Cloud Access Manager 8.1.3

Configuration Guide

Legend

WARNING: A WARNING icon indicates a potential for property damage, personal injury, or death.

CAUTION: A CAUTION icon indicates potential damage to hardware or loss of data if instructions are not followed.

IMPORTANT, NOTE, TIP, MOBILE, or VIDEO: An information icon indicates supporting information.

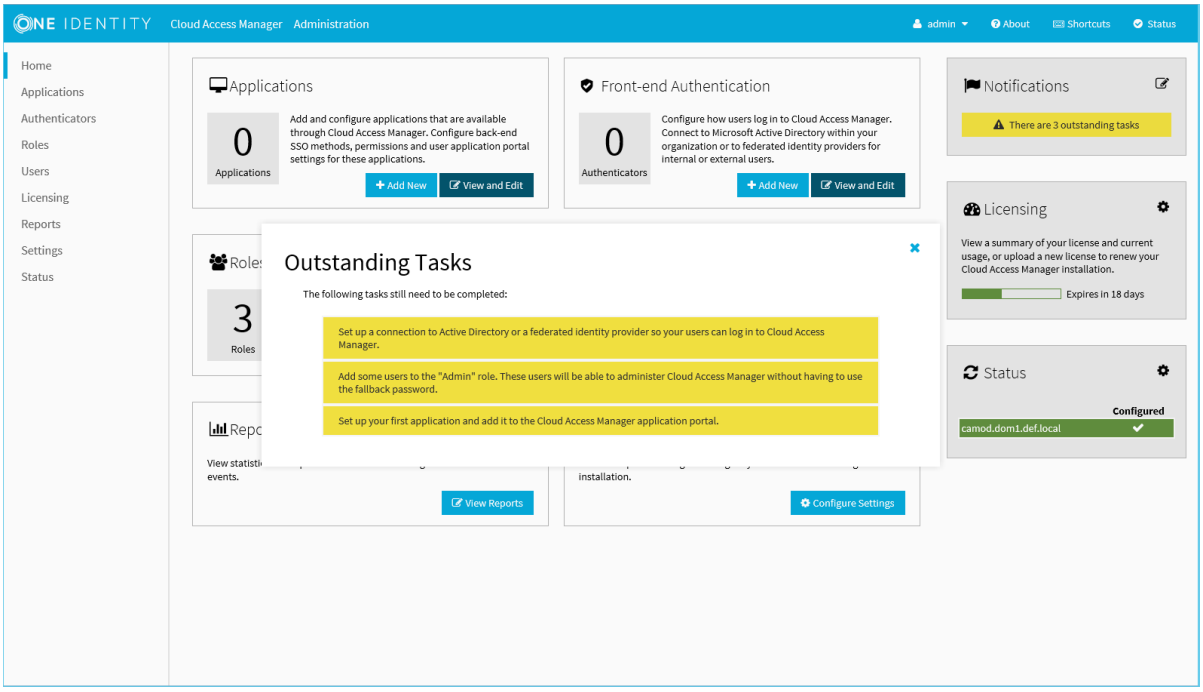
Cloud Access Manager Configuration Guide
Updated - October 2017
Version - 8.1.3

Contents

Configuring a front-end authentication method

Before users and administrators can log in to Cloud Access Manager you will need to configure a Front-end Authentication method. Typically this will involve configuring the Microsoft Active Directory authenticator to authenticate users to your domain, but equally you can configure the SAML or WS-Federated authenticator to authenticate users to a different security authority.

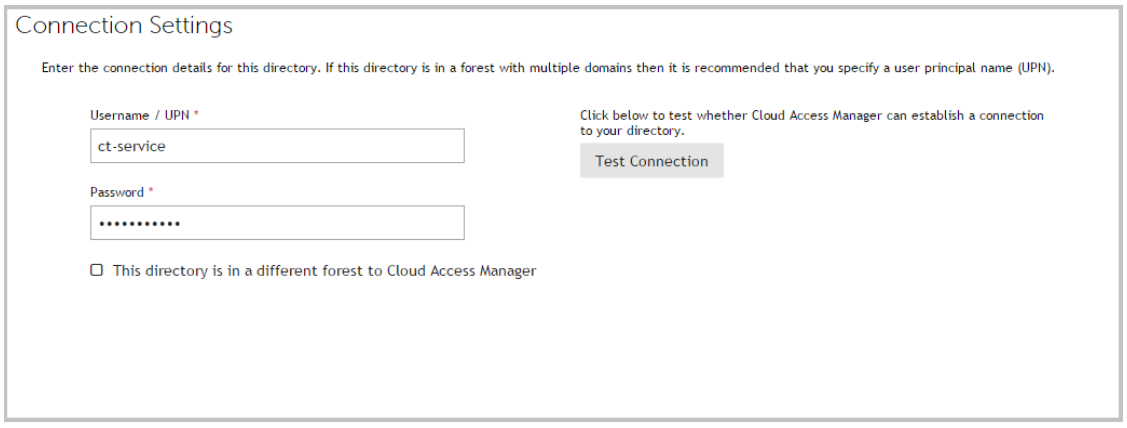
Figure 1: Cloud Access Manager



Microsoft Active Directory authentication

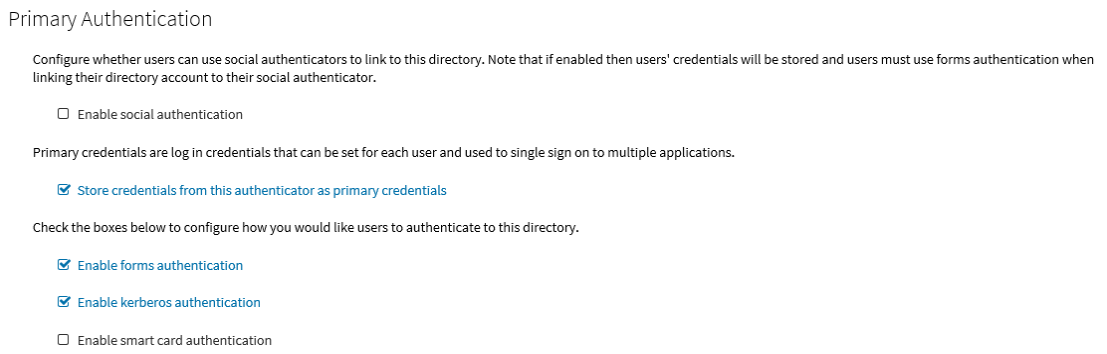
To configure Microsoft Active Directory front-end authentication

1. Log in to the Administration Console and select **Add New** from the **Front-end Authentication** section on the home page.
2. Select **Microsoft Active Directory**, then click **Next**.
3. Enter the full user principal name (UPN), for example *johndoe@domain.org*, and password of a Windows domain account that has read access to all user and group objects in the forest (usually a regular user account belonging to the Domain Users group is sufficient).
4. Click **Test Connection**. This will test that Cloud Access Manager can connect to the domain using the credentials provided. When successful, click **Next**.



5. The settings on the **Primary Authentication** screen are split into three sections. When complete, click **Next**.

   The first section is used to determine whether or not users are allowed to use social authenticators, for example Facebook or Google, and link to the selected authenticator when authenticating to Cloud Access Manager.



The second section determines whether users' credentials are stored for accessing other applications. If selected, the credentials used to authenticate to Cloud Access Manager are stored as the Primary Credentials in the user's **Password Wallet**. Please refer to Primary credentials on page 37 for details.

The third section is used to determine how users are challenged for their Windows credentials, you must choose at least one option. Cloud Access Manager checks for credentials presented in the following order of precedence:

a. **Enable kerberos authentication** — Cloud Access Manager will check for a Kerberos ticket generated during Windows domain login and supplied by the browser. If the Kerberos ticket is present and valid, then the user will be successfully logged in.

   Successful Kerberos authentication requires correct configuration of the user's browser. Please refer to Integrated Windows Authentication on page 41 for details. In addition some browsers do not support Kerberos authentication. Please refer to the *One Identity Cloud Access Manager Installation Guide* for browsers that support Integrated Windows Authentication.

b. **Enable smart card authentication** — Users are given the opportunity to present an X.509 certificate in order to log in to Cloud Access Manager. The X.509 certificate may be located on a smart card or in the client computer's certificate store. If the certificate is invalid or expired the login attempt will be rejected. Please refer to the section Configuring smart card authentication on page 15 for details.

c. **Enable forms authentication** — Users are prompted for their Active Directory username and password using a login form.

NOTE: If you enable social authentication, storing credentials from the authenticator is required, this in turn requires that forms authentication is the only enabled authentication method. Storing credentials is required as Cloud Access Manager needs to verify if the linked account used for primary authentication is still valid, for example the account is not disabled, or the password has not expired when authenticating using a social authenticator. If a user attempts to authenticate with a social authenticator and the linked account is not valid, the user will be prompted to enter the correct credentials for the primary authenticator.

NOTE: If you enable social authentication, we recommend that you set linked accounts to have a long password expiry, this allows seamless authentication using the social authenticator.

6. If you require two factor authentication each time users authenticate to Cloud Access Manager, select **Use two factor authentication for all applications** from the **Two factor authentication mode** list. Select the method of authentication from the **Type of two factor authentication** list.

   For information on how to configure the various authentication types or how to configure two factor authentication only for specific users or applications, refer to Configuring step-up authentication on page 83. When complete, click **Next**

7. In the **Authenticator Name** field, enter the name that will be used to identify the authenticator within Cloud Access Manager, then click **Finish**.

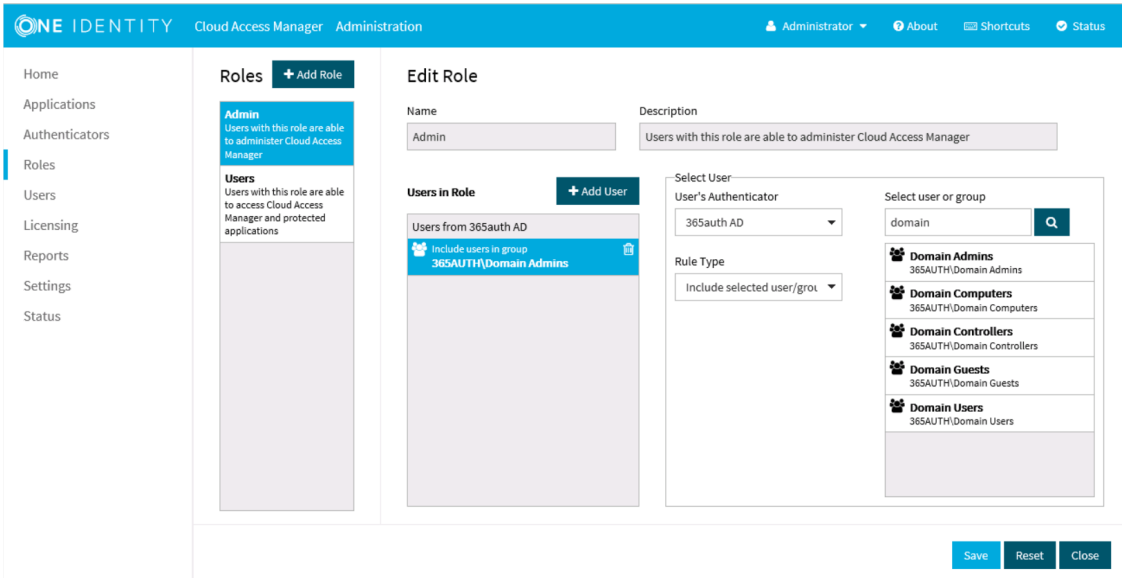
   NOTE: This name will be seen by Cloud Access Manager users during authentication if multiple authentication methods have been configured.

8. You have now created the front-end authentication method. Click **Edit Roles**.

   Before Cloud Access Manager administrators and users can log in to Cloud Access Manager using their Active Directory credentials, you must tell Cloud Access Manager how to identify administrators and users based on their Active Directory group membership. For example, the Domain Admins group for Cloud Access Manager administrators and the Domain Users group for regular Cloud Access Manager users.

9. Click **Admin**.

10. Click +**Add User**.



11. Select the new Active Directory authentication method from the list.
12. Use the search box to locate the group whose members are to be granted access to the Cloud Access Manager Administration application, then select the group from the list.
13. Click **Save**.
14. Now repeat the process for the Cloud Access Manager users. Click **Users**.
15. Click +**Add User**.
16. Select the new Active Directory authentication method from the list.
17. Select a group from the list whose members are to be granted access to the Cloud Access Manager application portal.
18. Click **Save**.
19. Click **Close** to return to the Cloud Access Manager Administration Console. The configuration is now complete. Cloud Access Manager administrators and users can now log in to Cloud Access Manager using their Active Directory credentials.

NOTE: The Active Directory authentication method supports single sign-on (SSO) using Integrated Windows Authentication (IWA) to the Cloud Access Manager application portal for users signed into a domain connected workstation. The next section describes how to configure Cloud Access Manager for Integrated Windows Authentication.

Configuring Cloud Access Manager for Integrated Windows Authentication

The Active Directory front-end authentication method supports Integrated Windows Authentication (IWA) to provide single sign-on (SSO). This allows users signed into a domain connected workstation using their Active Directory account access to their Cloud Access Manager portal without re-entering their credentials.

To enable users to access the Cloud Access Manager portal without entering their Active Directory credentials, the following additional steps are required.

Configuring the Cloud Access Manager service account for SSO

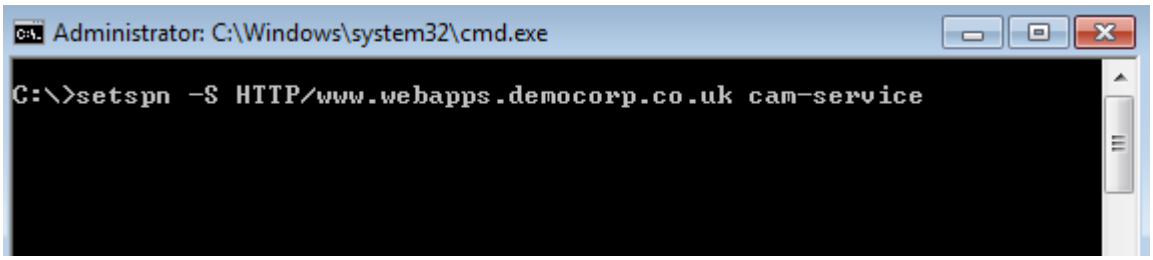
NOTE: This step is not required when using the Proof Of Concept (POC) installation type. This uses the built-in *system* account rather than a dedicated service account.

- To enable the user's web browser to authenticate the user with Cloud Access Manager using Kerberos, the browser must first identify the service account used to run Cloud Access Manager. The user's browser must authenticate with the Cloud Access Manager service account. This is achieved by configuring a Service Principal Name (SPN) for the service account that maps the Cloud Access Manager Proxy hostname to the Cloud Access Manager service account name.
- To create an SPN, you need the name of the service account specified during the Cloud Access Manager installation and the hostname assigned to the proxy for the Cloud Access Manager portal. This is the account name and the hostname entered in the section *Installing Cloud Access Manager*, in the *One Identity Cloud Access Manager Installation Guide*.
- On the Cloud Access Manager STS host, open a command prompt window and run the following command:

  ```
  setspn -S HTTP/<hostname> <account>
  ```

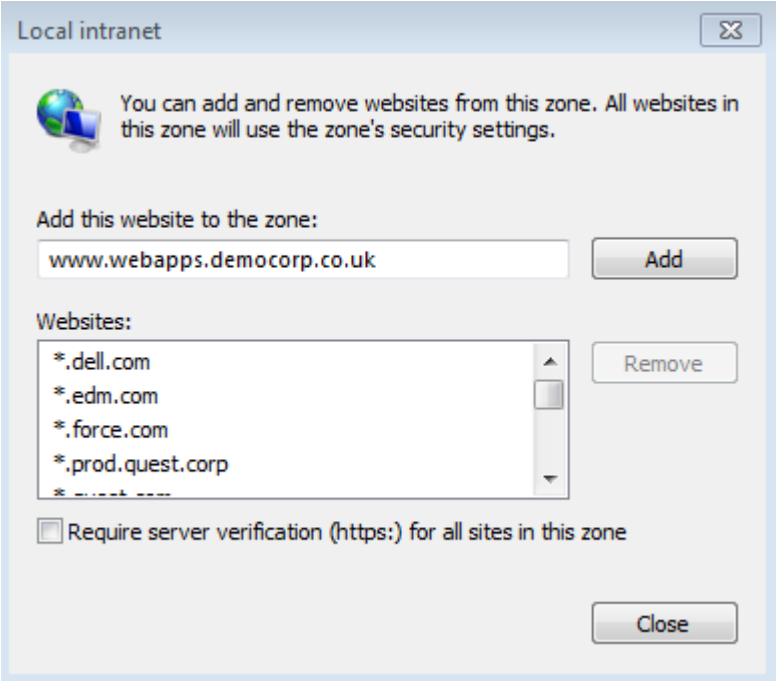
  Where `<hostname>` is the hostname assigned to the proxy for the Cloud Access Manager portal and `<account>` is the name of the service account specified during the Cloud Access Manager installation.

  For example:

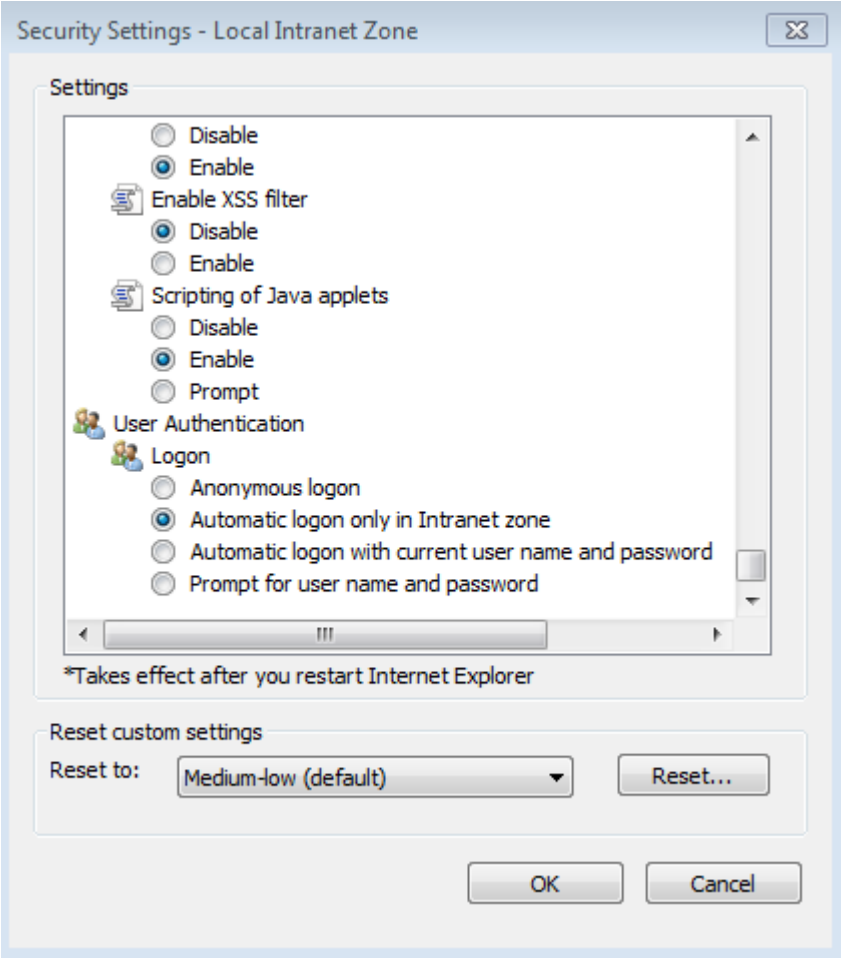


Configuring Microsoft Internet Explorer to single sign-on to the Cloud Access Manager portal

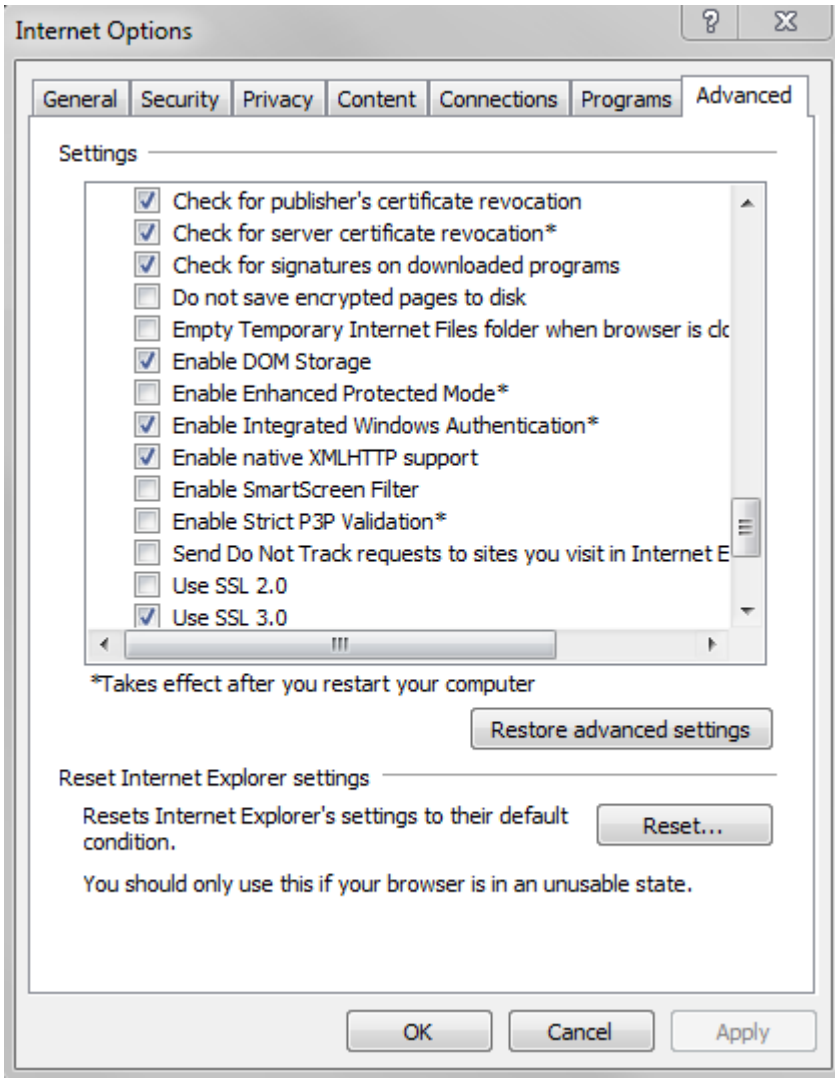
1. Open the Cloud Access Manager application portal using Internet Explorer.
2. Make a note of the Cloud Access Manager application portal hostname displayed in the address bar.
3. Add Cloud Access Manager to Internet Explorer's Local intranet zone. To do this, click **Tools** |**Internetoptions** | **Security** | **Local Intranet** | **Sites** | **Advanced**. Verify that the website address displayed matches the Cloud Access Manager application portal address noted in Step 2, then click **Add**.



4. Next, click **Tools | Internet options | Security | Local Intranet | Custom level** to verify that the local Intranet zone has the default logon option of **Automatic logon only in Intranet zone** selected.



5. Check that **Enable Integrated Windows Authentication** is selected in the Internet Explorer **Advanced** panel.



6. Close Internet Explorer.
7. You should now be able to access Cloud Access Manager without providing a username and password in Internet Explorer.

Configuring Google Chrome to single sign-on to the Cloud Access Manager portal

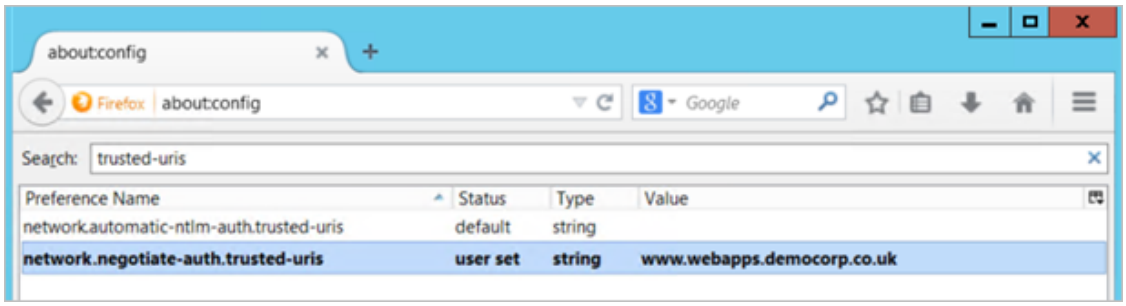
1. Open the Cloud Access Manager application portal using Google Chrome.
2. Make a note of the Cloud Access Manager application portal hostname displayed in the address bar.
3. Add Cloud Access Manager to the Google Chrome local intranet zone. To do this, click the Chrome menu **Customize and control Google Chrome | Settings | Show Advanced Settings... | Network | Change proxy settings...** This will open Internet Explorer's **Tools |Internet options | Connections**.

   NOTE: Google Chrome uses Internet Explorer's Integrated Windows Authentication settings for Kerberos configuration.

4. Next, click **Security | Local Intranet | Sites | Advanced**. Verify that the website address displayed matches the Cloud Access Manager application portal address noted in Step 2, then click **Add**.
5. Next, follow from Step 4 in the section Configuring Microsoft Internet Explorer to single sign-on to the Cloud Access Manager portal.
6. Close Google Chrome.
7. You should now be able to access Cloud Access Manager without providing a username and password in Google Chrome.

Configuring Mozilla Firefox for Integrated Windows Authentication

1. Open the Cloud Access Manager application portal using Firefox.
2. Make a note of the hostname displayed in the address bar.
3. Type **about:config** into the address bar and press **enter**.
4. Search for the setting `network.negotiate-auth.trusted-uris`
5. Double click the `network.negotiate-auth.trusted-uris` setting and edit it by entering the hostname of the Cloud Access Manager application portal noted in Step 2. If the setting already contains a value, separate the existing and new values with a comma. The value entered into this setting should contain only the hostname of the application portal and should not contain the https:// protocol prefix or the portal path.



6. Close Firefox.
7. You should now be able to access Cloud Access Manager without providing a username and password in Firefox.

Configuring smart card authentication

Cloud Access Manager can be configured to require your Microsoft Active Directory users to present an X.509 Certificate as a means of authentication. The certificate can be stored on a standards-based smart card or in the user's local certificate store.

NOTE: You must be logged in to Cloud Access Manager using the fallback admin login if you wish to enable smart card authentication.

To configure smart card authentication

1. On the **Primary Authentication** page ensure that **Enable smart card authentication** is selected. In this example we will not allow other authentication methods.

   NOTE: If smart card authentication is enabled, users primary credentials will not be populated on user login. Users will need to populate their primary credentials manually in a one time task, either in the Cloud Access Manager Password Wallet, or when they first access an application that requires their primary credentials. This applies to other Front-end Authenticators that do not provide a user password on login, for example Kerberos authentication or SAML federation.

2. Select the **Enable certificate revocation list checking** box to cause Cloud Access Manager to check the Certificate Authority's Certificate Revocation List (CRL) to ensure the user's certificate has not been revoked. If the user's certificate has been revoked, the login request will be denied.

   NOTE: Cloud Access Manager will use the CRL location defined within the user's certificate to check whether it has been revoked. The location can be found in the certificate field "CRL Distribution Points". The STS host must have network connectivity to the host specified in this field in order to perform a CRL check. If the host is not contactable then all X.509 certificate authentications will fail if this option is switched on.

   NOTE: If certificate revocation list checking is enabled then the root CA certificate must be installed in the Local Computer\Trusted Root Certification Authorities folder in the Windows certificate store of all Cloud Access Manager STS hosts. Refer to Microsoft documentation on how to use the Certificates.msc snap-in to import certificates into the Windows certificate store. If you are using the Microsoft Certificate Authority the root CA certificate will be installed automatically on all hosts within the domain.

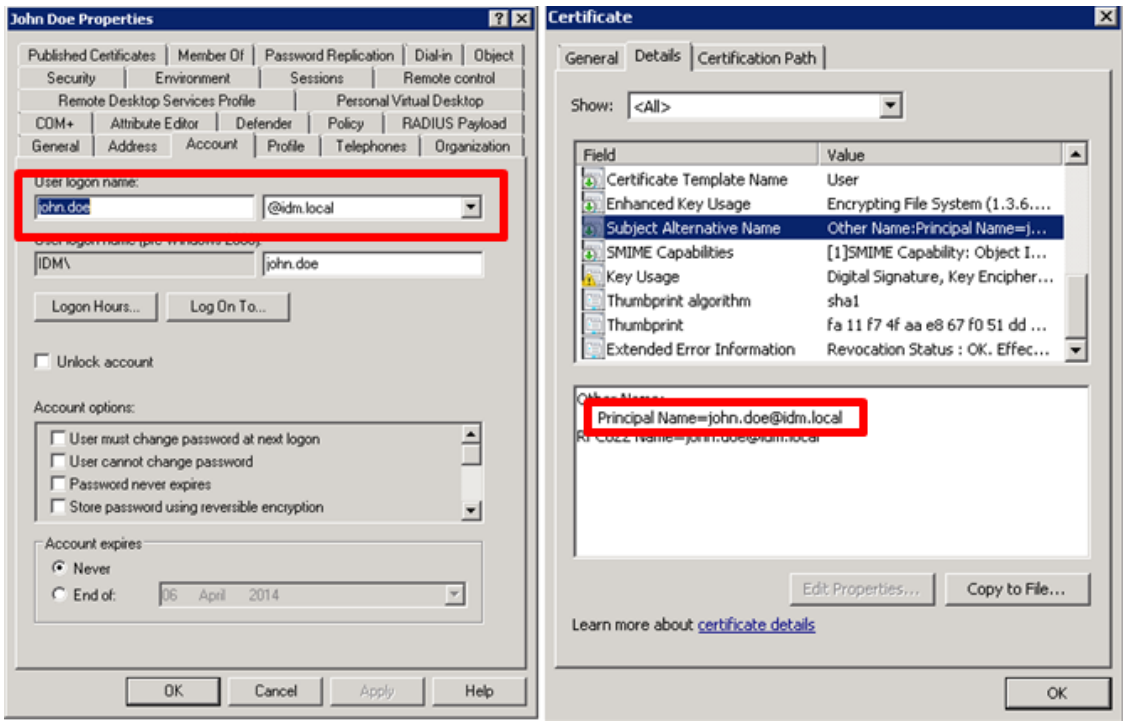
   Cloud Access Manager must redirect the user's browser to another port in order to perform an X.509 certificate authentication, the default port used is 8443. You can change this if 8443 is being used by another service on the Cloud Access Manager host. Your firewall(s) will also need to be updated to allow users to access the proxy hosts using 8443 in addition to the standard ports 80 and 443.

3. Export the certificate from your Certificate Authority in .pem or base-64 encoded format, copy it to the Cloud Access Manager STS host and upload it using the **Choose File** control on this page. In order to perform X.509 certification authentication, the public signing certificate of the root Certificate Authority must be uploaded into Cloud Access Manager.
4. Follow the front-end authentication wizard to completion.

Configuration of smart card authentication for Microsoft Active Directory front-end authentication is now complete.

To test your smart card authentication setup

1. Follow Microsoft documentation to enroll a certificate for a user in your Active Directory forest.
2. Open the certificate and verify that the Subject Alternative Name contains a Principal Name value that matches the user's User Principal Name (UPN) defined in Active Directory. This certificate value is used to locate the user in Active Directory.
3. To determine the User Principal Name, open the user object in **Active Directory Users and Computers**.
4. Click the **Account** tab and concatenate the User logon name with the domain name that follows it, as shown below. Click **Next**.



LDAP authentication

To configure LDAP front-end authentication

1. Log in to the Administration Console and select **Add New** from the **front-end Authentication** section on the home page.
2. Select **LDAP**, then click **Next**. The **Connection Settings** page is displayed.
    a. In the **Comma separated list with optional ports** field, enter either a single host or a comma separated list of hosts, including the port. The default port is 389 (or 636 if the **Use secure LDAP?** box is selected). If you want to use a different port, it should be explicitly specified on a per host basis.

b. In the **DN of User to Bind With** field, enter a user account to use to connect to the directory, for example,

   cn=administrator,cn=users,dc=company,dc=com

   The account must have the appropriate privileges to allow it to read user and group information from the directory.

c. In the **Object Class of Users** field and **Object Class of Groups** field, enter appropriate object classes to distinguish users and groups.

d. In the **Attribute Mappings** section, complete the attributes as required. The **User's Unique ID Attribute** field and **Group's Unique ID Attribute** field are usually left empty. They will default to the object's distinguished name.

   NOTE: The **User's Unique ID Attribute** is used to link to the **Group's Members Attribute**, so it is important that they are in the same format.

e. When you have entered the required configuration information, click **Test Connection** to verify the configuration. Click **Next.**

3. The settings on the **Primary Authentication** screen are split into two sections. When complete, click **Next**.

   a. **Store credentials from this authenticator as primary credentials** — Determines whether the user's credentials are stored for accessing other applications. If the box is selected, the credentials used to authenticate to Cloud Access Manager are stored as the Primary Credentials in the user's Password Wallet. Please refer to Primary credentials on page 37 for details.

   b. **Enable social authentication** — Determines whether users are allowed to use social authenticators, for example Facebook or Google, and link to the selected authenticator when authenticating to Cloud Access Manager.

   NOTE: If you enable social authentication, it may be beneficial to set linked accounts to have a long password expiry, this allows seamless authentication using the social authenticator.

   NOTE: If you enable social authentication, storing credentials from the authenticator is required, this in turn requires that forms authentication is the only enabled authentication method. Storing credentials is required as Cloud Access Manager needs to verify if the linked account used for primary authentication is still valid, for example the account is not disabled, or the password has not expired when authenticating using a social authenticator. If a user attempts to authenticate with a social authenticator and the linked account is not valid, the user will be prompted to enter the correct credentials for the primary authenticator.

4. Click **Next**.

5. If you require two factor authentication each time users authenticate to Cloud Access Manager, select **Use two factor authentication for all applications** from the **Two factor authentication mode** list. Select the method of authentication from the **Type of two factor authentication** list.

For information on how to configure the various authentication types or how to configure two factor authentication only for specific users or applications, refer to the Configuring step-up authentication section. When compete, click **Next**.

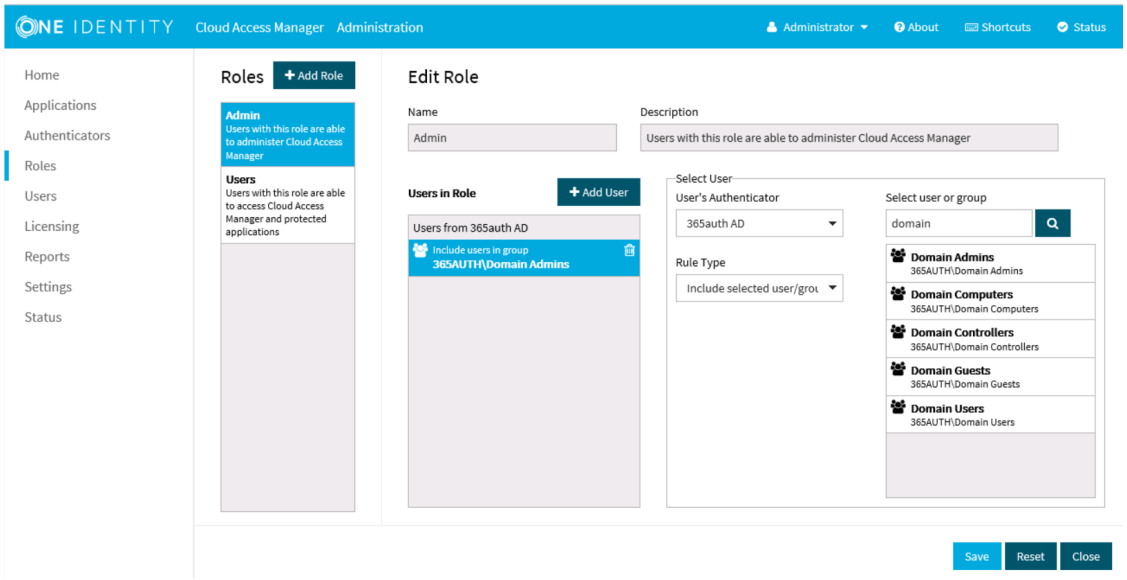
6. In the **Authenticator Name** field, enter the name that will be used to identify the authenticator within Cloud Access Manager, then click **Finish**.

   NOTE: This name will be seen by Cloud Access Manager users during authentication if multiple authentication methods have been configured.

7. You have now created the front-end authentication method. Click **Edit Roles**.

   Before Cloud Access Manager administrators and users can log in to Cloud Access Manager using their directory credentials, you must tell Cloud Access Manager how to identify administrators and users based on their directory group membership. For example the admins group for Cloud Access Manager administrators and the users group for regular Cloud Access Manager users.

8. Click **Admin**.
9. Click +**Add User**.



10. Select the new Active Directory authentication method from the list.
11. Use the search box to locate the group whose members are to be granted access to the Cloud Access Manager Administration application, then select the group from the list.
12. Click **Save**.
13. Click **Users**, now repeat Step 9 through Step 12 for Cloud Access Manager users.
14. Click **Close** to return to the Cloud Access Manager Administration Console. The configuration is now complete. Cloud Access Manager administrators and users can now log in to Cloud Access Manager using their directory credentials.

Microsoft Active Directory LDS

This option is an LDAP authenticator with pre-configured attributes for use with Microsoft Active Directory Lightweight Directory Service, please see LDAP authentication on page 17, for configuration options.

389 Directory Service

This option is an LDAP authenticator with pre-configured attributes for use with 389 Directory Service, please see LDAP authentication on page 17, for configuration options.

Novell eDirectory

This option is an LDAP authenticator with pre-configured attributes for use with Novell eDirectory, please see LDAP authentication on page 17, for configuration options.

Windows Azure Active Directory authentication

This section describes how you can configure Cloud Access Manager to use Windows Azure Active Directory for authentication. Before you begin you need to configure an application for Cloud Access Manager so it can authenticate users and obtain lists of users and groups. In addition:

- The Azure Active Directory Provider currently supports a single domain and single tenant application in Azure Active Directory, you should have an Active Directory instance created in your Microsoft Azure cloud subscription.
- The free version of Azure Active Directory can be used, but the Azure Active Directory self-serve password management will not be available. In all cases, newly-created users must not have the **Force Change Password On Next Login** set.
- Users must login with their full Azure Active Directory username, which should resemble an email address containing your full Azure Active Directory domain name. Azure Active Directory domain names should be of the form, <name>.onmicrosoft.com. Any external users added to your Azure Active Directory cannot be authenticated.

NOTE: A newly-created Azure Active Directory user account is assigned a temporary password. For the user to access Cloud Access Manager, they must first change their password so that it is no longer expired. Cloud Access Manager does not accept logons from users with expired passwords.

To configure an Azure Active Directory application for use with Cloud Access Manager

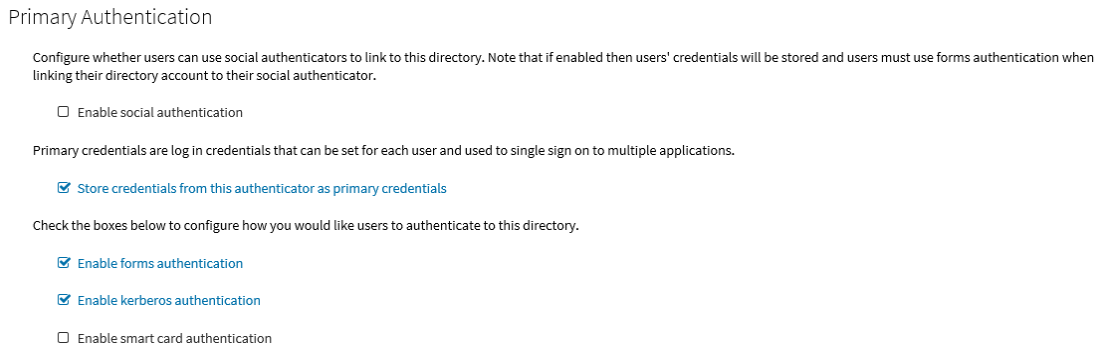
1. Log in to the Azure management portal.
2. From the dashboard navigate to the directory management page.
3. Select the **Applications** tab.
4. Click **Add** in the bottom toolbar and in the dialog that appears, select **Add an application my organization is developing.**
5. Enter a name for the application, this is simply a description. Ensure that **WEB APPLICATION AND/OR WEB API** is selected and click the arrow to move to the next page.
6. Enter http://CloudAccessManager in both the **SIGN-ON URL** and **APP ID URI** fields (these values are not used). Click the tick button to complete the dialog.
7. When you have created the application, select the **Configure** tab and scroll down until you see the **CLIENT ID** field, you will need to copy and save the value here to use when configuring Cloud Access Manager.
8. Scroll down to the **permissions to other applications** section, click the **Application Permissions** list and check **Read directory data**.
9. Scroll back up to the **keys** section, select the required duration for the key and click **SAVE**. At this point you **MUST** copy and save the key value as it will not be available again and you need it to configure Cloud Access Manager.
10. Click **VIEW ENDPOINTS** in the bottom toolbar, copy and save the **MICROSOFT AZURE AD GRAPH API ENDPOINT** and **OAUTH 2.0 TOKEN ENDPOINT** values.

To configure Azure Active Directory authentication

1. Log in to the Administration Console and select **Add New** from the **Front-end Authentication** section on the home page.
2. Select **Azure Active Directory**, then click **Next**. The **Connection Settings** page is displayed.
    a. In the **Client ID** field, enter the client ID from the Azure portal.
    b. In the **Application Key** field, enter the key that you created in the Azure portal.
    c. In the **Windows Azure AD Graph API Endpoint** field, enter the Windows Azure AD Graph API Endpoint from the Azure portal App Endpoints page.
    d. In the **OAuth 2.0 Token Endpoint**, enter the OAuth 2.0 Token Endpoint from the Azure portal App Endpoints page.
    e. When you have entered the required configuration information, click **Test Connection** to verify the configuration. Click **Next**.

3. The settings on the **Primary Authentication** screen are split into three sections. When complete, click **Next**.

   The first section is used to determine whether or not users are allowed to use social authenticators, for example Facebook or Google, and link to the selected authenticator when authenticating to Cloud Access Manager.



   The second section determines whether users' credentials are stored for accessing other applications. If selected, the credentials used to authenticate to Cloud Access Manager are stored as the Primary Credentials in the user's **Password Wallet**. Please refer to Primary credentials on page 37 for details.

   The third section is used to determine how users are challenged for their Windows credentials, you must choose at least one option. Cloud Access Manager checks for credentials presented in the following order of precedence:

   a. **Enable kerberos authentication** — Cloud Access Manager will check for a Kerberos ticket generated during Windows domain login and supplied by the browser. If the Kerberos ticket is present and valid, then the user will be successfully logged in.

      Successful Kerberos authentication requires correct configuration of the user's browser. Please refer to Microsoft Active Directory authentication on page 7 for details. In addition some browsers do not support Kerberos authentication. Please refer to the *One Identity Cloud Access Manager Installation Guide* for browsers that support Integrated Windows Authentication.

   b. **Enable smart card authentication** — Users are given the opportunity to present an X.509 certificate in order to log in to Cloud Access Manager. The X.509 certificate may be located on a smart card or in the client computer's certificate store. If the certificate is invalid or expired the login attempt will be rejected. Please refer to the section Configuring smart card authentication on page 15 for details.

   c. **Enable forms authentication** — Users are prompted for their Azure Active Directory username and password using a login form.

NOTE: If you enable social authentication, storing credentials from the authenticator is required, this in turn requires that forms authentication is the only enabled authentication method. Storing credentials is required as Cloud Access Manager needs to verify if the linked account used for primary authentication is still valid, for example the account is not disabled, or the password has not expired when authenticating using a social authenticator. If a user attempts to authenticate with a social authenticator and the linked account is not valid, the user will be prompted to enter the correct credentials for the primary authenticator.

NOTE: If you enable social authentication, we recommend that you set linked accounts to have a long password expiry, this allows seamless authentication using the social authenticator.

4. If you require two factor authentication each time users authenticate to Cloud Access Manager, select **Use two factor authentication for all applications** from the **Two factor authentication mode** list. Select the method of authentication from the **Type of two factor authentication** list.

   For information on how to configure the various authentication types or how to configure two factor authentication only for specific users or applications, refer to Configuring step-up authentication on page 83. When compete, click **Next**.

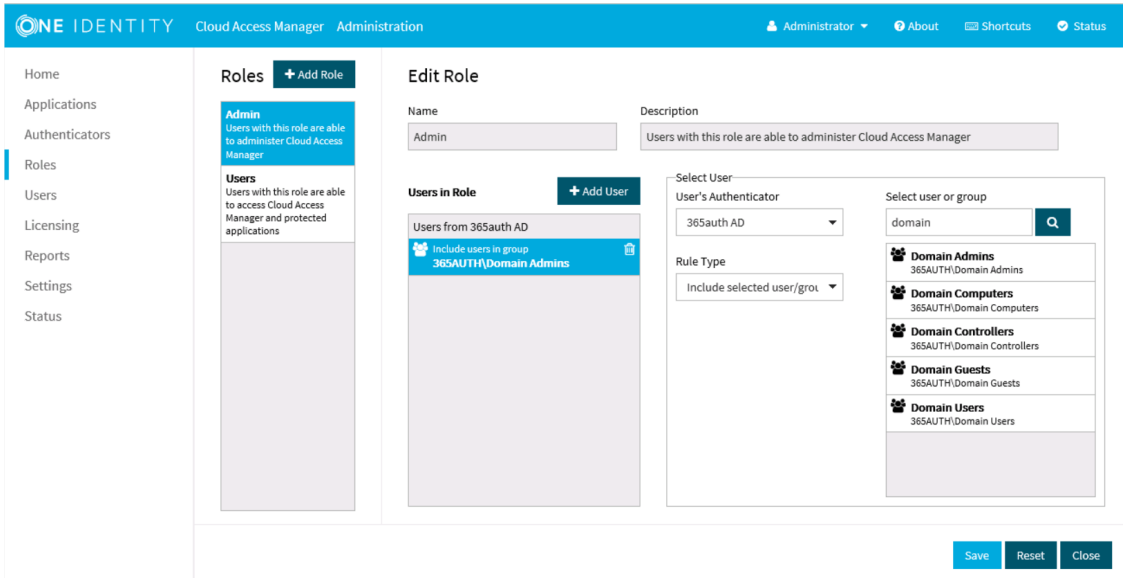
5. In the **Authenticator Name** field, enter the name that will be used to identify the authenticator within Cloud Access Manager, then click **Finish**.

   NOTE: This name will be seen by Cloud Access Manager users during authentication if multiple authentication methods have been configured.

6. You have now created the front-end authentication method. Click **Edit Roles**.

   Before Cloud Access Manager administrators and users can log in to Cloud Access Manager using their Azure Active Directory credentials, you must tell Cloud Access Manager how to identify administrators and users based on their Azure Active Directory group membership. For example, the Domain Admins group for Cloud Access Manager administrators and the Domain Users group for regular Cloud Access Manager users.

7. Click **Admin**.
8. Click +**Add User**.



9. Select the new Azure Active Directory authentication method from the list.
10. Use the search box to locate the group whose members are to be granted access to the Cloud Access Manager Administration application, then select the group from the list.
11. Click **Save**.
12. Now repeat the process for the Cloud Access Manager users. Click **Users**.
13. Click +**Add User**.
14. Select the new Azure Active Directory authentication method from the list.
15. Select a group from the list whose members are to be granted access to the Cloud Access Manager application portal.
16. Click **Save**.

Click **Close** to return to the Cloud Access Manager Administration Console. The configuration is now complete. Cloud Access Manager administrators and users can now log in to Cloud Access Manager using their Azure Active Directory credentials

SAML federated

This example uses Microsoft AD FS v2 to federate users using SAML 2.0. However, it should be possible to use any SAML 2.0 compliant IDP.

To configure SAML federated authentication

1. Log in to the Administration Console and select **Add New** from the **Front-end Authentication** section on the home page.

NOTE: If SAML federation is enabled, users primary credentials will not be populated on user login, this is because SAML tokens do not include the password. Users will need to populate their primary credentials manually in a one time task, either in the Cloud Access Manager Password Wallet, or when they first access an application that requires their primary credentials. Please note, this applies to other Front-end Authenticators that do not provide a user password on login, for example Smart card authentication or Kerberos authentication.

2. Select **SAML Federated**, then click **Next**.
3. In the **Federation Metadata URL** field, enter the federation metadata URL provided by your IDP. The example URL shown below can be found in **ADFS Management Console | Service | Endpoints** | **Metadata**.

   https://<*Host FQDN*>/FederationMetaData/2007-06/FederationMetaData.xml

   Alternatively, click **Browse** to locate the file containing federation metadata. Refer to your IDP configuration interface for assistance with locating this information.

   If you have entered the federation metadata URL as described above, click **Next** and skip to Step 6.
4. In the **IDP Login URL** field, enter the **Issuer/IDP Service URL**. For ADFS v2, this will be https://<*FederationServiceName*>/adfs/ls/. To find the **Federation Service Name** from the AD FS host, open the **AD FS 2.0** management console and click **Edit Federated Service Properties**.
5. Click **Browse** to locate and upload the contents of the IDP's public signing certificate in Base-64 encoded X.509 (.CER) format.

   For AD FS v2, this certificate is known as the Token-signing certificate. To obtain the certificate from the AD FS host, open the **AD FS 2.0 management console**, click **Service | Certificates**, then double-click the Token-signing certificate. From here, click **Copy to file** to save the certificate.
6. On the **User Identity Claims** page, click **Next**.
7. In the **Authenticator Name** field, enter the name that will be used to identify the authenticator within Cloud Access Manager, then click **Finish.**

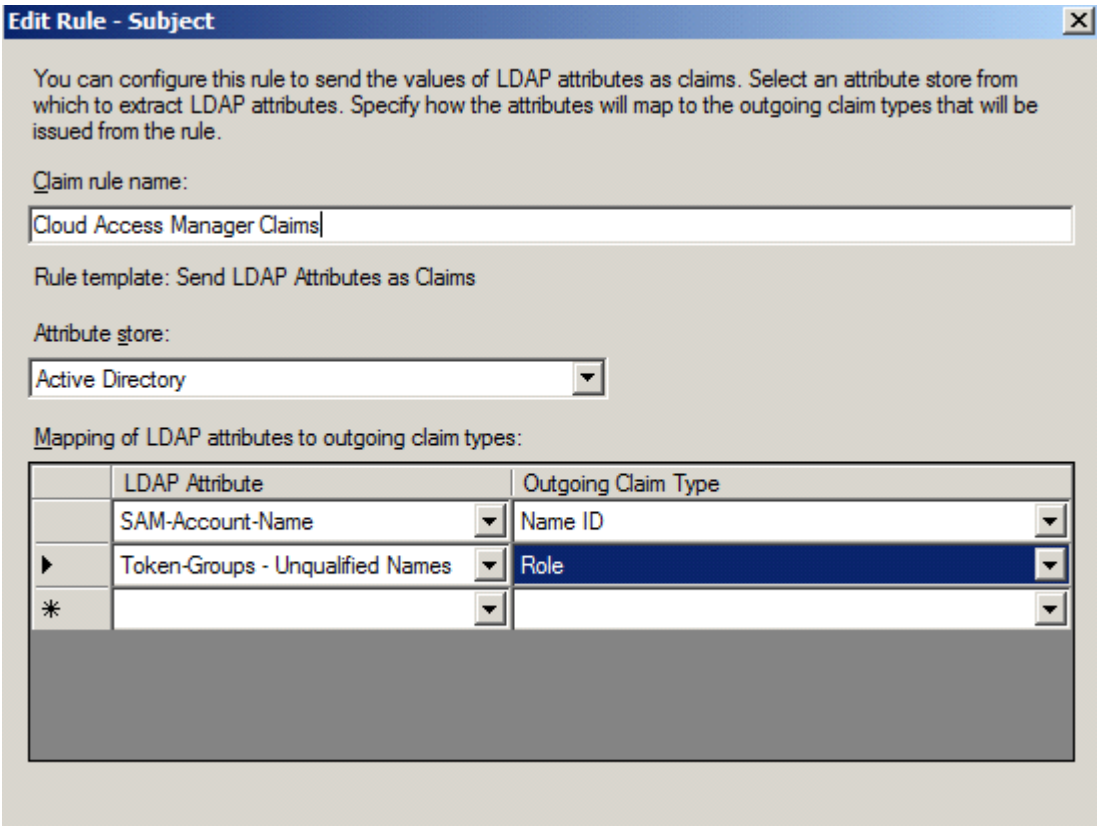
   The **Front-end Authentication Method Created** page is now displayed. Leave this page open.
8. Before you click **Edit Roles**, we will switch to the SAML host (AD FS v2 in this case) and configure Cloud Access Manager as a Relying Party.
9. On the AD FS host, open the **AD FS 2.0** management console and click **Add Relying Party Trust**.
10. Click **Start** to launch the wizard.
11. It is recommended that you use the metadata produced by Cloud Access Manager to configure the trust relationship with the STS.
    - To configure this in AD FS, select **Import data about the relying party published on a local network** and enter the metadata URL shown in the

Cloud Access Manager console.

- Or download the Cloud Access Manager metadata from the console and select **Import data about the relying party from a file** to upload the file to AD FS and click **Next**.

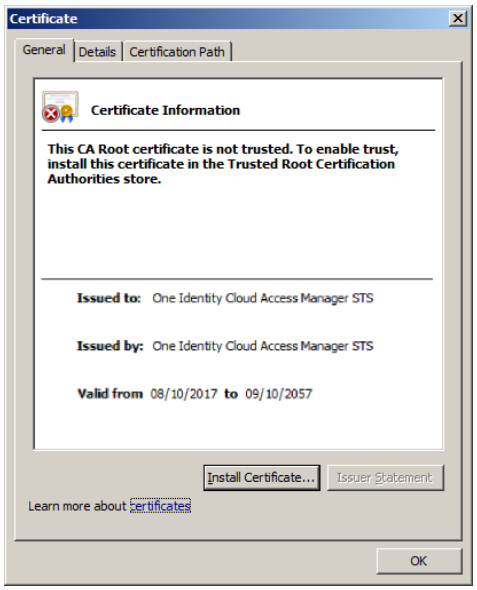
Otherwise, select **Enter data about the relying party manually** and click **Next**.

12. Enter a name for the trust, for example *Cloud Access Manager*. If using metadata skip to Step 18.
13. Select **AD FS 2.0 profile**, then click **Next**.
14. Click **Next** on the **Optional Token Encryption** page to skip this step.
15. Select **Enable support for the SAML 2.0 WebSSO protocol**.
16. Enter the **Relying Party SAML 2.0 SSO service URL**. To find this URL, switch back to the Cloud Access Manager console and copy the Recipient URL. For example, https://www.webapps.democorp.com/CloudAccessManager/RPSTS/Saml2/Login.aspx. Click **Next**.
17. Enter the **Relying party trust identifier**. To find this information, switch back to the Cloud Access Manager console and copy the **Audience/SP Identity value**. For example, urn:CloudAccessManager/RPSTS. Click **Add**, then **Next**.
18. Select **Permit all users to access this relying party**, then click **Next**.
19. Review the configuration information, then click **Next**.
20. Ensure that the **Open the Edit Claim Rules dialog** option is selected and click **Close**.
21. Click **Add Rule**.
22. Select **Send LDAP Attributes as Claims** and click **Next**.
23. Enter a claim rule name, for example *Cloud Access Manager Claims*.
24. Set the Attribute store to **Active Directory**.
25. Select an LDAP Attribute of **SAM-Account-Name** and an Outgoing Claim Type of **Name ID** for the first claim mapping. This claim is required to identify the user to Cloud Access Manager.
26. On the second row, select an LDAP Attribute of **Token-Groups - Unqualified Name** and an Outgoing Claim Type of **Role**.
27. Click **Finish**.



NOTE: By default Cloud Access Manager will perform authorization based on claims of the type Role. If you use a different claim type you will also need to change the claim type within Cloud Access Manager during the Role Mappings configuration.

28. The **Edit Claim Rules for...** page is displayed. Click **OK**.
29. To support single sign out, logout requests from Cloud Access Manager need to be signed. If you use metadata to configure AD FS then the certificate will have been loaded already.
    If you are configuring manually:
    a. Switch back to the Cloud Access Manager console and download the Signing certificate
    b. Copy the certificate to the computer running AD FS
    c. In AD FS view the **Properties** for the newly created Relying Party Trust and switch to the **Signature** tab
    d. Click **Add...** and select the Cloud Access Manager signing certificate.
30. You now need to add the certificate to Trusted Root Certification Authorities so that it is trusted by ADFS. Go to the **Signature** tab in the **ADFS Properties** and click **View...** if the certificate is not trusted you will see a warning similar to the one below.



To allow the certificate to be trusted:

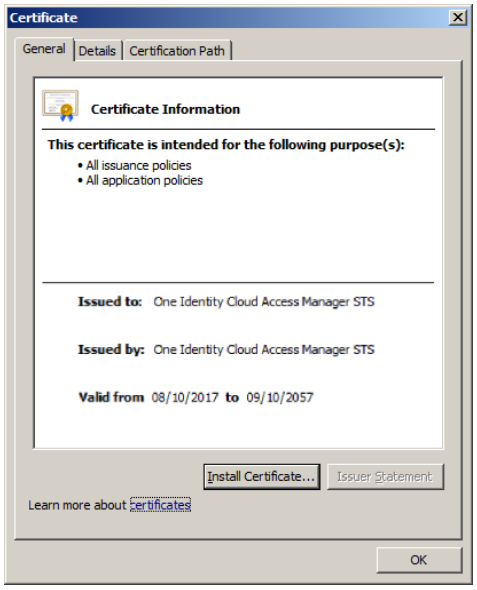
a. Click **Install Certificate...**

b. In the wizard click **Next**, then select **Place all certificates in the following store**.

c. Click **Browse...** and select **Trusted Root Certification Authorities**, click **OK**, then **Next** and **Finish**.

d. When the **Security Warning** dialog is displayed, confirm that the certificate is for Cloud Access Manager STS Trust and click **Yes**.

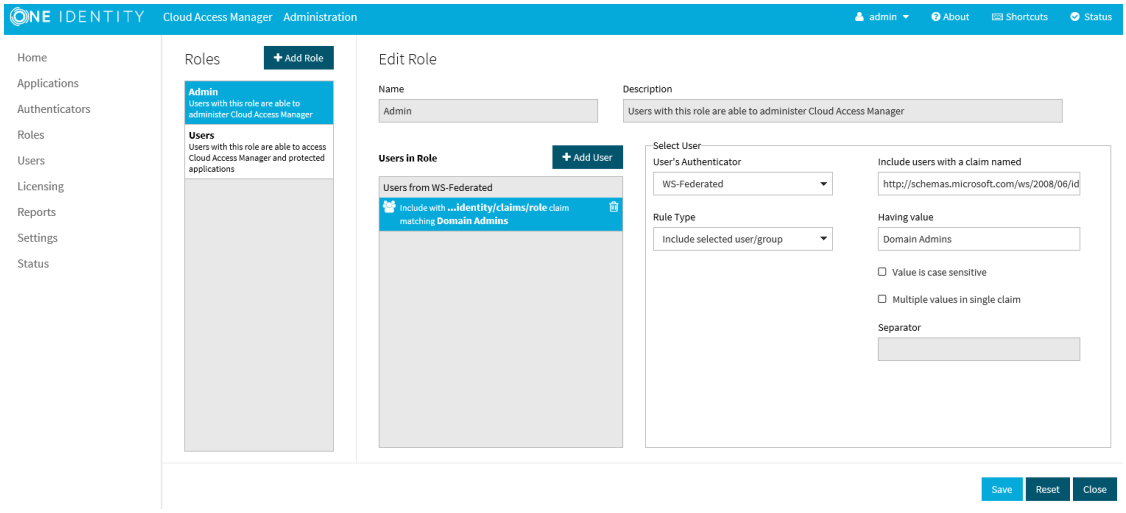
The certificate is now trusted. You can confirm this by closing and re-opening the certificate dialog. The certificate will now appear as below.



31. To complete the configuration you will need to set the hash algorithm for the relying party trust. In AD FS view **Properties** for the newly created Relying Party Trust, switch to the **Advanced** tab and select **SHA-1** from the list of secure hash algorithms. Click **OK**.
32. Configuration of the Relying Party Trust on the AD FS host is now complete. Switch back to the Cloud Access Manager console to complete the configuration.
33. On the **Front-end Authentication Method Created** page, click **Edit Roles**.
34. Before Cloud Access Manager administrators and users can log in to Cloud Access Manager using their federated identity, you must tell Cloud Access Manager how to identify administrators and users based on their claims. For AD FS v2, the claim could be an AD group name in a role claim. For example, the Domain Admins group for Cloud Access Manager administrators and the Domain Users group for regular Cloud Access Manager users.

    NOTE: By default Cloud Access Manager will look for claims of the type *role*. If you configured claims of a different type, update the **Allow users with a claim named** field with the different type.

35. Click **Admin**.
36. Click +**Add User**.
37. Select the new SAML front-end authentication method from the list.
38. Select **Include selected user/group** from the **Rule Type** list.
39. Enter **Domain Admins** into the **Having value** field.
40. Click **Save**.



41. Now repeat the process for the Cloud Access Manager users. Click **User**.
42. Click **Add User**.
43. Select the new **SAML** front-end authentication method from the list.
44. Enter **Domain Users** into the **Having value** field.
45. Click **Save**.

46. Click **Close** to return to the Cloud Access Manager Administration Console.

The configuration is now complete. Cloud Access Manager administrators and users can now log in to Cloud Access Manager using their Active Directory federated credentials. For example, users who belong to the AD *Domain Admins* security group will be able to log in and configure Cloud Access Manager and all domain users will be able to log in to the Cloud Access Manager portal using their federated identity.

NOTE: To fully support logout from AD FS, you must configure AD FS to not use integrated authentication. Once an NTLM connection has been established it will be retained in the browser for its lifetime, and will be used for all connections between the browser and AD FS. Logout from AD FS will appear to work, but on the next connection to AD FS the browser will use the cached connection details and you will be logged on without requiring re-authentication.

WS-Federated

This example uses Microsoft AD FS v2 to federate users using WS-Federation.

To configure WS-Federated authentication

1. Log in to the Administration Console and select **Add New** from the **Front-end Authentication** section on the home page.
2. Select **WS-Federated**, then click **Next**.
3. In the **Federation Metadata URL** field, enter the federation metadata URL provided by your IDP. The example URL shown below can be found in **ADFS Management Console | Service | Endpoints** | **Metadata**.

   https://dc01.qctest.local/FederationMetaData/2007-06/FederationMetaData.xml

   Alternatively, click **Browse** to locate the file containing federation metadata. Refer to your IDP configuration interface for assistance with locating this information.

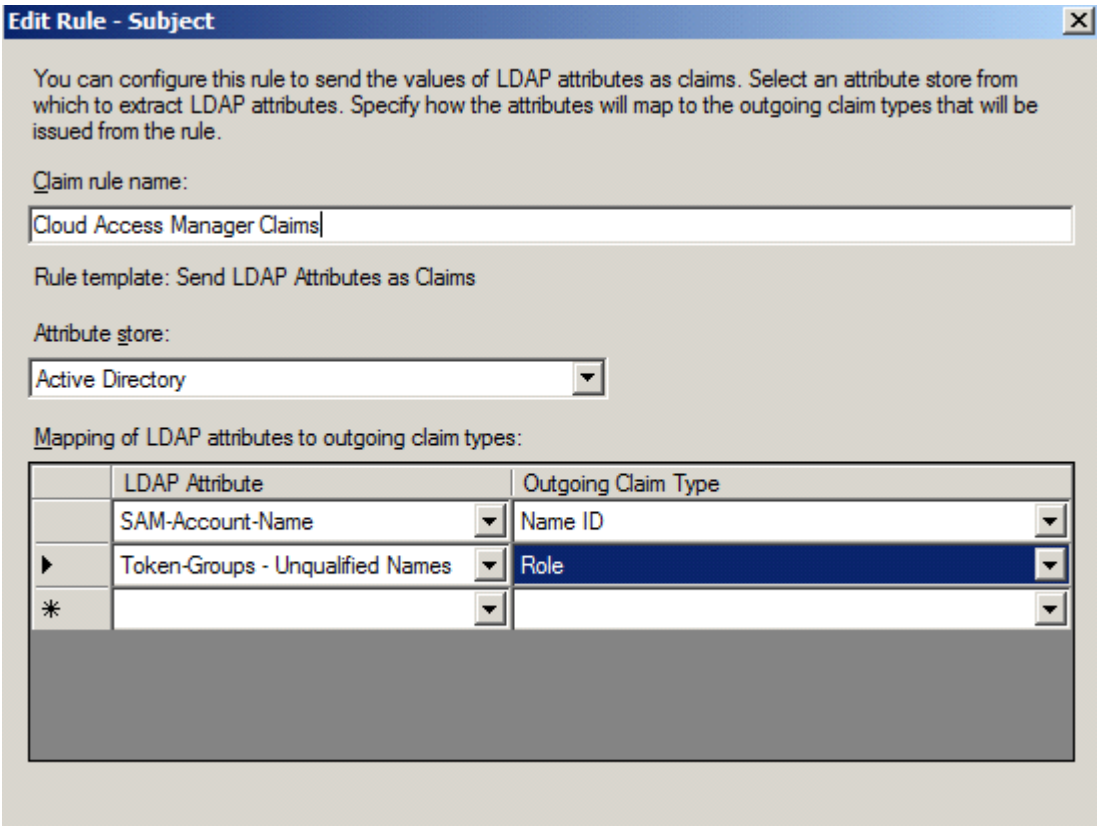
   If you have entered the federation metadata URL as described in this step, skip to Step 6.
4. Enter the **Endpoint URL**. For AD FS v2 this will be https://<*FederationServiceName*>/adfs/ls/ . To obtain the **Federation Service Name** from the AD FS host, open the **AD FS 2.0** management console and click **Edit Federated Service Properties**.
5. Click **Browse** to locate and upload the contents of the IDP's public signing certificate in Base-64 encoded X.509 (.CER) format.

   For AD FS v2 this certificate is known as the Token-signing certificate. To obtain the certificate from the AD FS host, open the AD FS 2.0 management console, click **Service Certificates**, then double-click the Token-signing certificate. Click **Copy to file** to save the certificate.

6. From the **WS-Federation Trust Settings** page which you configured earlier, click **Next**.
7. The **User Identity Claims** page is displayed. Click **Next**.
8. In the **Authenticator Name** field, enter the name that will be used to identify the authenticator within Cloud Access Manager, then click **Finish**.
9. The **Front-end Authentication Method Created** page is displayed. Leave this page open.
10. We will now switch to AD FS and configure Cloud Access Manager as a Relying Party.

    On the ADFS host, open the **AD FS 2.0** management console and click **Add Relying Party Trust**.
11. Click **Start** to launch the wizard.
12. It is recommended that you use the metadata produced by Cloud Access Manager to configure the trust relationship with the STS.
    - To configure this in AD FS, select **Import data about the relying party published on a local network** and enter the metadata URL shown in the Cloud Access Manager console
    - Or download the Cloud Access Manager metadata from the console and select **Import data about the relying party from a file** to upload the file to AD FS, and click **Next**.
13. Enter a name for the trust, for example Cloud Access Manager, then click **Next**. If using metadata skip to Step 19.
14. Select **AD FS 2.0 profile**, then click **Next**.
15. Click **Next** on the optional token encryption page to skip this step.
16. Select **Enable support for the WS-Federation Passive protocol**.
17. Enter the **Relying Party WS-Federation Passive protocol URL**. To find this URL, switch back to the Cloud Access Manager console and copy the Relying Party Endpoint URL. For example, https://www.webapps.democorp.com/CloudAccessManager/RPSTS/WSFed/Login.aspx . Click **Next**.
18. Enter the **Relying party trust identifier**. To find this information, switch back to the Cloud Access Manager console and copy the Relying Party Realm/Identity value. For example urn:CloudAccessManager/RPSTS. Click **Add**, then **Next**.
19. Select **Permit all users to access this relying party** and click Next.
20. Review the configuration information, then click **Next**.
21. Ensure that the **Open the Edit Claim Rules dialog** option is selected, then click **Close**.
22. Click **Add Rule**.
23. Select **Send LDAP Attributes as Claims**, then click **Next**.
24. Enter a claim rule name, for example *Cloud Access Manager Claims*.
25. Set the Attribute store to **Active Directory**.

26. Select an LDAP Attribute of **SAM-Account-Name** and an Outgoing Claim Type of **Name ID** for the first claim mapping. This claim is required to identify the user to Cloud Access Manager.
27. On the second row, select an LDAP Attribute of **Token-Groups - Unqualified Names** and an Outgoing Claim Type of **Role**.



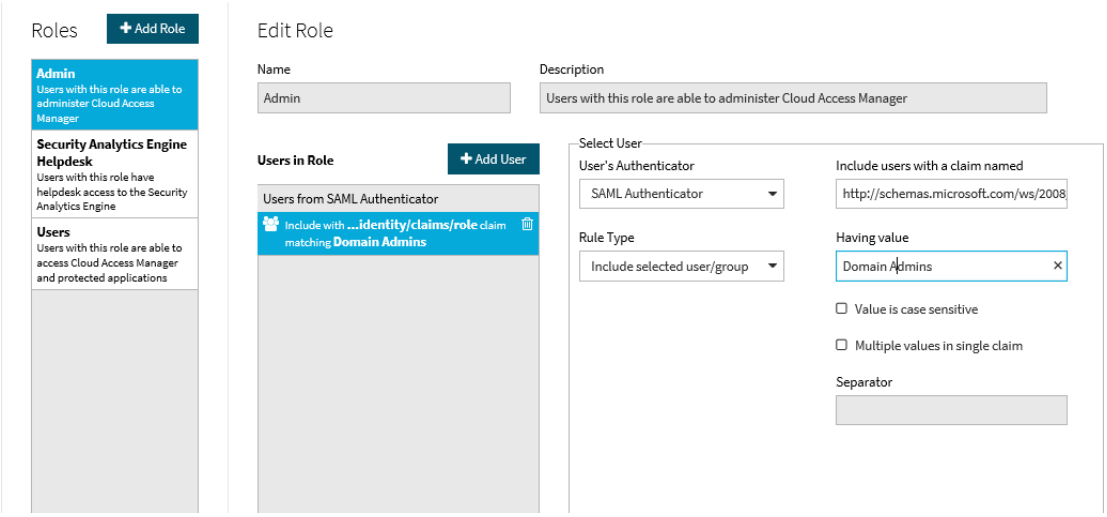
> NOTE: By default Cloud Access Manager will perform authorization based on claims of the type Role. If you use a different claim type, you will also need to change the claim type within Cloud Access Manager during the Role Mappings configuration.

28. Click **Finish**.
29. The **Edit Claims Rules for...** page is displayed. Click **OK**.
30. The Relying Party Trust configuration on the AD FS host is now complete. Switch back to the Cloud Access Manager console to complete the configuration.
31. On the **Front-end Authentication Method Created** page, click **Edit Roles**.
32. Before Cloud Access Manager administrators and users can log in to Cloud Access Manager using their federated identity, you must tell Cloud Access Manager how to identify administrators and users based on their claims. For AD FS v2, the claim could be an Active Directory group name in a role claim. For example, the Domain Admins group for Cloud Access Manager administrators and the Domain Users group

for regular Cloud Access Manager users.

> NOTE: By default, Cloud Access Manager will look for claims of the type *role*. If you configured claims of a different type, update the **Allow users with a claim named** field with the different type.

33. Click **Admin**.
34. Click **+Add User**.
35. Select the new WS-Federated front-end authentication method from the list.
36. Select **Include selected user/group** from the **Rule Type** list.
37. Enter **Domain Admins** into the **Having value** field, then click **Save**.



38. Now repeat the process for the Cloud Access Manager users. Click **Users**.
39. Click **Add User**.
40. Select the new WS-Federated front-end authentication method from the list.
41. Enter **Domain Users** into the **Having value** field, then click **Save**.
42. Click **Close** to return to the Cloud Access Manager Administration Console.

The configuration is now complete. Cloud Access Manager administrators and users can now log in to Cloud Access Manager using their Active Directory federated credentials. For example, users who belong to the Active Directory *Domain Admins* security group will be able to log in and configure Cloud Access Manager, and all domain users will be able to log in to the Cloud Access Manager portal using their federated identity.

> NOTE: To fully support logout from AD FS, you must configure AD FS to not use integrated authentication. Once an NTLM connection has been established it will be retained in the browser for its lifetime, and will be used for all connections between the browser and AD FS. Logout from AD FS will appear to work, but on the next connection to AD FS the browser will use the cached connection details and you will be logged on without requiring re-authentication.

Social authenticators

Social authenticators allow users to link third party authenticators, for example Facebook, Google, Twitter and Microsoft Live ID, with their Cloud Access Manager account.

Social authenticators are presented to users as links on the login page. The first time a user clicks one of these links they will authenticate with the third party web site. On returning to Cloud Access Manager, the user is asked to authenticate using their Cloud Access Manager credentials in order to link the two accounts together.

For future logons, the user will only need to authenticate using either third party credentials or their Cloud Access Manager credentials. The user can unlink the accounts using the **Manage Links** option on the **Navigate Menu** on the Cloud Access Manager home page. The following example uses Microsoft Live ID as the third party authenticator.

NOTE: In order to use Microsoft Live ID as a third party authenticator, a Microsoft account Developer Center application is required.

To configure Microsoft Live ID as a social authenticator

1. Log in to the Cloud Access Manager Administration Console and select **Add New** from the **Front-end Authentication** section on the home page.
2. Select **Microsoft Live Id** under **Social Authenticators**, then click **Next**.
3. On the **Provider Settings** page, complete the **Client Id** field with the client Id for your application in the Microsoft account Developer Center.
4. In the **Shared Secret** field, enter the shared secret for your application in the Microsoft account Developer Center, then click **Next**.
5. In the **Authenticator Name** field, enter the name that will be used to identify the authenticator within Cloud Access Manager, then click **Finish**.

   NOTE: This name will be seen by Cloud Access Manager users during authentication as an alternative authentication option.
6. From the displayed **Provider Settings**, copy the **Redirect URL** and enter it as a redirect URL in the **API Settings** for the application in the Microsoft account Developer Center, then click **Close**.

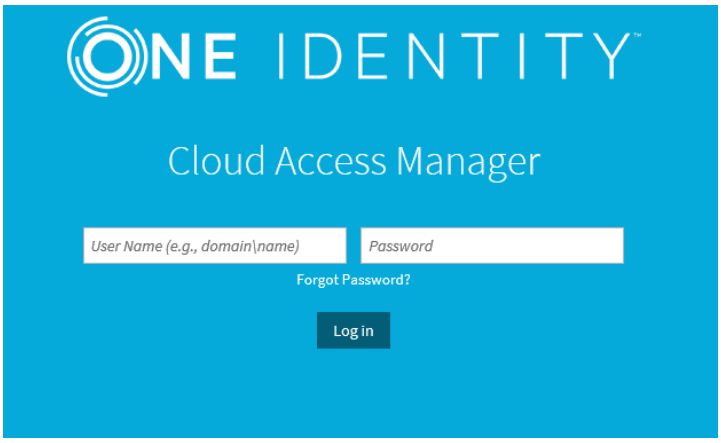
Configuration of Microsoft Live ID as a social authenticator is now complete.

NOTE: Use of social authenticators must be enabled per directory authenticator, please refer to Microsoft Active Directory authentication on page 7 and LDAP authentication on page 17 for details on enabling social authentication for a directory authenticator.

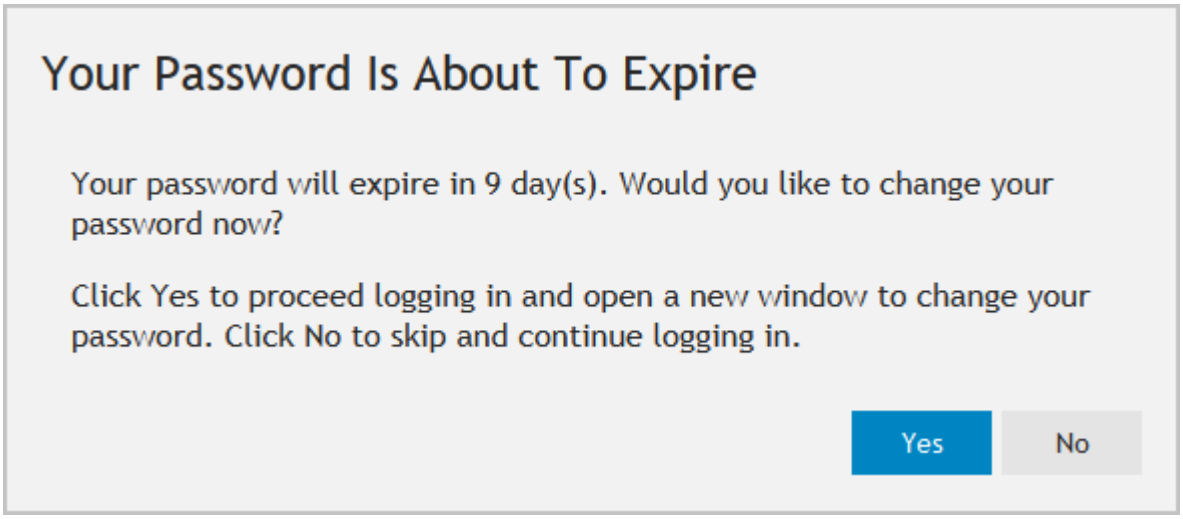
Integration with password management applications

When users authenticate using a directory authenticator, for instance Active Directory or Lightweight Directory Access Protocol (LDAP), Cloud Access Manager can link to password management software to allow users to reset their passwords or unlock their accounts.

If the link between Cloud Access Manager and the password management software is configured, users will see a Forgot Password? link either before authentication, or if authentication fails. In either situation, clicking the link will redirect the user to the password management application.



If password expiry reminders are enabled, and a user's password is due to expire soon, the following dialog is displayed:



Click **Yes** to open a new tab in the browser and load the password management application. If a user specific URL is configured, information about the authenticated user is communicated to the password management application.

To configure a password management application

1. Log in to the Administration Console, navigate to the **Settings** page and then to **Turn Features On/Off**.
2. In the **Password Management Options** section enter the URL in the **URL of password management application** field.

To configure a user specific URL for when the user's password is about to expire

1. Log in to the Administration Console, navigate to the **Settings** page and then to **Turn Features On/Off**.
2. In the **Password Management Options** section enter a URL in the **URL of password management application with user information** field. The following parameters may be inserted into the URL.

Table 1: Password management URL parameters

Parameter	Functionality
`{id}`	The unique identifier for the user
`{username}`	The user part of the user's login name
`{domain}`	The domain part of the user's login name
`{displayName}`	The user's display name
`{emailAddress}`	The user's email address

The following example shows how to pass the domain and user name to Cloud Access Manager; replace password.host with the Domain Name Service (DNS) name of your Password Manager installation:

https://password.host/QPMUser/EntryPoint/?ActionName=ResetPassword&IdentificationDomain={domain}&IdentificationAccount={username}

To configure the user's password expiry alert

1. Log in to the Administration Console, navigate to the **Settings** page and then to **Turn Features On/Off**.
2. In the **Password Management Options** section enter the number of days before the password expiration reminder will be displayed. To prevent users being notified that their password is about to expire, set this value to zero.

NOTE: If you do not enter a user specific URL, but password expiry notifications are enabled, Cloud Access Manager will use the standard password management application URL.

Primary credentials

In many cases, the directory which an application uses to authenticate a user is the same as the directory used by Cloud Access Manager to authenticate the user. In this situation, the username and password entered by the user to sign in to Cloud Access Manager, can be captured and re-used to automate sign in to the application.

Primary Credentials are the username and password that were used to authenticate to Cloud Access Manager. They will only be captured and saved when a login form, through the local built-in identity provider is used to authenticate the user. Authentication using either Kerberos, a smart card, or a federated identity provider will not update Primary Credentials.

Configuring user front-end authentication method selection

If Cloud Access Manager is configured with multiple front-end authenticators, the first time users authenticate through Cloud Access Manager they will have to select which front-end authenticator they want to use. This section describes how you can configure the selection method and what the user will experience.

NOTE: If you do not want users to specify the Home Realm by selecting from a list or entering text to match on, you can add the `ProviderID=authname` parameter to the Cloud Access Manager URL, for example:

https://www.webapps.cam.com/CloudAccessManager?ProviderID=authname

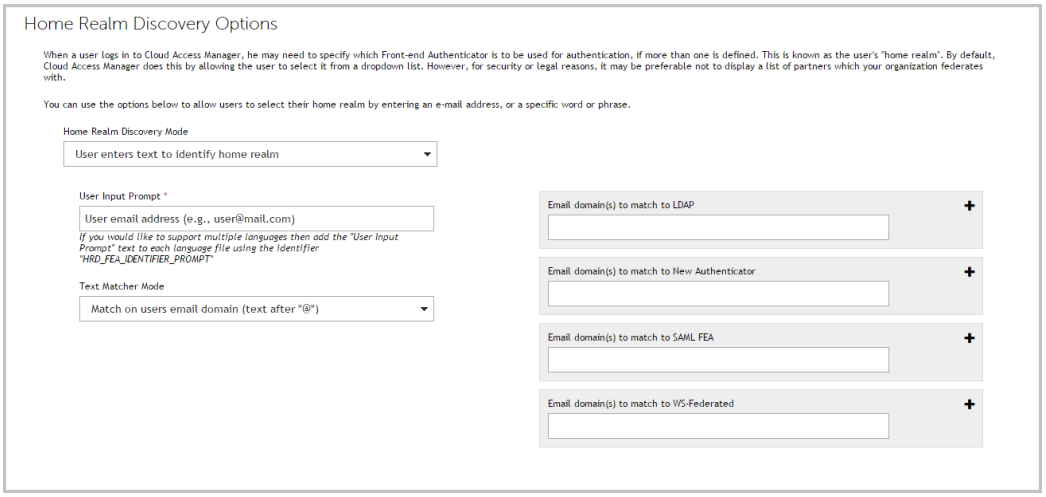
This will send users directly to the User login page with the front-end authenticator already selected.

You can embed the URL containing the `ProviderID=authname` parameter in your existing user portal or distribute it to your users in an email, from which they can create a bookmark.

To configure how the user selects which front-end authenticator to use

1. Log in to the **Administration Console**, navigate to the **Settings** page and click **Home Realm Discovery Options.**
2. Select **Home realm discovery mode**. There are two options with this mode:
    - **User selects home realm from a list** — the user can either select the home realm from a list.
    - **User enters text to identify home realm** — the user can enter text to match against. Typically, this will be an email address, where the domain of the email address is used to determine the front-end authenticator.

If you select **User selects home realm from a list** then no further configuration is required; if you select **User enters text to identify home realm**, you will see the following.



3. In the **User Input Prompt** field, enter the prompt that will be displayed to your users the first time they authenticate through Cloud Access Manager.
4. Select an option from the **Text Matcher Mode** list to determine how the text entered by the user will be matched.
5. If matching is by email domain, enter the domain for each front-end authenticator. To enter multiple domains for each front-end authenticator, click **Add Email Domain**.
6. If matching is on a word or phrase, enter the word or phrase for each front-end authenticator. To enter multiple words or phrases for each front-end authenticator, click the **+** icon.

NOTE: If you need to configure the matching text for a front-end authenticator after the initial configuration, you can either navigate back to the **Home Realm DiscoveryOptions** page in **Settings**, or go to the **Front-end Authentication** page and edit the required front-end authenticator. You will then see an extra **Home Realm Discovery** tab that allows you to edit the matching text for that front-end authenticator. In addition, if you have previously configured home realm discovery to use text matching, you will see the **Home Realm Discovery** tab as part of the wizard when you add a new front-end authenticator.

To always show the Home Realm Discovery choice

By default, the user will only be shown the **Home Realm Discovery** choice the first time they authenticate through Cloud Access Manager.

To show the Home Realm Discovery page each time the user authenticates

1. Navigate to the **Settings** page.
2. Click **Turn Features On/Off**.
3. In the **Log in Options** section, select **Always show front-end authentication**

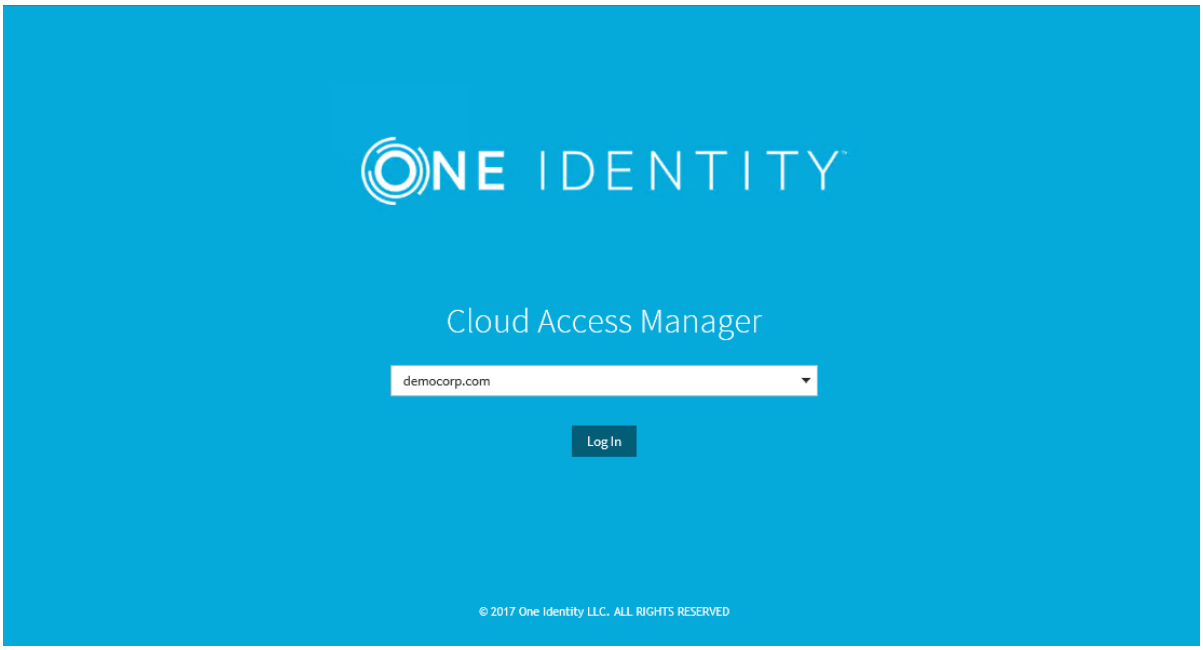
choice.

If this option is selected then the user's previous choice or word/phrase will be displayed the next time they authenticate.

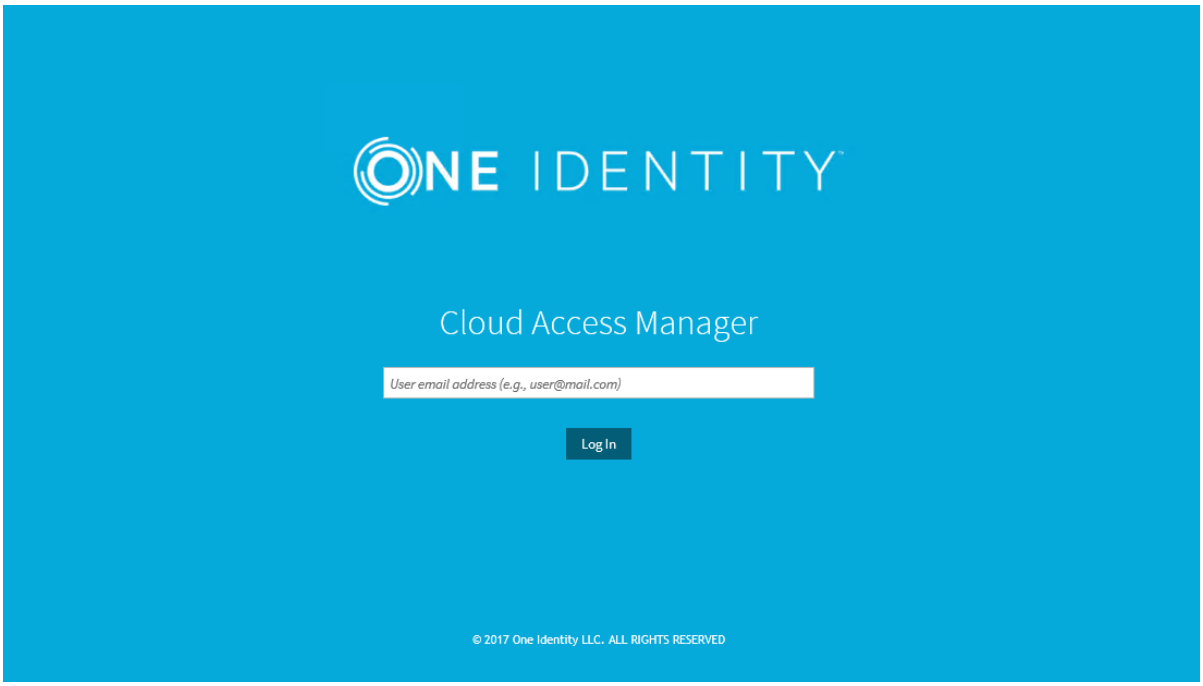
Home Realm Discovery user experience

By default the user will only need to select the authentication method the first time they authenticate through Cloud Access Manager.

If Home Realm Discovery is configured to display a list of front-end authenticators, the user will see a screen similar to that displayed below. The user must select the correct front-end authenticator from the list and click **Log in**. Authentication will then be directed to the selected front-end authenticator.



If Home Realm Discovery is configured to select the front-end authenticator using text matching, the user will see a screen similar to that displayed below.



NOTE: The prompt in the text box is configured in the **Home Realm Discovery** settings page. The user must enter the text as prompted and click **Log in**. Authentication will then be directed to the selected front-end authenticator.

If the default setting for displaying the home realm discovery page is still in place then the next time the user authenticates through Cloud Access Manager, they will be directed straight to the previously selected front-end authenticator.

NOTE: The choice of which front-end authenticator to use is stored in a cookie on the user's browser. If for any reason the user needs to use a different front-end authenticator, they must delete the cookie named CTC_HRD before attempting to authenticate.

Adding a web application

Before adding an application to Cloud Access Manager you must first identify which method of authentication the application is using; the most common methods are Integrated Windows Authentication (IWA) and form fill authentication. The following sections describe how to configure an application for each of the supported authentication methods.

Integrated Windows Authentication

This section will guide you through the steps required to configure single sign-on for One Identity Active Roles which uses Integrated Windows Authentication (IWA).

To configure Integrated Windows Authentication

1. Log in to the Administration Console using the desktop shortcut **Cloud Access Manager Application Portal** and select **Add New** from the **Applications** section on the home page.

   Cloud Access Manager provides a set of application templates to automatically configure common applications. This example describes how to configure an application manually, rather than using a template.
2. Click **Configure Manually**.
3. Select **Integrated Windows Authentication**, then click **Next**.

   NOTE: Additional user attributes can be sent in HTTP headers. In this example, we only need to send the authentication header.
4. Enter the protocol and Fully Qualified Domain Name (FQDN) used by the application you wish to Single Sign-On (SSO). Click **Next**.

   NOTE: The protocol and FQDN can be obtained from the URL used to access the application. For example, if the application is normally accessed through https://ars.prod.local/ARServerAdmin the FQDN would be ars.prod.local and the protocol would be Secure HTTP (HTTPS).
5. In this step, Cloud Access Manager needs to know how to proxy the application.

Typically, this involves configuring Cloud Access Manager to proxy the entire web server used by the application through a new fully qualified domain name. This is the preferred method and the method with the most applications. To configure Cloud Access Manager in this way, simply enter a new public FQDN into the field provided on the **Proxy URL** page, and click **Next**.

The new FQDN should be within the wildcard DNS subdomain created during the Cloud Access Manager installation, which will resolve to the Public IP address used by the proxy. For example, if you created the wildcard DNS subdomain *.webapps.democorp.com during the installation you could use the FQDN owa.webapps.democorp.com to proxy Outlook Web App. If you did not create a wildcard DNS subdomain for Cloud Access Manager during the installation you need to add this new FQDN into your public DNS manually. The new FQDN should be covered by the wildcard SSL certificate you are using.

Alternatively, some applications are installed entirely within their own virtual directory on the web server where they reside. One example of such an application is One Identity Active Roles which installs into the virtual directory /ARServerAdmin. In this case you may be able to configure Cloud Access Manager to proxy the application's virtual directory only, rather than the whole web server, and reuse the FQDN of the proxy. To configure this option, select the proxy's FQDN from the list, then enter the virtual directory where the application is installed into the field below and click **Next**.

NOTE: Take care to ensure that the path entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the path must be ARServerAdmin.

6. You will now see the **Permissions** page, which enables you to control which users can access the application. By default, all Cloud Access Manager users have access to the application. You can restrict access to the application to users who belong to a specific role, but for this example, simply click **Next** to allow all users to access the application.
7. Enter an Application Name, for example ARS.
8. Select **Use primary credentials to log into this application** and click **Next**. This will ensure that ARS uses the user's Active Directory domain credentials rather than a different username or password unique to the application, for example the same credentials that the user used to authenticate to Cloud Access Manager. For applications that require different credentials, make sure this option is deselected.
9. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the **Title** and **Description** you want to display on the Cloud Access Manager Portal. Many applications will require you to configure a particular entry point, for example for Active Roles Server you would need to add ARServerAdmin in the URL field of the **Application Portal** page.

NOTE: Take care to ensure that the URL entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the URL must be ARServerAdmin. In addition the **Add application to application portal home** and **Allow user to remove application from application portal home** options allow you to specify whether the application should automatically appear on each user's portal page and how the user can manage the application from the application portal. The options are shown in the table below.

Table 2: Application portal options

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the application catalog.
✓	✓	application is added to the portal and it can be removed by the user through the application catalog.
✗	✗	application is not automatically added to the portal. The user can add or remove the application to/from the portal through the application catalog.

To access the application catalog from the application portal, the user simply needs to click their username, then select **Application Catalog**. Depending on the settings in the **Add application to application portal home** and **Allow user to remove application from application portal home** options, the user can add or remove applications to/from the application portal.

10. Configuration of the application is now complete. Click **Finish**.

To verify that the application is configured correctly

1. Close Internet Explorer to end your Cloud Access Manager session.
2. Use the desktop shortcut **Cloud Access Manager Application Portal** to open the Cloud Access Manager Portal.
3. Log in to the Cloud Access Manager application portal and click the **Active Roles Server** application.

4. If the user's application credentials, the user's primary credentials in this case, have not yet been stored in Cloud Access Manager you will be prompted to enter them.
5. You will be signed in to One Identity Active Roles automatically.

Configuration of One Identity Active Roles for SSO is now complete.

Further considerations

When you have added an application to Cloud Access Manager, you may want to ensure users only access the application using Cloud Access Manager. This may be required if you use Cloud Access Manager to enforce strong authentication for the application, or want to use Cloud Access Manager's auditing features to monitor application usage. For further information on how to ensure that users access the application using Cloud Access Manager, please refer to *Preventing direct access to applications protected by Cloud Access Manage*r in the *One Identity Cloud Access Manager Security and Best Practices Guide*.

Form fill authentication

This example will guide you through the steps required to configure single sign-on for Microsoft Outlook Web App using the form fill authentication method.

Log in to the Administration Console using the desktop shortcut **Cloud Access Manager Application Portal** and select **Add New** from the **Applications** section on the home page.

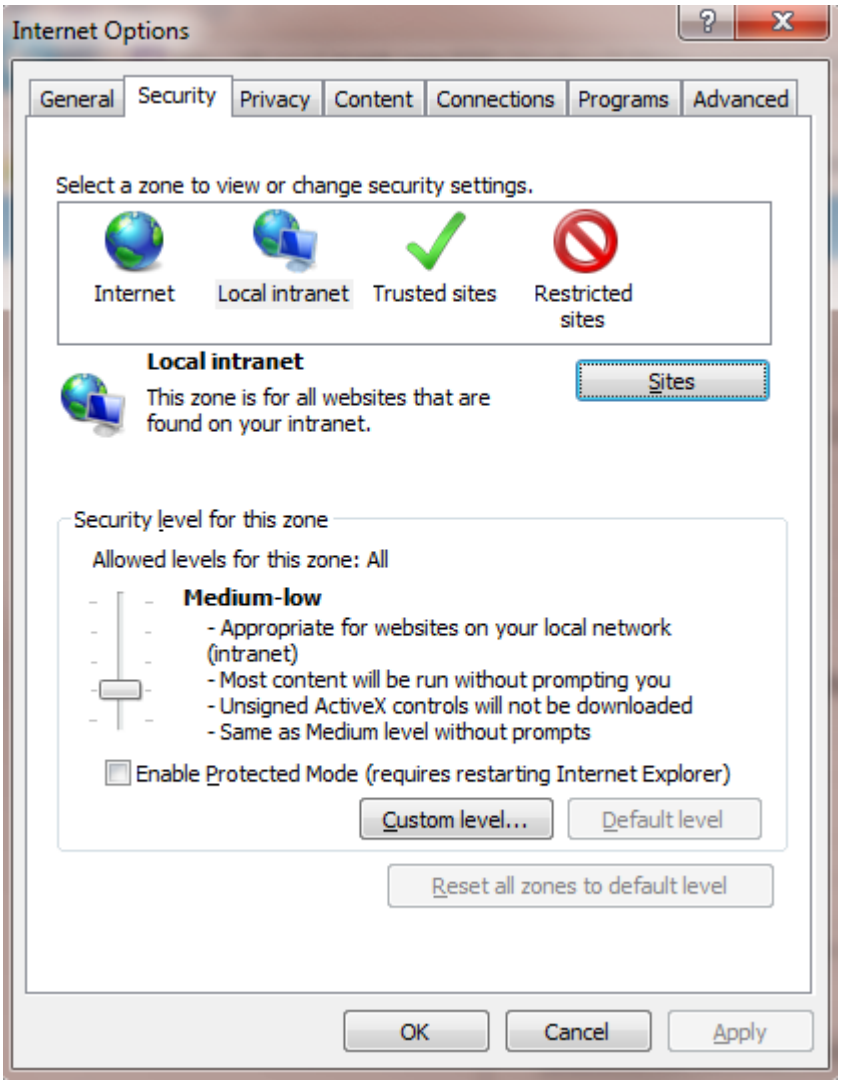
Cloud Access Manager provides a set of application templates to automatically configure common applications. This example describes how to configure an application manually, rather than using a template.

To configure single sign-on for Microsoft OutlookWeb App using form fill authentication

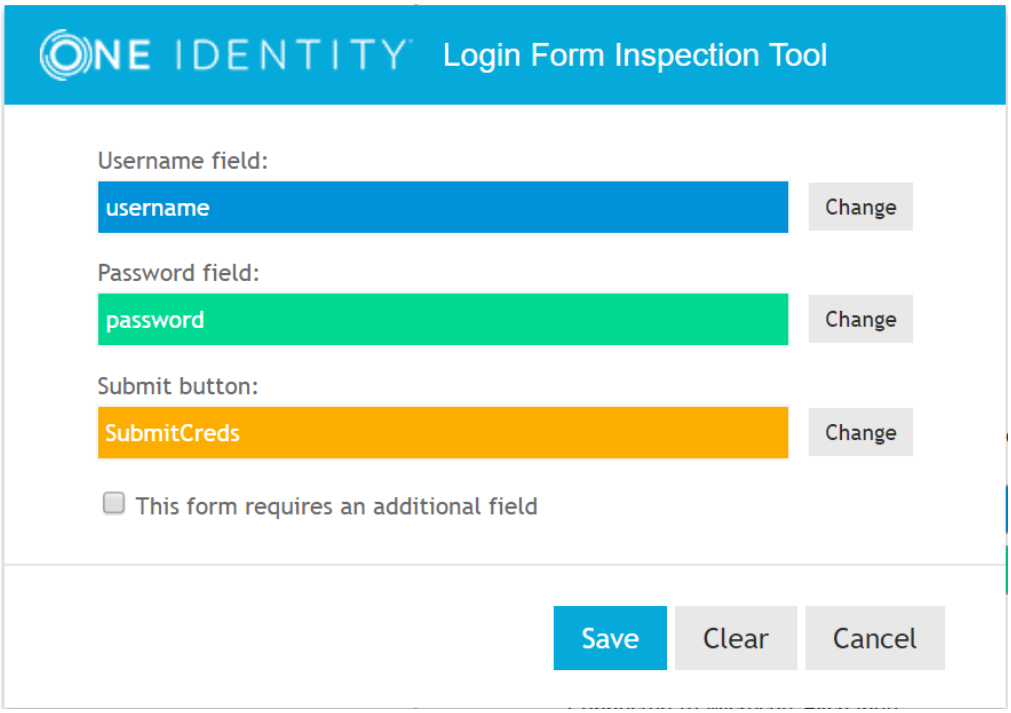
1. Click **Configure Manually**.
2. Select **Form Fill**, then click **Next**.
3. If you have not already done so while adding a previous Form Fill application, save the **Inspect Login Form** bookmarklet to your browser's favorites. To do this, right-click the **Inspect Login Form** link, then click **Add to favorites**.
4. Enter the URL of the application into the box provided and click **Go**. For example, for Microsoft Outlook Web App (OWA) enter https://webmail.prod.local/owa, where webmail.prod.local is the hostname of the host running OWA. This will take you to the application's login page. If you are taken directly to the application, check that you are not already signed in and if necessary, sign out.
5. With the application's login page displayed, click the browser's **Favorites** icon and

click **Inspect Login Form**. The **Cloud Access Manager Login Form Inspection Tool** will now appear in the bottom-right corner of the browser window.

NOTE: If you are using the Cloud Access Manager **Login Form Inspection Tool** in Internet Explorer, your Cloud Access Manager website will need to be in the **Local intranet** zone. This can be selected by going to the **Internet Options | Security** tab in Internet Explorer while viewing your site. If Local Intranet is not highlighted as shown, click **Local intranet |Sites | Advanced** then add your Cloud Access Manager site.



6. Use the tool to obtain the field IDs for the login form. For example, click in the **Username** field, then click in the **Password**field, then finally click the **Submit** button.



7. Click **Save** to save the form IDs and return to the Cloud Access Manager configuration wizard.
8. After using the **Login Form Inspection Tool** to identify the username and password fields and action URL, you are presented with the **Form Fill Method** page. This is where you choose whether or not to proxy the application with Cloud Access Manager, if you choose not to proxy the application Step 11 and Step 12 will not apply. Click **Next**.
9. Review the detected **Form Fill Details**.

   NOTE: If the application displays the password field on a separate page to the username field, check the box titled **The password field is located on a separate page**. You will then be able to manually enter the field identifiers for the password field and submit button.

10. For OWA, leave the **Form Fill URLs** with their detected values. Click **Next**.

NOTE: Some applications use URLs where only the query string portion of the URL changes when navigating between pages. For example, pages in an Oracle application may only differ by a function id in the query string. The home page might have the ID of 150, for example https://server/OA_HTML/RF.jsp?functionId=150 and the login page an ID of 200, for example https://server/OA_HTML/RF.jsp?functionId=200.

To configure this type of application you need to select the box labelled **Information in the query string is required to identify the login page of the application**. Cloud Access Manager will then allow you to select the query string parameter that identifies the login page, for example the **functionId=200** parameter used in the previous Oracle example. If an application uses multiple query string parameters, only check the parameters that identify the login page. For example, some applications use additional parameters to store information unique to a particular user or access attempt. These parameters should not be selected as they would prevent the login page being detected for all users/requests.

NOTE: If the password field is located on a separate page, you will need to manually specify the URL of the password page. Cloud Access Manager requires the application to use a different URL for the password page to that of the login page containing the username field.

11. Verify the detected application URL is correct. The URL should contain the correct protocol for the application, for example https followed by the Fully Qualified Domain Name (FQDN) used by the application and optionally a port number if the application uses a non standard port. The URL should not contain a path, for example /OWA. Simple hostnames and IP addresses can also be used, but if the application has been configured to use a particular FQDN/alias then this must also be used in Cloud Access Manager, click **Next**.

    NOTE: The protocol and FQDN and port can be obtained from the URL used to access the application. For example, if the application is normally accessed through https://webmail.prod.local:8443/OWA the protocol would be HTTPS and the FQDN would be webmail.prod.local and the port would be 8443.

12. In this step, Cloud Access Manager needs to know how to proxy the application. Typically this involves configuring Cloud Access Manager to proxy the entire web server used by the application through a new FQDN. This is the preferred method and the method compatible with the most applications. To configure Cloud Access Manager in this way, simply enter the new public FQDN to proxy the application into the field provided on the **Proxy URL** page.

    The new FQDN should be within the wildcard DNS subdomain created during the installation, which will resolve to the Public IP address used by the proxy. For example, if you created the wildcard DNS subdomain *.webapps.democorp.com during the installation you could use the FQDN owa.webapps.democorp.com to proxy Microsoft Outlook Web App. If you did not create a wildcard DNS subdomain for Cloud Access Manager during the installation you will need to add this new FQDN into

your public DNS manually. The new FQDN should be covered by the wildcard SSL certificate you are using.

Alternatively, some applications are installed entirely within their own virtual directory on the web server where they reside. One example of such an application is One Identity Active Roles which installs into the virtual directory /ARServerAdmin, in this case you may be able to configure Cloud Access Manager to proxy the application's virtual directory only, rather than the whole web server, and reuse the FQDN of the proxy. To configure this option, select the proxy's FQDN from the list, then enter the virtual directory where the application is installed into the field below. When your configuration is complete, click **Next**.

> NOTE: Take care to ensure that the path entered is unaltered, even down to subtle changes such as character case. In the example for Active Roles Server, the path must be ARServerAdmin.

13. You will now see the **Permissions** page, which enables you to control which users can access the application. By default, all Cloud Access Manager users have access to the application. You can restrict access to the application to users who belong to a specific role, but for this example simply click **Next** to allow all users to access the application.
14. Enter a name for the application.
15. Select **Use primary credentials to log into this application**. This will ensure that OWA uses the user's Active Directory domain credentials rather than a different username or password unique to the application, for example the same credentials that the user used to authenticate to Cloud Access Manager. For applications that require different credentials make sure this option is left clear. Click **Next.**
16. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the **Title** and **Description** you want to display on the Cloud Access Manager Portal. Many applications will require you to configure a particular entry point, for example with Microsoft Outlook Web App you may need to append the URL with OWA if Outlook is not configured to automatically redirect to /OWA when no path is specified in the URL.

> NOTE: Take care to ensure that the URL entered is unaltered, even down to subtle changes such as character case. In the example Microsoft Outlook Web App, the URL must be appended with OWA. The **Add application to application portal home** and **Allow user to remove application from application portal home** options allow you to specify whether the application should appear automatically on each user's portal page, and how the user can manage the application from the application portal. The options are shown in the table below.

Table 3: Application portal options

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the Application Catalog.
✓	✓	application is added to the portal and it can be removed by the user through the Application Catalog.
✗	✗	application is not automatically added to the portal. The user can add or remove the application to/from the portal through the Application Catalog.

To access the application catalog from the application portal, the user simply needs to click their username, then select **Application Catalog**. Depending on the settings in the **Add application to application portal home** and **Allow user to remove application from application portal** options, the user can add or remove applications to/from the application portal.

17. Configuration of the application is now complete. Click **Finish**.

To verify that the application is configured correctly

1. Close Internet Explorer to end your Cloud Access Manager session.
2. Open the Cloud Access Manager Portal by using the desktop shortcut **Cloud Access Manager Application Portal**.
3. Log in to the Cloud Access Manager Portal and click the **OWA** application.

   NOTE: The first time an application using form fill authentication is accessed by each user, they are presented with the application's login page as normal. The user must enter their credentials for the application as they normally would to log in. Their credentials are then captured and securely stored within Cloud Access Manager so that they can be automatically entered the next time they access the application from the Cloud Access Manager application portal.

   Assuming a user's application credentials, the user's primary credentials in this case, have not yet been stored in Cloud Access Manager, they will be prompted to enter them.

4. Enter your credentials into the OWA login page as normal and click **Sign In**.
5. From OWA, click **Sign Out** and close Internet Explorer.
6. Re-open the Cloud Access Manager Portal and log in as the same user.
7. Click the **OWA** application and you are signed in automatically.

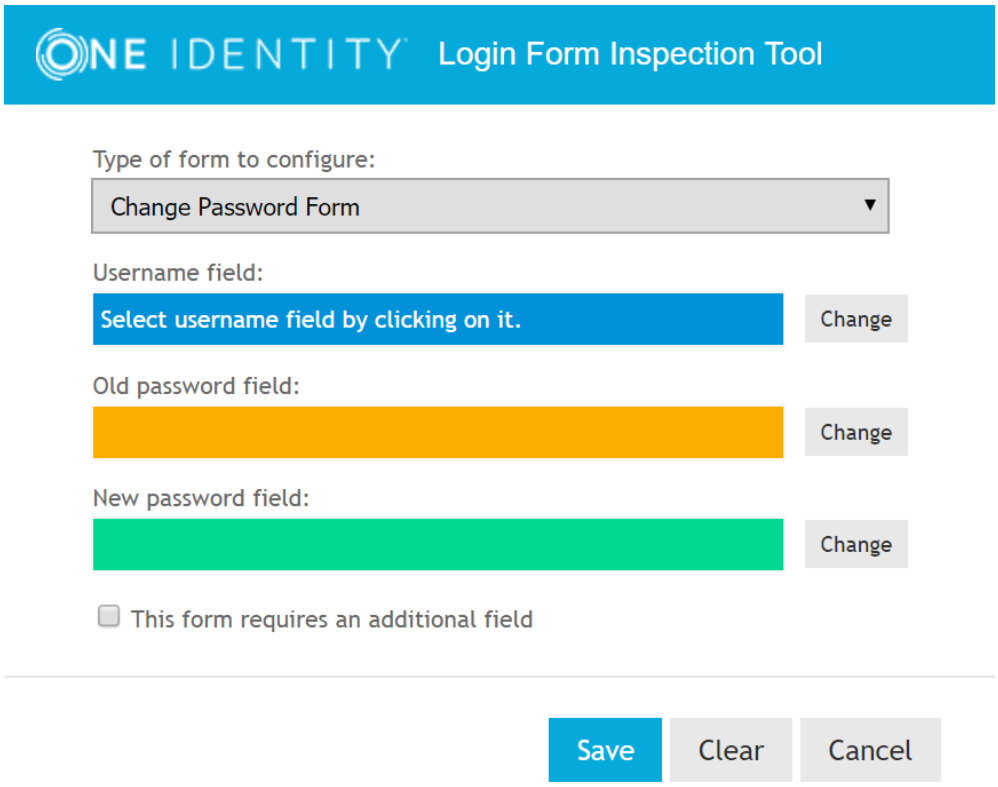
Configuration of Microsoft Outlook Web App for SSO is now complete.

NOTE: While the majority of applications can be configured automatically, some applications will require manual configuration. For further information on advanced form fill configurations, please refer to *One Identity Cloud Access Manager How To Configure Advanced Form Fill Authentication*.

To configure single sign-on for the form fill application change password page (optional)

If a web application supports change password or expired password pages, you can configure Cloud Access Manager to fill and capture these pages.

1. Log in to the Cloud Access Manager administrator console using the desktop shortcut **Cloud Access Manager Application Portal**.
2. Enter the URL of the application into another tab in the browser. For example, for OWA enter https://webmail.prod.local/owa, where webmail.prod.local is the hostname of the host running the Microsoft Outlook Web App.
3. Navigate to the change password page.
4. With the application's change password page displayed, click the browser's favorites icon and click **Inspect Login Form**. The **Cloud Access Manager Login Form Inspection Tool** is now displayed in the bottom-right corner of the browser window. The tool will detect that the application is already known to Cloud Access Manager and display a **Change Password Form/Expired Password Form** list. Select the type of form you want to configure.
5. Use the tool to obtain the field IDs for the login form. For example, if required click in the **Username field** for the field where a username needs to be entered, then if required click in the **Old password field** for where to enter the old password, and finally click in the **New password field** for where to capture the new password from.



6. Click **Save** to return to the Cloud Access Manager configuration wizard with your additional configuration.
7. Review the detected field IDs and click **Save**.

Further considerations

When you have added an application to Cloud Access Manager, you may want to ensure users only access the application using Cloud Access Manager. This may be required if you use Cloud Access Manager to enforce strong authentication for the application, or want to use Cloud Access Manager's auditing features to monitor application usage. For further information on how to ensure that users access the application using Cloud Access Manager, please refer to *Preventing direct access to applications protected by Cloud Access Manager* in the *One Identity Cloud Access Manager Security and Best Practices Guide*.

Configuring Single Log Out (SLO) for proxied applications

You may need to configure Single Log Out for some proxied applications, for example, Outlook Web App in a Cloud Access Manager for Defender deployment.

A user may unknowingly leave their Cloud Access Manager user session active, which a subsequent user could access using the same client and browser. This can occur when a user has connected directly to the proxied application URL rather than accessing it via the Application Portal; the user is redirected to Cloud Access Manager for login and then redirected back to the application where Single Sign-On (SSO) occurs. The Cloud Access Manager user session is not closed automatically when a proxied application session is logged out.

Cloud Access Manager includes proxy parameters that you can set for any proxied application. The parameters cause the browser to redirect to the Cloud Access Manager `/EndWebSession` URL when a target URL is seen by the proxy.

Depending on the logout routine of the application, you may need to apply the `cam.endSessionURLs` only, or `cam.endSessionURLs` and `cam.allowEndSessionURLToBeProxied` may be required. Please refer to the following steps and examples.

To configure SLO for a proxied application

1. Login to the Admin UI as the Fallback Administrator.
2. On the **Cloud Access Manager Proxy** page, select **Settings**, then **Tune**.
3. If used, set both parameters to apply to **All Applications**.

Example 1

For Outlook Web App 2010 which redirects to the standard "You have successfully signed out…close all browser windows" page, use the following configuration:

```
cam.endSessionURLs = /owa/auth/logoff.aspx?Cmd=logoff&src=exch
```

Example 2

For Outlook Web App 2010 when it is configured to redirect to the login page or other SSO location, or is protected by the Microsoft Threat Management Gateway (TMG), you cannot use the end URL as the SLO trigger or Cloud Access Manager could log out users when attempting SSO. Instead, you should use the OWA logout start URL. To ensure that the OWA logout routine is completed before the Cloud Access Manager redirect occurs, send this URL to the client browser using the following configuration:

```
cam.endSessionURLs = /owa/logoff.owa
```

```
cam.allowEndSessionURLToBeProxied = True
```

Proxy-less form fill authentication

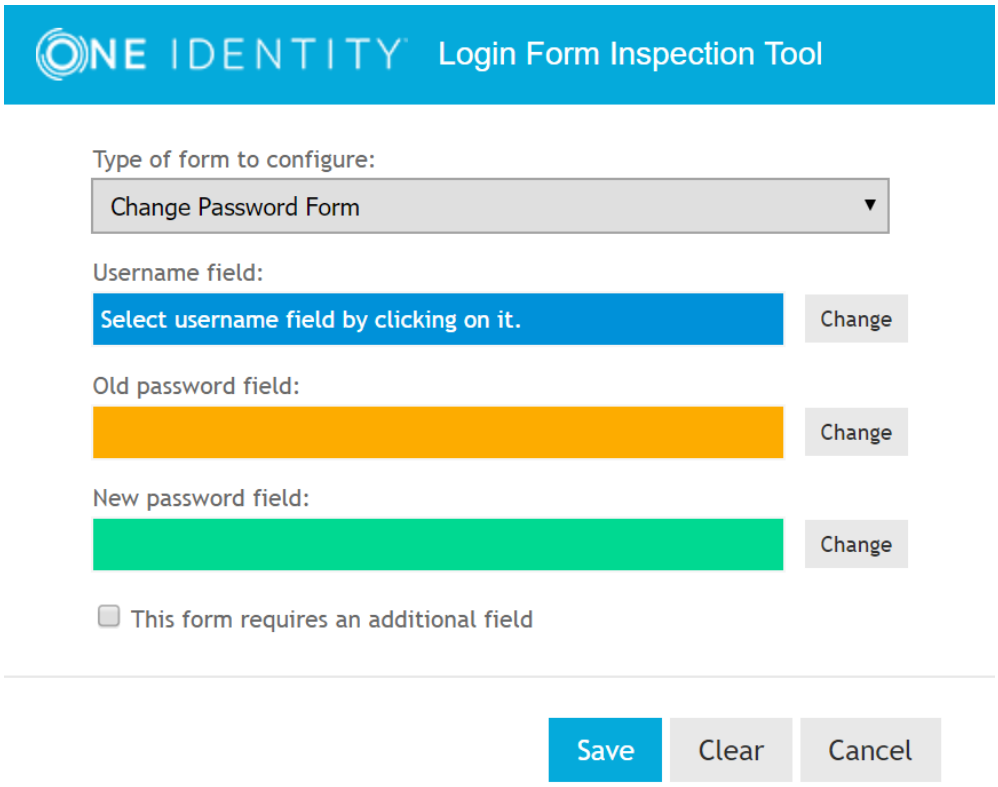
In proxy-less form fill, Cloud Access Manager attempts to emulate the application's login form with an unsolicited post to the action URL within the login form. Configuring an application in this way involves fewer steps than the form fill authentication method described in Form fill authentication on page 44. This example guides you through the steps

required to configure single sign-on to an application using the proxy-less form fill authentication method.

Log in to the **Administration Console** using the desktop shortcut **Cloud Access Manager Application Portal**, and select **Add New**from the **Applications** section on the home page. Cloud Access Manager provides a set of application templates to automatically configure common applications. This example describes how to configure an application manually, rather than using a template.

To configure single sign-on using proxy-less form-fill authentication

1. Click **Configure Manually**.
2. Select **Form Fill**, then click **Next**.
3. If you have not already done so while adding a previous form fill application, save the **Inspect Login Form** bookmarklet to your browser's favorites. To do this, right-click the **Inspect Login Form** link. Click **Add to favorites**.
4. Enter the URL of the application into the box provided and click **Go**, this will take you to the application's login page. If you are taken directly to the application, check that you are not already signed in and if necessary, sign out.
5. With the application's login page displayed, click the browser's **Favorites** icon and click **Inspect Login Form**. The **Cloud Access Manager Login Form Inspection Tool** is now displayed in the bottom-right corner of the browser window.
6. Use the tool to obtain the field IDs for the login form. For example, click in the **Username field**, for example, Domain\user name, then click in the **Password field**, then finally, click the **Submit button**, for example, **Sign in**.



7. Review the detected form IDs and click **Save** to save the form IDs and return to the Cloud Access Manager Configuration wizard.
8. After using the **Login Form Inspection Tool** to identify the username and password fields, proxy-less form fill does not use the submit button, and action URL, you are presented with the **Form Fill Method** configuration page, which is where you choose whether or not to proxy the application with Cloud Access Manager.

   NOTE: Proxy-less form fill only emulates basic elements of login forms, it is therefore not compatible with login forms that rely on cookies, dynamic hidden variables, session handling functions or view states.
9. The next page contains the form fill details (the **Username Field ID/Name** and **Password Field ID/Name**) and the **Login Form Action URL** (the login form's action URL) configuration detected by the **Login Form Inspection Tool**.
10. The next page enables you to customize permissions for the new application by configuring which Cloud Access Manager Roles have access, by default all users have access.
11. Enter a name for the application.
12. Choose whether or not to **Use primary credentials to log into this application**. If selected, this feature will use Active Directory domain credentials rather than a different username or password unique to the application. For example, the same credentials that the user used to authenticate to Cloud Access Manager. For applications that require different credentials make sure this option is left clear.

13. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the Title and Description you want to display on the Cloud Access Manager Portal. Many applications will require you to configure a particular entry point, for example for One Identity Active Roles you need to add ARServerAdmin in the URL field of the application portal page.

    NOTE: Take care to ensure that the URL entered is unaltered, even down to subtle changes such as character case, in the example for Active Roles Server the URL must be ARServerAdmin.

    NOTE: The **Add application to application portal home** and **Allow user to remove application from application portal home** options allow you to specify whether the application should appear automatically on each user's portal page, and how the user can manage the application from the application portal. The options are shown in the table below.

    Table 4: Application portal options

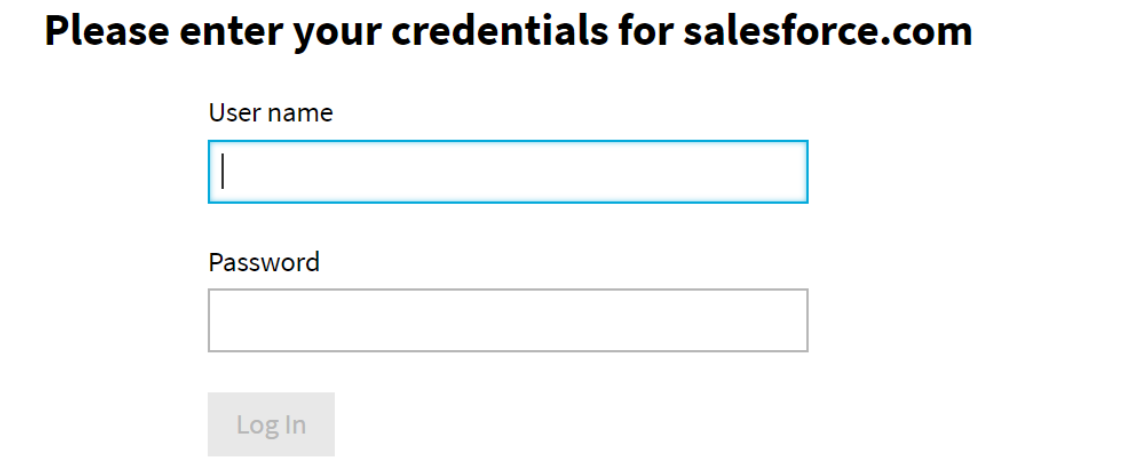
Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the Application Catalog.
✓	✓	application is added to the portal and it can be removed by the user through the Application Catalog.
✗	✗	application is not automatically added to the portal. The user can add or remove the application to/from the portal through the Application Catalog.

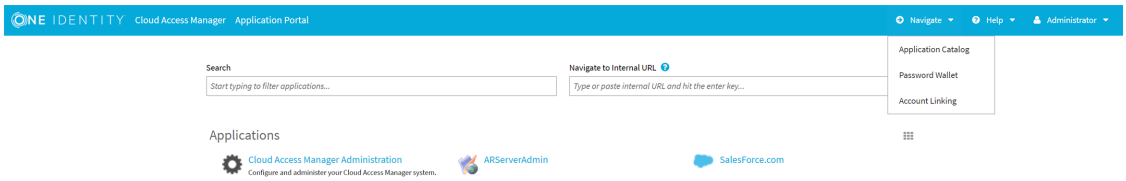
    To access the application catalog from the application portal, the user simply clicks their username, then selects **Application Catalog**. Depending on the settings in the **Add application to application portal home** and **Allow user to remove application from application portal home** options, the user can add or remove applications to/from the application portal.

14. Configuration of the application is now complete. Click **Finish**.

To verify that the application is configured correctly

1. Close Internet Explorer to end your Cloud Access Manager session.
2. Open the Cloud Access Manager Portal using the desktop shortcut **Cloud Access Manager Application Portal**.
3. Log in to the **Cloud Access Manager Portal** and click the application. When a user first accesses an application configured for proxy-less form fill they are presented with a pop-up to enter their login credentials. Cloud Access Manager will then pass the credentials to the application's target URL and store them in the user's Password Wallet for future access.





NOTE: If the user enters invalid credentials, they must be edited in Cloud Access Manager's Password Wallet. The user can access the Password Wallet from the Application Portal with their user ID.

4. Enter your credentials into the login page as normal and click **Save**.
5. From the application, click **Sign Out** and close Internet Explorer.
6. Re-open the Cloud Access Manager Portal and log in as the same user.
7. Click the application and you are signed in automatically.

Configuration of an application for proxy-less form fill is now complete.

Further considerations

When you have added an application to Cloud Access Manager, you may want to ensure users only access the application using Cloud Access Manager. This may be required if you use Cloud Access Manager to enforce strong authentication for the application, or want to use Cloud Access Manager's auditing features to monitor application usage. For further information on how to ensure that users access the application using Cloud Access Manager, please refer to *Preventing direct access to applications protected by Cloud Access Manage*r in the *One Identity Cloud Access Manager Security and Best Practices Guide*.

SAML federation

This example will guide you through the steps required to configure single sign-on for Google Apps service which uses SAML Authentication.

To configure single sign-on for Google Apps service using SAML authentication

1. Log in to the Administration Console using the desktop shortcut **Cloud Access Manager Application Portal** and select **Add New** from the **Applications** section on the home page.

   Cloud Access Manager provides a set of application templates to automatically configure common applications. The following example describes how to configure an application manually, rather than using a template.
2. Click **Configure Manually**.
3. Select **SAML**, then click **Next**.
4. If your service provider provides metadata for configuration, follow the instructions in this step to automatically configure the federation settings in Cloud Access Manager. Otherwise proceed to Step 5 to manually configure the federation settings.

   In the **Federation Metadata URL** field enter the federation metadata URL provided by your service provider. Alternatively, click **Browse** to locate the file containing federation metadata. Please refer to your service provider's configuration interface for assistance locating this information.
5. Enter the Recipient value for your SAML application, for example:

   https://www.google.com/a/<your_google_domain>/acs for Google Apps service

   If your service provider provides multiple Assertion Consumer Service (ACS) endpoints then you can add multiple entries by supplying an Index and Recipient for each entry. Click **Add ACS Entry** to add a new entry. Select the **Default** check box for the entry that will be the default if no Assertion Consumer Service URL or Index is specified in the SAML Authentication request.
6. Enter the **Audience / SP Identity** value for your SAML application, for example, google.com for Google Apps™ service.
7. If your service provider supports SAML logout enter the logout URL in the

Application Logout URL field.

NOTE: For logout requests to be sent to federated applications you must enable the **Log out of federated applications on session termination** option in **Settings | Configuration Settings**.

8. Some service providers sign their SAML Authentication requests or require that SAML Authentication Responses are encrypted. Both of these scenarios require Cloud Access Manager to be configured with a public certificate supplied by the service provider. These certificates can be uploaded using the controls at the bottom of the **Federation Settings** page.
   - If your service provider signs their SAML Authentication requests, click the first **Choose File** button to upload the certificate used to sign the request. This certificate will then be used by Cloud Access Manager to verify the SAML Authentication requests have come from a trusted source.
   - If your service provider requires SAML Authentication responses to be encrypted, click the second **Choose File** button to upload the certificate used to encrypt the response. This certificate will be used by Cloud Access Manager to encrypt the assertion element of the SAML response. To proceed, click **Next**.

   NOTE: Consult your documentation, or application administrative interface for the values to enter.
9. Select **Do not proxy this application**, then click **Next**.
10. Select the **Derive the username from an attribute** option and enter an attribute name of **mail**, then click **Next**.

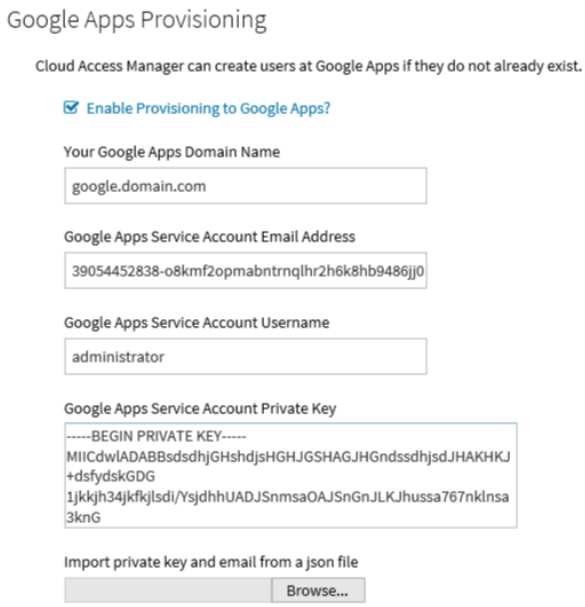
    NOTE: This option uses the user's email address stored in Active Directory as their application username, known as the user's SAML subject. You can change the suffix if required to match your Google domain.

    NOTE: Cloud Access Manager allows users to request their own application accounts. If the user is in a group that is authorized to access a particular application, the user can have a user account automatically created for them as they select it from their application catalog and add it to their portal page.

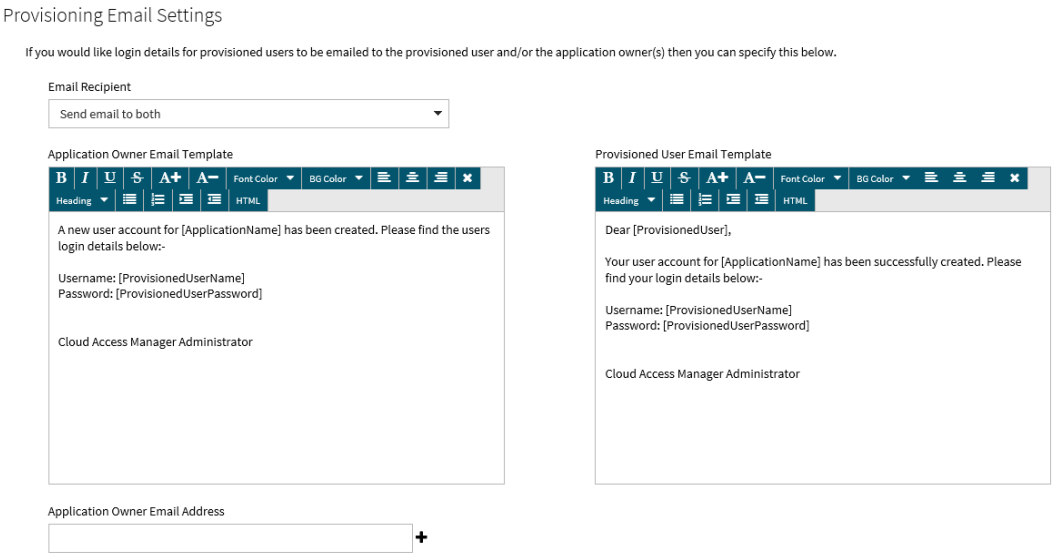
    Cloud Access Manager includes directory connectors, which allow user accounts to be provisioned from Cloud Access Manager into Google Apps service, Salesforce.com and Microsoft Office 365. When a user adds an application to their portal page by selecting it from their application catalog, Cloud Access Manager automatically checks whether they already have a user account in that application's directory. If the user does not, then an account is created for him or her through one of its directory connectors.

    The following three steps are for just-in-time provisioning of users and will only be displayed for applications for which Cloud Access Manager can provision users, such as Google Apps and Salesforce.
11. Enter the credentials of a user account to provision new user accounts. Use the **Test**

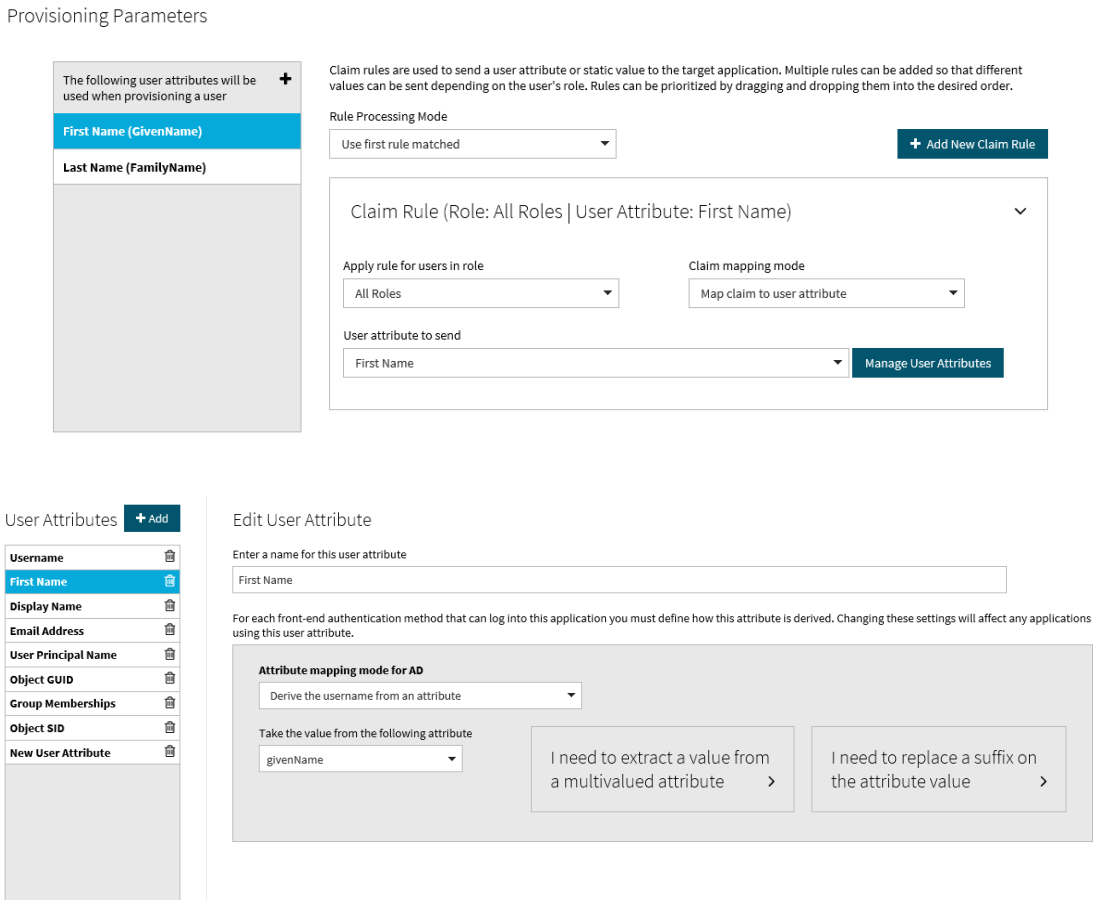
Connection button to validate the credentials before clicking **Next**.



12. Select who will receive an email when a new account is provisioned and enter the text to include in the email.



13. In order to provision a user, the application will typically require a number of provisioning parameters to be defined. For example, Google Apps requires the user's first and last name. For each parameter, configure a claim rule to map the provisioning parameter to a user attribute containing the required value. For example, add a claim called **Last Name**, where its value is derived from the Active Directory user attribute **sn**.



NOTE: There is an 8 character limit on the Alias provisioning parameter. However, there is no such limit on the sAMAccountName attribute that the Salesforce template maps Alias to by default. This results in a failure to provision any user who has a sAMAccountName of greater than 8 characters.

We recommend that either your Salesforce users are limited to a sAMAccountName of 8 characters or less, or the mapping of the Alias field is changed to use a different attribute that does meet this criteria. This is not limited to Salesforce, and can occur for any application that uses the Alias parameter for provisioning. In addition, using mapped attributes with NULL values will also result in a provisioning failure.

NOTE: If the application you are provisioning provides a user provisioning API, please refer to Manual user provisioning on page 67.

14. Click **Next** to continue.
15. You will now see the **Permissions** page, which enables you to control the users who can access the application. By default, all Active Directory users have access to the application. You can restrict access to the application to users who belong to a specific Active Directory security group, but for this demonstration deployment, simply click **Next** to allow all Active Directory users access to the application.
16. Enter an Application Name, for example, Google Apps, then click **Next**.

17. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the **Title** and **Description** you want to display on the Cloud Access Manager Portal.
18. Enter the URL that you want your users to be initially logged in to, for example https://mail.google.com/a/<*your_google_domain*>
19. Click **Fetch icon from application** to locate and display the icon of the application.
20. Click **Finish** to complete the configuration of the application.
21. Click **Download Certificate** to download the certificate created by Cloud Access Manager to import into your SAML application. In addition, make a note of the Issuer/IDP Service URL as this may be required by your SAML application. Click **Close**.

NOTE: The **Add application to application portal home** and **Allow user to remove application from application portal home** options allow you to specify whether the application should appear automatically on each user's portal page, and how the user can manage the application from the application portal. The options are shown in the table below.

Table 5: Application portal options

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the application catalog.
✓	✓	application is added to the portal and it can be removed by the user through the application catalog.
✗	✗	application is not automatically added to the portal. The user can add or remove the application to/from the portal through the application catalog.

To access the application catalog from the application portal, the user simply needs to click their username, then select **Application Catalog**. Depending on the settings in the **Add application to application portal home** and **Allow user to remove**

application from application portal home options, the user can add or remove applications to/from the application portal.

Cloud Access Manager configuration is now complete.

To configure your Google Apps account to authenticate your users using SAML

1. Log in to your Google account using your Google administrator credentials. For example, log in using the following URL: https://www.google.com/a/<*your_google_domain*>
2. Click **Advanced Tools**.
3. Click **Set up Single Sign-on (SSO)**.
4. Select **Enable Single-Sign-on**.
5. Enter the Cloud Access Manager Issuer/IDP Service URL that you noted in Step 21, into the **Sign-in page URL** and **Change password URL** fields. For example, enter:

   https://CloudAccessManager.democorp.local/CloudAccessManager/RPSTS/Saml2/Default.aspx
6. In the **Sign-out page** field, enter the URL:

   https://CloudAccessManager.democorp.local/CloudAccessManager
7. From the **Verification certificate** section, click **Browse**. Navigate to the Cloud Access Manager certificate obtained in the previous section, then click **Upload**.
8. Click **Save Change**.

For information on how to use the user mapping tool, please refer to the guide entitled *One Identity Cloud Access Manager How To Configure User Mapping*.

To verify that the application is configured correctly

1. Close Internet Explorer to end your Cloud Access Manager session.
2. Use the desktop shortcut **Cloud Access Manager Application Portal** to open the Cloud Access Manager portal.
3. Browse to the **Application Catalog** and add the **Google Apps** application to the application portal.
4. Browse to the **Application Portal** and click the **Google Apps** application. You are signed in automatically.

Configuration of Google Apps for SSO is now complete.

Configuring advanced SAML token settings

In most situations the SAML token produced by Cloud Access Manager in response to an authentication request is accepted by the service provider. If the service provider has

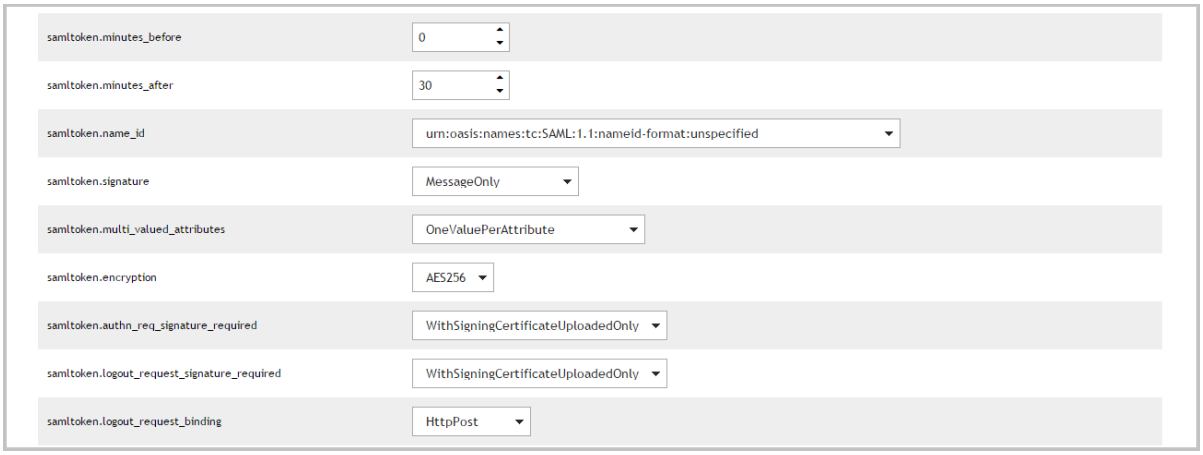
special requirements for the way the token is configured then you may modify the token options on the **SAML Token Settings** tab for the application.

Any settings changed on this page will only affect the selected application.

NOTE: To change the settings for all SAML applications, follow these steps:

1. Navigate to the **Settings** page.
2. Click **Show Advanced Settings**.
3. Click **Advanced Application Settings**.
4. Modify the settings as required.

For a description of the available configuration options, please refer to the table below.



NOTE: The settings for an individual application take precedence over global settings.

Table 6: SAML token advanced configuration options

Name	Description	Default
`samltoken.minutes_before`	The number of minutes before the token `IssueInstant` to set the `NotBefore` attribute in the Conditions element.	`0 minutes`
`samltoken.minutes_after`	The number of minutes after the token `IssueInstant` to set the `NotOnOrAfter` attribute in the Conditions and `SubjectConfirmationData` elements.	`30 minutes`
`samltoken.name_id`	The value of the Format attribute of the `NameID` element in the Subject.	`urn:oasis:names:tc:SAML:1.1:nameid-format:unspecified`
`samltoken.signature`	How the SAML token is signed. There are three options: • `MessageOnly` - Sign the outer message	`MessageOnly`

Name	Description	Default
	• AssertionOnly - Sign the assertion element • MessageAndAssertion - Sign both the outer message and the assertion element.	
samltoken.multi_valued_attributes	How attributes with multiple values are output in the SAML token. There are two options: • OneValuePerAttribute - Each value for a claim type is output in its own attribute, for example <Attribute Name="urn:example/role" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"> <AttributeValue>CN=Staff,CN=Users,DC=democorp,DC=co,DC=uk</AttributeValue> </Attribute> <Attribute Name="urn:example/role" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"> <AttributeValue>CN=Administrators,CN=Users,DC=democorp,DC=co,DC=uk</AttributeValue><AttributeValue> </Attribute> • OneAttributeWithMultipleValues - All values for a claim type are output in one attribute, for example <Attribute Name="urn:example/role" NameFormat="urn:oasis:names:tc:SAML:2.0:attrname-format:uri"> <AttributeValue>CN=Staff,CN=Users,DC=democorp,DC=co,DC=uk</AttributeValue> <AttributeValue>CN=Administrators,CN=Users,DC=democorp,DC=co,DC=uk</AttributeValue><AttributeValue> </Attribute>	OneValuePerAttribute
samltoken.encryption	How the Assertion element is encrypted, there are two options: AES256 - Encryption algorithm: AES256, encryption key:	AES256

Name	Description	Default
	RSA-OAEP `AES128` - Encryption algorithm: AES128, encryption key: RSAES-PKCS1-v1_5, key length: 128	
`samltoken.authn_req_signature_required`	When the authentication request is expected to be signed. There are two options: • `WithSigningCertificateUploadedOnly` - If the application has a signing certificate uploaded, the authentication request must be signed. • `Never` - The authentication request does not need to be signed. However, if the request is signed and the signing certificate is present, the signature will be checked and it must validate correctly to enable authentication.	`WithSigningCertificateUploadedOnly`
`samltoken.logout_request_signature_required`	When the logout request is expected to be signed. There are two options: • `WithSigningCertificateUploadedOnly` - If the application has a signing certificate uploaded, the logout request must be signed. • `Never` - The logout request does not need to be signed. However, if the request is signed and the signing certificate is present, the signature will be checked and it must validate correctly to enable logout.	`WithSigningCertificateUploadedOnly`
`samltoken.logout_request_binding`	The binding that will be used when sending logout requests to the application. Select `Disabled` to not send logout requests.	`HttpPost`

Configuring advanced WS-Federation token settings

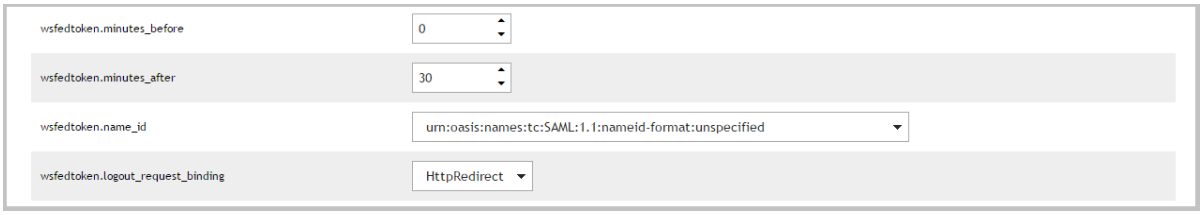
In most situations the WS-Federation token produced by Cloud Access Manager, in response to an authentication request is accepted by the service provider. However, if a service provider has special requirements for the way the token is configured, then you can modify the token options on the **WS-Fed Token Settings** tab for the application.

Any settings changed on this page will only affect the selected application.

NOTE: To change the setting for all WS-Federation applications, follow these steps:

1. Log in to the Administration Console, navigate to the **Settings** page.
2. Click **Show Advanced Settings.**
3. Click **Advanced Application Settings.**
4. Modify as required.

For a description of the available configuration options, please refer to the table below.



NOTE: The settings for an individual application take precedence over global settings.

Table 7: WS-Federation token advanced configuration options

Name	Description	Default
wsfedtoken.minutes_before	The number of minutes before the token IssueInstant to set the NotBefore attribute in the Conditions element.	0 minutes
wsfedtoken.minutes_after	The number of minutes after the token IssueInstant to set the NotOnOrAfter attribute in the Conditions element.	30 minutes
wsfedtoken.name_id	The value of the Format attribute of	urn:oasis:names:tc:SAML:1.1:nameidformat:unspecified

Name	Description	Default
	the NameIdentifier element in the Subject.	
wsfedtoken.logout_request_binding	If HttpRedirect is selected then logout requests will be sent to the application. If Disabled is selected then logout requests will not be sent.	HttpRedirect

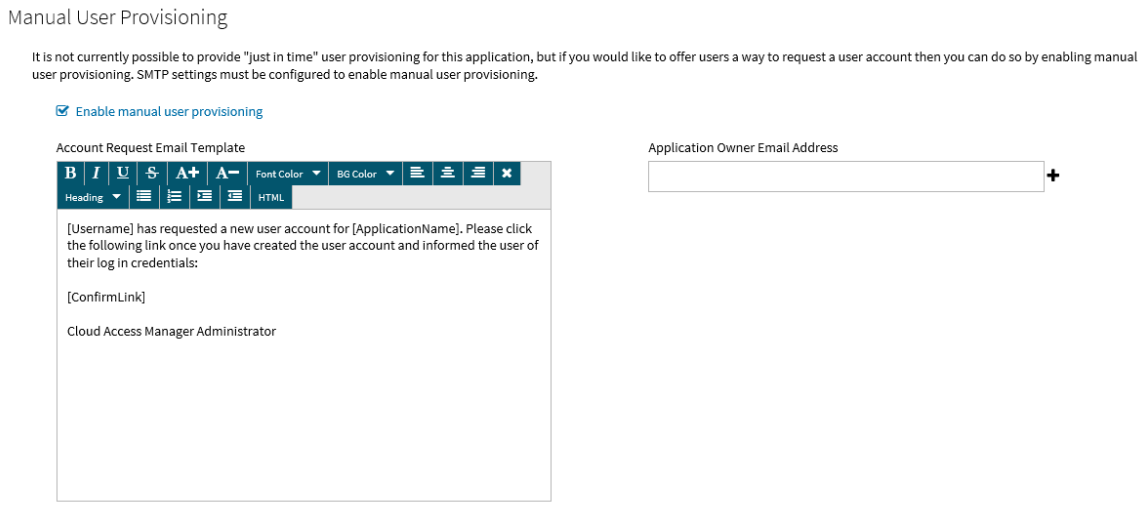
OpenID Connect/OAuth 2.0

For information on how to use Cloud Access Manager as an OAuth v2.0 or OpenID Authorization Server, please refer to the document entitled *One Identity Cloud Access Manager How To Develop OpenID Connect Apps*.

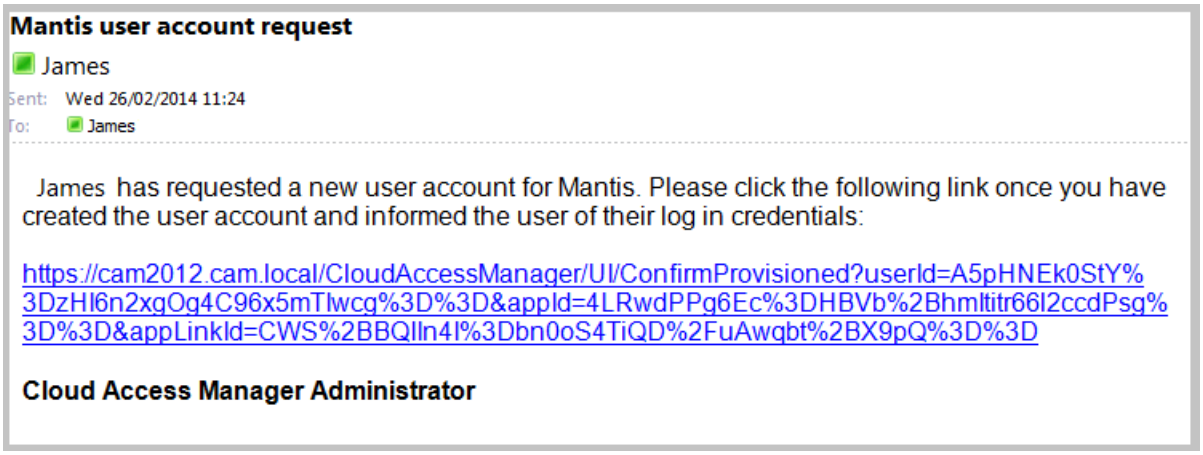
Manual user provisioning

If the application you are configuring does not provide a user provisioning API, you can use Cloud Access Manager as an intermediary between the user and the manual process of creating a user account for the application.

Manual user provisioning enables users to request a user account for an application from their application catalog. Cloud Access Manager then sends an email to the owner of the application advising them that the user requires an account.



The application owner manually creates the user account within the target application. When the user account has been created, the application owner returns to the email received from Cloud Access Manager and clicks the confirmation link contained in the email to confirm that they have created the user account.



Alternatively, a Cloud Access Manager administrator can view any outstanding manual provisioning requests. To do this, go to **Cloud Access Manager Application Portal | Users |Manual Provisioning Requests** and confirm that the requests have been dealt with.

When the user account request is confirmed as complete, the application is displayed on the user's application portal home page within Cloud Access Manager.

NOTE: You must have SMTP settings configured within Cloud Access Manager to enable manual user provisioning.

HTTP basic authentication

To configure HTTP Basic Authentication

1. Log in to the Administration Console using the desktop shortcut **Cloud Access Manager Application Portal** and select **Add New** from the **Applications** section on the home page.

   Cloud Access Manager provides a set of application templates to automatically configure common applications. This example describes how to configure an application manually, rather than using a template.

2. Click **Configure Manually**.

3. Select **HTTP Basic Authentication** and click **Next**.

   NOTE: Additional user attributes can be sent in HTTP headers. In this example we want to send the username and password only.

4. Enter the protocol and Fully Qualified Domain Name (FQDN) used by the application you wish to Single Sign-On (SSO). Click **Next**.

   NOTE: The protocol and FQDN can be obtained from the URL used to access the application. For example, if the application is normally accessed using https://ars.democorp.local/ARServerAdmin, the protocol would be Secure HTTP (HTTPS) and the FQDN would be ars.democorp.local

5. In this step, Cloud Access Manager needs to know how to proxy the application. Typically this involves configuring Cloud Access Manager to proxy the entire web server used by the application using a new fully qualified domain name (FQDN). This is the preferred method and the method compatible with the most applications. To configure Cloud Access Manager in this way, simply enter a new public FQDN into the field provided on the **Proxy URL** page, and click **Next**.

   The new FQDN should be within the wildcard DNS subdomain created during the installation, which will resolve to the public IP address used by the proxy. For example, if you created the wildcard Domain Name Service (DNS) subdomain *.webapps.democorp.com during the installation you could use the FQDN owa.webapps.democorp.com to proxy Microsoft Outlook Web App. If you did not create a wildcard DNS subdomain for Cloud Access Manager during the installation you will need to manually add this new FQDN into your public DNS. The new FQDN should be covered by the wildcard SSL certificate you are using.

   Alternatively, some applications are installed entirely within their own virtual directory on the web server where they reside. One example of such an application is One Identity Active Roles which installs into the virtual directory /ARServerAdmin. In this case you may be able to configure Cloud Access Manager to proxy the application's virtual directory only, rather than the whole web server, and reuse the FQDN of the proxy. To configure this option, select the proxy's FQDN from the list, then enter the virtual directory where the application is installed into the field below and click **Next**.

NOTE: Take care to ensure that the path entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the path must be ARServerAdmin.

6. You will now see the **Permissions** page, which enables you to control the users who can access the application. By default all Cloud Access Manager users have access to the application. You can restrict access to the application to users who belong to a specific role, but for this example, simply click **Next** to allow all users to access the application.
7. Enter a name for the application.
8. If the application requires users to log in using their primary credentials, for example their domain account, select **Use primary credentials to log into this application** and click **Next**. If the application requires users to use a different username or password, leave the option clear and click **Next**.
9. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the Title and Description you want to display on the Cloud Access Manager Portal. Many applications will require you to configure a particular entry point, for example for Active Roles Server you would need to add ARServerAdmin in the URL field of the **Application Portal** page.

   NOTE: Take care to ensure that the URL entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the URL must be ARServerAdmin. The **Add application to application portal home** and **Allow user to remove application from application portal** options allow you to specify whether the application should appear automatically on each user's portal page, and how the user can manage the application from the application portal. The options are shown in the table below.

Table 8: Application portal options

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the application catalog.
✓	✓	application is added to the portal and it can be removed by the user through the application catalog.

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✗	✗	application is not automatically added to the portal. The user can add or remove the application to/from the portal through the application catalog.

To access the application catalog from the application portal, the user simply needs to click their username, then select **Application Catalog**. Depending on the settings in the **Add application to application portal home** and **Allow user to remove application from application portal home** options, the user can add or remove applications to/from the application portal.

10. Configuration of the application is now complete. Click **Finish**.

Further considerations

When you have added an application to Cloud Access Manager, you may want to ensure users only access the application using Cloud Access Manager. This may be required if you use Cloud Access Manager to enforce strong authentication for the application, or want to use Cloud Access Manager's auditing features to monitor application usage. For further information on how to ensure that users access the application using Cloud Access Manager, please refer to *Preventing direct access to applications protected by Cloud Access Manage*r in the *One Identity Cloud Access Manager Security and Best Practices Guide*.

HTTP header value

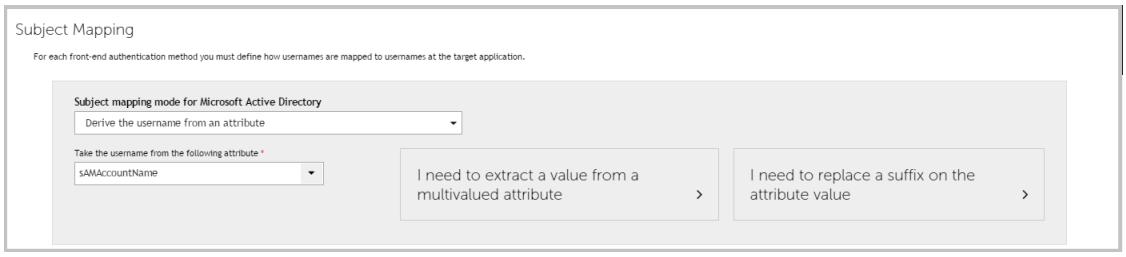
To configure the HTTP header value

1. Log in to the Administration Console using the desktop shortcut **Cloud Access Manager Application Portal** and select **Add New** from the **Applications** section on the home page.

   Cloud Access Manager provides a set of application templates to automatically configure common applications. This example describes how to configure an application manually, rather than using a template.

2. Click **Configure Manually**.
3. Select **HTTP Header** and click **Next**.
4. You now need to configure how Cloud Access Manager will derive the user's

username that will be used to authenticate to the application. This step will vary depending on which front-end authentication method you are using. In this example, we will run through the steps required for Active Directory front-end authentication. Select **Derive the username from an attribute**. A text box is displayed for you to enter the Active Directory attribute to use. Enter **sAMAccountName** and click **Next**.



5. Enter the name of the header you wish to use to send the derived username. The application's web server may prefix this header name with HTTP_. If this is the case, the application must include this prefix when referencing the header. Click **Next**.

   NOTE: Additional user attributes can also be sent in HTTP headers. In this example we will send the user's username only.

6. Enter the protocol and Fully Qualified Domain Name (FQDN) used by the application you wish to Single Sign-On (SSO). Click **Next**.

   NOTE: The protocol and FQDN can be obtained from the URL used to access the application. For example, if the application is normally accessed using https://ars.democorp.local/ARServerAdmin, the protocol would be HTTPS and the FQDN would be ars.democorp.local.

7. In this step, Cloud Access Manager needs to know how to proxy the application. Typically this involves configuring Cloud Access Manager to proxy the entire web server used by the application using a new fully qualified domain name. This is the preferred method and the method which is compatible with the most applications. To configure Cloud Access Manager in this way, simply enter a new public FQDN into the field provided on the **Proxy URL** page, and click **Next**.

   The new FQDN should be within the wildcard DNS subdomain created during the installation, which will resolve to the public IP address used by the proxy. For example, if you created the wildcard DNS subdomain *.webapps.democorp.com during the installation you could use the FQDN owa.webapps.democorp.com to proxy Microsoft Outlook Web App. If you did not create a wildcard DNS subdomain for Cloud Access Manager during the installation you will need to add this new FQDN into your public DNS manually. The new FQDN should be covered by the wildcard SSL certificate you are using.

   Alternatively, some applications are installed entirely within their own virtual directory on the web server where they reside. One example of such an application is One Identity Active Roles which installs into the virtual directory /ARServerAdmin. In this case you may be able to configure Cloud Access Manager to proxy the application's virtual directory only, rather than the whole web server, and reuse the FQDN of the proxy. To configure this option, select the proxy's FQDN from the list,

then enter the virtual directory where the application is installed into the field below and click **Next**.

> NOTE: Take care to ensure that the path entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the path must be ARServerAdmin.

8. You will now see the **Permissions** page, which enables you to control the users who can access the application. By default, all Cloud Access Manager users have access to the application. You can restrict access to the application to users who belong to a specific role, but for this example, simply click **Next** to allow all users to access the application.
9. Enter a name for the application, then click **Next**.
10. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the Title and Description you want to display on the Cloud Access Manager Portal. Many applications will require you to configure a particular entry point, for example for Active Roles Server you would need to add ARServerAdmin in the URL field of the **Application Portal** page.

    > NOTE: Take care to ensure that the URL entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the URL must be ARServerAdmin. The **Add application to application portal home** and **Allow user to remove application from application portal** options allow you to specify whether the application should appear automatically on each user's portal page, and how the user can manage the application from the application portal.

    The options are shown in the table below.

Table 9: Application portal options

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the application catalog.
✓	✓	application is added to the portal and it can be removed by the user through the application catalog.
✗	✗	application is not automatically added to

Add application to application portal home	Allow users to remove application from application portal home	Functionality
		the portal. The user can add or remove the application to/from the portal through the application catalog.

To access the application catalog from the application portal, the user simply needs to click their username, then select **Application Catalog**. Depending on the settings in the **Add application to application portal home** and **Allow user to remove application from application portal home** options, the user can add or remove applications to/from the application portal.

11. Configuration of the application is now complete. Click **Finish**.

To ensure that users are securely authenticated, you must configure applications that use header authentication to prevent users accessing the application directly. For further information on how to ensure that users access the application using Cloud Access Manager, please refer to *Preventing direct access to applications protected by Cloud Access Manager* in the *One Identity Cloud Access Manager Security and Best Practice Guide*.

No back-end SSO

To configure an application that uses no back-end SSO

1. Log in to the Administration Console using the desktop shortcut **Cloud Access Manager Application Portal** and select **Add New** from the **Applications** section on the home page.

   Cloud Access Manager provides a set of application templates to automatically configure common applications. This example describes how to configure an application manually, rather than using a template.

2. Click **Configure Manually**.
3. Select **Cloud Access Manager should not log the user in** and click **Next**.
4. You can now configure the application for external access.
   a. If the application is only accessible within your corporate network, select the internal option and click **Next**. This option will proxy the application so that users accessing Cloud Access Manager from outside of your corporate network can use the application.

NOTE: If users require access to the application before they have authenticated, or do not require authentication to access the application, then you can select the **Allow un-authenticated access to this application** box to allow un-authenticated access.

b. If your application is already accessible from outside of your corporate network, select the external option and click **Next**. This option will not configure the proxy, you may skip to Step 7.

5. Enter the protocol and Fully Qualified Domain Name (FQDN) used by the application you wish to Single Sign-On (SSO). Click **Next**.

   NOTE: The protocol and FQDN can be obtained from the URL used to access the application. For example, if the application is normally accessed using https://ars.democorp.local/ARServerAdmin, the protocol would be HTTPS and the FQDN would be ars.democorp.local.

6. In this step, Cloud Access Manager needs to know how to proxy the application. Typically this involves configuring Cloud Access Manager to proxy the entire web server used by the application using a new fully qualified domain name. This is the preferred method and the method compatible with the most applications. To configure Cloud Access Manager in this way, simply enter a new public FQDN into the field provided on the **Proxy URL** page, and click **Next**.

   The new FQDN should be within the wildcard DNS subdomain created during the installation, which will resolve to the public IP address used by the proxy. For example, if you created the wildcard DNS subdomain *.webapps.democorp.com during the installation you could use the FQDN owa.webapps.democorp.com to proxy Microsoft Outlook Web App. If you did not create a wildcard DNS subdomain for Cloud Access Manager during the installation you will need to add this new FQDN into your public DNS manually. The new FQDN should be covered by the wildcard SSL certificate you are using.

   Alternatively, some applications are installed entirely within their own virtual directory on the web server where they reside. One example of such an application is One Identity Active Roles which installs into the virtual directory /ARServerAdmin. In this case, you may be able to configure Cloud Access Manager to proxy the application's virtual directory only, rather than the whole web server, and re-use the FQDN of the proxy. To configure this option, select the proxy's FQDN from the list, then enter the virtual directory where the application is installed into the field below and click **Next**.

   NOTE: Take care to ensure that the URL entered is not altered, even down to subtle changes such as character case. In the example Active Roles Server, the URL must be ARServerAdmin.

7. You will now see the **Permissions** page, which enables you to control the users who can access the application. By default, all Cloud Access Manager users have access to the application. You can restrict access to the application to users who belong to a specific role, but for this example, simply click **Next** to allow all users to access the application.

8. Enter a name for the application, then click **Next**.

9. You can now configure how the application is displayed on the Cloud Access Manager Portal. Enter the Title and Description you want to display on the Cloud Access Manager Portal. Many applications will require you to configure a particular entry point, for example, for Active Roles Server you would need to add ARServerAdmin in the URL field of the **Application Portal** page.

   NOTE: Take care to ensure that the URL entered is unaltered, even down to subtle changes such as character case, in the example Active Roles Server the URL must be ARServerAdmin. The **Add application to application portal home** and **Allow user to remove application from application portal home** options allow you to specify whether the application should appear automatically on each user's portal page, and how the user can manage the application from the application portal.

   The options are shown in the table below.

   Table 10: Application portal options

Add application to application portal home	Allow users to remove application from application portal home	Functionality
✓	✗	application is added to the portal and it cannot be removed by the user through the application catalog.
✓	✓	application is added to the portal and it can be removed by the user through the application catalog.
✗	✗	application is not automatically added to the portal. The user can add or remove the application to/from the portal through the application catalog.

   To access the application catalog from the application portal, the user simply needs to click their username, then select Application Catalog. Depending on the settings in the **Add application to application portal home** and **Allow user to remove application from application portal home** options, the user can add or remove applications to/from the application portal.

10. Configuration of the application is now complete. Click **Finish**.

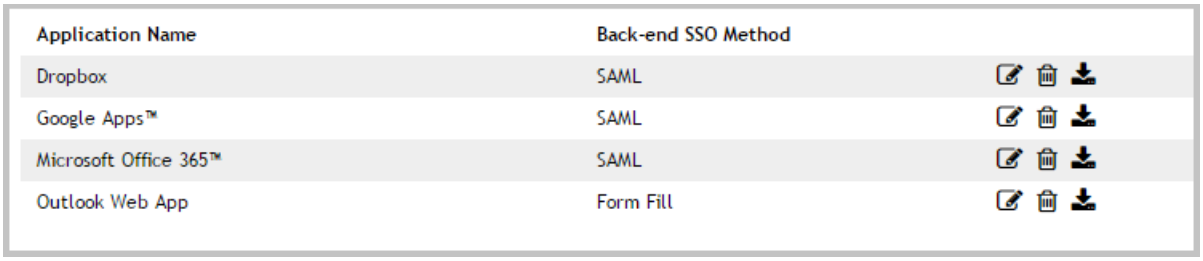
Further considerations

When you have added an application to Cloud Access Manager, you may want to ensure users only access the application using Cloud Access Manager. This may be required if you use Cloud Access Manager to enforce strong authentication for the application, or want to use Cloud Access Manager's auditing features to monitor application usage. For further information on how to ensure that users access the application using Cloud Access Manager, please refer to *Preventing direct access to applications protected by Cloud Access Manager* in the *One Identity Cloud Access Manager Security and Best Practices Guide*.

Exporting an application configuration template

When you have configured an application, you can export the configuration as a template file that can be imported into other Cloud Access Manager installations and used to re-create your configured application.

To export an application configuration as a template, in the list of applications click the download icon next to the application you want to export.



Before you can create a template, you need to know whether there are any environment or account specific variables within your application configuration. For example, any applications that have been purchased and installed on your local network are highly likely to have environment specific URLs within the configuration.

For SaaS applications such as Salesforce.com there may also be account specific variables within URLs, such as a unique domain name or account identifier.

To setup applications with no environment or account specific variables

If there are no environment or account specific variables in the application's configuration, you can:

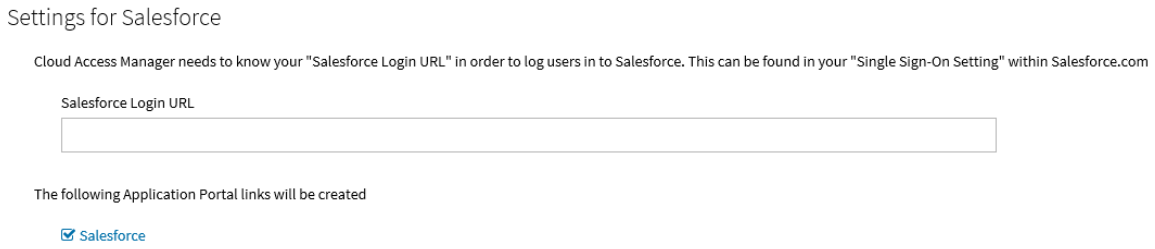
1. Enter the template name.
2. Clear the box shown in the image below:

3. Click **Create Template**.
4. When prompted to save or open the file, select **Save**.

To setup applications with existing environment or account specific variables

If there are environment or account specific variables within the application's configuration, you need to define the **template settings** page.

The **template settings** page is the first page of the application setup wizard that you will see when you click on a template. The image below shows an example of the Salesforce template settings page within Cloud Access Manager.



The text box in the image above is referred to in the **Export Application as Template** page as the **Input Field**. The value entered into this field by the administrator using the template is referred to as the **Input Field Value**. The **Template settings page instructions** should outline what the administrator must enter into the **Input Field**.

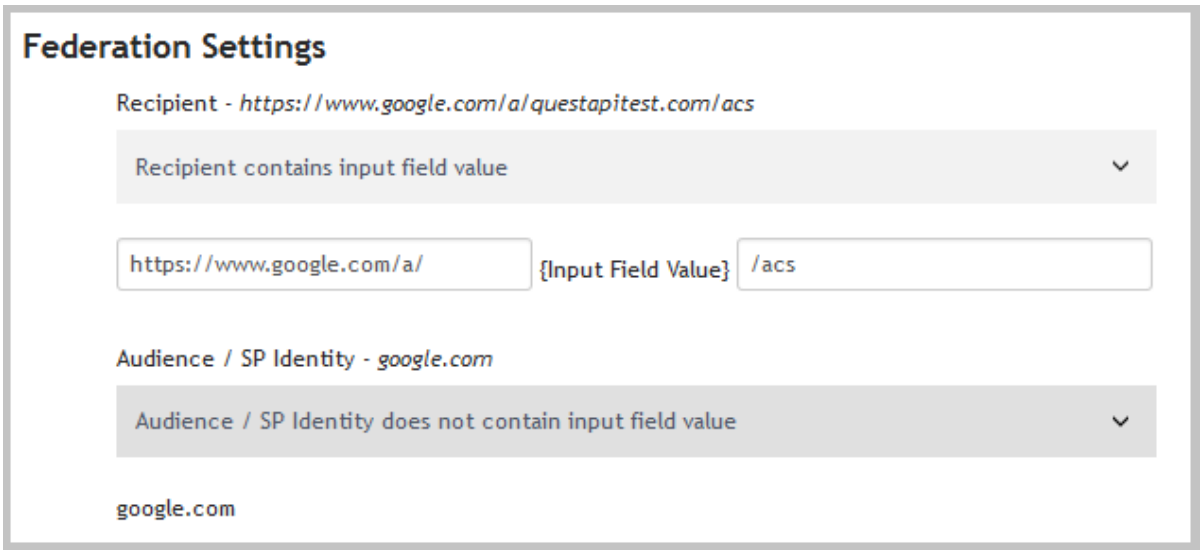
In the Salesforce example, this instruction is displayed below the **Settings for Salesforce** heading. The **Input Field Label** is used to identify the **Input Field** text box and is displayed just above the input field text box. The **Format of Input Field Value** list allows you to select the required format of the **Input Field Value**. For example, on the Salesforce template settings page, the **Input Field Value** must be a valid URL, so the format of the **Input Field Value** would be set to **Full URL**.

The final step is to identify the part of the application you are currently exporting as a template which is unique to your environment/account. For example, when configuring Google Apps the recipient field will be in the format https://www.google.com/a/yourgoogledomain.com/acs where yourgoogledomain.com is an account specific domain.

In this example you would enter yourgoogledomain.com into the Environment/account specific value to be replaced text box. This text will be replaced by the value users enter into the input field when using the template.

Example application configuration template

In the following example configuration for Google Apps, the **Input Field Value** would be **questapitest.com** as this is an account specific variable. The text in italics is the value of each field in your application's configuration. So, to make the **Recipient** field generic you need to tell Cloud Access Manager where the **Input Field Value** would be. The **Audience / SP Identity** field does not contain **questapitest.com**, so we can select that option.



When you have configured all fields, click **Create Template**. You are prompted to save or open the file. Select **Save**. The application configuration template file is now ready to use.

Forwarding claims to federated applications

Applications using SAML 2.0, WS-Federation or OpenID Connect/OAuth 2.0 to perform single sign-on (SSO) may receive claims from Cloud Access Manager, which are then delivered to the application as part of the assertion which tells the application the user has successfully authenticated. A claim is a piece of information about a user, which the application can use to tailor its interface or to make authorization decisions.

To configure Cloud Access Manager to send claims to the application, you must choose a name for each claim, and then map that name to an information source.

Cloud Access Manager can be configured to:

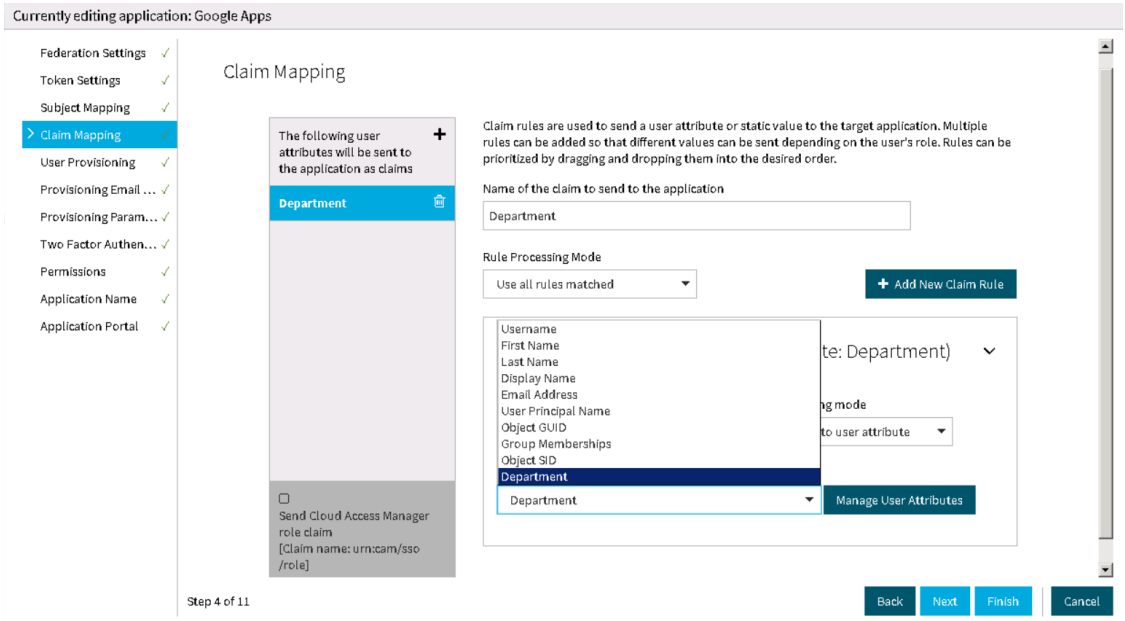
- Forward claims authored by the front-end authenticator which authenticated the user.
- Send static claims.
- Send the names of the user's Cloud Access Manager roles as claims.

These configuration options can be used individually or in any combination. Additionally, you can make the transmission of a claim dependent on a user's role membership.

To configure an application to receive claims from Cloud Access Manager

NOTE: You cannot configure claim mappings when you create the application definition. The facility to configure claim mappings is only available in edit mode, after you have created the application definition.

1. From the Cloud Access Manager Administration Menu, under **Applications**, click **View and Edit**.
2. Click the edit icon  next to the application you want to send claims to.
3. In the navigation bar, click **Claim Mapping**.
4. To add a new claim mapping, click the add icon  in the top right hand corner of the claim list pane.
5. Complete the **Name of the claim to send to the application**. For OpenID Connect/OAuth 2.0 applications, you can select from a preset list of standard claims.
6. **Rule Processing Mode**: Mapping rules can be applied to users who have a certain role. You can use the **Rules Processing Mode** setting to determine whether only the first rule matching the user(s) should apply, or whether all rules should apply.
    - **Use first rule matched** - return the result of the first rule where the user is a member of the role set on the rule.
    - **Use all rules matched** - return the results of all rules where the user is a member of the role set on the rule.
7. In the **Claim Rule** box:
    a. Select the roles that you want to apply the rule to.
    b. Choose the **Claim mapping mode**:
        - If you want the claim to be derived from a claim from an identity provider, choose **Map claim to user attribute**.
        - If you want Cloud Access Manager to set the claim to a constant value, choose **Map claim to static value**.
8. If you have chosen **Map claim to static value**, enter that value in the box provided. If you have chosen **Map claim to user attribute**, choose the attribute holding the information you want to send as a claim from the dropdown. To add more attributes to the list displayed in the dropdown, click **Manage User Attributes**.



9. If the same claim can be derived from different attributes depending on the user's role, you can add another **Claim Rule** by clicking the **Add New Claim Rule** button. If you have defined multiple **Claim Rules** you can order them by dragging and dropping the rules into the correct position, so that the correct rule is processed for users in a given role.
10. If you want to configure Cloud Access Manager to send more claims, click the add icon + in the top right hand corner of the claim list pane.
11. You can send the names of the user's Cloud Access Manager roles to the application as claims. To do this, select the **Send Cloud Access Manager role claim** box at the bottom of the claim list panel.

NOTE: If you select **Group Memberships** for a claim rule and you are using Active Directory the user's Primary Group is not returned. In default installations, the user's Primary Group is Domain Users. The Primary Group is not returned because the claim rule returns the values in the **memberOf** attribute and the Primary Group is determined using the **primaryGroupID** attribute.

Adding HTTP headers to proxy applications

If you are using Cloud Access Manager to proxy an application and authenticate users to that application you have the option of configuring HTTP headers to be sent as part of the authentication. The following type of applications can be configured to send extra HTTP headers:

- Form Fill
- Integrated Windows Authentication
- HTTP Basic Authentication
- HTTP Header.

Headers are pieces of information about a user, which the application can use to tailor its interface or to make authorization decisions. The mechanism for configuring additional HTTP headers uses the same process of building information from claim rules that is used in claim mapping. For detailed instructions on how the mapping interface works please refer to Forwarding claims to federated applications on page 79, where you see **Claim Mapping** replace this with **Header Mapping**.

To configure an application to receive claims from Cloud Access Manager

NOTE: You cannot configure header mappings when creating the application definition. The facility to configure header mappings is only available in edit mode, after the application definition has been created.

1. From the Cloud Access Manager Administration Menu, under **Applications**, click on **View and Edit**.
2. Click on the edit icon next to the application you want to send claims to.
3. In the navigation bar, click **Header Mapping**.
4. Follow the instructions in Forwarding claims to federated applications on page 79, where you see **Claim Mapping** replace this with **Header Mapping**.

Configuring step-up authentication

When you configure an Active Directory or Lightweight Directory Access Protocol (LDAP) front-end authenticator you can also configure two-factor authentication. Configuring a front-end authentication method describes how to configure two-factor authentication for all users, for all applications.

This section describes how to modify the configuration in three ways:

- Users are only prompted for two-factor authentication for some applications. This is known as step-up authentication as users will only be prompted for two-factor authentication when required.
- Only external users are prompted for two-factor authentication.
- Access control decisions are made based on the threat level determined by the Security Analytics Engine.

Configuring front-end authenticators

If your users are authenticating using one of the **Directory Authenticators** (Active Directory or one of the LDAP type authenticators), you can configure Cloud Access Manager to use a second factor of authentication in addition to a password. The secondary authentication methods available are:

- RADIUS server
- Smart card
- Starling 2FA

The configuration options for these methods are described in the following sections.

RADIUS server

Complete the **RADIUS Connection Settings** to allow Cloud Access Manager to connect to an authentication service using the Remote Authentication Dial-In User Service (RADIUS) protocol. Please refer to the table below for a detailed explanation of each feature.

Table 11: RADIUS connection settings

Field	Functionality
Hostname/IP Address (including port)	Enter the fully-qualified domain name or the IP address of your authentication service host and the UDP port number on which the authentication service is listening. The IANA-registered port number for RADIUS is 1812. For example radius.example.com:1812
Shared Secret	Enter the password or passphrase used to encrypt sensitive information in the RADIUS traffic sent to the authentication service. The authentication service must be configured with the same shared secret.
Challenge/Response Server	Many RADIUS authentication services are capable of maintaining an authentication session with multiple requests and responses. This allows challenge-response authentication tokens to be used, as well as other features like password expiry and token time window resynchronization. If your authentication service supports challenge/response mode, then select the **Challenge/Response Server** box.
Attribute to use for RADIUS username	Enter the name of the Active Directory attribute whose value is to be relayed to the RADIUS authentication service to identify the user. The default, sAMAccountName, contains the login username.
Test Connection	To determine whether Cloud Access Manager has connectivity to the RADIUS authentication service.

Smart card

The configuration procedure is similar whether you are using smart card as a primary or secondary factor authentication method. The following steps describe how to configure

Cloud Access Manager for smart card authentication:

1. Select the **Enable certificate revocation list checking** box. This will prompt Cloud Access Manager to check the Certificate Authority's Certificate Revocation List (CRL) to ensure the user's certificate has not been revoked. If the user's certificate has been revoked, the login request will be denied.
2. Cloud Access Manager must redirect the user's browser to another port in order to perform an X.509 certificate authentication. The default port is 8443. If port 8443 is already in use by another service on the Cloud Access Manager host, you can choose a different port number.
3. Export the certificate from your Certificate Authority in .pem or base-64 encoded format, then copy it to the Cloud Access Manager Secure Token Server (STS) host and upload it using the **Browse...** control.

For detailed instructions on smart card configuration, please refer to Configuring smart card authentication on page 15.

Starling 2FA

Starling 2FA is a cloud based authentication service that allows users to self-register and then access their one time passwords on both mobile and desktop devices. For further information on accessing Starling 2FA and using Cloud Access Manager to authenticate Starling 2FA users, please refer to Configuring each application on page 86.

When you have obtained a Starling 2FA subscription, you must complete the following fields to allow Cloud Access Manager to connect to the service:

Table 12: Starling 2FA

Field	Functionality
Starling 2FA subscription key	Enter the subscription key that was supplied when you registered for Starling 2FA. You can also obtain this key from the dashboard of your Starling 2FA instance administration portal.
Attribute to use for mobile phone number	Enter the name of the attribute from the primary directory (Active Directory / LDAP) whose value is to be relayed to the Starling 2FA authentication service to identify the user. The default attribute is **mobile**, this usually contains the user's mobile telephone number.
Default country code for phone numbers	Select the country for which mobile telephone numbers can be specified without the country code prefix. If you have telephone numbers in your directory that are not in the default region they must begin with a plus sign followed by the numeric region code.

Configuring each application

Configuring step-up authentication for an application is a two stage procedure. The:

- Front-end authenticator must be configured to support two factor authentication
- Application must be configured for step-up authentication.

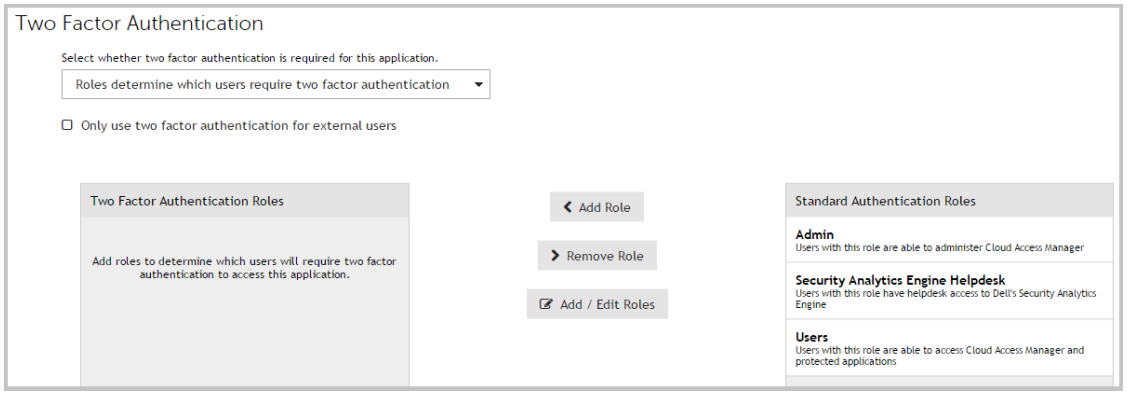
NOTE: If multiple front-end authenticators are configured, step-up authentication is only available for users who authenticate with front-end authenticators that have two-factor authentication configured.

To configure the front-end authenticator for step-up authentication

1. Navigate to the **Front-end Authentication** page and click the name of the authenticator that you want to configure.
2. Click the **Two Factor Authentication** tab.
3. Select **Use two factor authentication for specific applications**.
4. Configure the **RADIUS connection settings** if not already configured, please refer to Configuring a front-end authentication method on page 6 for further information.

To configure the application for step-up authentication

1. Navigate to the **Applications** page and click the name of the application that you want to configure.
2. Click the **Two Factor Authentication** tab.
3. From the list, select the users who will require two factor authentication to access the application. This will be either:
    - **All users of this application require two factor authentication,** or
    - **Roles determine which users require two factor authentication**.
4. If you are configuring role based access, select the required roles from the **Standard authentication roles** list and click **Add Role** to add the role to the list of **Two factor authentication roles**.



Configuring for external users

Two factor authentication may also be applied only for external users. In this context, external users are defined as users whose IP addresses do not fall in the following ranges:

- 10.0.0.0 - 10.255.255.255
- 172.16.0.0 - 172.31.255.255
- 192.168.0.0 - 192.168.255.255

Two factor authentication for external users may be configured either to apply to all applications, or it can be configured on a per application basis.

To configure two factor authentication for external users for all applications

1. Navigate to the **Front-end Authentication** page and click on the name of the authenticator that you want to configure.
2. Click the **Two Factor Authentication** tab.
3. Select **Use two factor authentication for all applications for external users only**.
4. Configure the **RADIUS Connection Settings** if not already configured, please refer to Configuring a front-end authentication method on page 6 for further information.

To configure two factor authentication for external users for specific applications

1. To configure the front-end authenticator follow the steps in To configure the front-end authenticator for step-up authentication.
2. To configure the application follow the steps in To configure the application for step-up authentication on page 86.
3. Select the **Only use two factor authentication for external users** check box on the application's **Two Factor Authentication** tab.

Configuring Starling 2FA with Cloud Access Manager

Starling 2FA is a cloud based authentication service that allows users to self-register and access their one time passwords on both mobile and desktop devices.

Purchasing a subscription

If you do not currently have a Starling 2FA subscription please email sales@oneidentity.com or telephone: 1-800-306-9329 for assistance.

Starling 2FA Dashboard

When your subscription is confirmed, you will receive an email containing your subscription details. This will include the:

- Starling 2FA instance name
- Subscription key
- address and credentials required to access the **Starling 2FA dashboard**.

To access the Starling 2FA dashboard

1. Open an Internet browser and navigate to the address in your email.
2. You will be prompted for your email address and password, use the values in your confirmation email.
3. You will then be prompted for a token response.

   NOTE: To obtain this response you must install the **Starling 2FA App**. Links to download the app are included in your confirmation email.
4. When downloaded and configured, the app will display a one-time password that you can enter into the browser.
5. When you are logged in, you are presented with a short video that explains the functionality of the dashboard.

To configure Cloud Access Manager to use Starling 2FA

Starling 2FA can be used as a two-factor authentication provider from within Cloud Access Manager. For instructions on how to configure two-factor authentication, please refer to Configuring front-end authenticators.

Using the Security Analytics Engine

The Security Analytics Engine calculates a risk score based on various factors relating to the user at the time of authentication. These factors can include the user's browser, their IP address, the time of day and any historical data from previous authentications. This risk score is reported to Cloud Access Manager as a value from 0 to 100 percent. You can configure Cloud Access Manager to change the authentication requirements for the user based on their risk score.

This section describes how to configure Cloud Access Manager to use the Security Analytics Engine. The example takes you through the steps required to configure Cloud Access Manager to adapt its access control based on the risk score reported by the Security Analytics Engine. For more information on how to configure the Security Analytics Engine, please refer to the following chapters in the *Security Analytics Engine User Guide*:

- Plugins
- Conditions
- Applications
- Auditing
- Policy Overrides
- Glossary
- Shared Policies
- Issued Alerts

IMPORTANT: Any additional chapters within the *Security Analytics Engine User Guide* will not apply to Cloud Access Manager and using them may cause complications for both Cloud Access Manager and the Security Analytics Engine.

NOTE: If multiple front-end authenticators are configured, step-up authentication will only be available for users authenticating with front-end authenticators which have two-factor authentication configured. In addition, when Cloud Access Manager is configured to use the Security Analytics Engine it is not important which two factor authentication mode is selected in the front-end authenticator configuration as all modes will be treated equally. The decision on when two factor authentication is required will be determined by the risk score returned.

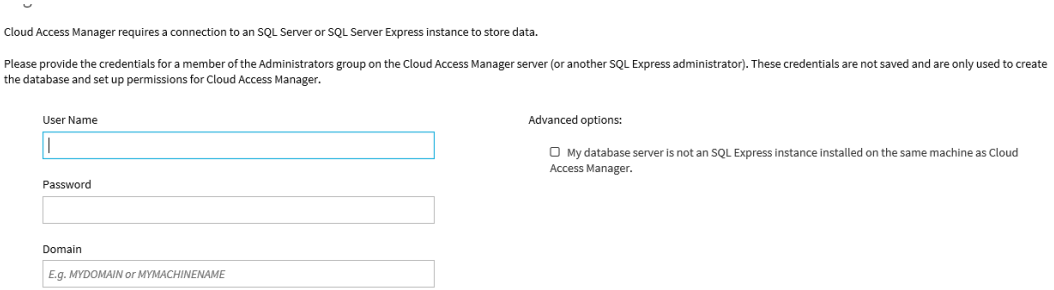
Enabling the Security Analytics Engine

The Security Analytics Engine is installed automatically when you install Cloud Access Manager, but the application is dormant until you enable it from within Cloud Access Manager.

To enable the Security Analytics Engine

1. Navigate to the **Settings** page and then to the **Externalized Access Control** page.
2. Select **Security Analytics Engine will manage step-up authentication decisions** from the list.
3. If you have upgraded from a version of Cloud Access Manager that did not include the Security Analytics Engine, and you are not using the LocalDB database, then you will be prompted to enter credentials allowing Cloud Access Manager to create a new database catalog. The credentials should be those of a member of the sysadmin role on the database server. In addition, the existing data source used by Cloud Access

Manager will be set as the default data source. You may change this data source to store data for the Security Analytics Engine in a separate database to that used for storage of the main Cloud Access Manager data.

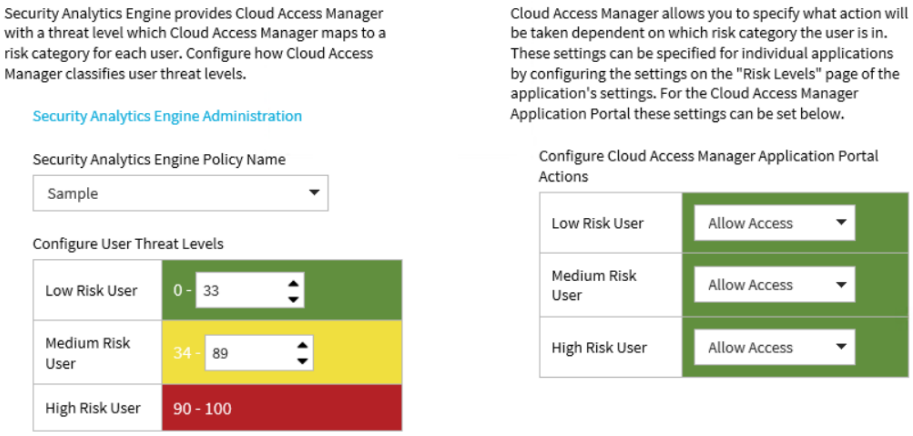


4. Cloud Access Manager then configures the Security Analytics Engine for use with Cloud Access Manager. This configuration will:
    - Configure the Security Analytics Engine to use Cloud Access Manager as a federated authenticator for allowing access to the Security Analytics Engine administration interface.

      NOTE: Users should always access the Security Analytics Engine Administration web site through the links provided in Cloud Access Manager, please refer to the section To access the Security Analytics Engine user interface on page 93.
    - Set the fallback password for the Security Analytics Engine to be the same as the fallback password for Cloud Access Manager.
    - Create a new Cloud Access Manager role named **Security Analytics Engine Helpdesk**.

      NOTE: Users assigned the Security Analytics Engine Helpdesk role should refer to the following chapters of the *Security Analytics Engine Help Desk User Guide* for information on using the Security Analytics Engine:
        - Auditing
        - Policy Overrides
        - Issued Alerts
    - Configure the Security Analytics Engine to allow members of the Cloud Access Manager Admin and Security Analytics Engine Helpdesk roles to access the Security Analytics Engine administration interface.
    - Create a Cloud Access Manager application within the Security Analytics Engine. When configuring policies within the Security Analytics Engine for use with Cloud Access Manager, you must edit policies within this application.

- On completion you will see the page shown below:



Configuring Cloud Access Manager to use the Security Analytics Engine for access control

When a user attempts to access an application that is controlled by Cloud Access Manager a query is made to the Security Analytics Engine to evaluate the risk for the user. This risk score is a value from 0 to 100 percent, Cloud Access Manager maps this to one of three threat levels for low risk, medium risk and high risk users and then uses the threat level to decide what action to take on a per application basis. The action is either:

- Allow Access — user only requires a single level of authentication
- Step Up Authentication — user needs to be two factor authenticated
- Deny Access — user cannot access the application

NOTE: For guidance on deploying Security Analytics Engine, SonicWALL and Cloud Access Manager together, please refer to the document entitled *One Identity Cloud Access Manager SonicWALL Integration Overview*.

To configure which Security Analytics Engine policy to use

Each application within the Security Analytics Engine can have multiple policies associated with it. To select which of these policies Cloud Access Manager will use as the default policy, perform the following steps:

1. Navigate to the **Settings** page and then to the **Externalized Access Control** page.
2. In the **Security Analytics Engine Policy Name** list select the required policy.

NOTE: Additional Security Analytics Engine policies can be configured from within the **Security Analytics Engine Administration** page.

To configure threat levels

1. Navigate to the **Settings** page and then to the **Externalized Access Control** page.
2. In the **Configure User Threat Levels** table set the upper bound of the risk score for low risk users.
3. In the **Configure User Threat Levels** table set the upper bound of the risk score for medium risk users.

NOTE: The lower bounds for medium and high risk users will automatically change based on the values you enter for low and medium risk users.

To configure the action to take when logging on to the Cloud Access Manager Application portal

1. Navigate to the **Settings** page and then to the **Externalized Access Control** page.
2. In the **Configure Cloud Access Manager Application Portal Actions** table, set the desired actions for low, medium and high risk users.

To configure the action to take when a user accesses an application

1. Navigate to the **Applications** page and click the name of the application that you wish to configure.
2. Click the **Threat Levels** tab.
3. You can either:
   - Select **Use the configured user threat levels to set user access to this application** to configure access based on the user threat levels configured on the **Externalized Access Control** page.
   - Select **Override configured user threat levels to set user access to this application** to configure specific threat levels for this application.
   - Select **Roles determine which users are assessed by the Security Analytics Engine** to configure specific threat levels for this application and only use the Security Analytics Engine to authorize users in particular roles.
4. If you are using standard threat levels, set the desired actions for low, medium and high risk users.
5. If you are overriding the standard threat levels:
   a. Select the Security Analytics Engine policy to use.
   b. Set the upper bound of the risk score at which users will be allowed access with a single level of authentication.
   c. Set the upper bound for the risk score at which users must be two factor authenticated to access the application.
6. If you are using roles to determine which users are assessed by the Security Analytics Engine:
   a. Configure the Security Analytics Engine policy and risk score bounds as in Step 5.

b. Select the required roles for users to be authorized by the Security Analytics Engine from the **Standard authentication roles** list, then click **Add Role** to add the role to the list of **SAE authentication roles**. Users not in roles required for authorization by Security Analytics Engine will be authorized for application access based on roles set on the **Permissions** tab.

The scores at which users will be denied access to the application will be set automatically based on the other scores.

NOTE: The default action for all threat levels is to allow access. You may edit the threat levels for multiple apps at once by using the commands on the **Options** menu on the **Applications** page.

To configure Security Analytics Engine administrators and helpdesk users

1. Navigate to the **Roles** page.
2. To add a user as a Security Analytics Engine administrator, add them to the Admin role.
3. To add a user as a Security Analytics Engine helpdesk user, add them to the Security Analytics Engine Helpdesk role.

To access the Security Analytics Engine user interface

1. From the Cloud Access Manager Administration portal, navigate to the **Settings** page and then to the **Externalized Access Control** page.
2. Click the **Security Analytics Engine Administration** link. If you are logged on as the fallback user you will be prompted to authenticate. Use the same password that you used to authenticate to Cloud Access Manager. If you are logged on as a federated user you will be taken directly to the Security Analytics Engine administration page without the need to re-authenticate.

From the Cloud Access Manager Application portal:

1. Authenticate to the Cloud Access Manager Application portal as a user who is a member of either the Admin or Security Analytics Engine Helpdesk roles.
2. Click the **Security Analytics Engine Administration** link.

Managing your SSL certificate

When you install Cloud Access Manager, a temporary self-signed certificate is created for the proxy and stored in the database. This section describes how to replace the temporary certificate with a fully signed, trusted certificate.

Obtaining a signed certificate

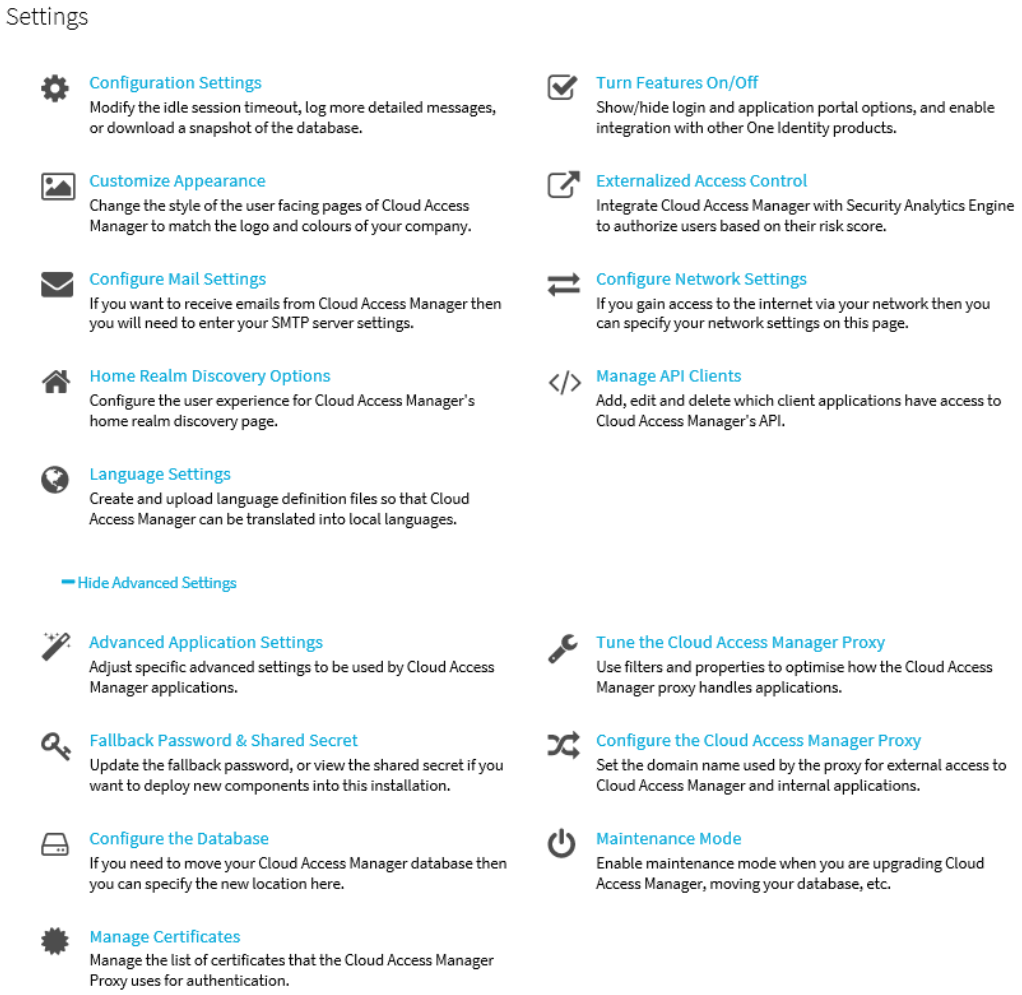
To obtain a signed certificate you must generate a Certificate Signing Request (CSR) and then install the resulting certificate as described in the following steps.

To generate a certificate signing request

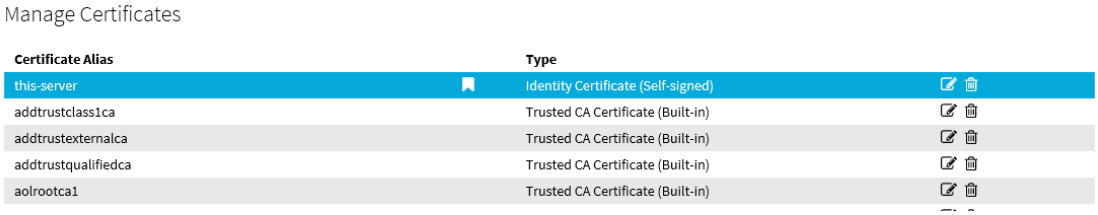
1. Log on to the **Administration Console** using the fallback login shortcut and navigate to the **Settings** page, then select **Show Advanced Settings**.

   NOTE: The **Settings** page is accessed from the gear icon.

2. Click **Manage Certificates**.



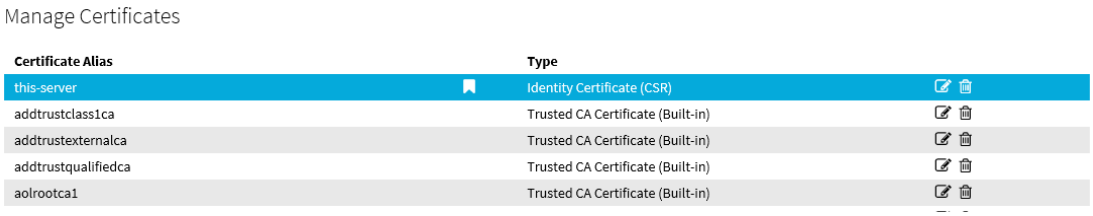
3. The **Certificates** page is displayed.
4. The proxy certificate is displayed at the top of the list of certificates with the alias **this-server**. After installation the proxy certificate is displayed as **Self signed**. If you do not already have a signed certificate to use for the proxy you will need to create a certificate signing request and submit it to your Certificate Authority. To do this, click **Generate Key Pair and CSR**.



5. Complete the **Fully Qualified Server DNS Name** field. This must match the wildcard DNS subdomain created for the Cloud Access Manager Proxy, for example, *.webapps.company.com. For further information, please refer to the Prerequisites section in the One Identity Cloud Access Manager Installation Guide.

6. If you want to specify additional information that will be displayed on your certificate, select the **Supply Additional Certificate Information** check box and complete the fields as required.
7. Click **Generate**.
8. When the certificate has been generated, click **Download Certificate** or copy and paste the information shown to a file.

   When the certificate signing request was generated, the certificate entry in the **Certificates Alias** list on the **Certificates** page changed from **Self-signed** to **CSR**. At this stage, you can click **Download CSR** to retrieve the certificate signing request if required.



9. You now need to request a wildcard Secure Sockets Layer (SSL) certificate, using the generated certificate signing request, from a Certificate Authority, for example, VERISIGN, Thawte or Go Daddy.
10. When your certificate has been signed, download the complete certificate chain in PKCS#7 format, ensuring that your Certificate Authority's root certificate, any intermediate certificates they may use, and your signed certificate are included in a single PKCS#7 certificate file.

    NOTE: If your Certificate Authority does not have a PKCS#7 complete chain option, select the option for a **Tomcat Web Server** certificate.

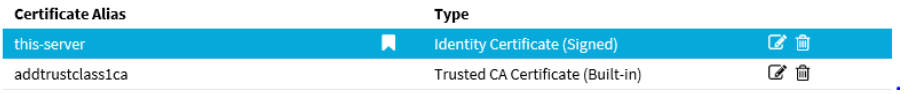
11. If you downloaded the signed certificate in PKCS#7 format containing the complete chain, on the **Certificates** page, click **Install CSR Reply**.

    If you did not download the complete certificate chain in a single PKCS#7 file, you will need to install the Certificate Authority's root certificate and any of its intermediate certificates prior to installing your signed certificate. The Certificate Authority's root certificate and any intermediate certificates are typically included in the download containing your signed certificate.

    NOTE: Cloud Access Manager will only support base64 encoded certificates, with the exception of importing a PKCS12 for **this-server**, both .crt and .cer files can be either PEM encoded (base64) or DER encoded (raw binary file), Cloud Access Manager will only support them if they are PEM encoded.

    Depending on your Certificate Authority, you may be given a separate root certificate and an intermediate certificate or a bundle containing both the root and intermediate certificates. To install these, use the **Install Trusted CA Certificate** option on the **Certificates** page. When these have been installed, click **Install CSR Reply** from the **Certificates** page to install your signed certificate.
12. Click **Save**. When the certificate has been installed, it is displayed in the

Certificates Alias list as signed.



Replacing an expiring certificate

You can create a new certificate signing request before your current certificate expires.

To replace an expiring certificate, from the **Certificates** page, click **Generate Expiry Key Pair and CSR**. The procedure for generating the replacement certificate is the same as when you created the original certificate, refer to Obtaining a signed certificate on page 94. Your current certificate is only overwritten when the replacement certificate is fully signed.

Installing a fully signed certificate from a certificate archive file

If you already have a signed certificate to use for the proxy, from the **Certificates** page, click **Import PKCS12 / PFX file**, and upload the certificate.

Installing a certificate authority certificate

To install a certificate authority certificate

1. From the **Certificates** page, click **Install Trusted CA Certificate**, specify the certificate alias in the **Certificate Alias** field, click **Browse** to import the public certificate from a file.
2. Click **Save** to install the certificate.

NOTE: Cloud Access Manager will only support base64 encoded certificates, with the exception of importing a PKCS12 for **this-server**, both .crt and .cer files can be either PEM encoded (base64) or DER encoded (raw binary file), Cloud Access Manager will only support them if they are PEM encoded.

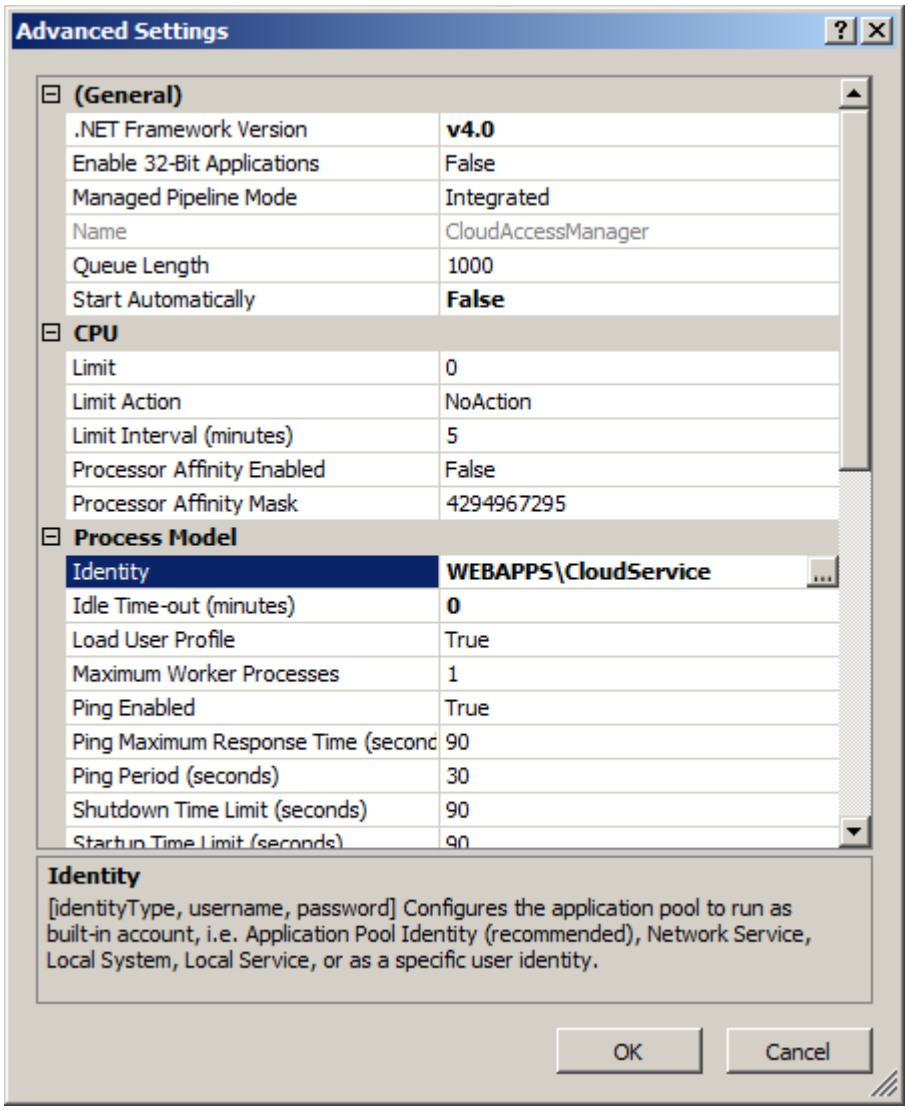
Changing the Cloud Access Manager service account password

If Cloud Access Manager is configured using an account with a password that is allowed to expire, when the password has expired you will need to reconfigure Cloud Access Manager with a new password. To do this, complete all three stages of the procedure described below.

Cloud Access Manager IIS Application Pool

To set the password for the One Identity Cloud Access Manager IIS Application Pool

1. Start **Internet Information Services (IIS) Manager**.
2. Expand the **Connections** tree on the left and select **Application Pools**.
3. In the list of Application Pools, click the **CloudAccessManager** entry and then select **Advanced Settings...** from the **Actions** menu.
4. In the **Advanced Settings** dialog, click **Identity** and then the button that is displayed containing the ellipsis.

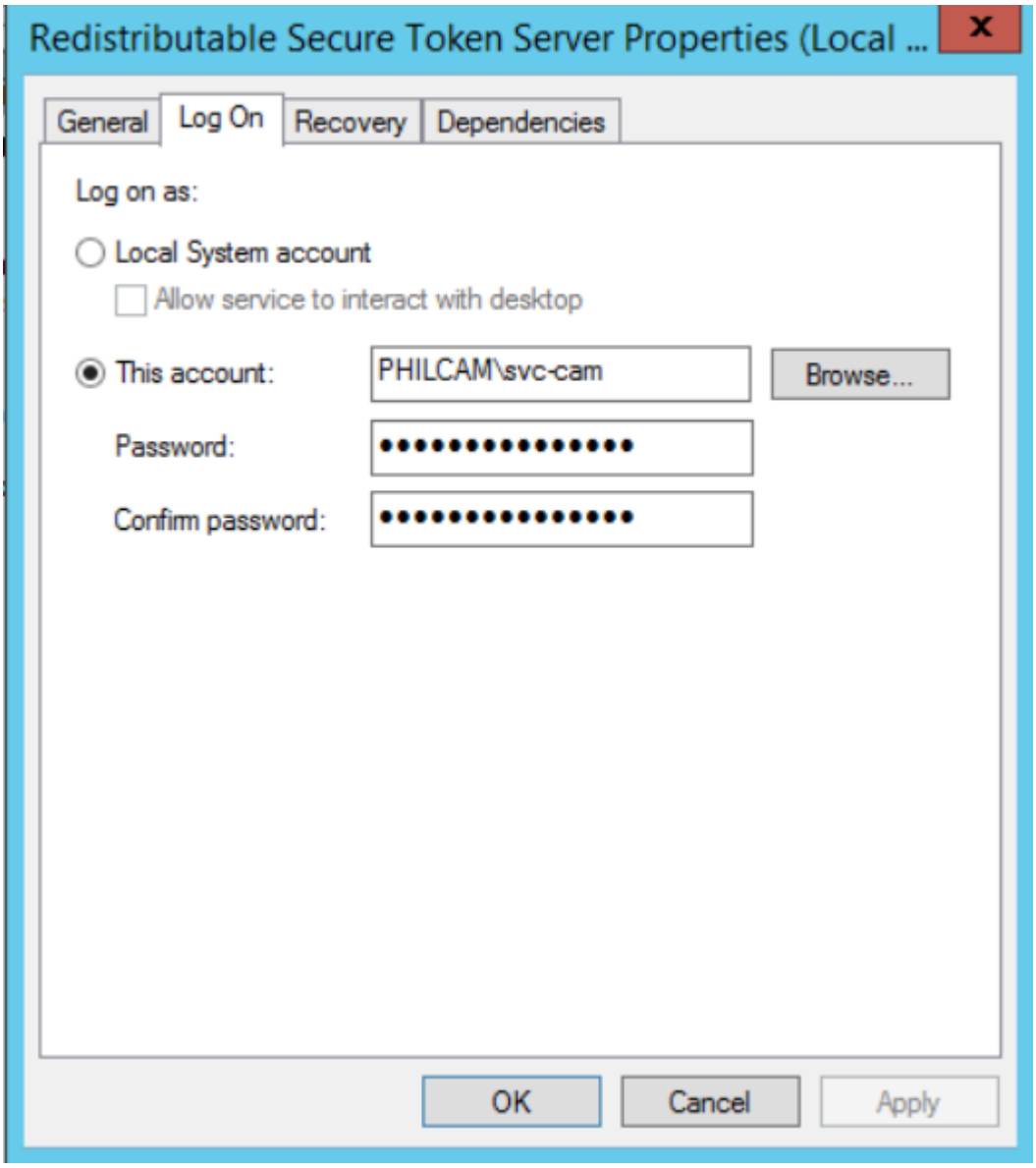


5. In the **Application Pool Identity** dialog, click **Set...** and then set the new credentials to use.
6. Close all dialogs. In the list of **Application Pools,** click **Start** on the **Actions** menu to start the application pool.

Redistributable Secure Token Server

To set the password for the Redistributable Secure Token Server

1. Start the **Windows Services Manager** (services.msc).
2. Right click on the **Redistributable Secure Token Server** service and select **Properties.**
3. Switch to the **Log On** tab and set the new password.



4. Click **Apply**.
5. Switch back to the **General** tab and click **Start** to start the service.
6. Close the dialog.

Front-end authenticators

To set the password for any front-end authenticators that use the service account

1. Log in to Cloud Access Manager using the **Cloud Access Manager Administration (fallback login)**.
2. Select **View and Edit** in the **Front-end Authentication** section on the homepage.
3. For each authenticator of type **Microsoft Active Directory** that uses the service account, click the **edit** button and set the password on the **Connection Settings** page.

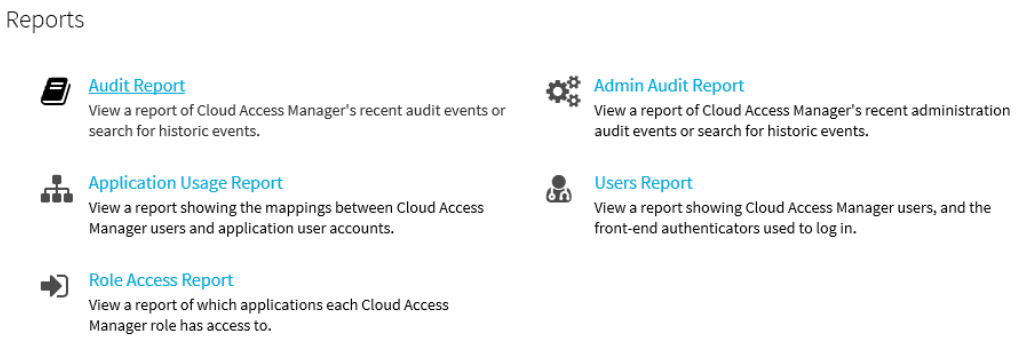
6

Reporting

Cloud Access Manager provides a number of reports to display audit events, usage, and statistics. To access Reports, log in to the Administration Console using the desktop shortcut **Cloud Access Manager Administration** and select **Reports** on the home page.

All reports can be exported in the following formats:

- Comma-delimited (.csv)
- Microsoft Excel (.xls)
- Microsoft Word (.doc)
- PDF (.pdf)



Audit report

Allows you to view successful and failed user audit events for the following request types:

- Authentication
- Access grant
- Provisioning
- Deprovisioning

Date and variable filters can be applied to search for specific events.

Admin audit report

The admin audit report enables you to view successful administration audit events. The report lists only changes made through the Cloud Access Manager Administration console and indicates whether the changes were add, modify or delete operations. Changes to the following Cloud Access Manager elements are reported:

- Application
- Front-end authenticator
- Role
- Provisioned user
- License
- Setting
- User
- Proxy
- Custom File.

Date and variable filters can be applied to search for specific events.

Application usage report

The application usage report displays:

- Which users have been provisioned and/or signed in to an application through Cloud Access Manager
- When the users last signed in to the application through Cloud Access Manager.

Date and text search filters can be applied to search for specific users within each application.

Users report

The users report displays the:

- Front-end authenticators each user has used to log in to One Identity Cloud Access Manager
- Last time the user logged in.

Date and text search filters can be applied to search for specific users within each front-end authenticator.

Role access report

The role access report displays which applications each Cloud Access Manager role has access to. This report is an amalgamation of the **Permissions** page from each application configuration.

NOTE: Specific settings for front-end authenticators and applications alongside dynamic user variables may also affect whether a role gains access to an application. Text search and variable filters can be applied to search for roles and/or applications.

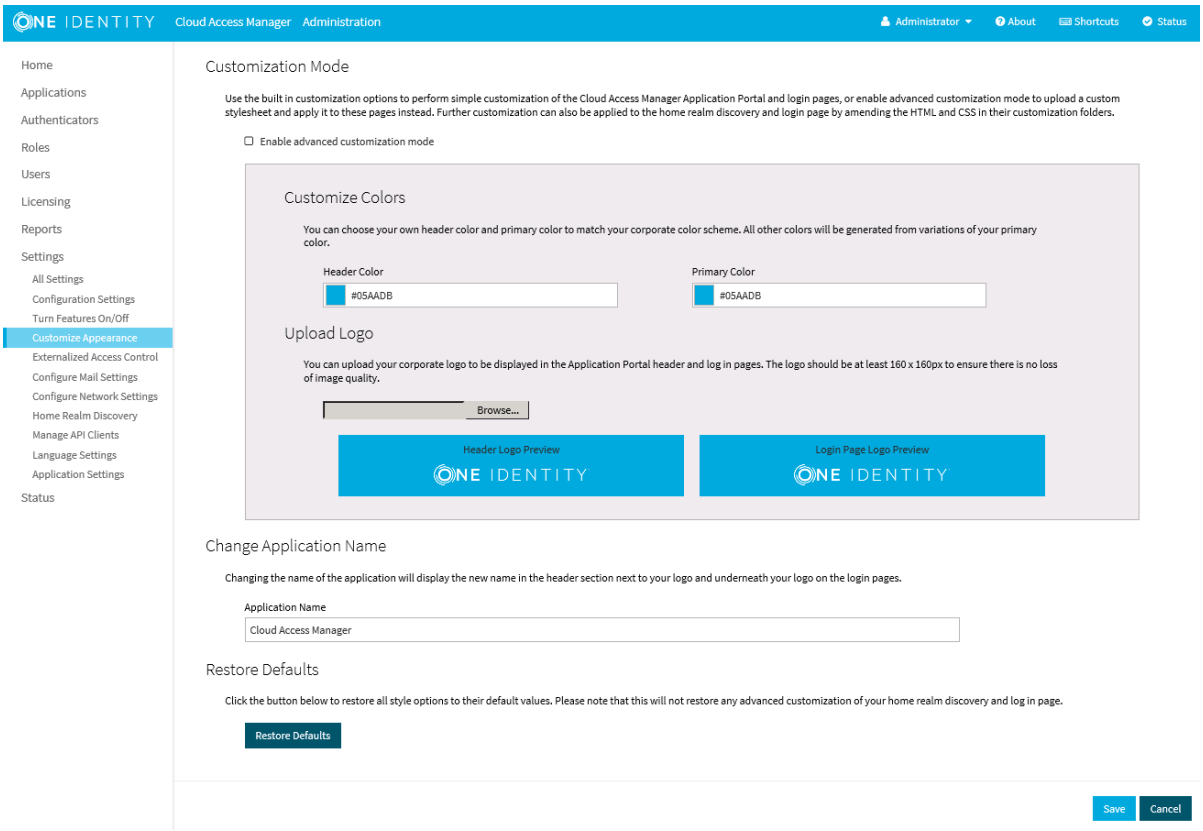
7

Customizing One Identity Cloud Access Manager

You can customize the appearance of Cloud Access Manager to meet the needs of your users and to match your corporate branding.

You can easily change common aspects of the look and feel using the **Customize Appearance** options in the Cloud Access Manager Administration User Interface (UI). For example, you can change the colors, company name and logo in the Admin UI. For more extensive changes, you can manually edit the Cascading Style Sheet (CSS) file that is used to style Cloud Access Manager from the Admin UI. If you cannot achieve the look you require by editing the CSS file, you can also edit the HTML of the Login and Home Realm Discovery (HRD) pages.

You can find the customization settings in the Admin UI under **Settings | Customize Appearance**.



To manually edit the CSS file, select **Enable advanced customization mode** in the Admin UI. You will then be able to download and upload the CSS file.

If you want to manually edit the HTML for the Login and HRD pages, you need to log in to each of the Security Token Service (STS) hosts and edit the following files manually:

- C:\Program Files\One Identity\Cloud Access Manager\Customization\Login\Login.htm
- C:\Program Files\One Identity\Cloud Access Manager\Customization\HRD\UserIdentity.html

When you edit these HTML files, you must make sure that the changes are applied to each STS host. If you need to include JavaScript or image files, you can place these files in the same directory and reference them using a relative path.

About us

Contacting us

For sales or other inquiries, visit https://www.oneidentity.com/company/contact-us.aspx or call +1-800-306-9329.

Technical support resources

Technical support is available to One Identity customers with a valid maintenance contract and customers who have trial versions. You can access the Support Portal at https://support.oneidentity.com/.

The Support Portal provides self-help tools you can use to solve problems quickly and independently, 24 hours a day, 365 days a year. The Support Portal enables you to:

- Submit and manage a Service Request
- View Knowledge Base articles
- Sign up for product notifications
- Download software and technical documentation
- View how-to-videos
- Engage in community discussions
- Chat with support engineers online
- View services to assist you with your product