LeadingRE – Confidential: for Limited Release

Member User Management & Authentication Service

Web Service Documentation

access | LEADINGRE

INSTITUTE
Education by LeadingRE

Version: ***2.1***

Prepared by: ***Neil Elver, LeadingRE***

Leading REAL ESTATE COMPANIES OF THE WORLD™

Published: 1/20/2017

TABLE OF CONTENTS

Introduction

This guide encompasses all technical details required to develop, test, and deploy user account synchronization between a member firm's Intranet user management system and LeadingRE's access and Institute 2.0 suite of networks and sites.

The LeadingRE 'access / Institute' Member User Management Web Service has been developed as a next-generation single-sign-on (SSO) mechanism for LeadingRE member real estate firms, as well to keep staff rosters up-to-date with the LeadingRE Extranets. The User Management Service allows for Security Token Authentication to the access / Institute suite of sites, documents, and applications including Institute. Additionally, user accounts can be created, modified, and disabled remotely through the web service implementation. A standard SOAP protocol web service securely exposes several methods for user management for use by member firms.

This guide also describes the mirrored set of authentication and user management web services provided for debug and testing environments.

PLEASE NOTE THE FOLLOWING REQUIREMENTS:

- This implementation must be done by an experienced programmer who has knowledge of web services.
- Implementation must follow the specifications and instructions outlined in this document.
- Testing of at least two user account creations and a completed agreement form is required prior to live site access.
- Testing and debugging shall be performed on the test system only.

High Level Details

- In order to utilize any of these services, the member firm will be issued a Security ID code to fulfill a required parameter for every call made to the production web service.

Your company Security ID Code is: _________provided via email__________

- All method parameters are required for all web service method calls. Failure to supply a parameter will result in a general web service error response.
- All parameters and return values are simple data types: numeric, string
- Per web service standards, each request will be followed by a response containing the return response value(s).
- Authentication tokens will be valid forever in testing scenarios, five minutes in production (to allow for success during rare production debugging scenarios)

Revision History

Date	Version	Person Responsible	Description
1/18/2011	0.5	Neil Elver	Initial document outline for LeadingRE SSO 2.0
1/21/2011	0.9	Neil Elver	Most details covered, sent to beta testing companies
2/9/2011	1.0	Neil Elver	Testing utility screenshots and code samples added
3/18/2011	1.1	Neil Elver	Minor additions to page 3 on activation requirements for additional beta companies
3/23/2011	1.2	Neil Elver	Replaced Appendix Screenshots with Actual Text
5/9/2011	1.3	Neil Elver	Added additional Role options to user create
8/10/2011	1.4	Neil Elver	Added three new fields for user creation – Bio, OfficeName, PhotoURL
8/19/2011	1.5	Neil Elver	Removed references related to Inside.LeadingRE conversion to avoid confusion.
11/1/2012	1.6	Neil Elver	Added ability to send a 0 (zero) as the RoleID for UserUpdate function.
7/15/2013	1.7	Neil Elver	Minor text and graphic updates.
2/24/2015	1.8	Neil Elver	Changed redirect URLs to reflect new secured server.
10/19/2016	1.9	Shriyal Padte	Added new Create/Update methods for Agent License Number
12/28/2016	2.0	Kelsey Winzeler	Updated logos & replaced Our World references with access.
1/20/2017	2.1	Kelsey Winzeler	Replaced all Create/Update with new KeyValCSV parameters.

Production Web Service Methods

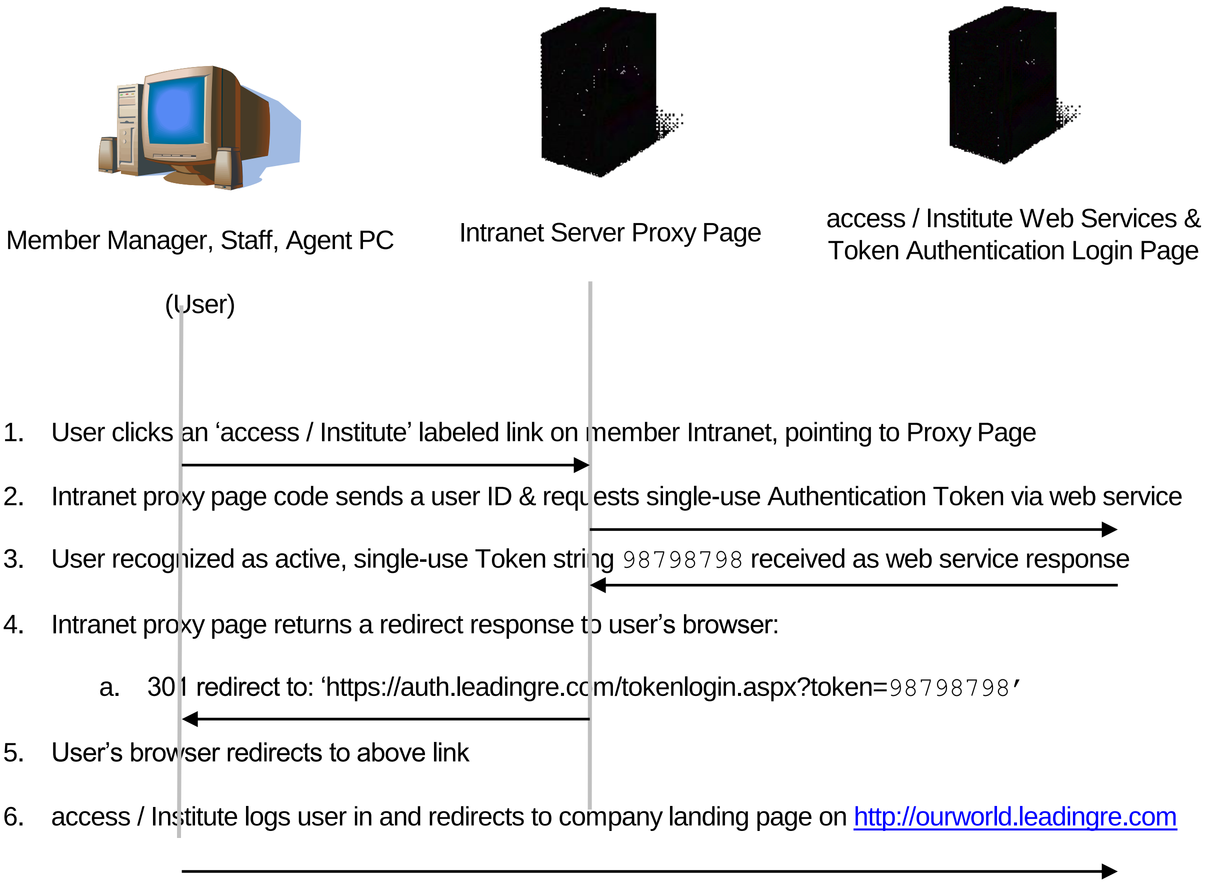
1.0 Introduction

The LeadingRE Member User Management Service consists of two web services – one testing and one production service – which mirror each other in methods but not functionality. The production web service will make updates to live users and allow real user logins. The test service, on the other hand will allow for practice users and data to be manipulated without affecting real users. The test Token Authentication steps will show a basic 'Login Success' page upon successful login. The production Token Authentication will prompt new users to accept a Terms and Conditions agreement and record the login history, which we need to prevent during debugging and testing.

2.0 Web Service Token Authentication - Practical Usage

The single-sign-on implementation details are entirely up to the member's technical staff or contractors, but will typically operate using a dynamic web proxy page which invokes a series of web service calls to authenticate and possibly create a user. The proxy page will respond with a redirect response pushing the user to the authentication token page, fully logging them in to access / Institute.

Example of a Recognized User Using Token Authentication for SSO:


Member Manager, Staff, Agent PC
(User)
Intranet Server Proxy Page
access / Institute Web Services &
Token Authentication Login Page

1. User clicks an 'access / Institute' labeled link on member Intranet, pointing to Proxy Page
2. Intranet proxy page code sends a user ID & requests single-use Authentication Token via web service
3. User recognized as active, single-use Token string 98798798 received as web service response
4. Intranet proxy page returns a redirect response to user's browser:
    a. 301 redirect to: 'https://auth.leadingre.com/tokenlogin.aspx?token=98798798'
5. User's browser redirects to above link
6. access / Institute logs user in and redirects to company landing page on http://ourworld.leadingre.com

3.0 Use Case Process Flows

Token Authentication for 'new' users

1. Attempt to request token from Authentication Token request service by hitting web service
    a. Security ID and User's unique User ID passed in
2. 'Error: UNKNOWNUSER' response is received because User ID not recognized by LeadingRE
3. Create the user with **CreateNewUserKeyValCSV** web service method, passing in data from user setup data in Intranet
4. Retry request of login token string (step 1)
5. Redirect user to token login URL with token received above as parameter:

    https://auth.leadingre.com/tokenlogin.aspx?token=< insert provided one-time token here >

Token Authentication for existing users

1. Attempt to request token from token request service by hitting web service
2. Redirect user to token login URL with token received above as parameter:

    https://auth.leadingre.com/tokenlogin.aspx?token=< insert provided one-time token here >

4.0 Member Intranet Proxy Page Pseudocode Example

```
If UserCookie is Not Empty then
    tokenstring = GetTokenFromLREService('mysecurityidstring', UserCookieUserID)
Else Throw Error – 'User Not Logged In to Intranet'
End If
If Not tokenstring contains 'Error' then
    302 Redirect to 'https://auth.leadingre.com/tokenlogin.aspx?token=' + tokenstring
Else If tokenstring contains 'UNKNOWNUSER' then
    CreateUserFromLREService(UserCookieUserName, UserCookeLastName, UserCookieEmail, …
    tokenstring = GetTokenFromLREService('mysecurityid', UserCookieUserID)
    302 Redirect to 'https://auth.leadingre.com/tokenlogin.aspx?token=' + tokenstring
Else Throw Error – 'Authentication Error'
```

5.0 Web Service Methods

These web services expose the following methods over an industry standard XML SOAP web service protocol. See appendix for examples.

GetToken

- SID – Security ID
  - Provided once during setup by LeadingRE with this documentation
  - Always unique per member company
  - Two steps to initial company setup – security ID creation for testing, later an activation of the same ID for production
- UniqueID – Intranet login name or other ID unique to users in the local (Intranet) system.
  - This is any behind-the-scenes Intranet user identifier string which cannot be changed by the user. In many cases an email address can be used, if it is a permanent setting for the user account. If your users can edit their email addresses a user database ID number or GUID would be a better choice. This value is created by the member company Intranet as users are created and must be unique per user, unchanging, and never null.
  - **If Inside.LeadingRE SSO was implemented previously, please utilize the same UniqueID parameter to ensure user history and records are seamlessly transitioned.**
- **Response**: String representing a valid security token, or error messaging
- **Example**: GetToken('7862384762828', 'jsmith') returns 'sdif9s8df98sdfh9s8dfh' as single-use login token parameter allowing a single login for user jsmith

Web Service Methods (cont.)

CreateNewUserKeyValCSV

(note – old CreateNewUser is being deprecated)

- SID – Security ID (Provided)
- UniqueID – String, 100 Chars
- FirstName – String, 50 Chars
- LastName – String, 50 Chars
- Email – String, 100 Chars
- Title – String, 50 Chars
- Accreditations – String, 15 Chars
- RoleID (see table on next page) – Integer
- Biography – String, unlimited (may be truncated during display)
- Office Name – String, 50 Chars
- PhotoURL – String, 100 Chars
- keyValCSV – to send Agent License number, pass

  `license=99999`

  where `license` is literal passed as is, `99999` is the license number up to 25 characters. If there is no license number to be assigned, leave whole parameter blank.

  This parameter value is ignored if the literal `license=` is not found.

- **Response**: String, 'True' or 'False' for success or failure

Example: CreateNewUserKeyValCSV ('7862384762828', 'jsmith, 'John', 'Smith', 'jsmith@abc.com', 'Training Manager', 'CRB, CRS, RCC', 2, 'Best Salesperson Ever', 'Jonestown Office', 'http://mybroker.com/profilephoto.aspx?id=23423423', 'license=99999') returns 'True'

UpdateUserKeyValCSV

(note – email address is a key to other systems and is not currently editable)

- SID – Security ID (Provided)
- UniqueID – **Used to match to existing user, not updateable**
- FirstName – String, 50 Chars
- LastName – String, 50 Chars
- Title – String, 50 Chars
- Accreditations – String, 15 Chars
- RoleID (see table on next page) – Integer
  - You may send a 0 (zero) as the RoleID to allow the LeadingRE set roles to remain intact, which may be more granular than those set up in your Intranet.
- Bio (Profile Biography) – String, unlimited (may be truncated during display)
- Office Name – String, 50 Chars
- PhotoURL – String, 100 Chars – will be fetched once and saved to access / Institute profile
- keyValCSV – to send Agent License number, pass

  `license=99999`

  where `license` is literal passed as is, `99999` is the license number up to 25 characters. If there is no license number to be changed, leave whole parameter blank.

  This parameter value is ignored if the literal `license=` is not found.

- **Response**: String, 'True' or 'False' for success or failure

Example: UpdateUserKeyValCSV ('7862384762828', 'jsmith, 'John', 'Smith', 'Training Manager', 'CRB, CRS, RCC', 2, 'Best Salesperson Ever', 'Jonestown Office', 'http://mybroker.com/profilephoto.aspx?id=23423423', 'license=99999') returns 'True'

DisableUser

- SID – Security ID (Provided)
- UniqueID – String, 100 Chars

Production Web Service Connection Address

The production web service connection will be accessed through the following address. Please do all testing using the test service address, detailed later in this document.

https://authservice.ourworld.leadingre.com/AuthService.asmx

Note: web service will require port 443 open outbound on firewalls blocking outbound ports

Error Conditions

Invalid Security ID - Example Error Response String: ***Error: BADSECURITYID***

NOTE: if you get this error on production but not test, you likely have not had your ID activated on the production server. Please contact LeadingRE to check your rollout activation status.

This error will be returned if the SecurityID string passed into the web service is not recognized as matching a known LeadingRE Security ID. Only companies participating in the token authentication have security IDs issued. Tokens are generated randomly or hand-created to be impossible to guess, lookup, or iterate. If this error is experienced during testing, it's likely that your company has not been fully 'activated' in access / Institute for launch. Please coordinate with LeadingRE to use the production (AuthService.asmx) web service.

Unknown User - Example Error Response String: ***Error: UNKNOWNUSER***

This error will be encountered if the Unique User ID passed into the web service does not match a 'known foreign user' in the LeadingRE SSO system. **This error will be frequently encountered when new users attempt to log in via token authentication, since we don't yet have a user record to match them to. See Page 6 for use case example.**

Technical Notes

- Security IDs will be temporarily disabled and technical engineers contacted if test accounts are created on the production system. Please utilize the test web service for all testing.

- Token Login URL: https://auth.leadingre.com/tokenlogin.aspx?token=______________

Role Lookup for Creating User

Role ID	Role
1	Agent
2	Corporate Staff / Leadership
3	Branch Manager
4	Relocation Staff
0	**Special case for UpdateUser only**: if you do not wish to update roles upon every UpdateUser web service method call, send a 0 as the RoleID value. This is necessary, for instance, if your Intranet only tracks a singular role and then your access / Institute administrator or LeadingRE staff marks some of your users as Broker, Relocation Staff, or other more granular roles and you want to run an UpdateUser() upon every login without

	clearing the more granular role details.

Note: if detailed roles are not available in the local system, please use option 1 as default – accounts can be given additional permissions within access / Institute once created. An intranet with only staff and agent differentiation should use options 1 and 2.

Member Intranet URL

LeadingRE will need the URL of the login page for each member intranet. This will be utilized as the redirect action following a Logout link click from within access / Institute. Additionally, access / Institute email messages containing a link to 'View Discussion' will prompt for user login if the user does not have an active session. In the case of SSO member companies, we will direct the user to their own Intranet login page rather than the stock access / Institute login page, which the user will not have credentials for. *The users will need to be trained that they will have to 're-click' the link within their email after authenticating with their Intranet.* This is a workaround that may be resolved through some more advanced SSO functionality to be determined.

Testing Web Service Methods

6.0 Introduction to Testing

This Member User Management system is a conduit to not one but many integrated LeadingRE membership and agent software systems, which play many roles in our online referral and communication sites. For this reason, **we will not permit 'test' users, 'test' token login practice, or other non-production user creation or management on our live systems.** We have therefore created a test 'playground' for development and debug use. This will allow for system integration testing without fear of disabling, creating, or changing usage history records of live users.

All code designed for production use will be developed against the test service, and all method names, parameters, data types, and variables are identical between test and live. This will enable the transition from 'test' to 'live' to be made by simply changing the web service connection URL:

https://authservice.ourworld.leadingre.com/Auth**Test**Service.asmx

becomes

https://authservice.ourworld.leadingre.com/AuthService.asmx

for production use

Web Service Connection Address

Please do all testing using the test service address:

https://authservice.ourworld.leadingre.com/AuthTestService.asmx

Note: this must be requested over SSL, which will require port 443 open outbound on firewalls blocking outbound ports

Deviations From Production WS

The testing Web Service works identically to the production version, with the following exceptions:

- Returned security tokens are valid forever to allow day-to-day testing and debugging, and temporarily hard-coding URLs for test cases. In production functionality they would be single use, and expire nearly immediately if unused.

- Token authentication will not result in any security access, but will simply display a 'login success' page indicating basic known information for that user. This will be sufficient for testing and will aid in troubleshooting by displaying diagnostic information.

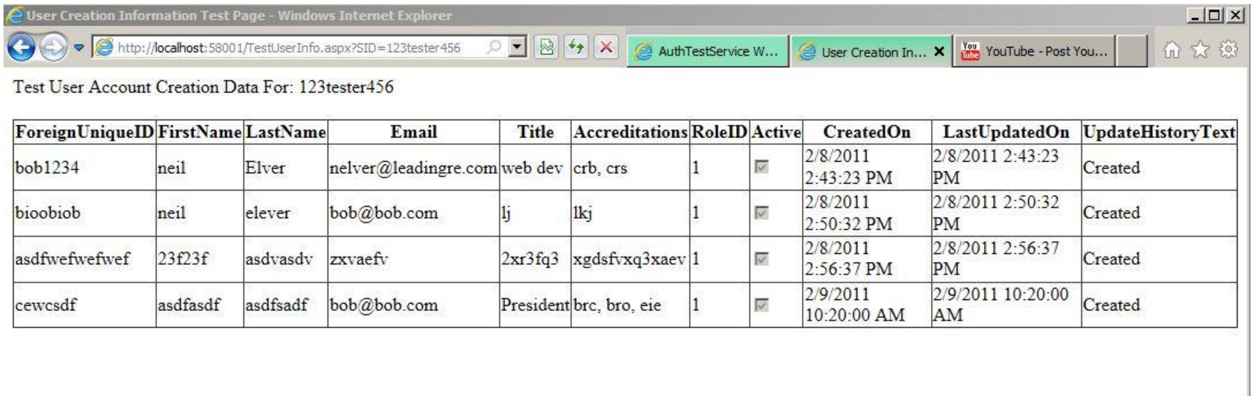
- 'Create User' method will create temporary, non-production users which will be cleared every few months, and 'Disable User' method will only apply to these temporary users. These test accounts can be verified using the User Creation Information Testing Page described next.

User Creation Information Testing Page

A special page has been created specifically for viewing the result of data transfers taking place while testing the User Create process. The testing URL is shown below. Replace YOURSECURITYID with the provided security ID code to see your companies test user records.

https://authservice.ourworld.leadingre.com/TestUserInfo.aspx?SID=YOURSECURITYID

Example:


User Creation Information Test Page - Windows Internet Explorer
http://localhost:58001/TestUserInfo.aspx?SID=123tester456
AuthTestService W...
User Creation In...
YouTube - Post You...

Test User Account Creation Data For: 123tester456

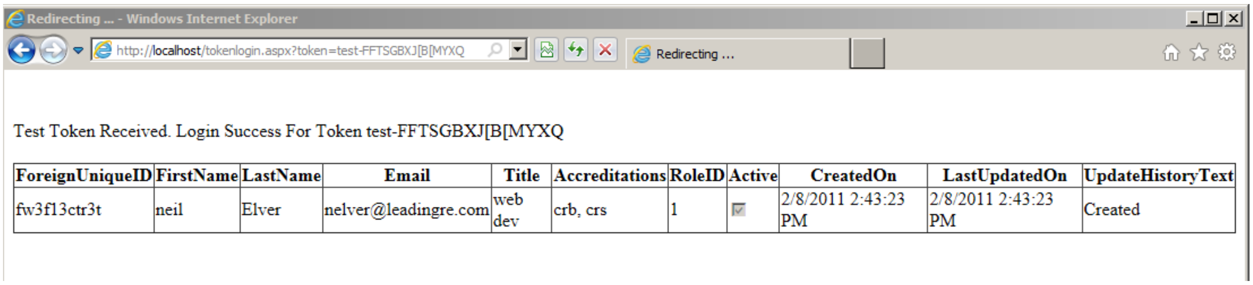
ForeignUniqueID	FirstName	LastName	Email	Title	Accreditations	RoleID	Active	CreatedOn	LastUpdatedOn	UpdateHistoryText
bob1234	neil	Elver	nelver@leadingre.com	web dev	crb, crs	1	☑	2/8/2011 2:43:23 PM	2/8/2011 2:43:23 PM	Created
bioobiob	neil	elever	bob@bob.com	lj	lkj	1	☑	2/8/2011 2:50:32 PM	2/8/2011 2:50:32 PM	Created
asdfwefwefwef	23f23f	asdvasdv	zxvaefv	2xr3fq3	xgdsfvxq3xaev	1	☑	2/8/2011 2:56:37 PM	2/8/2011 2:56:37 PM	Created
cewcsdf	asdfasdf	asdfsadf	bob@bob.com	President	brc, bro, eie	1	☑	2/9/2011 10:20:00 AM	2/9/2011 10:20:00 AM	Created

Token Login Testing

The token authentication address is the same regardless of Testing or Production. Test tokens will begin with the word ‘test’ for clarity and result in the display of test user data on the Token Login web page itself. Use of the token URL with production tokens will result in a redirect into access / Institute.

https://auth.leadingre.com/tokenlogin.aspx?token=______________

Test Token Login Result


Redirecting ... - Windows Internet Explorer
http://localhost/tokenlogin.aspx?token=test-FFTSGBXJ[B[MYXQ
Redirecting ...

Test Token Received. Login Success For Token test-FFTSGBXJ[B[MYXQ

ForeignUniqueID	FirstName	LastName	Email	Title	Accreditations	RoleID	Active	CreatedOn	LastUpdatedOn	UpdateHistoryText
fw3f13ctr3t	neil	Elver	nelver@leadingre.com	web dev	crb, crs	1	☑	2/8/2011 2:43:23 PM	2/8/2011 2:43:23 PM	Created

Post-Beta Phase Future Plans

Additional Security:

- Allow only known IP addresses to access the LeadingRE AuthService
    - To be implemented after beta phase
- Allow login access only with known referring URL

Appendix

Web Service Call Examples

Visit **https://authservice.ourworld.leadingre.com/AuthService.asmx**

for a complete list of web service operations.

GetToken

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /AuthService.asmx HTTP/1.1
Host: authservice.ourworld.leadingre.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction:
"https://authservice.ourworld.leadingre.com/AuthService.asmx/GetToken"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetToken
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <_securityID>string</_securityID>
      <_uniqueUserID>string</_uniqueUserID>
    </GetToken>
  </soap:Body>
</soap:Envelope>
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    <GetTokenResponse
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <GetTokenResult>string</GetTokenResult>
    </GetTokenResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /AuthService.asmx HTTP/1.1
Host: authservice.ourworld.leadingre.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetToken
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <_securityID>string</_securityID>
      <_uniqueUserID>string</_uniqueUserID>
    </GetToken>
  </soap12:Body>
</soap12:Envelope>
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <GetTokenResponse
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <GetTokenResult>string</GetTokenResult>
    </GetTokenResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /AuthService.asmx/GetToken HTTP/1.1
Host: authservice.ourworld.leadingre.com
Content-Type: application/x-www-form-urlencoded
Content-Length: length

_securityID=string&_uniqueUserID=string
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<string
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">string</st
ring>
```

SOAP 1.1

The following is a sample SOAP 1.1 request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /AuthService.asmx HTTP/1.1
Host: authservice.ourworld.leadingre.com
Content-Type: text/xml; charset=utf-8
Content-Length: length
SOAPAction: "https://authservice.ourworld.leadingre.com/AuthService.asmx/
CreateNewUserKeyValCSV"

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    < CreateNewUserKeyValCSV
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <_securityID>string</_securityID>
      <_uniqueuserID>string</_uniqueuserID>
      <_firstname>string</_firstname>
      <_lastname>string</_lastname>
      <_email>string</_email>
      <_title>string</_title>
      <_accreditations>string</_accreditations>
      <_roleID>int</_roleID>
      <_bio>string</_bio>
      <_officeName>string</_officeName>
      <_photoURL>string</_photoURL>
      <_keyValCSV>string</_keyValCSV>
    </ CreateNewUserKeyValCSV>

  </soap:Body>
</soap:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
  <soap:Body>
    < CreateNewUserKeyValCSVResponse
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <CreateNewUserKeyValCSVResult>string</ CreateNewUserKeyValCSVResult>
    </CreateNewUserKeyValCSVResponse>
  </soap:Body>
</soap:Envelope>
```

SOAP 1.2

The following is a sample SOAP 1.2 request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /AuthService.asmx HTTP/1.1
Host: authservice.ourworld.leadingre.com
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <CreateNewUserKeyValCSV
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <_securityID>string</_securityID>
      <_uniqueuserID>string</_uniqueuserID>
      <_firstname>string</_firstname>
      <_lastname>string</_lastname>
      <_email>string</_email>
      <_title>string</_title>
      <_accreditations>string</_accreditations>
      <_roleID>int</_roleID>
      <_bio>string</_bio>
      <_officeName>string</_officeName>
      <_photoURL>string</_photoURL>
      <_keyValCSV>string</_keyValCSV>
    </CreateNewUserKeyValCSV>
  </soap12:Body>
</soap12:Envelope>
HTTP/1.1 200 OK
Content-Type: application/soap+xml; charset=utf-8
Content-Length: length
```

```
<?xml version="1.0" encoding="utf-8"?>
<soap12:Envelope xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://www.w3.org/2003/05/soap-envelope">
  <soap12:Body>
    <CreateNewUserKeyValCSVResponse
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">
      <CreateNewUserKeyValCSVResult>string</CreateNewUserKeyValCSVResult>
    </CreateNewUserKeyValCSVResponse>
  </soap12:Body>
</soap12:Envelope>
```

HTTP POST

The following is a sample HTTP POST request and response. The **placeholders** shown need to be replaced with actual values.

```
POST /AuthService.asmx/ CreateNewUserKeyValCSV HTTP/1.1
Host: authservice.ourworld.leadingre.com
Content-Type: application/x-www-form-urlencoded
```

```
Content-Length: length

_securityID=string&_uniqueuserID=string&_firstname=string&_lastname=string&_em
ail=string&_title=string&_accreditations=string&_roleID=string&_bio=string&_of
ficeName=string&_photoURL=string
HTTP/1.1 200 OK
Content-Type: text/xml; charset=utf-8
Content-Length: length

<?xml version="1.0" encoding="utf-8"?>
<string
xmlns="https://authservice.ourworld.leadingre.com/AuthService.asmx">string</st
ring>
```

Web Services Resources

- http://en.wikipedia.org/wiki/Web_service

by Language

Coldfusion

- http://www.techrepublic.com/article/invoke-soap-web-services-the-easy-way-using-coldfusion-mx/5034367
- http://www.bennadel.com/blog/1809-Making-SOAP-Web-Service-Requests-With-ColdFusion-And-CFHTTP.htm

PHP

- http://php.net/manual/en/refs.webservice.php
- http://developer.apple.com/internet/webservices/soapphp.html

Microsoft .NET

- http://msdn.microsoft.com/en-us/library/t745kdsh.aspx

Java

- http://java.sun.com/developer/technicalArticles/WebServices/WSPack/