafeNet Authentication Client (SAC) is a public key infrastructure (PKI) middleware that provides a secure method for exchanging information based on public key cryptography, enabling trusted third-party verification of user identities. Gemalto's certificate-based tokens and smart cards provide secure remote access, as well as other advanced functions, in a single token, including digital signing, password management, network logon, and combined physical/logical access.

The tokens come in different form factors, including USB tokens, smart cards, and software tokens. All of these form factors are interfaced using a single middleware client, SafeNet Authentication Client (SAC). The SAC generic integration with CAPI, CNG, and PKCS#11 security interfaces enables out-of-the-box interoperability with a variety of security applications, offering secure web access, secure network logon, PC and data security, and secure email. PKI keys and certificates can be created, stored, and used securely with the hardware or software tokens.

This document provides guidelines for deploying certificate-based authentication (CBA) for user authentication to Exchange 2016 with ADFS using Gemalto's tokens and smart cards.

It is assumed that the Exchange 2016 with ADFS environment is already configured and working with static passwords prior to implementing Gemalto multi-factor authentication.

Exchange 2016 with ADFS can be configured to support multi-factor authentication in several modes. CBA will be used for the purpose of working with Gemalto products.

Applicability

The information in this document applies to:

- **SafeNet Authentication Client (SAC) Typical installation mode** - SafeNet Authentication Client is public key infrastructure (PKI) middleware that manages Gemalto's tokens and smart cards.
- **SafeNet Authentication Client (SAC) IDGo800 Compatible mode** - DGo800 Minidriver based package, uses Microsoft Smart Card Base Cryptographic Provider to manage Gemalto IDPrime MD smart cards.

For more details about different SAC installation modes, please refer to the Customization section in SafeNet Authentication Client Administrator Guide.

- **Microsoft AD FS** 2012 R2
- **Microsoft Exchange server 2016** - Installed on Server 2012 R2.

Environment

The integration environment that was used in this document is based on the following software versions:

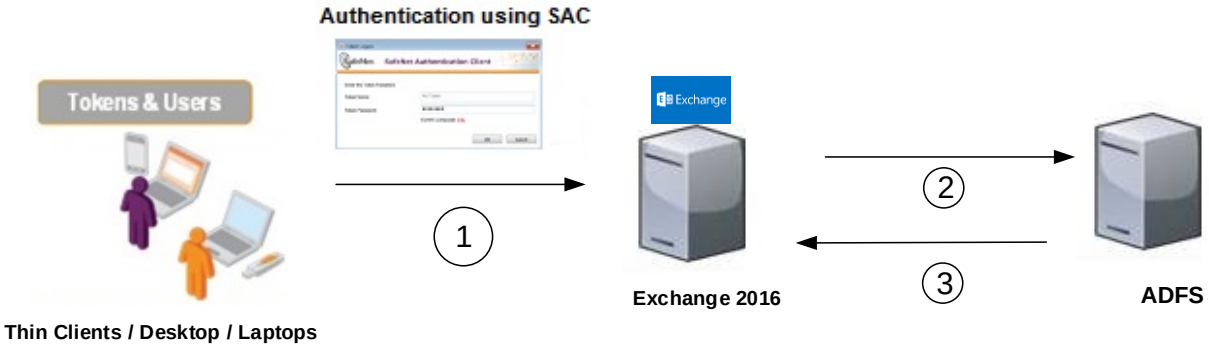
- **SafeNet Authentication Client (SAC*)*** - 10.3
- **Microsoft Exchange server 2016**
- **Microsoft AD FS 2012 R2**
- **Microsoft Active Directory & Certificate authority**

Audience

This document is targeted to system administrators who are familiar with Exchange 2016 with ADFS, and are interested in adding certificate-based authentication capabilities using Gemalto tokens and smart cards. See Supported Tokens and Smart Cards in SafeNet Authentication Client, on page 7.

CBA Flow using SafeNet Authentication Client

The diagram below illustrates the flow of certificate-based authentication:



1. A user attempts to connect to the Exchange 2016 with ADFS server using the Exchange 2016 with ADFS URL
2. The user is redirected to an ADFS authentication page, and is prompted to enter the Domain user name and password. After successfully entering the Domain user name and password, the user is prompted to enter the Token/Smart card password on which the certificate resides.
3. After successful authentication, the user is allowed to access to the Exchange portal.

Prerequisites

This section describes the prerequisites that must be installed and configured before implementing certificate-based authentication for Exchange 2016 with ADFS using Gemalto tokens and smart cards:

- To use CBA, the Microsoft Enterprise Certificate Authority must be installed and configured. In general, any CA can be used. However, in this guide, integration is demonstrated using Microsoft CA.
- If SAM is used to manage the tokens, Token Policy Object (TPO) must be configured with MS CA Connector. For further details, refer to the section “Connector for Microsoft CA” in the *SafeNet Authentication Manager Administrator’s Guide*.
- Users must have a Gemalto token or smart card with an appropriate certificate enrolled on it.
- AD FS is set-up as described: https://technet.microsoft.com/en-us/library/dn635116(v=exchg.150).aspx
- It is assumed that the environment is configured and working with domain users user name password authentication.
- SafeNet Authentication Client (10.3) must be installed on all client machines.

Supported Tokens and Smart Cards in SafeNet Authentication Client

SafeNet Authentication Client (10.3) supports the following tokens and smart cards:

Certificate-based USB tokens

- SafeNet eToken 5110 GA
- SafeNet eToken 5110 FIPS
- SafeNet eToken 5110 CC

Smart Cards

- Gemalto IDPrime MD 830
- Gemalto IDPrime MD 840

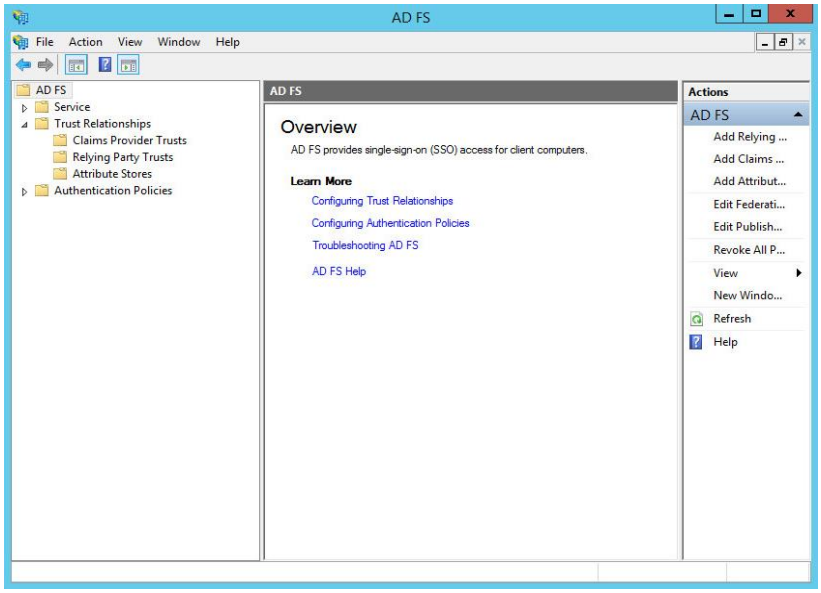
For a list of all supported devices, refer to SafeNet Authentication Client Customer Release Notes.

Configuring Exchange 2016 with ADFS

ADFS Configuration

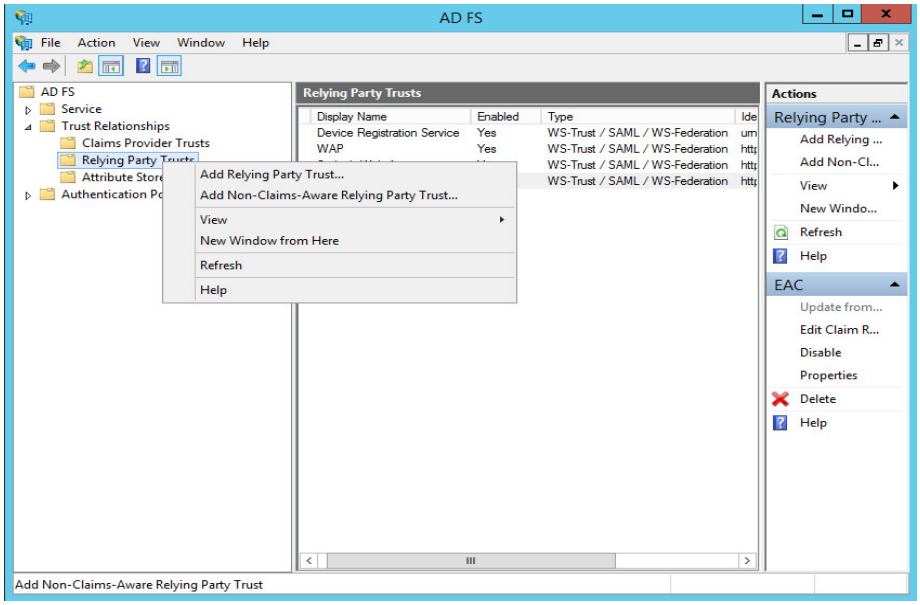
Creating a relying party trust for Outlook Web App and EAC

1. In **Server Manager**, click **Tools**, and then select **AD FS Management.**



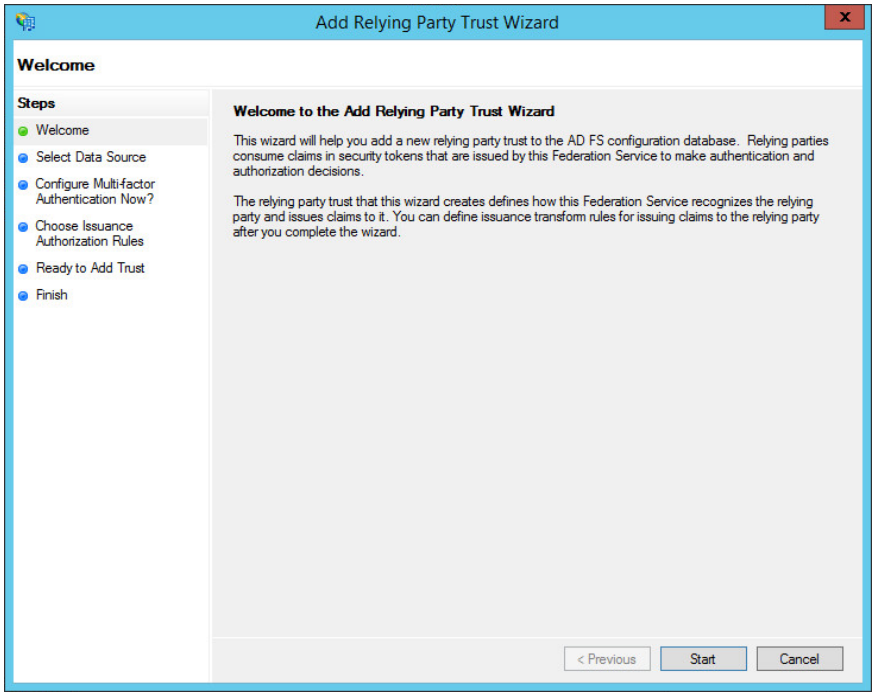
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

2. In **AD FS snap-in**, under **AD FS\Trust Relationships**, right-click **Relying Party Trusts**, and then click **Add Relying Party Trust** to open the **Add Relying Party Trust** wizard.



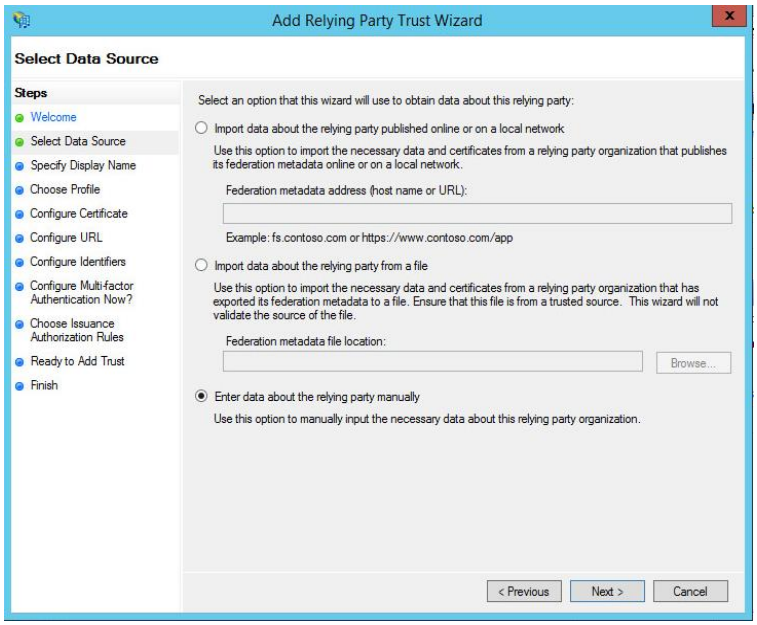
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

3. On the **Welcome** page, click **Start**.



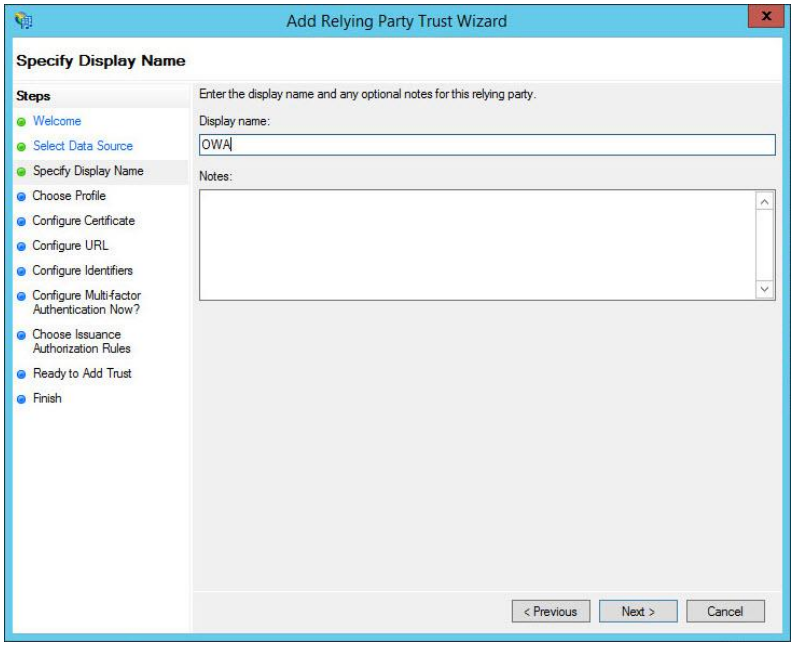
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

4. On the **Select Data Source** page, click **Enter data about the relying party manually**, and then click **Next**.



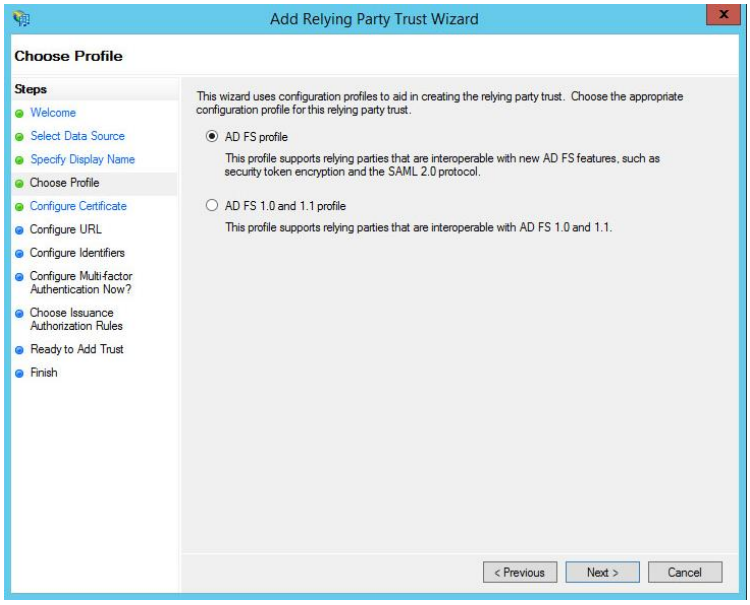
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

5. On the **Specify Display Name** page, in the **Display Name** box, type **Outlook Web App** or **OWA**, and then under **Notes**, you can type a description for this relying party trust (such as **This is a trust for https:// exchange.integ.com/owa**) and then click **Next**.



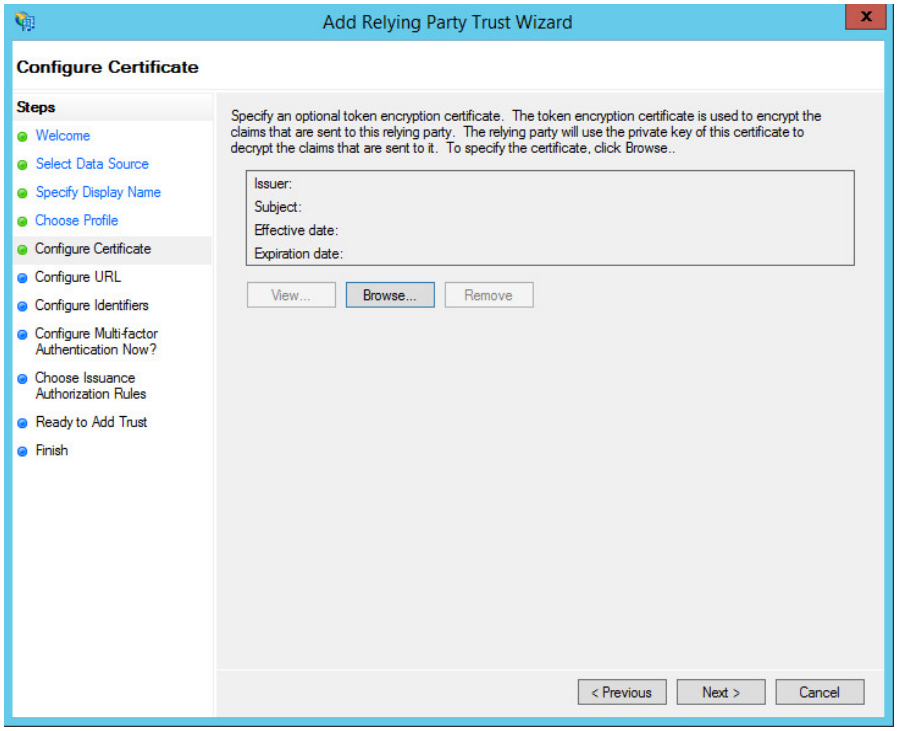
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

6. On the **Choose Profile** page, click **AD FS profile**, and then click **Next**.



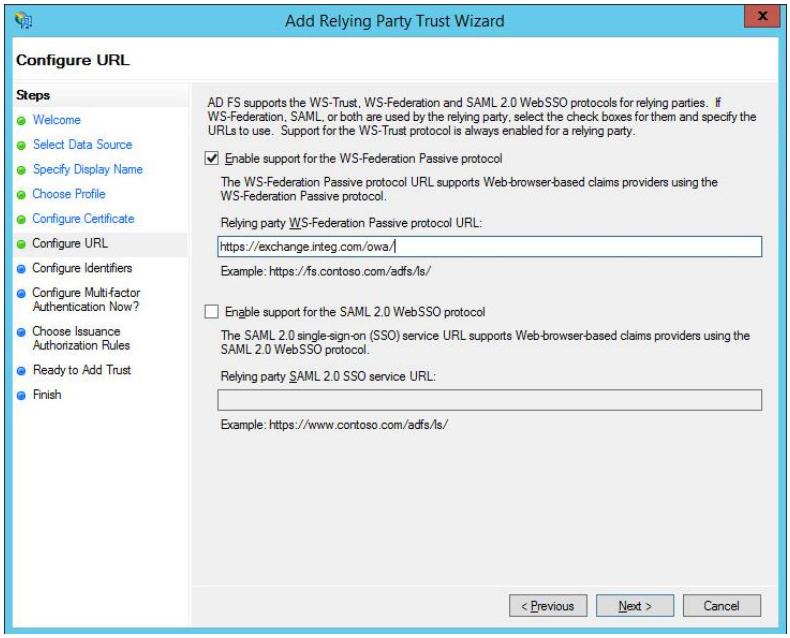
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

7. On the **Configure Certificate** page, click **Next**.



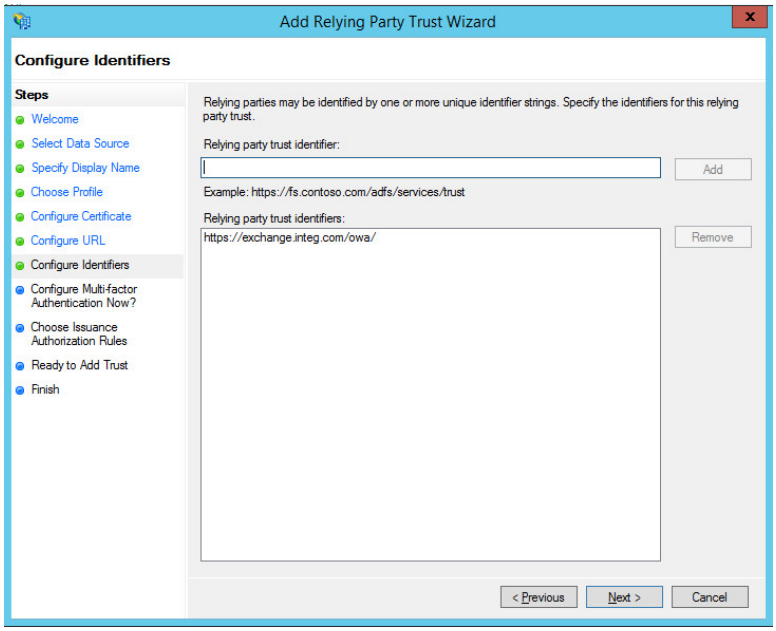
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

8. On the Configure URL page, **click Enable support for the WS-Federation Passive protocol**, and then under **Relying party WS-Federation Passive protocol URL**, type your OWA's URL (for example, https://exchange.integ.com/owa/), and then click **Next**.



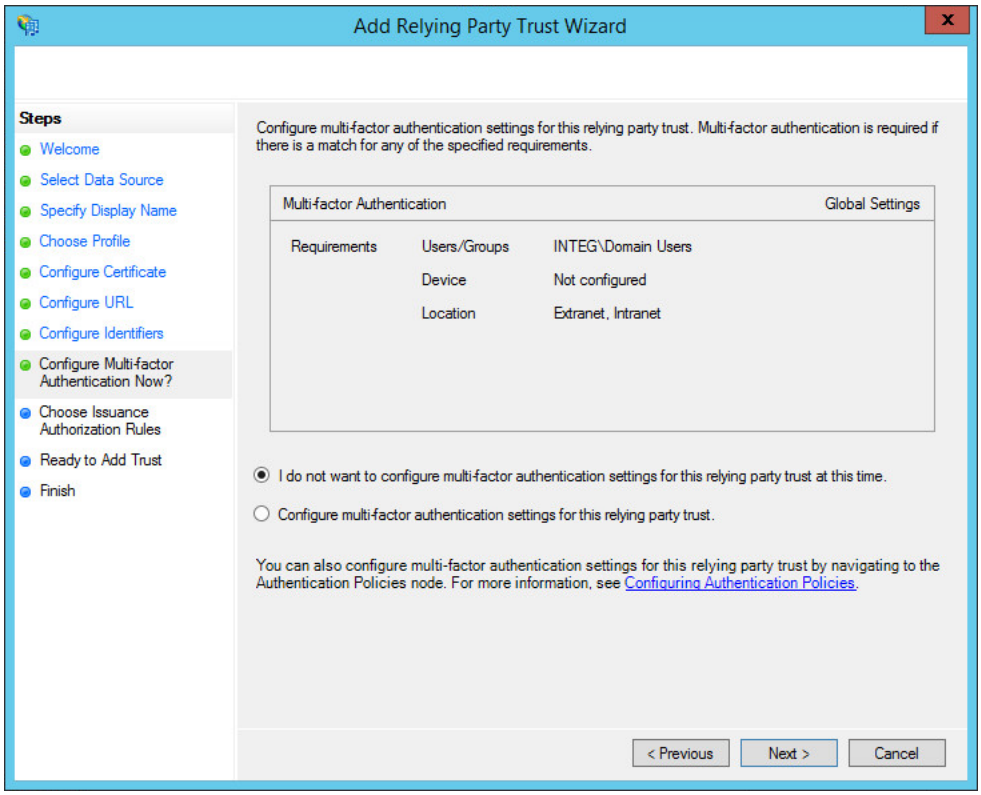
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

9. On the **Configure Identifiers** page, specify one or more identifiers for this relying party, click **Add** to add them to the list, and then click **Next**.



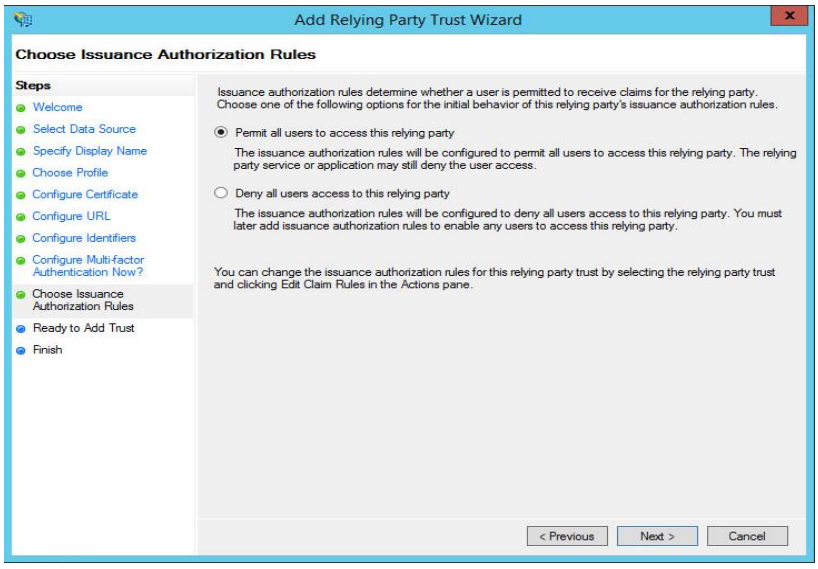
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

10. On the **Configure Multi-factor Authentication Now?** Page, leave the default configuration (we will configure MFA later on) and then click **Next**.



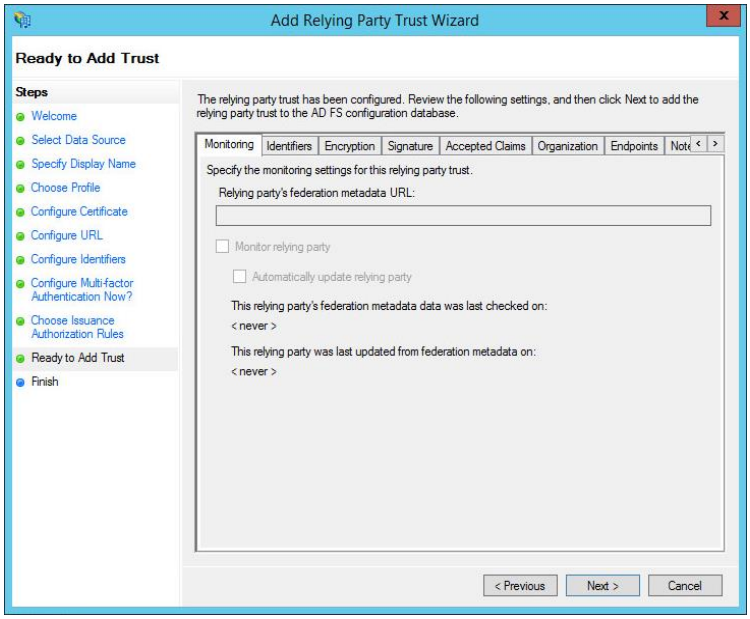
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

11. On the **Choose Issuance Authorization Rules** page, select **Permit all users to access this relying party**, and then click **next**.



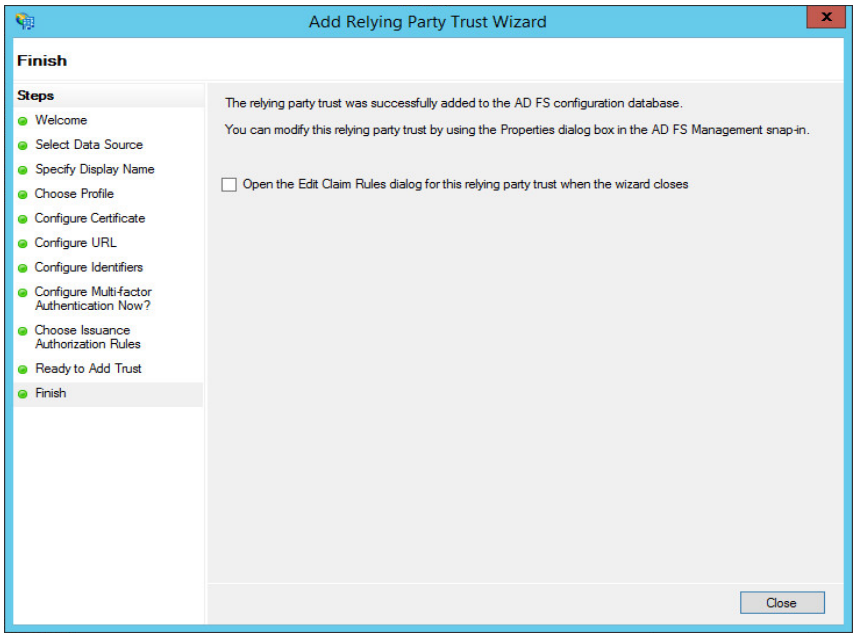
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

12. On the **Ready to Add Trust** page, review the settings, and then click **Next** to save your relying party trust information.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

13. On the **Finish** page, verify that **Open the Edit Claim Rules dialog for this relying party trust when the wizard closes** is unchecked, and click **Close**.

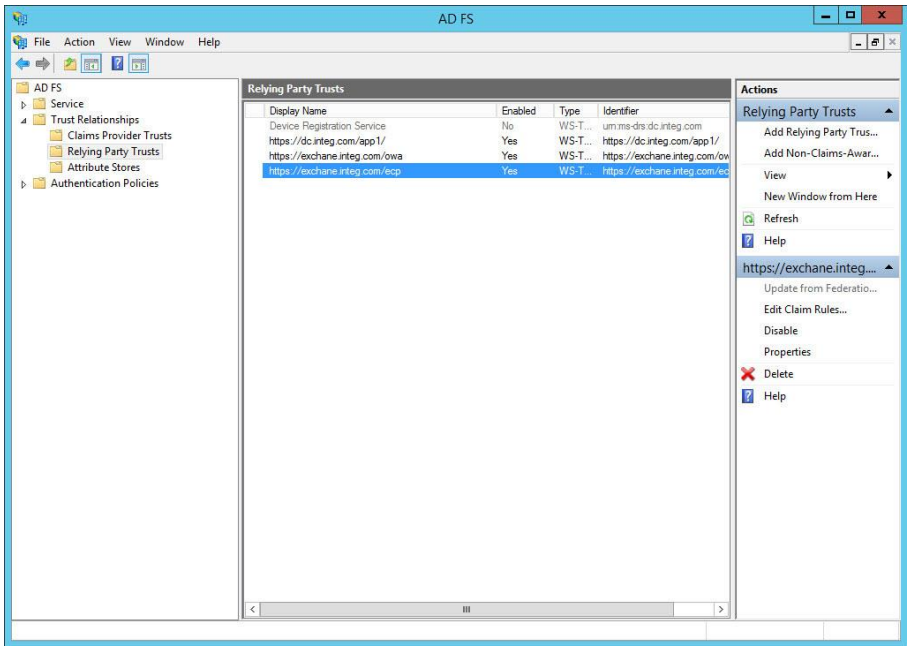


(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

14. To create a relying party trust for EAC, repeat steps 1-13 with the following differences:

- In step 5, enter **EAC** for the display name instead of Outlook Web App. For the description, you can enter, for example, **This is a trust for the Exchange Admin Center.**
- In step 8, the **Relying party WS-Federation Passive protocol URL** is ECP's URL (for example, **https://exchange.integ.com/ecp/**).

The ADFS **Relying Party Trusts** pane should now include two new trusts (OWA and ECP):



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

Adding ADFS claim rules for OWA and EAC

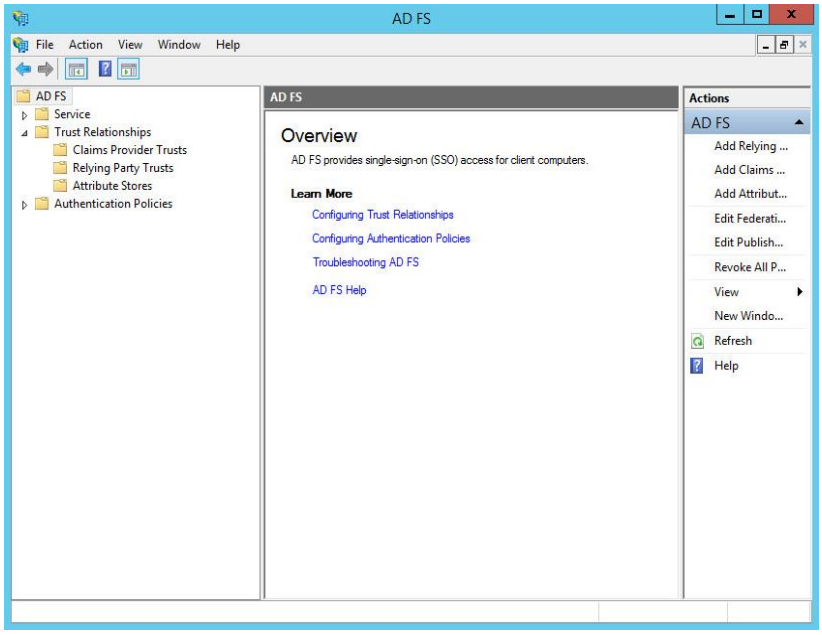
In a claims-based identity model, the function of Active Directory Federation Services (AD FS) as a federation service is to issue a token that contains a set of claims. Claims rules govern the decisions in regard to claims that AD FS issues. Claim rules and all server configuration data are stored in the AD FS configuration database.

You must create three claim rules:

- Active Directory user SID
- Active Directory group SID
- Active Directory UPN

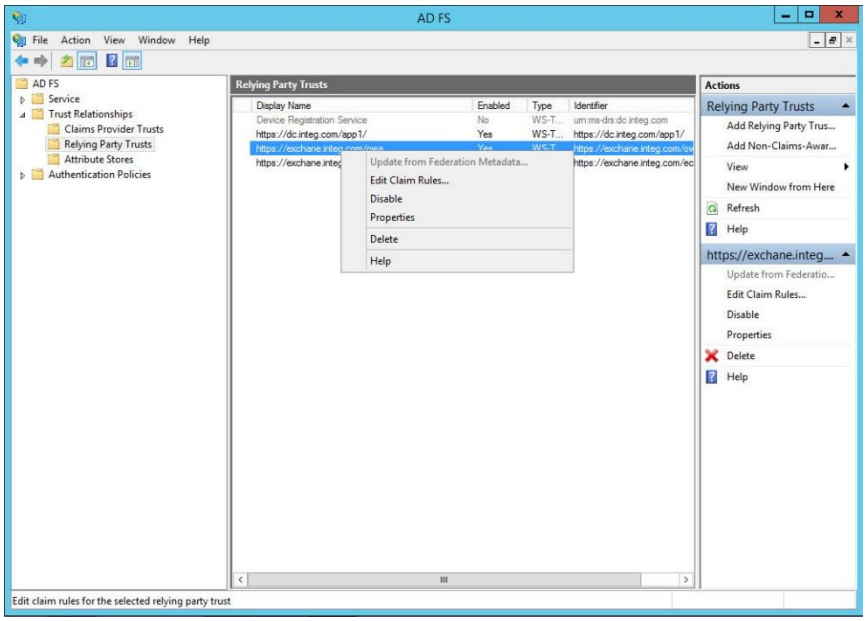
To add the required claims:

1. In **Server Manager**, click **Tools**, and then click **AD FS Management**.



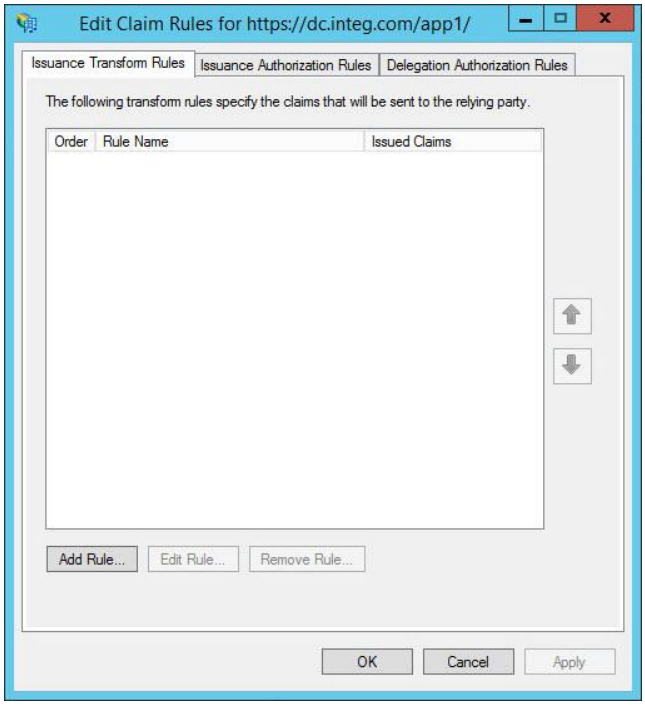
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

2. In the console tree, under **AD FS\Trust Relationships**, click the **Relying Party Trusts**, and then right-click the Outlook Web App trust, and then click **Edit Claim Rules**.



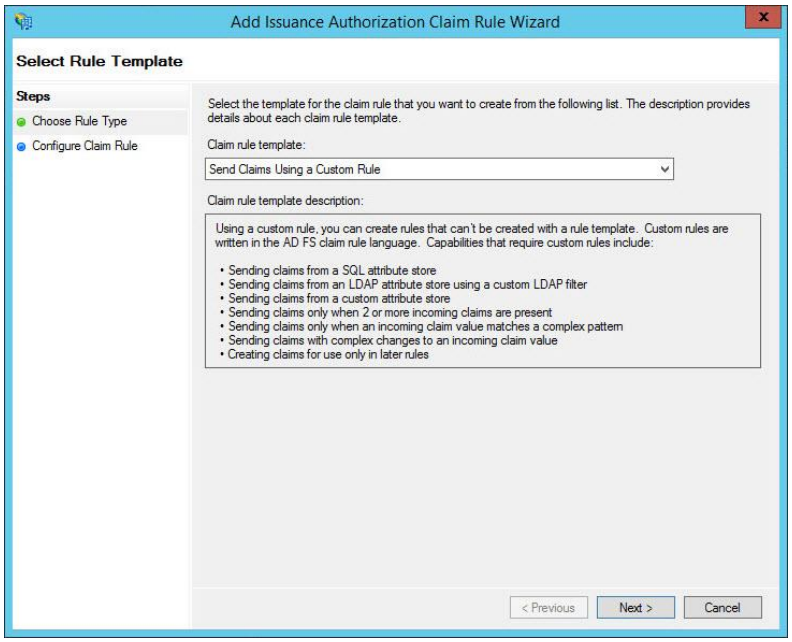
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

3. In the **Edit Claim Rules** window, on the **Issuance Transform Rules** tab, click **Add Rule** to start the Add Transform Claim Rule Wizard.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

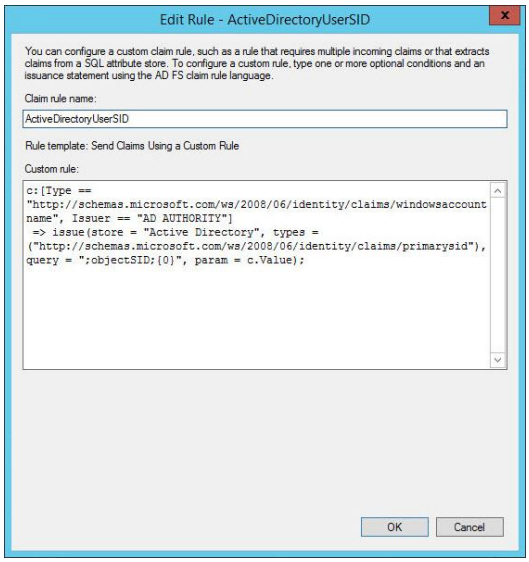
4. On the **Select Rule Template** page, under **Claim rule template**, select **Send Claims Using a Custom Rule**, and click **Next**.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

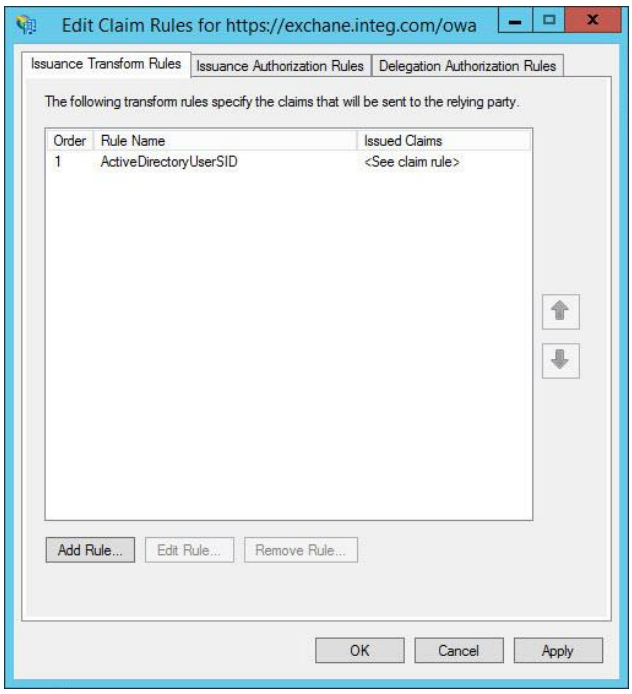
5. On the **Configure Rule** page, in the **Choose Rule Type** step, under **Claim rule name**, enter the name for the claim rule. Use a descriptive name for the claim rule, for example, **ActiveDirectoryUserSID**. Under **Custom rule**, enter the following claim rule language syntax:

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

=> issue(store = "Active Directory", types =
("http://schemas.microsoft.com/ws/2008/06/identity/claims/primarysid"), query = ";objectSID;{0}",
param = c.Value);
```



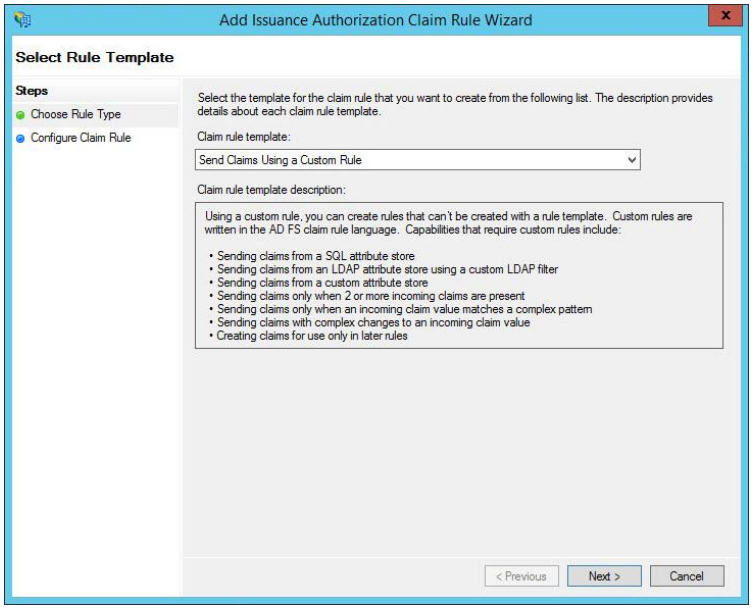
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

6. On the **Configure Rule** page, click **OK**.
7. In the **Edit Claim Rules** window, on the **Issuance Transform Rules** tab, click **Add Rule** to start the Add Transform Claim Rule Wizard.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

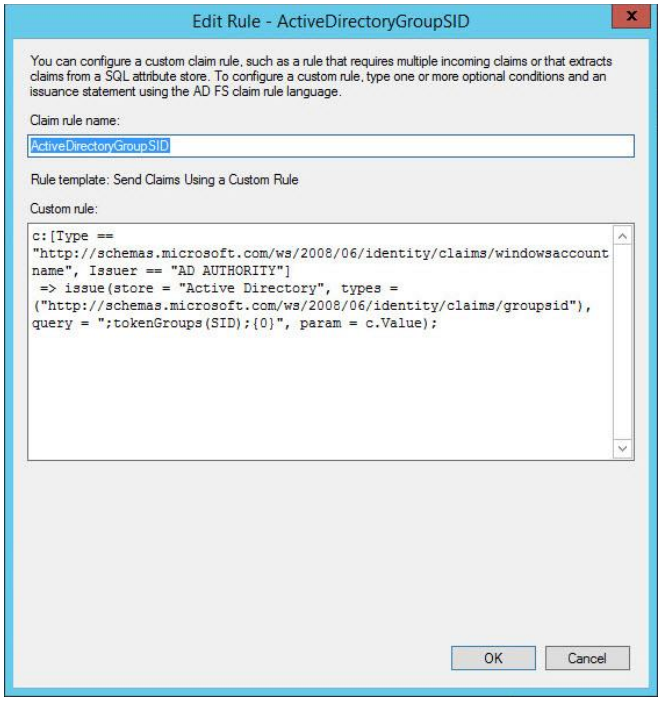
8. On the **Select Rule Template** page, under **Claim rule template**, select **Send Claims Using a Custom Rule**, and then click **Next**.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

9. On the **Configure Rule** page, on the **Choose Rule Type** step, under **Claim rule name**, enter the name for the claim rule. Use a descriptive name for the claim rule, for example, **ActiveDirectoryGroupSID**. Under **Custom rule**, enter the following claim rule language syntax for this rule:

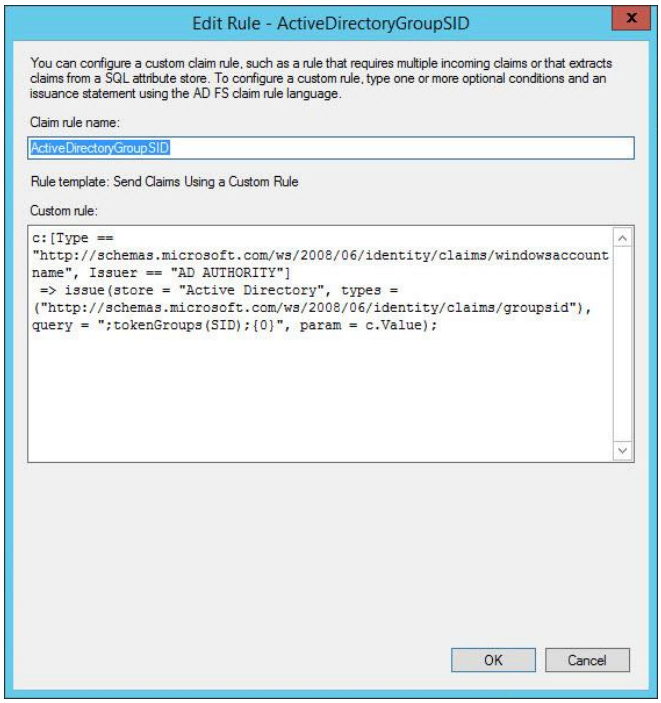
```
c:[Type == "http://schemas.microsoft.com/ws/2008/06/identity/claims/windowsaccountname",
Issuer == "AD AUTHORITY"]

=> issue(store = "Active Directory", types =
("http://schemas.microsoft.com/ws/2008/06/identity/claims/groupsid"), query =
";tokenGroups(SID);{0}", param = c.Value);
```



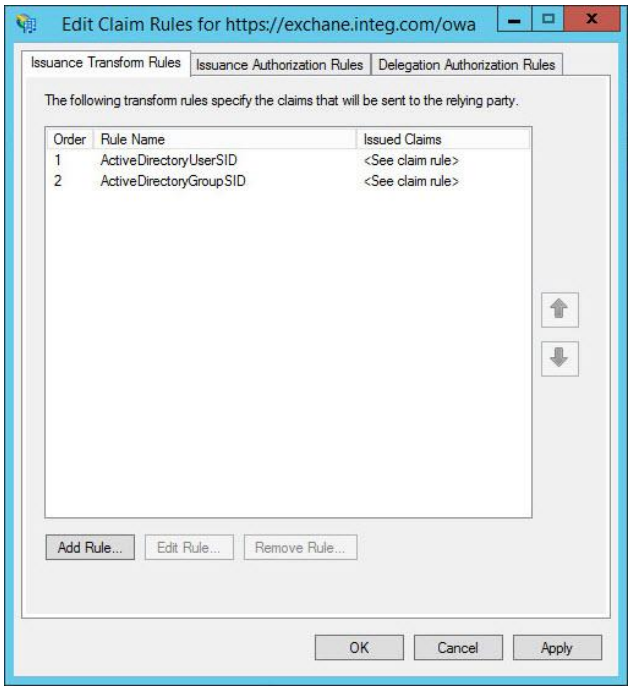
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

10. On the **Configure Rule** page, click **OK**.

11. In the **Edit Claim Rules** window, on the **Issuance Transform Rules** tab, click **Add Rule** to start the **Add Transform Claim Rule** wizard.

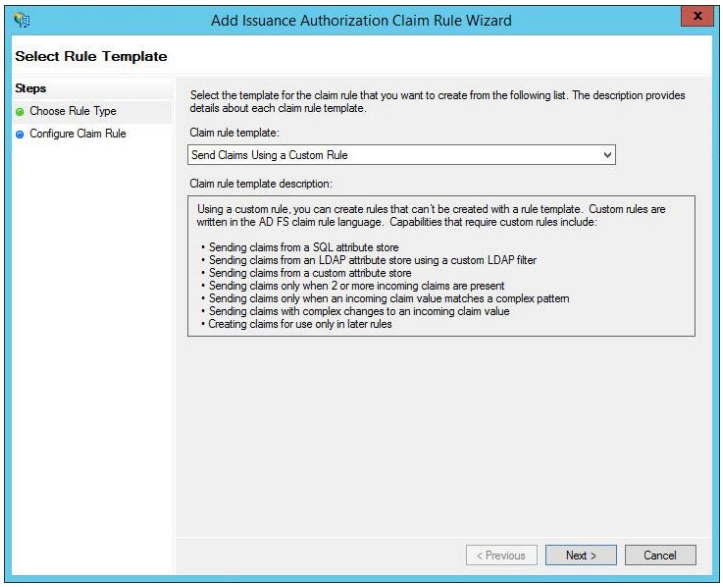


(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

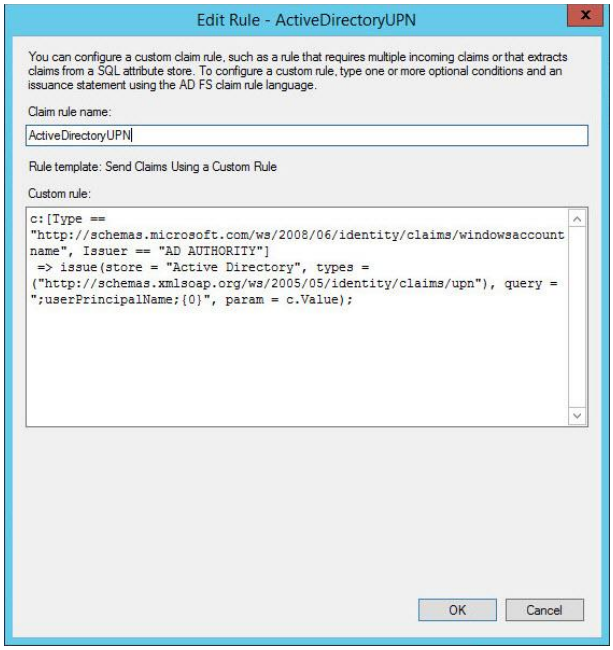
12. On the **Select Rule Template** page, under **Claim rule template**, select **Send Claims Using a Custom Rule,** and then click **Next**.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

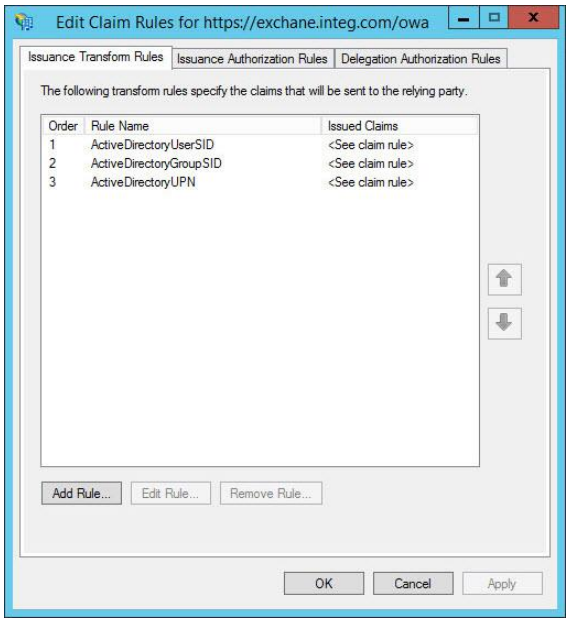
13. On the **Configure Rule** page, on the **Choose Rule Type** step, under **Claim rule name**, enter the name for the claim rule. Use a descriptive name for the claim rule, for example, **ActiveDirectoryUPN**. Under **Custom rule**, enter the following claim rule language syntax for this rule:

```
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);
```



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

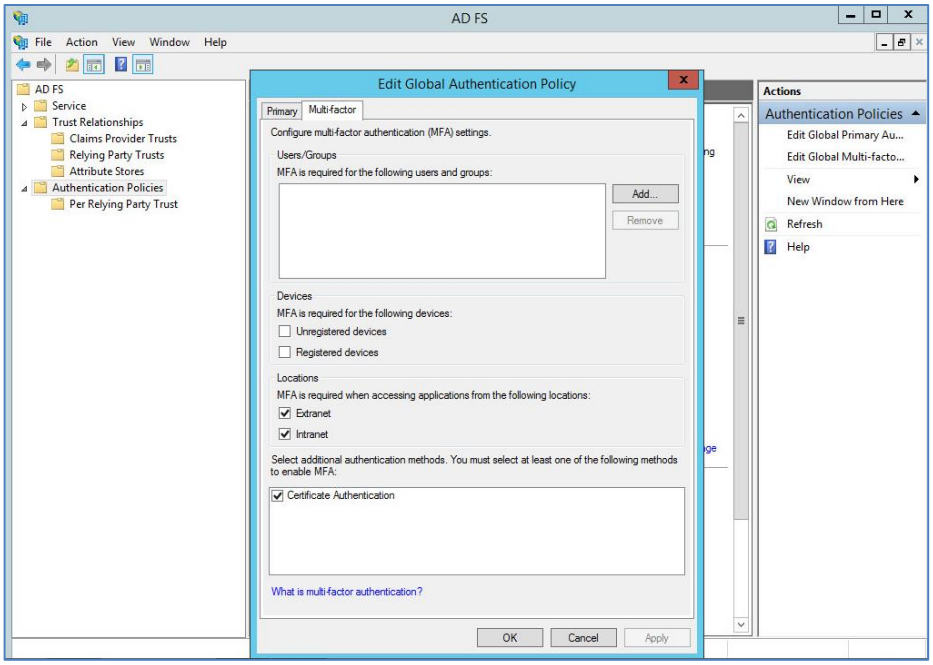
14. Click **OK**.
15. In the **Edit Claim Rules** window, click **Apply**, and then **OK**.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

Configuring AD FS Authentication Policy:

1. In the **ADFS Management** window, right click on the **Authentication Policies** and choose to **Edit Global Primary Authentication**.
2. In the **Primary** tab, make sure **Form Authentication** is checked both for Extranet and Intranet.
3. In the **Multi-factor** tab:
   a. Add the users/groups to be controlled by the MFA.
   b. Choose **Extranet/Intranet** (or both) according to your preferred configuration.
   c. Make sure **Certificate Authentication** is checked as additional authentication methods.
   d. Press **ok**.



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

Exchange Configuration

In this step, after configuring AD FS to use claims-based authentication with Outlook Web App and EAC, it is needed to enable AD FS on the exchange side:

1. Copy signing certificate thumbprint From AD FS Server
2. **ON ADFS Server** Locate the AD FS token signing certificate thumbprint by using Windows PowerShell on the AD FS server.

    a. Open Windows PowerShell and enter:
    `Get-ADFSCertificate -CertificateType "Token-signing"`
    b. Copy the signing certificate thumbprint

3. On **Exchange Server**, using the Exchange Management Shell enter and run (see the example and paste the signing certificate thumbprint where needed):

    Example:

    Open Exchange Management Shell and enter:

    $uris = @(" https://exchange.integ.com/owa/","https://exchange.integ.com/ecp/")
    Set-OrganizationConfig -AdfsIssuer "https://adfs.integ.com/adfs/ls/" -AdfsAudienceUris $uris -AdfsSignCertificateThumbprint"88970C64278A15D642934DC2987D9CCA5E65DS3B"

4. Enable ADFS authentication on the virtual directories.

5. **On the Exchange Server** using the Exchange Management Shell Enable ADFS authentication enter and run following:

   a. For ECP Open Windows PowerShell and enter:

      Get-EcpVirtualDirectory | Set-EcpVirtualDirectory -AdfsAuthentication $true -BasicAuthentication $false -DigestAuthentication $false -FormsAuthentication $false -WindowsAuthentication $false

   b. For OWA Open Windows PowerShell and enter:

      Get-OwaVirtualDirectory | Set-OwaVirtualDirectory -AdfsAuthentication $true -BasicAuthentication $false -DigestAuthentication $false -FormsAuthentication $false -WindowsAuthentication $false -OAuthAuthentication $false

   Note: ECP virtual directory must be configured before OWA

6. Perform **IIS Reset.**

Running the Solution

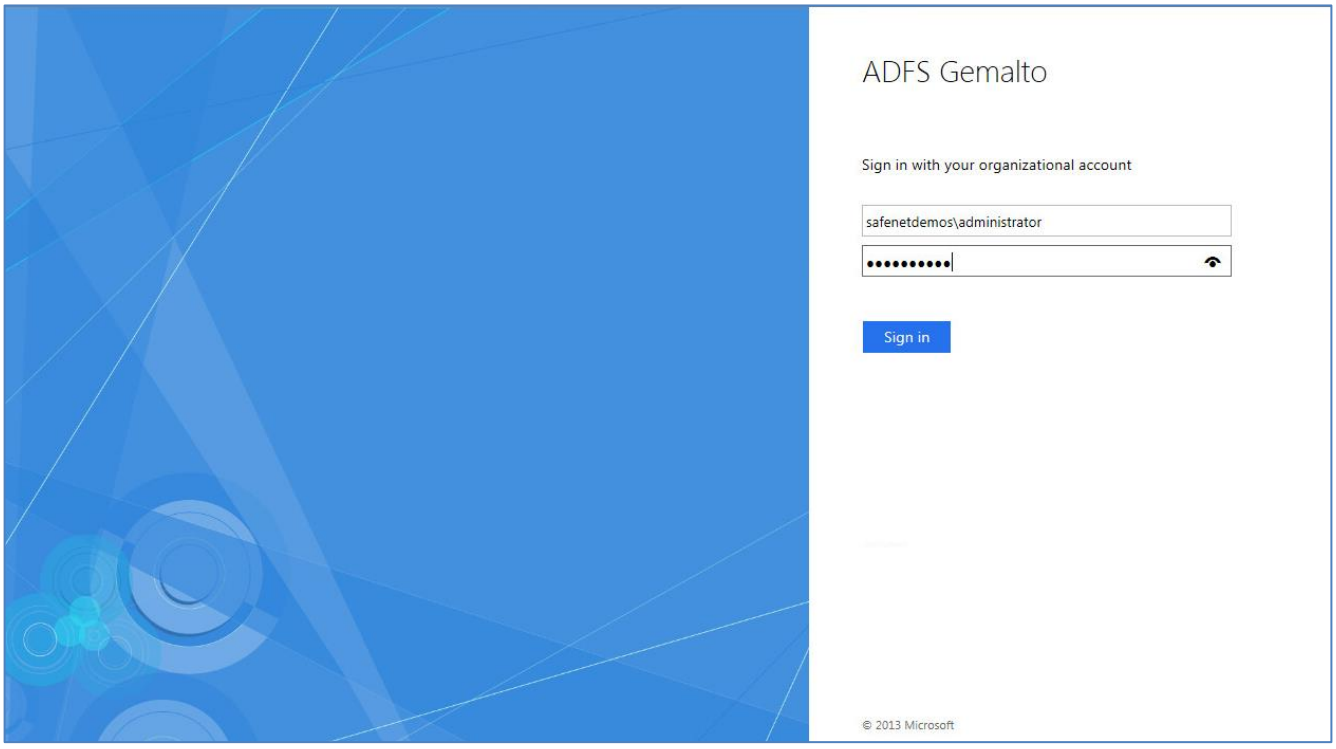
After ADFS and Exchange is configured to use authentication (through ADFS), users can log on to Exchange 2016 using Multi Factor Authentication.

Prerequisites:

- SafeNet Authentication Client is installed
- Token/smart card with smart card user certificate is connected.

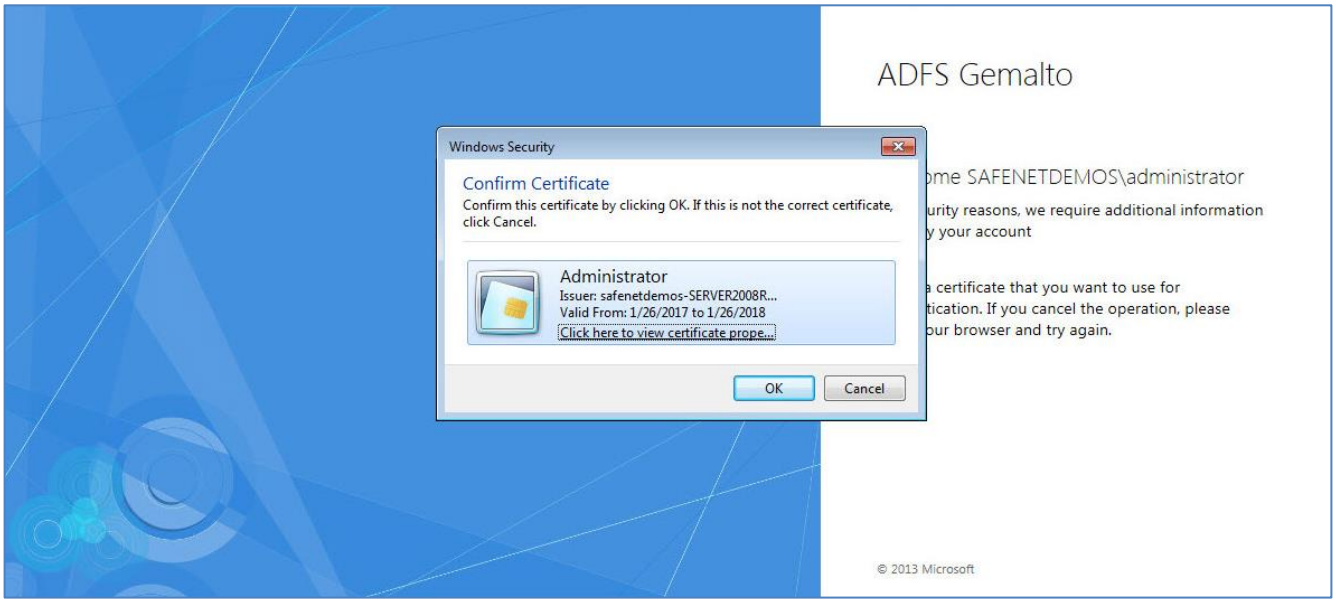
To run the solution:

1. The administrator browses to the exchange URL **Example**: https://exchange.integ.com/ecp/ and is redirected to the organization's ADFS login page.



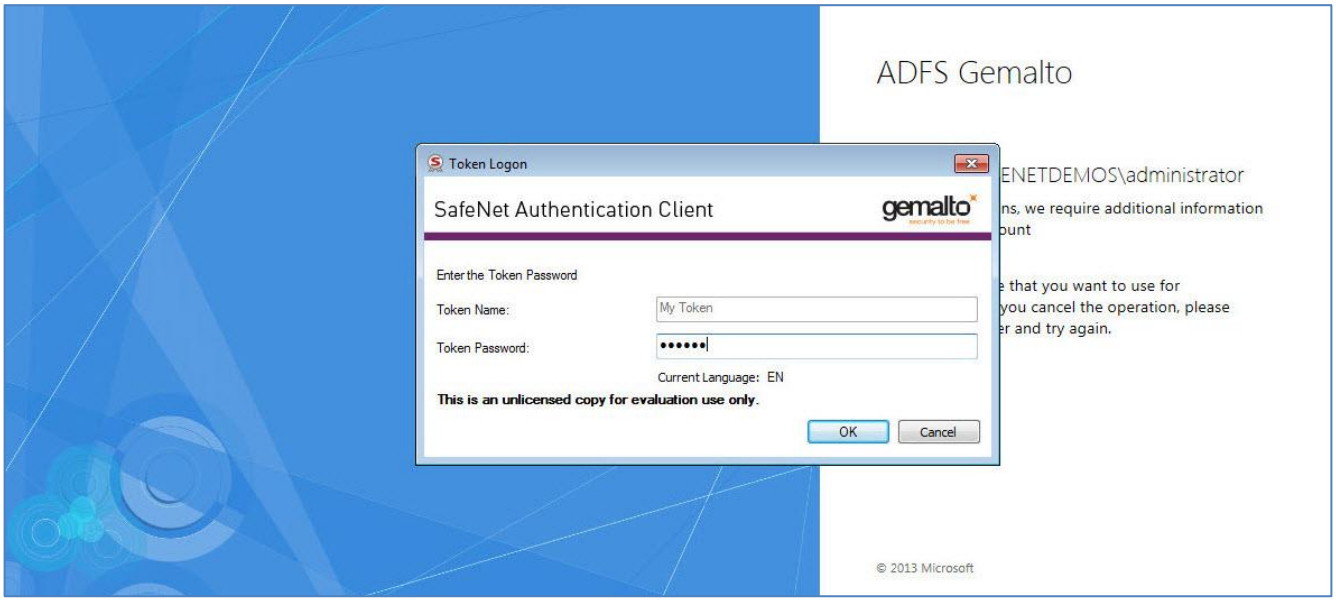
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

2. The administrator enters AD credentials and clicks **Sign in**.
3. After successful login, when prompted to confirm certificate, the administrator clicks **OK.**



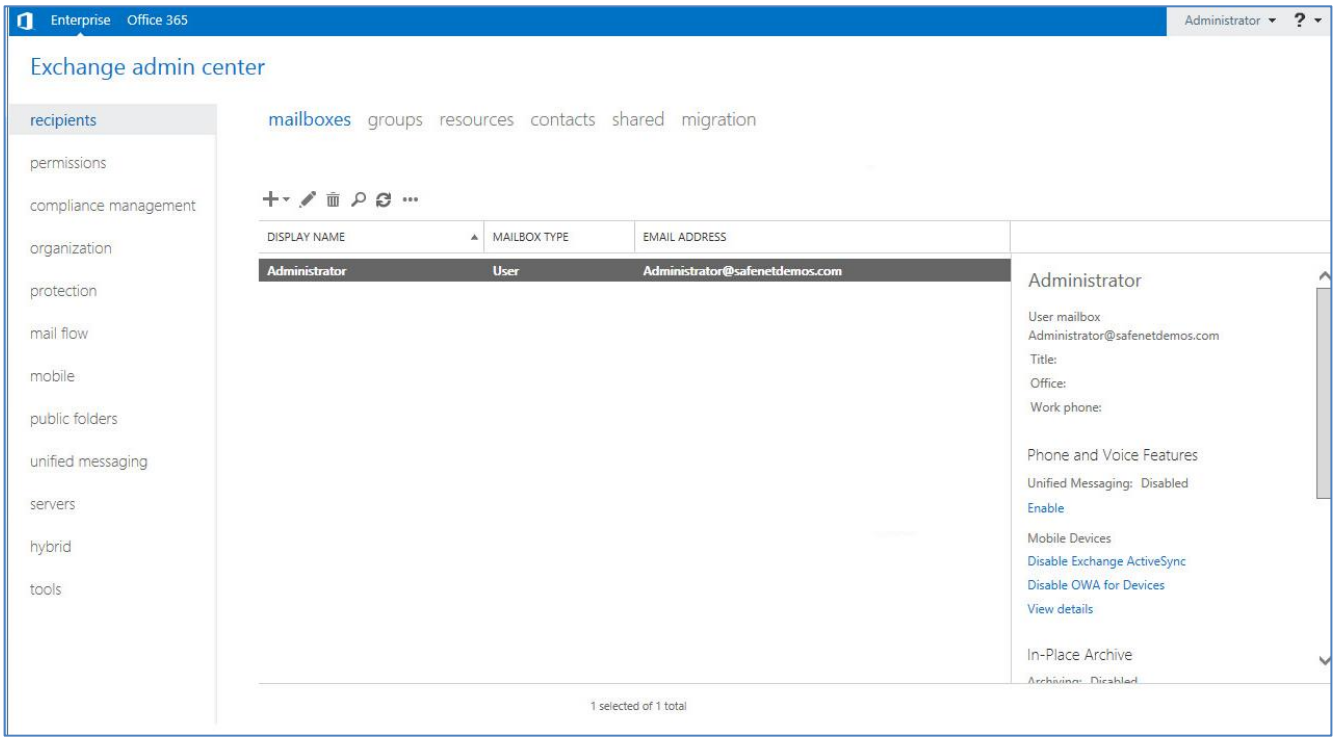
(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

4. After certificate confirmation, the smart card logon window opens. The user enters the token/smart card password and clicks **OK.**



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

5. After another successful login, the user is redirected to the organization's Exchange server (in this example: https://exchange.integ.com/ecp/)



(The screen image above is from Microsoft© software. Trademarks are the property of their respective owners.)

Support Contacts

If you encounter a problem while installing, registering, or operating this product, please make sure that you have read the documentation. If you cannot resolve the issue, contact your supplier or Gemalto Customer Support. Gemalto Customer Support operates 24 hours a day, 7 days a week. Your level of access to this service is governed by the support plan arrangements made between Gemalto and your organization. Please consult this support plan for further information about your entitlements, including the hours when telephone support is available to you.

Contact Method	Contact Information
Customer Support Portal	https://supportportal.gemalto.com Existing customers with a Technical Support Customer Portal account can log in to manage incidents, get the latest software upgrades, and access the Gemalto Knowledge Base.
Technical Support contact email	technical.support@gemalto.com