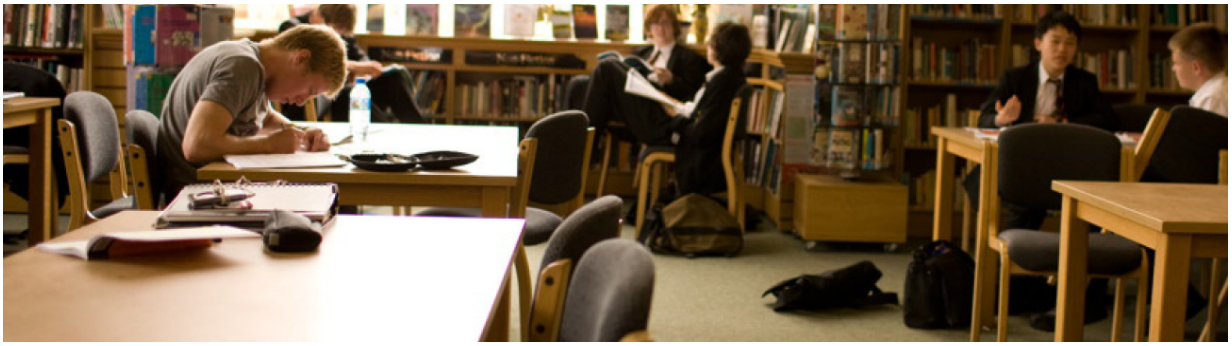
White Paper

Deploying OAuth with Cisco Collaboration Solution Release 12.0

Authors: Bryan Morris, Kevin Roarty (Collaboration Technical Marketing)

Last Updated: December 2017



This document describes the new OAuth deployment mode available with Unified Communications Manager, IM and Presence Server, Cisco Jabber and Expressway.

Introduction

This whitepaper has been created to help administrators understand the support for the OAuth standard in Cisco's collaboration solution. The reader will learn what OAuth is, the benefits of OAuth for their organization, what is required to use OAuth and the user experience OAuth delivers for Cisco Jabber users.

What is OAuth

OAuth is an authorization protocol. It is an open standard defined by the IETF OAuth Working group which was originally released in 2007. In 2010 OAuth 2.0 was released as RFC6749 which is the current version of the standard.

OAuth allows an end user to authorize an application to gain access to a third party service without sharing their credentials with the application. To grant access to a third party service a user authorizes an OAuth server via authentication to issue OAuth tokens to the third party application. The application can now present the OAuth token to access a protected resource rather than user credentials. OAuth tokens will expire after a period of time thus limiting the time the 3rd party application can access the resource. In some implementations OAuth can provide a method to refresh an expired token to provide continued access to information or a service.

How does OAuth Work

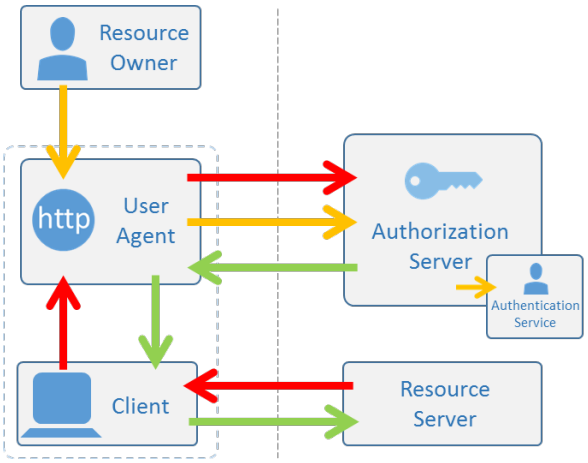
OAuth is heavily used on the Internet today. If we consider an example, it is a common scenario for an end user to authorize a 3rd party website (such as a travel site) to access information on a social media site (such as Facebook or Twitter). In this case the user typically clicks an "allow access to social media" button to authorize access to information (such as a contact). This will result in a web page for the social media site to be opened. The user will need to confirm their identity (Authentication) and maybe approve what information can be accessed. On a successful authentication the social media site allows an access token to be issued to the 3rd party using OAuth. The key benefit here is the user never gave their authentication credentials to the 3rd party. These were kept secret between the social media site and the user. The token can be defined so it has a limited scope, for example it can be used to view contacts on the social media site but doesn't allow to post information. Finally the token can be valid for a predefined duration.

The OAuth protocol is a framework specification. OAuth can be compared to a toolbox of authorization functions. The OAuth standard defines a protocol "**Flow**" where defined "**Roles**" take part in the authorization process. The OAuth roles are:

- Resource Owner (End User)
- Resource Server (i.e. Unified CM)
- Client (i.e. Cisco Jabber/User Agent)
- Authorization Server (i.e. Unified CM OAuth)

There are multiple OAuth flows, this diagram provides a summary of the flow used by Unified CM.

1. Resource server redirects client to authorization server
2. Resource owner required to authenticate to grant access
3. Client authorized to access resource server



When using the Cisco Jabber UC client we need to access multiple services offered by the Collaboration infrastructure. We need to access configuration information, instant message service, call control and voicemail. If the Collaboration infrastructure is configured to use OAuth, the Jabber client only has to authenticate once to get an OAuth token. Jabber will then use that token to access all these services. Only when the token expires do we need to authenticate again. This provides a more secure solution as the Jabber application never needs to know the user password. Jabber is also only authorized to access the services it needs using the token.
When talking about OAuth it is important to understand the difference between authorization and authentication. OAuth is a standard which supports authorization. A user must be authenticated before they can be authorized. Before granting authorization the OAuth authorization service will normally call or redirect to an authentication service such as a user database, LDAP directory or SAML 2.0 based Identity Provider (IdP).

Authentication

Authentication is the process of confirming a person (or thing's) identity. Traditionally this is using a username and password but could use a certificate or other proof of identity. Increasingly modern systems require multi-factor authentication where multiple proofs of identity are required. Authentication doesn't define what a user can do but just that they are the correct person. We can compare this to a hotel check-in: when you arrive at the hotel they will ask for proof of identity. This could be a passport, driving license or other document that can confirm your identity.

Authorization

Authorization is the process of defining access rights or privileges to an entity. If we again compare this to a hotel check-in, the hotel will authorize you to access a hotel room by providing you with a room key once they have confirmed your identity. The room key may provide you with access to additional facilities in the hotel such as the gym or swimming pool. You are not required to prove your identity again once you have the room key. Furthermore, anybody owning the room key can get access to the room using that key.

OAuth Flows

An authorization request is a set of interactions between the OAuth roles. OAuth provides different interaction models or "Flows" depending on the operating environment. OAuth provides the following protocol flows:

- Resource Owner Password Credentials Flow
- Client Credentials Flow
- Authorization Code Grant Flow
- Implicit Flow

The OAuth specification makes recommendations for when a developer should use each of these flows.

Cisco Unified Communications Manager implements the “Implicit” and “Authorization Code Grant” flows. The implicit flow was introduced with Unified CM 10.5(2) and the Authorization Code Grant flow was introduced with Unified CM 11.5(1) SU3.

Cisco recommends using Authorization Code Grant flow for 11.5(1) SU3 and above and the remainder of this document will focus on this OAuth flow.

Why use OAuth for Authorization

OAuth provides a number of benefits to an organization. In this section we examine the benefits important to different user types.

Why OAuth, the Benefits for… (By job role)

*… **the Information Security Officer***

- A user is not required to share credentials with a 3rd party application.
- Reduction in security attack surface.
- Allows for stronger authentication methods (multi-factor, biometric) when combining OAuth with SAML 2.0 based single sign-on.

… the UC administrator

- Information Security Officer accepted security
- Reduction in password support cases
- Allows for Expressway MRA user policy

… the End User

- Not required to authenticate when the client is restarted
- Authentication doesn’t fail when the password is changed
- Faster login process once authenticated

Cisco Collaboration Support for OAuth

Unified CM provides two different OAuth models.

Unified CM Pre 12.0

Unified CM has supported the OAuth protocol since release 10.5. In these earlier releases OAuth is only used when SAML 2.0-based single sign-on (SSO) has been deployed, and the grant flow used is the Implicit Flow.

OAuth Implicit Flow

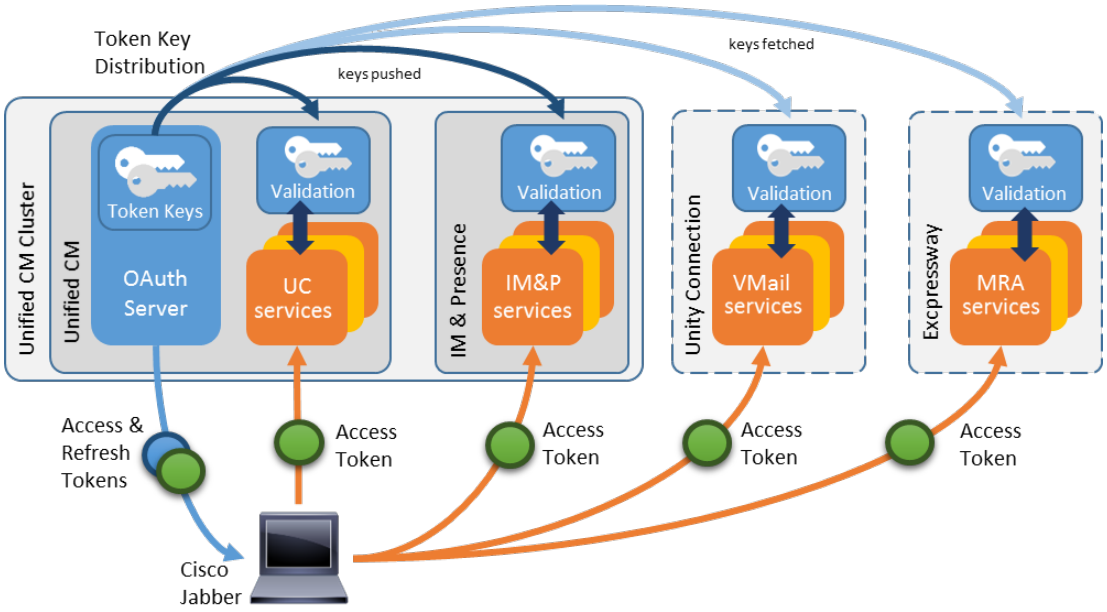
The implicit grant flow provides a method for a client to request authorization to access a resource server. (Unified CM, IM&P, Unity and Expressway services). The client will make the initial request to the authorization server using HTTPS. The OAuth server redirects the user via a web browser application to an external Identity Provider (IdP). Depending on the authentication method requested by the IdP the native OS web browser may be presented to the end user to authenticate themselves. A successful authentication will result in an “Access Token” being issued to the native OS web browser which is passed back to the client. The client then uses this token to access services. When the token expires the full OAuth/Authentication process must be repeated. The default lifetime of an access token is 60 minutes.

Unified CM 12.0 and later **(also back ported to 11.5.1SU3 and later releases)**

The 12.0 Collaboration architecture has been enhanced to provide support for OAuth with refresh tokens. OAuth is now also supported regardless of the user authentication method deployed. OAuth authorization can work with Local User, LDAP and SAML SSO based authentication models. All Unified CM nodes run the OAuth authorization service. Other infrastructure nodes (IM&P, Unity Connection and Expressway) are also able to validate tokens issued by Unified CM servers. The new architecture implements the OAuth Authorization Code grant flow, which supports access and refresh tokens. Refresh tokens allow new access tokens to be obtained without repeated authentication for the validity period of the refresh token. Access and refresh tokens are encrypted/signed by the Unified CM OAuth authorization service. Unified CM OAuth tokens are self-contained so they can be validated by other infrastructure nodes without requests being made to the OAuth server.

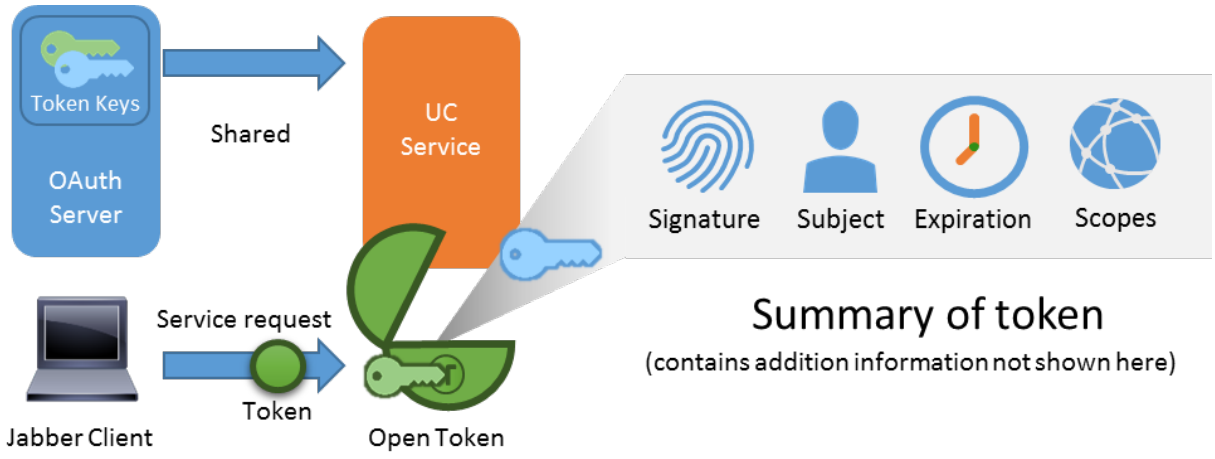
(Note: The 12.0 architecture is also able to support the pre 12.0 OAuth model for backward compatibility)

The following diagram shows the 12.0 OAuth architecture



The diagram above shows how the OAuth server generates a set of encryption and signing keys used for signing and encrypting OAuth tokens. These keys are automatically distributed within Unified CM clusters to call control and IM and presence nodes. Unity connection and Expressway-C servers are also able to fetch encryption and signing keys using a REST API. Once a server is in possession of the signing/encryption key set it is able to validate and decrypt access tokens presented by Cisco Jabber clients.

When a token is presented to a UC service it is decrypted and its signature is checked using the keys shared by the OAuth server. The tokens are self-contained, meaning that the UC service trusts tokens it can open and validate without communicating with the OAuth server. Tokens use the RFC-based JSON Web Token (JWT) format.

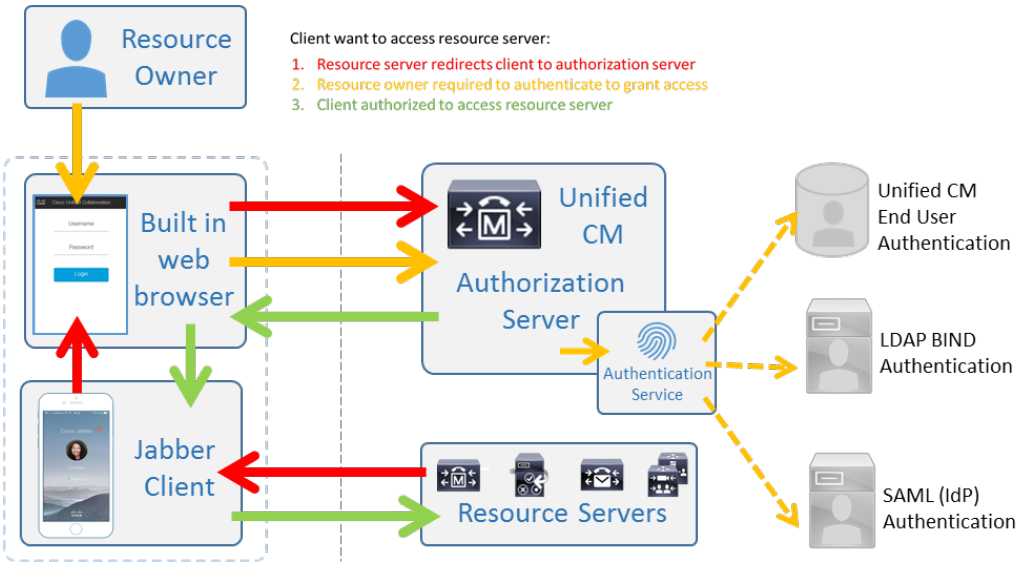


Tokens include a number of different data fields. This include:

Signature: Token signed/issued by
Subject: The user/device the token was issued to (Cisco Jabber user)
Expiration: When this token will expire

Scopes: What services this token is valid for (covered later in this document)

The following diagram shows the collaboration components mapped to the OAuth flow.



When OAuth has been enabled on a Cisco collaboration environment the following services are secured using OAuth.

	Unified CM HTTPS	Unified CM SIP	IM & Presence XMPP	Unity Connection HTTPS
Jabber **(corporate network)**	OAuth Token	Direct Connection ǂ	OAuth Token	OAuth Token
Jabber **(Mobile Remote Access)**	OAuth Token	OAuth Token	OAuth Token	OAuth Token

ǂ Certificate-based authorization can be used when Unified CM is running in mixed mode
SIP authorization for devices on the corporate network is planned for a future release of Unified CM

OAuth Authorization Code Grant Flow

The 12.0 OAuth architecture has been built on the OAuth Authorization Code grant flow rather than the implicit grant flow supported by previous versions of Unified CM.

The authorization code grant flow provides a method for a client to obtain access and refresh tokens to access a resource (Unified CM, IM&P, Unity and Expressway services). This flow is also based on redirection and thus requires the client to be able to interact with an HTTP user-agent (web browser) controlled by the user. The client will make an initial request to the authorization server using HTTPS. The OAuth server redirects the user to an authentication service. This may be running on Unified CM or an external IdP if SAML SSO is enabled. Depending on the authentication method being used, a web page view may be presented to the end user to authenticate themselves. (Kerberos authentication is an example that would not display a web page.) Unlike the implicit grant flow, a successful authentication code grant flow will result in the OAuth servers issuing an “Authorization

Code" to the web browser. This is a one-use, short-lived unique code that is then passed back from the web browser to the client. The client provides this "Authorization Code" to the authorization server together with a pre-shared secret and receives in exchange an "Access Token" and a "Refresh Token". The client secret used in this step enables the authorization service to limit the use to only registered and authenticated clients. The tokens are used for the following purposes:

Access Token: This token is issued by the authorization server. The client presents the token to a resource server when it needs to access protected resources on that server. The resource server is able to validate the token and trusts connections using the token. (Cisco access tokens default to a lifetime of 60 minutes)

Refresh Token This token again is issued by the authorization server. The client presents this token to the authorization server together with the client secret when the access token has expired or is due to expire. If the refresh token is still valid then the authorization server will issue a new access token without requiring another authentication. (Cisco refresh tokens default to a lifetime of 60 days). If the refresh token has expired then a new full OAuth authorization code grant flow has to be initiated to obtain new tokens.

Why OAuth Authorization Code Grant Flow is better

Unified CM has migrated from the OAuth implicit grant to the OAuth authorization code grant flow for a number of reasons:

1. In the implicit grant flow the access token is passed to the Jabber client via a HTTP user agent (browser). In the authorization code grant flow the access token is exchanged directly between the authorization server and the Jabber client. The token is requested from the authorization server using a time-limited unique authorization code. This direct exchange of the access token is more secure and reduces risk exposure.

2. The OAuth authorization code grant flow supports the use of refresh tokens. This delivers a better experience to the end user since they don't need to re-authenticate as frequently (by default 60 days)

3. As the user is authenticating less frequently users are more accepting of stronger/multi-factor authentication schemes.

4. The OAuth authorization code grant also reduces the load on the authentication/Identity provider, as the refresh token is used to gain new access tokens.

Better together

Unified CM 12.0 includes a number of features which work together to enhance the user experience for users of Jabber and support infrastructure.

Solution Level Authentication

OAuth provides the ability for a Jabber user to authenticate once and gain access to services of Unified CM, IM&P server, Expressway and Unity Connection. Access to services uses OAuth tokens, and user credentials are only required when a user's refresh token has expired. Token based access allows Jabber to reconnect to services much faster than previous releases.

Fast Login

When Jabber starts up in Fast Login mode it restores its contact and configuration information from a local encrypted data store. This makes reconnection times much faster than previous releases. Fast login also introduces a background configuration refresh process, which updates configuration once the client has connected to UC services.

Offline Login

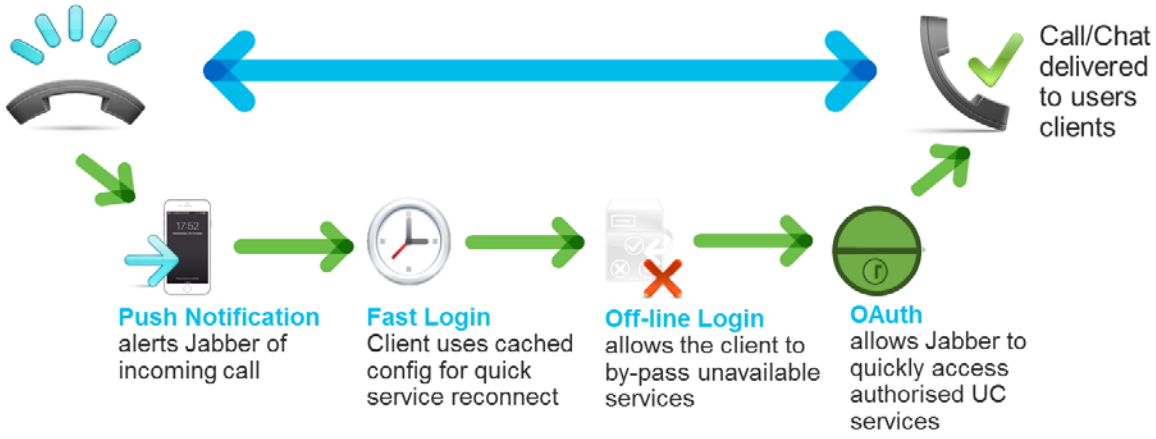
Offline login allows Jabber to handle offline services without affecting operation services. If a Jabber user was provisioned for chat, voice, video, voicemail and conferencing but on startup instant messaging services were offline, using offline login Jabber will allow users to still connect to the services which were available. Without offline login for Jabber, the IM service being offline would block voice and video services starting.

Apple Push notification

Support for Apple iOS Push notifications has been introduced into Jabber to align with upcoming changes expected in Apple iOS. When using push notifications for incoming chat and voice/video calls the notification is sent to the tablet/phone via the Apple cloud service. Push notification applications typically send less network traffic which can result in better battery life.

Example

The following provides an example of the 12.0 architecture components working together. In this case an Apple push notification is sent to an iOS device. This starts the Jabber application which needs to connect to UC services. Jabber restores its configuration from the local encrypted storage. Jabber by-passes any offline or inaccessible services. Jabber is authorized using OAuth tokens and finally presents a call to the user. This happens in a matter of seconds using the combination of new features introduced with the 12.0 architecture.



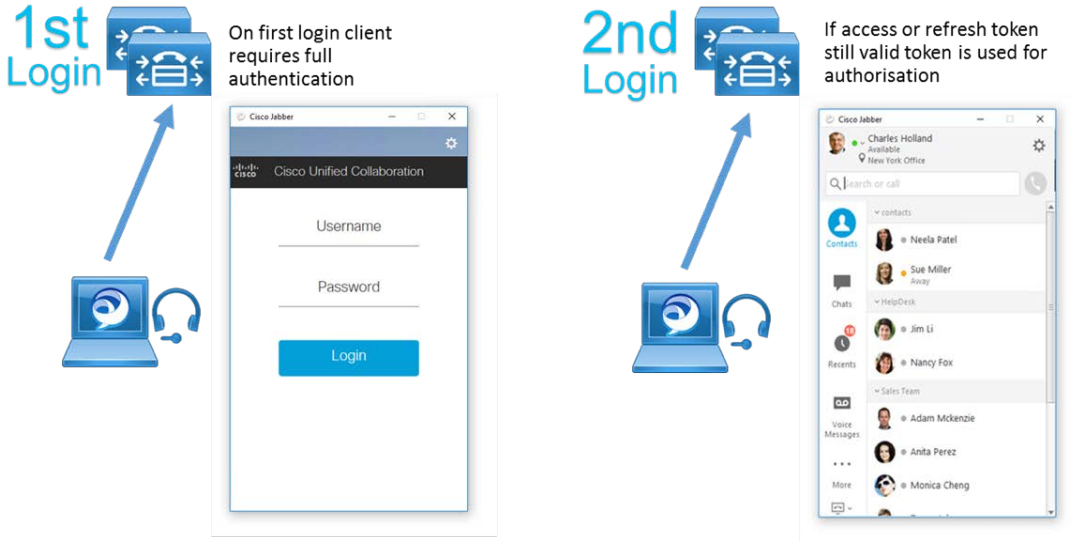
End User Experience with OAuth

Authentication Experience

Jabber users will experience a user friendly and streamlined login experience within a UC environment configured for OAuth.

Token based Authorization Experience

When using OAuth the connection to Unified CM services is based on presenting a valid access token, the Jabber login screen will only be displayed if the user needs to get a new access and refresh token pair. If the client is already in possession of a valid access token or a valid refresh token, which it can use to obtain a valid access token, the user is taken straight to their contact list when starting the Jabber client.



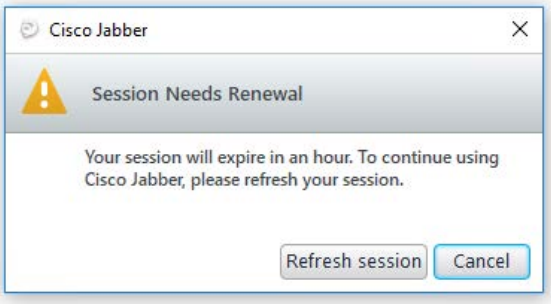
Token Refresh Experience

Tokens need to be refreshed when they expire. The Jabber client will manage the refreshing of tokens and only prompt the user for authentication when required. Jabber will typically

refresh an access token once it reaches 75% of its validity duration (i.e. a token with validity of one hour could be refreshed after 45 minutes)

Access Token: If the Access token expires but a valid refresh token still exists then the Jabber client will automatically refresh the access token using the refresh token in the background.

Refresh Token: When the Refresh token expires then a new OAuth authorization code grant flow will start from scratch and the user will be required to authenticate again. The Jabber client will advise the end user when the token is close to expiring. A box similar to the one shown below will be displayed, asking this user to "Refresh Session". Clicking the button will restart the authentication process.



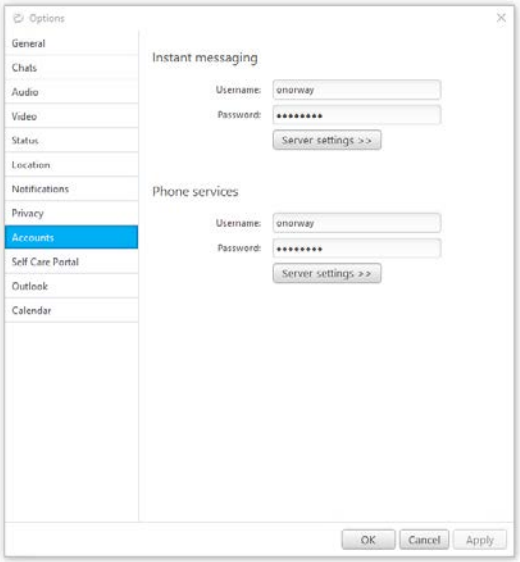
In a default operation (60 day refresh token) the following logic is used:

- First prompt when the refresh token has 3 days left to expire
- Second prompt is shown 24 hours later
- Third prompt is shown 24 hours later
- Fourth prompt is shown 1 hour before the token expires

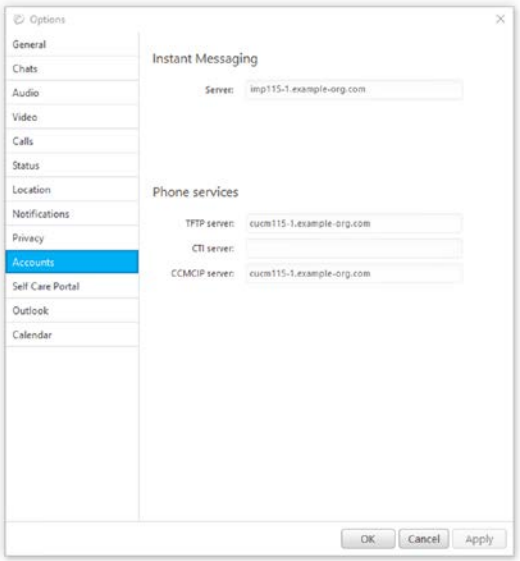
If the token has less than 3 days to expiry when the client is logged in the prompt is also displayed.

If the user is not logged in when the Refresh Token expires they will be asked to complete authentication at next login.
The default lifetime of an access token is 60 minutes and the lifetime of a refresh token defaults to 60 days. An administrator has the option to modify the lifetime of both the access and refresh tokens. (Information on how to change token lifetime is covered later in this paper)

Account Credentials
If all UC services are enabled for OAuth (including Cisco Unity Connection) the accounts tab in the options windows no longer provides the ability to update credential information. If Unity Connection has not been enabled for OAuth then only the Unity Connection credentials will be displayed and managed using the Accounts tab in Jabber.



Default accounts tab



OAuth accounts tab

How to Enable OAuth

The following sections discuss how to enabled OAuth within a Unified CM environment. All Unified CM servers, IM and Presence servers and Expressways must be upgraded to a suitable release before enabling OAuth.

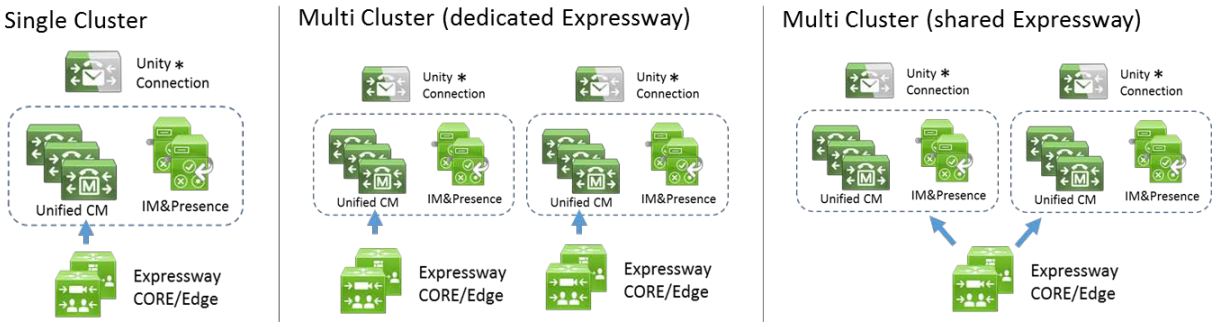
Required Infrastructure

- Cisco Unified Communications Manager 11.5.1(SU3) and later
- Cisco Unified Communications Manager IM and Presence Service 11.5.1(SU3) and later
- Cisco Expressway X8.10.1 and later
- Cisco Unity Connection Server 11.5.1(SU3)

Note: Cisco recommends all components are enabled for OAuth to deliver the best user experience

Deployment models

OAuth can be enabled for different Unified CM deployment models. A summary of deployments are shown below.



Single Cluster Architecture
In a single cluster, OAuth is enabled for Unified CM and IM&P using an enterprise parameter. Cisco Unity Connection can optionally be enabled for OAuth if running an OAuth supported release. If Expressway is providing Mobile and Remote access to users in this Unified CM cluster, then Expressway MUST be running a software release that supports and is enabled for OAuth. In the single cluster model the authorization method configured on Unified CM must match the method configured on Expressway. (i.e both using OAuth)

Multi Cluster (dedicated Expressway) Architecture
This model is the same a single cluster. In this model Expressway servers are dedicated to each Unified CM cluster. The Expressway authorization configuration must align with the Unified CM authorization configuration.

Multi Cluster (shared Expressway) Architecture
In this model the Expressway servers are shared across Unified CM clusters. If both Unified CM clusters are enabled for OAuth then OAuth must be enabled on the Expressway. If only one of the Unified CM clusters is enabled for OAuth, then "Check for internal authentication availability" needs to also be enabled on the Expressway. This will cause the Expressway to query clusters to confirm what kind of authentication and authorization are available for each user based on their home Unified CM cluster configuration. The recommended configuration is to enable the "fixed" authorization method whenever possible which is the default setting for Expressway.

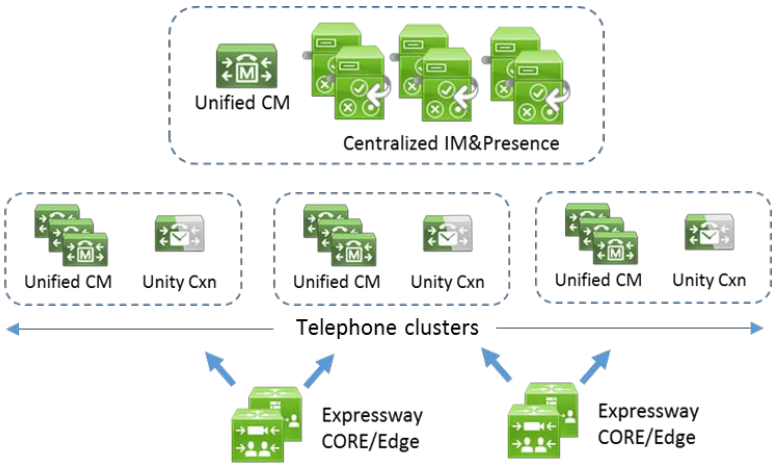
The diagram below shows the two methods of operation.

When the authorization discovery method is used, the Jabber client will identify itself to Expressway. Expressway will use this information to identify the correct home cluster for the client. Once the user's home cluster is known, Expressway will query the home cluster to identify the authorization method to use for this user. Expressway will then proceed to authorize the user with the correct method.

When the fixed authorization method is used the Expressway is pre-configured to use the same method of authorization for all clusters. This method is preferred as Expressway does not generate traffic on the internal network from Expressway before the client is authenticated.

Authorization discovery should normally only be enabled while Unified CM is being migrated to code release that allows OAuth to be enabled. Once all clusters are enabled for OAuth then authorization discovery should be disabled.

Distributed Telephony clusters with centralized IM&P cluster



This final model was introduced with Unified CM 12.0/11.5.1(SU4) and allows for clusters to provide different UC services. In this model a Jabber client would consume telephony services from a telephony cluster and IM services from the central cluster. Jabber clients always connect to a telephony cluster first and then to the central cluster. In this model all clusters must be configured to use the same method of authentication. The centralized cluster must also be populated with the signing and encryption keys from each of the telephony cluster to allow the central IM&P nodes to trust tokens from the telephony clusters.

Recommended Enablement Sequence

It is recommended that the administrator uses the following sequence to enable their deployment for OAuth support:

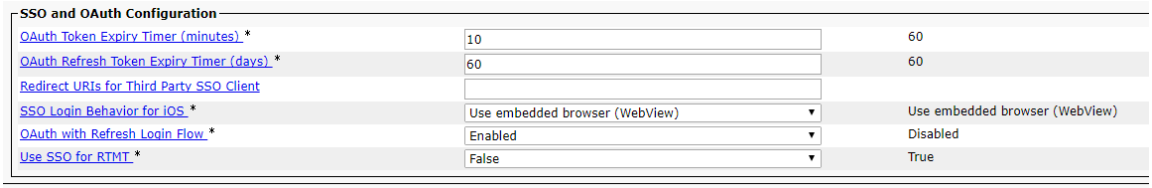
- Unified CM / IM and Presence nodes
- Unity Connection nodes
- Expressway nodes

The following information covers how to enable OAuth for each of the server types.

Enabling Unified CM & IM&P

To enable OAuth perform the following procedure

1. Go to Cisco Unified Communications Manager Admin > System > Enterprise Parameters > SSO and OAuth Configuration
2. "Select OAuth with Refresh Login Flow" set Enable/Disable support OAuth feature
3. Set "OAuth Access Token Expiry Timer (minutes)"
4. Set "OAuth Refresh Token Expiry Timer (days)"
5. Click "Save" button, OAuth will be effective immediately



Once the Unified CM is upgraded to 11.5(1) SU3 or later, the authorization service is available even without enabling the “OAuth with Refresh Login Flow" enterprise parameter. This parameter only controls the API response that Jabber and Expressway can use to determine whether or not the OAuth with refresh login flow should be attempted.

Enabling Cisco Unified Communications Manager IM and Presence Service

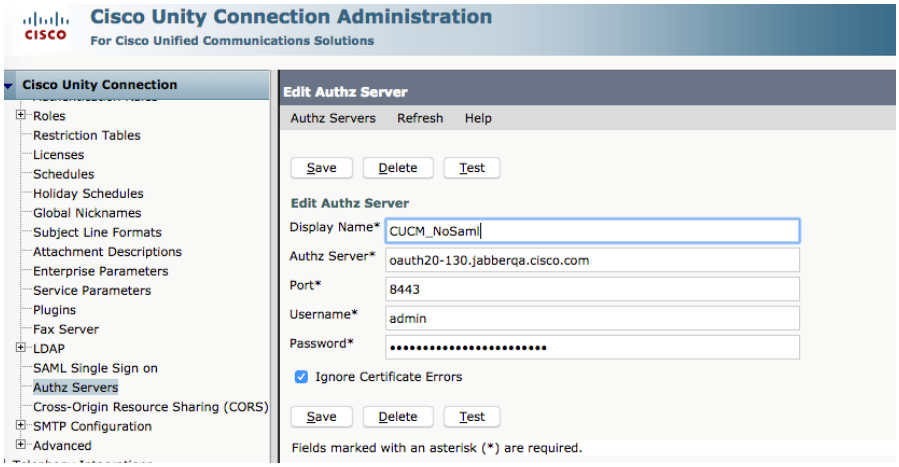
The IM and Presence service will follow the coupled Unified CM OAuth configuration. No specific action is required on the IM and P service.
Note: when deploying distributed telephony clusters with centralized IM&P cluster, additional configuration is required on the IM&P cluster to fetch the OAuth token keys from the telephony clusters, but that is beyond the scope of this document.

Enabling Unity Connection

To enable Unity Connection for OAuth support, the administrator must perform two steps.

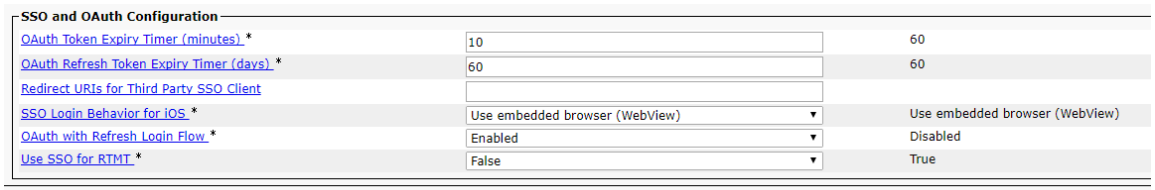
1) Configure the Unity connection Server to fetch OAuth token signing and encryption keys from Unified CM.
2) Enable OAuth services on the Unity connection server

To fetch the signing and encryption keys Unity must be configured with the Unified CM host details and a user account enabled for Unified CM AXL access
Go to AuthZ Servers > Add New



After fetching the signing and encryption keys from Unified CM, the Unity Connection server must be enabled for OAuth. This is configured as follows.

1. Go to System > Enterprise Parameters> SSO and OAuth Configuration
2. Select “OAuth with Refresh Login Flow" to enable OAuth with refresh support

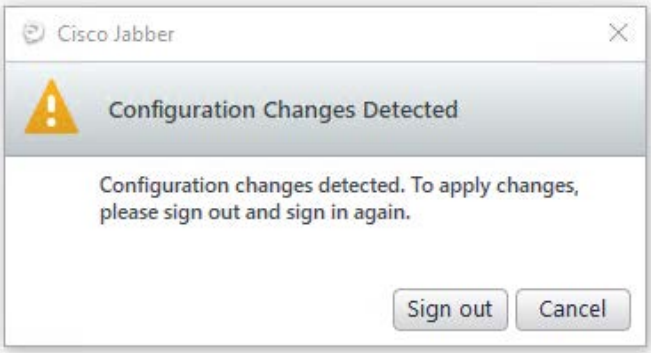


Enabling Cisco Jabber Client

There are no configuration changes required for Cisco Jabber. The Cisco Jabber 11.9 (or later) client will automatically detect that OAuth has been enabled on the Unified CM cluster. When the Jabber client connects to its home cluster it will make a request to Unified CM asking for the authentication and authorization configuration of the cluster. This Unified CM will reply with one of four outcomes:

1) Cluster is NOT enabled for OAuth nor SSO
2) Cluster is enabled for SSO and OAuth WITHOUT refresh token (Pre 12.0 model)
3) 12.0 Cluster is enabled for OAuth with refresh using Local or LDAP based authentication
4) 12.0 Cluster is enabled for SSO and OAuth WITH refresh tokens (12.0 model)

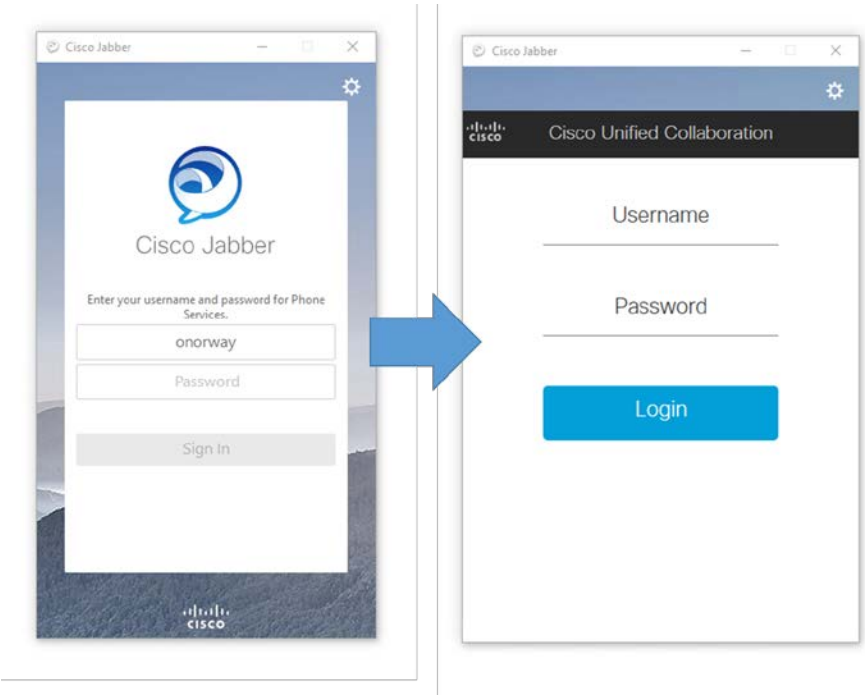
Once the administrator has enabled OAuth on the Unified CM cluster the Cisco Jabber clients must refresh their configuration. Cisco Jabber 11.9 with fast login uses a background update process to refresh its configuration after login is complete. After a new login Jabber will refresh its configuration within one to five minutes. Logged in Jabber clients also refresh their configuration every seven to nine hours using a random timer. If the Jabber client refreshes its configuration and finds OAuth has been enabled, it will ask the user to sign out and sign in again with the following prompt.



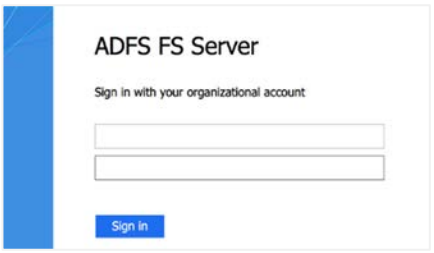
Jabber will then start the correct authorization flow enabled on the cluster.

A user/administrator can force the refresh early by selecting File>Help>Refresh configuration in the Jabber client.

Once Jabber has been successfully migrated to OAuth the login screen will change from the built in login screen to a web login screen provided by Unified CM.



If Unified CM has been configured to use SAML SSO with an external IdP the login screen of the IdP may be displayed rather than the Unified CM login screen.



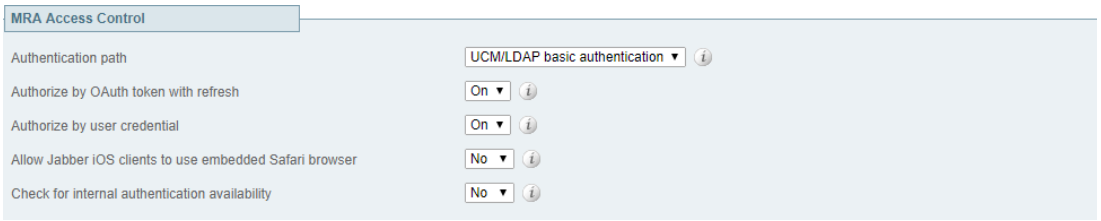
If using Mobile and Remote Access via Expressway, the request will be made to Expressway. Expressway will determine which authorization method to use and report that back to the Cisco Jabber client. It will also generate the web login screen if the new OAuth flow is selected.

Enabling Expressway

If Expressway is providing Mobile and Remote Access to Jabber clients, then the following procedure can be used to enable Expressway servers for OAuth with refresh.

1. Connect to the Expressway-C server and navigate to

   Configuration > Unified Communications > Configuration



2. Toggle "Authorize by OAuth token with refresh" to On to enable OAuth operation and click save at the bottom of the page.

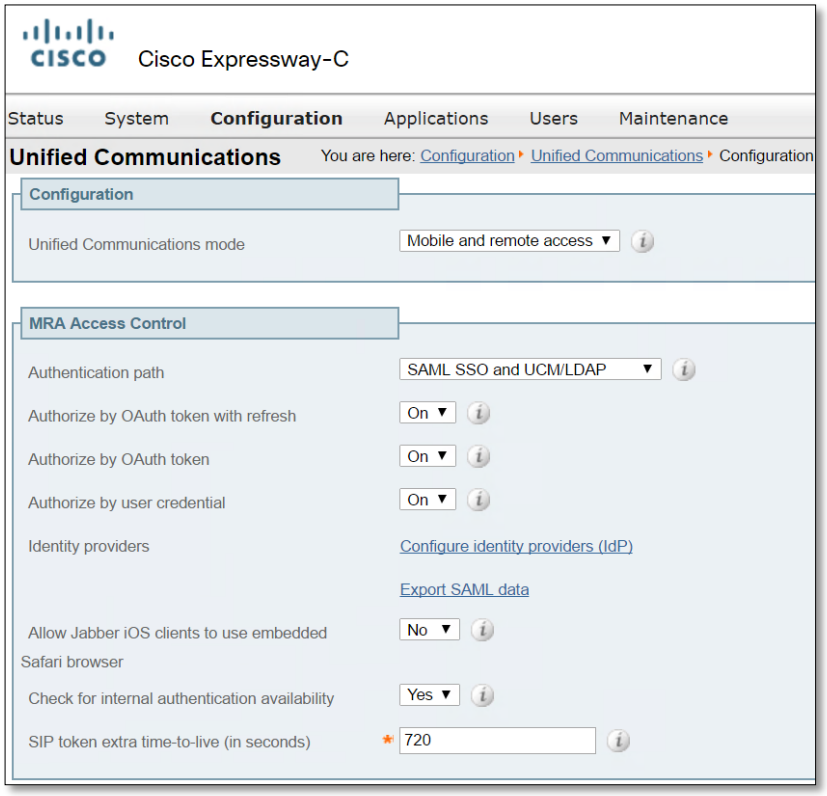
3. Navigate to the Configuration > Unified Communications > Unified CM servers and select each cluster by ticking the box next to the publisher address, and then click refresh servers. This allows the Expressway-C to fetch the signing and encryption keys from Unified CM that will be used by Expressway-C to verify each access and refresh token used by the Jabber clients connecting over MRA. This Unified CM refresh operation is a standard practice that needs to be performed after upgrading, or making significant configuration changes on Unified CM.

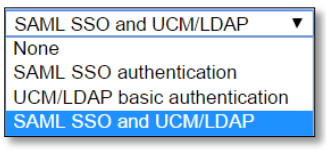
Additional MRA Access Control Details

The Expressway X8.10 release introduced a new MRA Access control menu that allows an administrator to enable the new OAuth with Refresh token login flow, and also featured an update to some of the existing authentication and authorization configuration options. The administrator needs to configure the Expressway access controls to be compatible with the

home Unified CM(s) SSO and OAuth configuration for all the MRA users that this Expressway will support. This includes both Jabber and non-Jabber MRA users.

The default values for these controls represent the most secure configuration option when paired with the Unified CM 12.0 architecture, but this is not typically the most practical configuration, depending on the MRA client and devices involved and the version(s) of Unified CM in the environment. For this reason, it is important to understand each of these controls and the MRA client/endpoint dependencies.





The “Authentication Path” dropdown selection dictates which settings will be available below.

If the authentication path is set to “SAML SSO authentication” only Jabber clients using an SSO enabled Unified CM cluster would be able to use MRA on this Expressway. This is an SSO only configuration.

Expressway MRA support for all IP phones, all TelePresence endpoints, and any Jabber clients homed to a Unified CM cluster not configured for SSO will require the

authentication path to include UCM/LDAP authentication. If there are no SSO enabled Unified CM clusters, select “UCM/LDAP authentication”. And if one or more of the Unified CM clusters supports Jabber SSO, select the “SAML SSO and UCM/LDAP” to allow for both SSO and basic authentication.

The “Authorize by OAuth token with refresh” setting only needs to be enabled when the Expressway is supporting Jabber 11.9 (or later) clients homed to a Unified CM cluster that is version 11.5(1)SU3 (or later) enabled for OAuth with refresh login flow.

The “Authorize by OAuth token” setting only needs to be enabled when the Expressway is supporting Jabber 10.6 (or later) clients homed to a Unified CM cluster that is version 10.5(2) (or later) enabled for SSO, and not yet upgraded or enabled for the OAuth with Refresh login flow.

The “Authorize by user credential” setting needs to be enabled to allow MRA functionality for all IP phones, TelePresence endpoints, or Jabber clients not using OAuth.

If this Expressway supports Jabber MRA users from multiple Unified CM clusters and not all the clusters are enabled for OAuth or all using a common authentication configuration, you should enable “Check for internal authentication availability”. This setting should be enabled only until all your Unified CM clusters are migrated to a release of Unified CM that supports OAuth with refresh and are all using a common authentication (either SSO or UCM/LDAP basic authentication). When enabled the Expressway-C will first determine a user’s home Unified CM cluster, and then use the same Unified CM API mentioned above to determine if the home cluster is one of these four states:

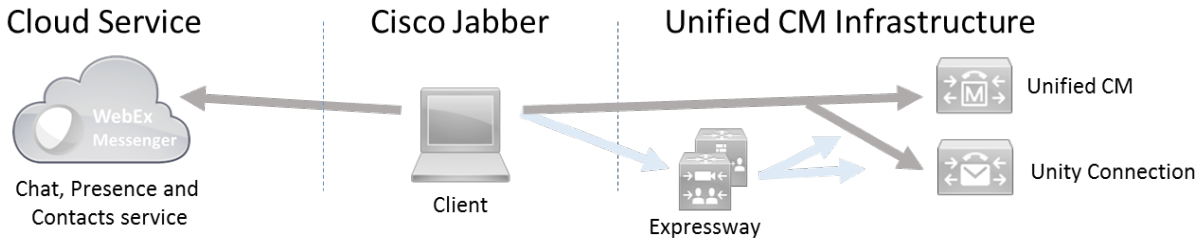
1) Cluster is NOT enabled for OAuth nor SSO
2) Cluster is enabled for SSO and OAuth WITHOUT refresh token (Pre 12.0 model)
3) 12.0 Cluster is enabled for OAuth with refresh using Local or LDAP based authentication
4) 12.0 Cluster is enabled for SSO and OAuth WITH refresh tokens (12.0 model)

This configuration allows for diverse Unified CM configurations and migrations in environments with several clusters, but also requires the Expressway to be configured with compatible authentication path and authorization settings mentioned previously in this section.

Hybrid deployment with WebEx Messenger

Cisco Jabber can also be run in a hybrid environment where instant message and presence services are provided by the Cisco WebEx Messenger cloud service. Telephony services in this model are provided by Unified CM. In this model OAuth operation is not supported.

Unified CM and all other UC services must have OAuth with refresh functionality disabled.



The exception to this is when both WebEx Messenger and Unified CM are configured to use SSO via an IdP. With a common IdP in place for both cloud and Unified CM services, OAuth can be enabled on Unified CM and other services. An IdP reverse proxy may also be required in this model for the cloud service to redirect the Jabber client to authenticate with the on-premises IdP.

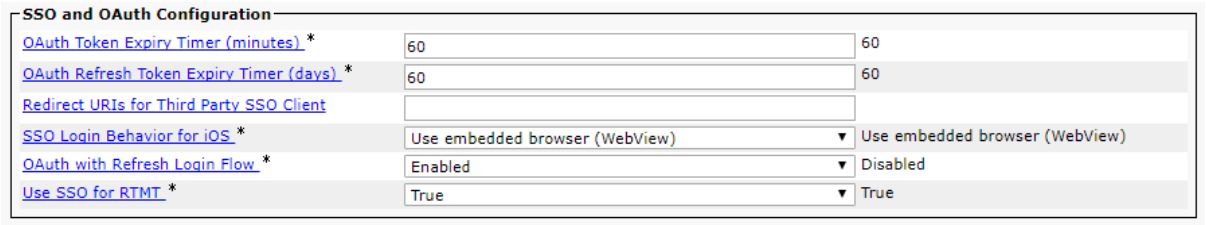
Key and Token management

This information is available in the Unified CM system configuration guide but included here for completeness.

Changing the duration of access and refresh tokens

If the administrator wants to change the duration of access and refresh tokens the following steps can be completed.

Go to System > Enterprise Parameters, SSO and OAuth Configuration.



“OAuth Token Expiry Timer” defines the validity duration in minutes of the OAuth access token. The default value is 60 minutes but this value can be configured from 1 to 1440 minutes.
“OAuth Refresh Token Expiry Timer” defines the validity duration in days of the OAuth refresh token. The default value is 60 days but this value can be configured from 1 to 365 days.
Note that changing the validity duration of the OAuth Refresh Token automatically revokes all previously-issued Refresh Tokens, thereby forcing all users to re-authenticate to obtain new tokens with the updated validity duration.

Regenerate the signing and encryption keys for OAuth Server
If an administrator believes the keys used for signing and encrypting OAuth tokens have been compromised then the following steps can be completed:

Connect to the Unified CM publisher using the command line interface using SSH.

To regenerate the encryption key use:
set key regen authz encryption
To regenerate the signing key use:
set key regen authz signing

Performing this action will make any existing tokens invalid and existing authorized users will be required to re-authenticate on their next authenticated connection to a UC service.

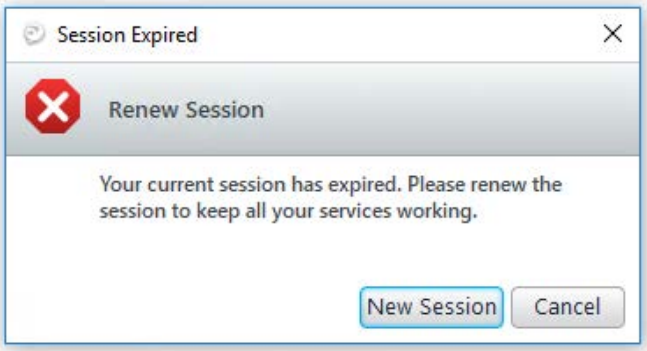
After regenerating these keys the Unified CM cluster details must be refreshed by all Expressway-C clusters providing MRA access to users from this Unified CM cluster, as well as by all Unity Connection servers.

Revoking OAuth Refresh Tokens
If the administrator needs to revoke a token a set of REST API are available on Unified CM for token management. An administrator can revoke a refresh token using a web browser or the curl utility. To revoke tokens for a user using the curl utility use the following command.

curl -k -u "admin:password"
https://<UCMaddress:8443/ssosp/token/revoke?user_id=<end_user>

Where:
admin:password is the login ID and password for the Cisco Unified Communications Manager administrator account.
UCMaddress is the FQDN or IP address of the Cisco Unified Communications Manager publisher node.
end_user is the user ID for the user for whom you want to revoke refresh tokens.

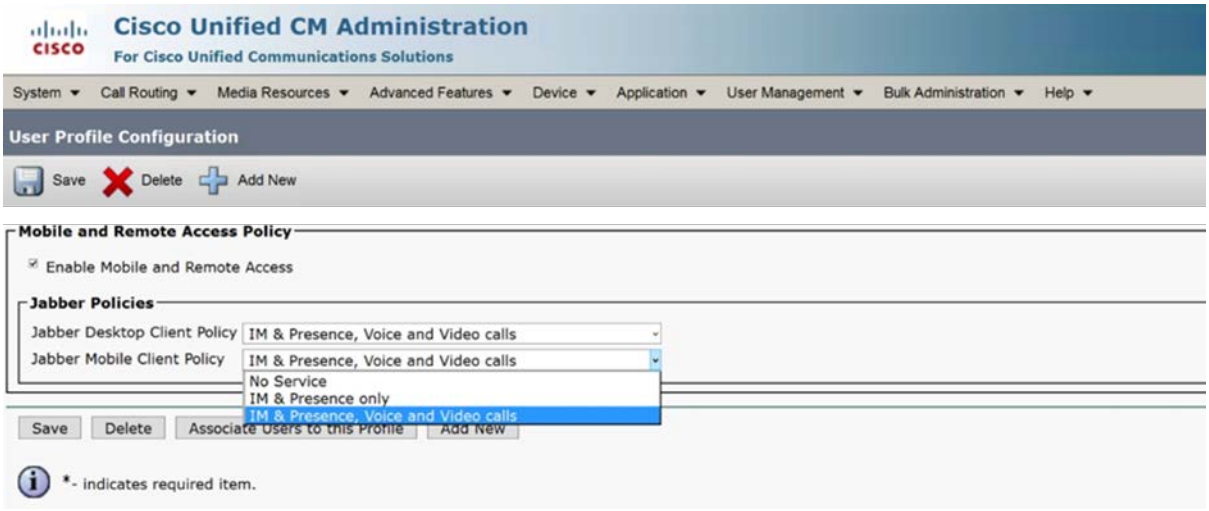
This example will revoke all refresh tokens associated with a user. The next time their Jabber client attempts to refresh their access token using the refresh token they will be required to re-authenticate. The Jabber client will show a message similar to the following:



Token Scopes (requires Cisco Jabber 12.0)

As previously highlighted, access tokens contain a scope element. The scope defines which UC services the token is valid for. This scope element is used to control access to the mobile remote access service provided by Cisco Expressway. The Unified CM administrator can create a user profile which defines which remote access services (chat/voice & video) by device type a user can access. A user can then be associated to the user profile.

The diagram below shows the User Profile Configuration in Unified CM 12.0.



The policy defined in the user profile is added to the access token by the OAuth server when the user authenticates and authorization request.

When a Cisco Jabber client tries to access the corporate network via Expressway the access token scope will be checked. The scope will define what services the Jabber user is authorized for and only connections to these servers will be established.

Further scope controls may be added to future releases.

Americas Headquarters
Cisco Systems, Inc.
San Jose, CA

Asia Pacific Headquarters
Cisco Systems (USA) Pte. Ltd.
Singapore

Europe Headquarters
Cisco Systems International BV Amsterdam,
The Netherlands

Cisco has more than 200 offices worldwide. Addresses, phone numbers, and fax numbers are listed on the Cisco Website at **www.cisco.com/go/offices.**