Date: **2/1/2017**

JAR Version: **1.0.4**

A Generic SSO Implementation for ADI

# Introduction

A generic servlet-based filter has been created by Axway Professional Services for use in some common ADI SSO implementations for customers where authentication is delegated. It has been targeted for typical use-cases and filter-based technologies (including Java Security Framework, SiteMinder, ClearTrust), but it may not work for all implementations (i.e. it does not support SAML). It requires some user data (the authenticated User ID and optionally the Roles for that user) to be passed on the HTTP header from a webserver to ADI over a trusted connection. The webserver does the authentication, adds the user data to the header, and then forwards the request to ADI. This SSO component seamlessly retrieves that user data from the header and interfaces with the ADI platform.

# Steps to enable SSO

1. Modify the file conf/platform.properties to accept SSO authentication. Example:

com.systar.photon.application.auth.ssoMode=strict

|  |  |
| --- | --- |
| **Parameter value** | **Description** |
| strict | SSO is activated, but if no SSO information is present in the request, an error page is displayed. |
| permissive | SSO is activated, but if no SSO information is present in the request, the login screen is displayed. This is the preferred setting for all environments, excluding production. |
| Off | SSO is disabled, the login screen is always displayed. |

1. Modify the file conf/platform.properties to enable SSO role provisioning if needed. Example:

com.systar.photon.application.auth.ssoRoleProvisioning=true

|  |  |
| --- | --- |
| **Parameter value** | **Description** |
| true | Default value. SSO provides the user roles via a HTTP header parameter. SSO user's roles modification through platform is forbidden. |
| false | SSO does NOT provide any user roles, and SSO user's roles are managed through platform (unless LDAP Role Provision is enabled). |

1. If SSO Role Provisioning is not possible, and LDAP Role Provision is desired instead, then there is a separate setting that can be enabled:

com.systar.photon.application.auth.ldapRoleProvisioning=true

|  |  |
| --- | --- |
| **Parameter value** | **Description** |
| false | Default value. No roles are retrieved from LDAP. Roles are managed through the platform, unless SSO Role Provisioning is enabled. |
| true | User's roles are retrieved from LDAP. Roles modification through the platform is forbidden. See “Configuring User Directories (LDAP)” documentation for additional configuration requirements required in conf/photon-authentication/settings.xml. |

1. Add the customer-specific HTTP header parameters and ADI roles to conf/platform.properties. Example:

sso.httpHeader.uid=HTTP\_UID

sso.httpHeader.roles=HTTP\_GROUPS

sso.ldap.mappings=L2SUPPORT:LG-FAK-CSPWIRE-DEV-L2SUPPORT;OPSADMIN:LG-FAK-CSPWIRE-DEV-OPSADMIN

sso.httpHeader.roles.delimiter=,

|  |  |
| --- | --- |
| **Property Name** | **Description** |
| sso.httpHeader.uid | Indicates the HTTP parameter name that contains the User ID (i.e. "sm\_user") |
| sso.httpHeader.roles | Indicates the HTTP parameter name that contains the LDAP Groups (i.e. "groups"). Role names must be alpha-numeric. |
| sso.httpHeader.roles.delimiter | Indicates how to parse the role data passed within the HTTP header (optional, uses a comma by default) |
| sso.ldap.mappings | Indicates the LDAP-to-ADI mapping for role names where pairs are mapped by colons and separated by semi-colons (i.e. <ADI-Group1>:<LDAP-Group1>;<ADI-Group2>:<LDAP-Group2>). This is only required when SSO Role Provision is enabled. |

1. Add the Single sign-on agent file “generic-ssoagent-X.jar“ to the lib/plugins directory. This agent is a JAR implementing an OSGi bundle, with a J2EE filter providing the authentication information. Source code can be provided.

# Isolating common problems

Additional logging can be enabled by dynamically adding the following loggers via the Log configuration Web interface, or by inserting the below two lines in log4j.properties file located under the “conf” folder and restarting the server:

log4j.logger.com.axway.generic.ssoagent=DEBUG

log4j.logger.photon-authentication=DEBUG

# Resources:

Authentication and authorization:

<https://techweb.axway.com/public/display/DOCTND/Authentication+and+authorization>

Configure Single sign-on (SSO):

<https://techweb.axway.com/public/pages/viewpage.action?pageId=21432374>

Develop a single sign-on agent:

<https://techweb.axway.com/public/display/DOCTND/Develop+a+single+sign-on+agent>

# Notes:

There are additional SSO-related parameters that are not covered here. Please read the documentation for these topics:

* Authentication error page
* Logout landing page
* Hide logout button
* Proxy settings
* HTTPS connections

Also, this agent does not currently support the additional SSO attributes, but this component can be easily enhanced to support these if needed:

* First Name
* Last Name
* Email
* Description