HTTP SSO

Overview

The Single Sign-on (SSO) add-on allows you to reuse existing HTTP-based SSO infrastructures with the JFrog Platform Unit (JPD), such as the SSO modules offered by Apache HTTPd.

You can have JPD authentication work with commonly available SSO solutions, such as native NTLM, Kerberos etc.

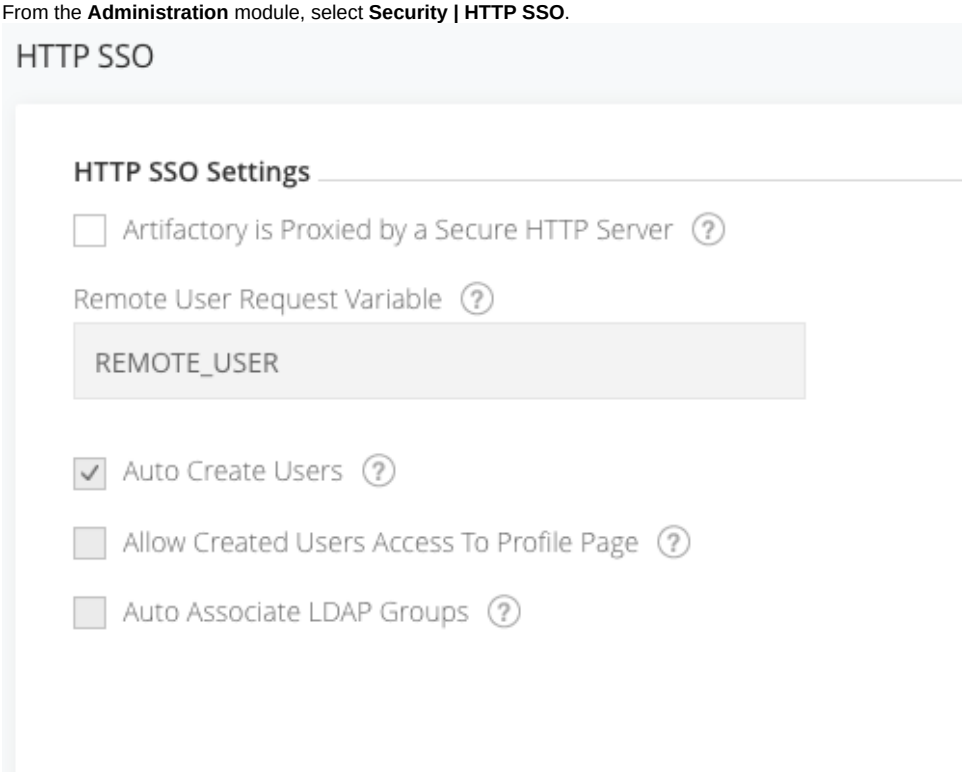
SSO works by letting JPD know what trusted information it should look for in the HTTP request, assuming this request has already been authenticated by the SSO infrastructure that sits in front of Artifactory.

Page Contents

- Overview
- Configuring HTTP SSO in JPD
- Integrating Apache and Tomcat
- Setting Up a Reverse SSL Proxy for SSO
  - Components and Versions
  - Modifying Your Webserver Configuration File
- Using API Key with HTTP-SSO Users

Configuring HTTP SSO in JPD

1. From the **Administration** module, select **Security | HTTP SSO**.



2. Select the **Artifactory is Proxied by a Secure HTTP Server** check box to indicate that Artifactory is running behind a secure HTTP server that forwards trusted requests to it.
3. Add the variable to look for trusted authentication information. The default is to look for a REMOTE_USER header or the request variable, which is set by Apache's AJP and JK connectors.
   You can choose to use any request attribute (as defined by the Servlet specification) by providing a different variable name.

> **Adding Your Own SSO Integration**
>
> You can write a simple servlet filter to integrate with custom security systems and set a request attribute on the request to be trusted by the SSO add-on.

4. Select **Allow Created Users Access to Profile Page** check box to instruct Artifactory to treat externally authenticated users as temporary users, so that Artifactory does not create them in its security database.
   In this case, permissions for such users are based on the permissions given to auto-join groups.
5. Select the **Associate LDAP Groups** check box to associate the user with groups returned in the LDAP login response.

Field Name	Description
Artifactory is Proxied by a Secure HTTP Server	When selected, Artifactory trusts incoming requests and reuses the remote user originally set on the request by the SSO of the HTTP server. This is extremely useful if you want to use existing enterprise SSO integrations, such as the powerful authentication schemes provided by Apache (mod_auth_ldap, mod_auth_ntlm, mod_auth_kerb, etc.). When Artifactory is deployed as a webapp on Tomcat behind Apache: • If using `mod_proxy_ajp`, make sure to set `tomcatAuthentication="false"` on the AJP connector. • If using mod_jk, make sure to use the `JkEnvVar REMOTE_USER` directive in Apache's configuration. Artifactory should be explicitly and exclusively binding to localhost and the reverse proxy collocated on the same machine if “Artifactory is Proxied by a secure HTTP server” is enabled.
Remote User Request Variable	The name of the HTTP request variable to use for extracting the user identity. Default is: REMOTE_USER.
Auto Create System Users	When not checked, authenticated users are not automatically created in the system. Instead, for every request from a SSO user, the user is temporarily associated with default groups (if such groups are defined) and the permissions for these groups apply. Without auto user creation, you must manually create the user inside Artifactory to manage user permissions not attached to its default groups.
Allow Created Users Access To Profile Page	When selected, users created after authenticating using HTTP SSO, will be able to access your User profile. This means they are able to generate their API Key and set your password for future use.
Auto Associate LDAP Groups	When selected, associate the user with groups returned in the LDAP login response. Note that the user's association with the returned groups is persistent if **Auto Create system user** is selected.

Custom URL base

For your HTTP SSO settings to work, make sure you have your Custom URL Base configured.

Integrating Apache and Tomcat

When Artifactory is deployed as a webapp on Tomcat behind Apache:

- If using `mod_proxy_ajp` - Make sure to set `tomcatAuthentication="false"` on the AJP connector.
- If using `mod_jk` - Make sure to use the `JkEnvVar REMOTE_USER` directive in Apache's configuration.
- If using `mod_proxy` (requires mod_proxy_http, mod_headers and mod_rewrite - There are two known working methods that forward the header:

```
RequestHeader set REMOTE_USER %{REMOTE_USER}e
```

or

```
RewriteEngine On
    RewriteCond %{REMOTE_USER} (.+)
    RewriteRule . - [E=RU:%1]
    RequestHeader set REMOTE_USER %{RU}e
```

Setting Up a Reverse SSL Proxy for SSO

You may set up a reverse SSL proxy on your webserver in order to run JPD supporting SSO.

To do this, you need to have the right components installed, modify your webserver configuration file, and then configure Artifactory for SSO.

When correctly set up,you should be able to login to Artifactory with your Windows credentials and stay logged in between sessions.

For best security, Artifactory and the reverse proxy webserver must be co-located on the same machine. Artifactory should be explicitly and exclusively bound to **localhost**.

Components and Versions

The following has been tested to work with Kerberos/NTLM SSO working with JPD using the following components.

- IBM Websphere 8.5.5 running on Windows 8 using the IBM Websphere Java 7 JDK Package.
- The mod_auth_sspi Apache module.

Modifying Your Webserver Configuration File

Once you have the right components and versions installed, you need to add the following lines to your *[HTTP_SERVER_HOME]/conf/httpd.conf* file:

httpd.conf file

```
<VirtualHost *:80>
ServerName yourhostname
DocumentRoot "C:/IBM/Installation Manager/eclipse/plugins/org.apache.ant_1.8.3.v20120321-1730"
ProxyPreserveHost on
ProxyPass /artifactory http://yourhostname:9080/artifactory
ProxyPassReverse /artifactory http://yourhostname:9080/artifactory
</VirtualHost>


<Location /artifactory>
AuthName "Artifactory Realm"
AuthType SSPI
SSPIAuth On
SSPIAuthoritative On
require valid-user
RewriteEngine On
RewriteCond %{REMOTE_USER} (.+)
RewriteRule . - [E=RU:%1]
RequestHeader set REMOTE_USER %{RU}e
</Location>
```

Then you need to enable the following modules in your *httpd.conf* file:

Modules to enable

```
LoadModule sspi_auth_module modules/mod_auth_sspi.so
LoadModule headers_module modules/mod_headers.so
LoadModule proxy_module modules/mod_proxy.so
LoadModule proxy_connect_module modules/mod_proxy_connect.so
LoadModule proxy_http_module modules/mod_proxy_http.so
LoadModule rewrite_module modules/mod_rewrite.so
```

Using API Key with HTTP-SSO Users

While HTTP-SSO provides access to the JPDI, it is also possible for HTTP-SSO users to generate an API Key that can be used instead of a password for basic authentication or in a dedicated REST API header, this is very useful when working with different clients, e.g. Docker, npm, maven, etc. or using Artifactory REST API.

In order to allow HTTP-SSO users access to an API key you will need to make sure that the "**Auto Create Artifactory Users**" and "**Allow Created Users Access To Profile Page**" check boxes are checked. This means that SSO users are also saved in Artifactory database and can access their User Profile in order to generate, retrieve and revoke their API key.