OPENSSO API WSDL

The specifications and information in this document are subject to change without notice. Companies, names, and data used in examples herein are fictitious unless otherwise noted. This document may not be copied or distributed by any means, in whole or in part, for any reason, without the express written permission of RCDevs Security.

WebADM and OpenOTP are trademarks of RCDevs. All further trademarks are the property of their respective owners.

No guarantee is given for the correctness of the information contained in this document. Please send any comments or corrections to info@rcdevs.com.

OpenSSO API WSDL

SOAP API SOAP

OpenSSO SOAP API Description

Usage

OpenSSO provides a very simple API to enable OpenOTP single sign-on across several web application. The API allows setting, removing and checking SSO sessions. The SSO session IDs should be given to the end users in Web browsers' cookies.

A typical usage of OpenSSO is:

User authenticates on Server1:
The web application on Server1 performs the following SOAP calls to the WebADM server.

	SOAP Calls	Response
1.	**openssoCheck**	
	failure	user not authenticated
2.	**openotpLogin**	
	failure	Do not start a SSO session
	success	Start a SSO session
3.	**openssoStart**	

User goes to Server2:
The web application on Server2 performs the following SOAP calls to the WebADM server.

	SOAP Calls	Response
1.	**openssoCheck**	
	success	Session valid - No need to re-authenticate user
	failure	Authenticate user again with OpenOTP
2.	**openotpLogin**	
3.	**openssoStart**	

The openssoStart returns a session ID. This session ID should be provided to the user browser in a cookie. This way the user will present his session ID to all the servers in your SSO system.

OpenSSO allows transporting and updating user data in the SSO sessions. This is a convenient way to pass work data from Server1 to Server2 in the context of an established SSO session.

OpenSSO provides 4 methods:

1. openssoStart

Used to start an SSO session.

The request contains the following attributes:

- username: User login name (mandatory).
- domain: User login domain (optional if OpenSSO has a default domain setting).
- data: Any serialized data to be stored in the SSO session.
- client: Client identifier (NAS) to be used in service logs (defaults to the client IP address).
- source: IP address of the end user system (optional).
- settings: List of OpenSSO settings which will override the user/group/application server-side settings (ex. *SessionTimeout=600,SessionRenew=Yes*).

The response contains the following attributes:

- code:
- 1 means session start success.
- 0 means session start failure.
- error: The error ID if code 0 was returned. The ID corresponds to the error message template names in `opensso.xml` (ex.

BadUser).

- message: The server reply message to be displayed to the user.
- session: OpenSSO session ID on success.
- timeout: SSO session time.

2. openssoStop

Used to stop an SSO session.

The request contains the following attributes:

- session: OpenSSO session ID.

The response contains the following attributes:

- code:
- 1 means session stop success.
- 0 means session stop failure.
- error: The error ID if code 0 was returned.
- message: The server reply message to be displayed to the user.

3. openssoCheck

Used to check an SSO session.

The request contains the following attributes:

- session: OpenSSO session ID.
- data: If non-empty, updated data to be stored in the SSO session.

The response contains the following attributes:

- code:
- 1 means session still valid.
- 0 means session not existing or expired.
- error: The error ID if code 0 was returned.
- message: The server reply message to be displayed to the user.
- data: The SSO session data if any.

4. openssoStatus

Used to query a server status.

The request does not contain any attribute.

The response contains the following attributes:

- status:
- 1 if the server is willing to accept requests.
- 0 if the server cannot accept new requests.
- message: The server status details.