Approaches to Single Sign-On

Developer Technical Note

WebEx Communications Inc.
3979 Freedom Circle, Santa Clara, CA 95054, U.S.A.

Corp.: +1.408.435.7000 Sales: 1.877.509.3239

www.webex.com

Introduction

This developer technical note describes approaches and considerations for effecting single sign-on (SSO) behavior for WebEx services. Whether one or more WebEx services are integrated with a corporate portal or with an enterprise application (such as SFA, CRM, LMS, etc.), there are a number of approaches to creating secure, SSO behavior, and alleviating users from explicitly logging into WebEx.

WebEx does not currently support centralized directory services for dynamic authentication from a server such as LDAP, because WebEx is a hosted service, and integration would force organizations to open up their LDAP service to WebEx (outside their firewall). This is typically not possible for security and policy reasons. However, using MediaTone Integration APIs, it is easily possible to achieve the desired SSO behavior, as described in this tech note.

There are a number of obvious benefits for an organization supporting SSO integration with WebEx services, including:

- Leverage your existing security and authentication policies.
- Simplify account creation and maintenance.
- Accelerate your ROI on your WebEx investment

Additionally, the WebEx Professional Services Organization (PSO) has an SSO offering that is available for purchase and maintenance fees. The WebEx PSO team can work with your IT department to build the integration with your portal or application in a turn-key fashion. Contact your WebEx Client Service Manager for more information.

WebEx User Identities and Login

All WebEx services (Meeting Center, Sales Center, etc.) share a single database of user identities and passwords. Each user in WebEx has a unique WebEx ID and password within their customer site. The first step in creating SSO behavior is to either make sure that the user identities and passwords in WebEx match their identities in their corporate environment (and/or enterprise applications being integrated with WebEx), or maintain separate WebEx user credentials. This can be done manually by the WebEx site administrator (which is obviously laborious for large user communities), semi-automatically using import tools on the WebEx site administration pages, or programmatically using the APIs.

Batch Import of User Identities

The WebEx site admin tool supports importing user identities from a CSV (comma separated variable) template file in the Edit User List function. Typically, applications or directory servers can export a CSV, spreadsheet, or other file containing user information. To import using this mechanism, first create an export file, and copy/paste the data into the CSV template following the instructions on the WebEx Edit User List pages. Once the file is created, it can be imported.

The drawback to this approach is that future changes or additions to the user list must be processed manually.

Programmatic Approaches

Both the URL API and the XML API possess capabilities for registering new users, removing users, and editing the properties of existing users in WebEx. There are two approaches to using either of these APIs to register users: a batch-oriented approach, and a “lazy” registration approach. In both cases, the user identity creating the users must have WebEx administration privileges.

- ***Batch Registration***
  In this approach, a script or mini-application is constructed that cycles through all the users who will be given WebEx capabilities, and creates them using on of the APIs for that purpose. This is a good way to quickly grant access to a large number of users, but still requires additional mechanisms to maintain and update users as their properties change, or as users need to be added or removed.

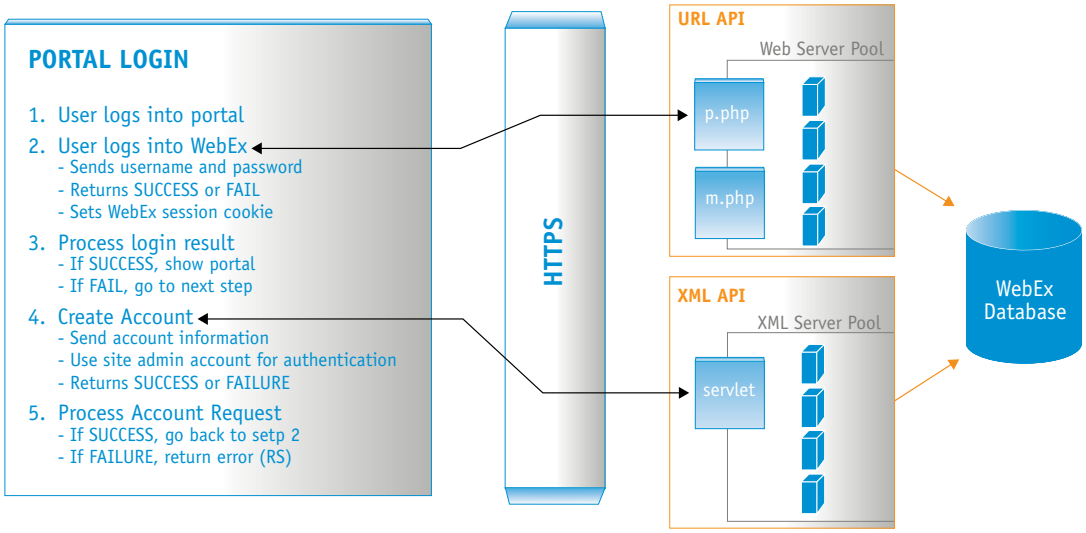
- ***On-Demand Registration***
  In this approach, the application or portal first attempts to log a user in when they log in to the application, and if they are not already a WebEx user, an error is generated. The application or portal can then trap the error, and dynamically create the user in WebEx on the spot, followed by a normal login. This approach has the advantage of being self-maintaining, and is most commonly implemented.

Security Considerations

Many companies will not want to populate the WebEx user database with their internal user's passwords, or even their internal user identities. Rather than simply passing along the authenticated user's names and passwords, you can take an approach that passes some arbitrary WebEx user names and WebEx passwords. There are a couple of ways to do this:

- Use a password algorithmically derived from the host login (but unknown to the user)
- Use a common password unknown to any users

The method you choose depends on whether you intend to have your employees access WebEx services only through your portal, or also directly through the Internet on your WebEx site. Using these approaches requires that the portal integration code maintain the alternate user identities and pass them to WebEx through API calls.



Typical Single Sign-on Integration

SSO with the URL API

The Partner Page service (p.php) of the URL API provides the needed commands to log users in and out, create and edit user accounts, and activate/deactivate users. Most other URL API commands (such as create meeting, add attendees, list meetings, etc.) require the user to be logged in. For example, the command to log a user in is "LI" with the following syntax:

p.php?AT=LI
&MU=*BackURL_or_goback*
&PW=*password*
&WID=*WebExID*
[&EM=*emailAdr*]
[&FN=*firstname*]
[&LN=*lastname*]
[&NPW=*newpassword*]
[&BU=*BackUrl*]

Refer to the URL API reference documentation for other commands' syntax. Use of the LI command should be over a secure HTTPS Post operation to prevent unencrypted passwords being passed to the WebEx servers openly.

On-Demand Registration with the URL API

To implement on-demand registration with the URL API, application or portal logic must do the following:

- Issue the Login (LI) command with the user's credentials. If this succeeds, the user is already a valid WebEx user.
- If LI fails, then create a new user account using the "Setup User" (SU) command. This command passes in a user identity and password as well as other information such as first and last name, e-mail address, and many other optional parameters. The SU command requires a confidential PartnerID parameter to validate the permission to create a new user. The PartnerID is supplied to developers by WebEx as part of their membership in the developer program.
- After successfully creating the user account, re-issue the Login (LI) command to log the user in.

Login Persistence

The login command (LI) logs the user in, and a cookie is set on the user's computer to maintain their logged-in status. However, the cookie has a 30 minute time-out expiration, and simply logging in once will typically be insufficient. Therefore, for uninterrupted SSO behavior, a modified approach is required.

To make sure general URL API commands succeed (scheduling a meeting, adding attendees, etc.), the Login (LI) command should be concatenated with each general command. The Login command's MU parameter specifies a destination URL when the login succeeds; this is where subsequent commands can be placed. Note that only the LI command supports the MU feature to concatenate an additional command.

For example, to add an attendee to a previously scheduled meeting, we concatenate the Meeting Page (m.php) Add Attendee (AA) command in the MU parameter of the Login (LI) command. When embedding a URL in the MU argument, it must be relative to your WebEx site, and certain characters in the URL need to be encoded according to the URL encoding rules specified in the URL API reference documentation. In the example below, "%26" is the "&" character, and "%3D" is the "=" character. The complete command would look like the following:

http://*YourWebExSite.webex.com/YourWebExSite/p.php?AT=LI*
&WID=UserWebExID&PW=UserPasswd&MU=
YourWebExSite/m.php?AT%3DAA%26MK%3DMeetingKey%26
FN%3DAttendeeFullName%26EM%3DAttendeeEmailAddr

SSO with the XML API

The behavior of the XML API is distinctly different from the URL API from a login standpoint. Each XML Request Document is a stateless transaction, and there is no concept of being “logged in” to the XML API service. Rather, each and every XML Request Document begins with a Security Context that provides authentication information for the request. Below is the standard Security Context segment that validates each request:

```
<header>
   <securityContext>
      <webExID>userid</webExID>
      <password>userpassword</password>
      <siteID>0000</siteID>
      <partnerID>9999</partnerID>
   </securityContext>
</header>
```

Note also that the Security Context also requires a SiteID and PartnerID for complete validation.

On-Demand Registration with the XML API

To implement on-demand registration with the XML API, the application or portal can use the XML Meeting Service to interrogate WebEx to determine if the user exists in WebEx. If not, the integration logic can then create the user. The approach is as follows:

- After the user logs into the application or portal, use the getUser Request (in the XML User Service) to see if the user already exists in WebEx. The Security Context of this command must specify a user with WebEx site administration privileges. If getUser returns SUCCESS, no further action is required.
- If getUser fails, then the user can be created using the createUser Request of the XML User Service. The Security Context of this command must specify a user with WebEx site administration privileges. Once the user is created, their WebExID and Password can be used in the Security Context of subsequent Request documents.

SSO with the XML API

To implement SSO behavior in an integration using the XML API, the integration code (in an application or corporate portal) must cache the user identity, password, SiteID, and PartnerID, and place this information in the Security Context of each XML Request being made. Of course, the user identity must have been previously established in WebEx.

If you use the XML API to log users in, but also want to provide access to the WebEx user interface for general activities like managing meetings, joining meetings, etc., then you will also have to use the URL API which can return web pages for these activities. The XML API only provides data exchange facilities, and does not return web pages to the caller.

Worldwide Sales Offices:

Americas & Canada
Tel: +1.877.509.3239
AmericasInfo@webex.com

Europe, Middle East & Africa
Tel: + 31 (0)20.4108.700
europe@webex.com

United Kingdom
Tel: 0800.389.9772
europe@webex.com

Australia & New Zealand
Tel: + 61 (0)3.9653.9581
AsiaPacInfo@webex.com

China (HK)
Tel: + 852.8201.0228
AsiaPacInfo@webex.com

India
Tel: 080.2228.6377/17030 9330
sales@cyberbazaarindia.com

Japan
Tel: + 81 3 5501 3272
JapanInfo@webex.com

TN-SSO-060905