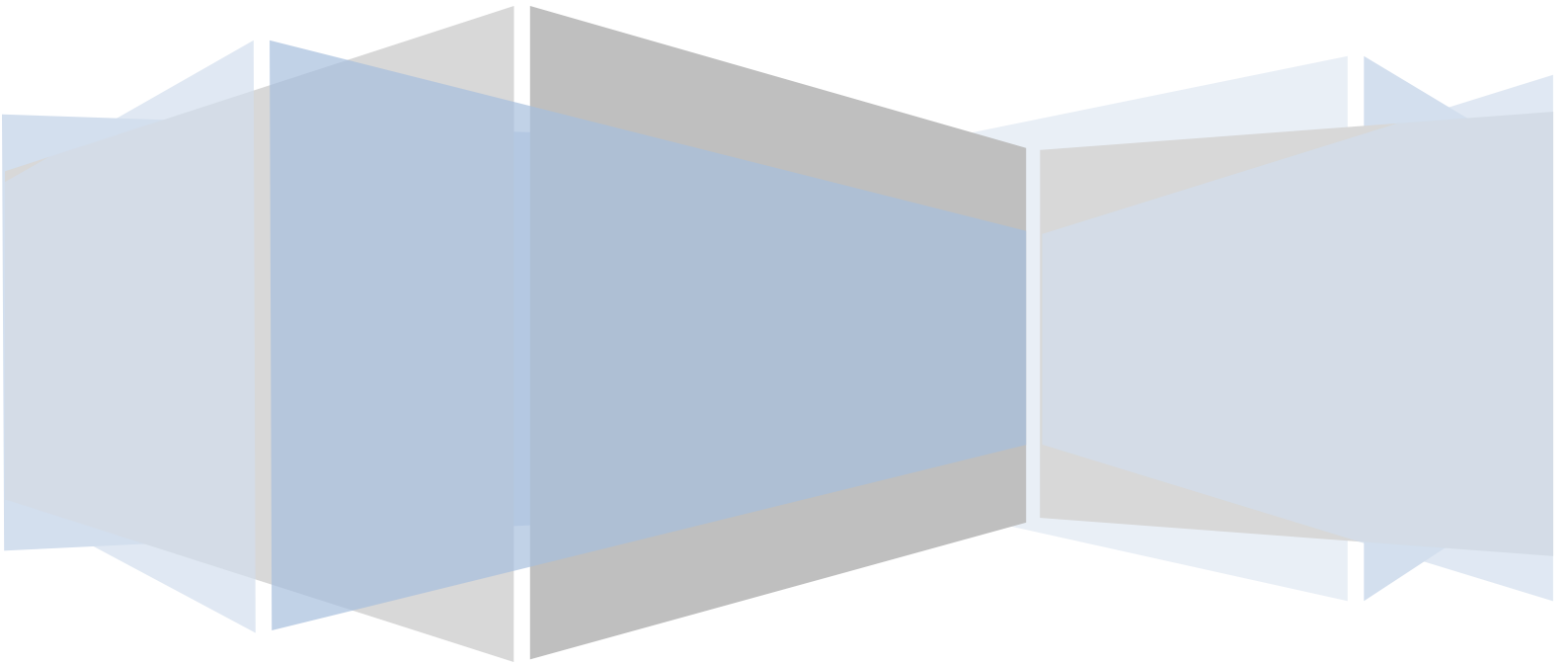
SSO Plugin

Installation for HP Service Manager

J System Solutions

http://www.javasystemsolutions.com

Version 4.0



Introduction

This document covers:

- Compatibility matrix and other introductory material for SSO Plugin.
- Installation and configuration of SSO Plugin for HP Service Manager 9.2+.
- Upgrading from previous versions.

Separate documents are available for other components, ie JasperServer Reports, SAP Business Objects XI, BMC Dashboards, BMC Analytics.

The JSS support website can be found here:

http://www.javasystemsolutions.com/jss/support

Providing new features

It is anticipated that features currently available for BMC AR System, but not included in the HP Service Manager build, will be available in a subsequent release. Whilst only a few features are missing, most notably the functionality to automatically raise an incident when a user has no SSO access to HP Service Manger, if any are of particular importance to an organisation then please contact JSS so we can prioritise development accordingly.

Compatibility

The product has been developed for Service Manager 7.1+ and will run on all platforms.

There are additional installation steps for SM 9.3 due to a change in the way SSO is enabled on the Service Manager server component (discussed in the enabling trusted sign on section).

Please note:

1. We support Tomcat 5.5+, Weblogic 11g+ and Websphere. If you use another Java servlet engine, please contact us to confirm supportability.

Single-sign on integrations and mechanisms

Please consult the Configuring Web Tier document for a full list of supported integrations/mechanisms.

Java compatibility

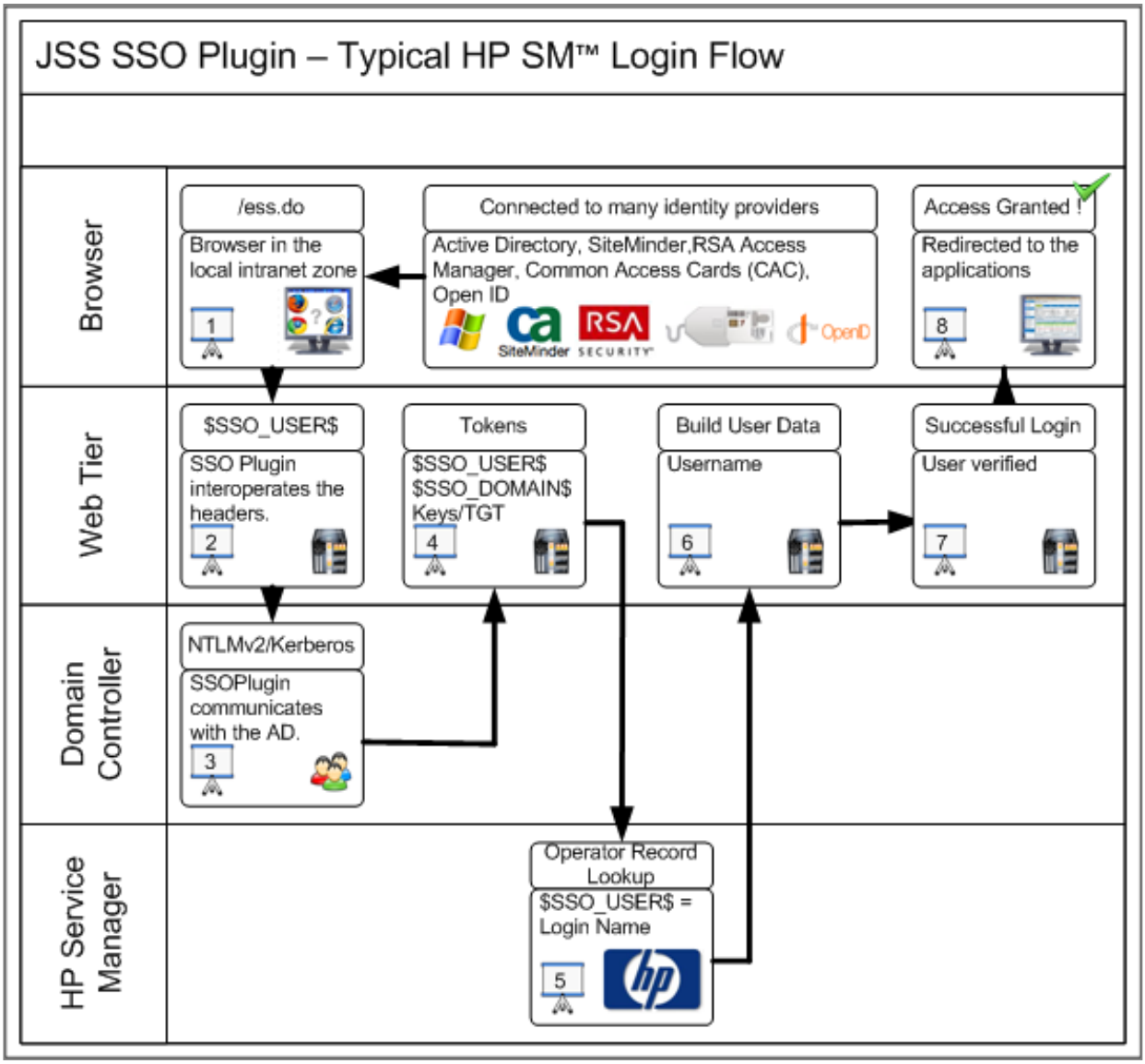
The Web Tier for Service Manager 7.1 requires Java 1.5 and SSO Plugin is supported on version 1.5 update 7 or greater, however we recommend the latest (and final) version of 1.5 which is 1.5 update 22.

The Web Tier for Service Manager 9.2+ runs on Java 1.6.

Overview of the SSO Plugin

The SSO Plugin is invoked by the SM when a user goes to /index.do (support console) or /ess.do (self service console).

If the relevant details were available on the incoming request for the SSO Plugin to operate correctly, then these details are passed back to the SM. SSO Plugin also ensures users have access to SM before proceeding with an SSO login, which includes checking they are not a self service user trying to access the support console.



Installation

To install the SSO Plugin for Service Manager, there are two steps:

1. Configuring the Service Manager server component.
2. Configuring the Web Tier component.

Configuring the Service Manager server component

There are two steps required to set up the SM server component:

1. Enable trusted sign on.
2. Modify the Operator webservice, to allow SSO Plugin running in Web Tier to interact with the Operator table.

Enabling trusted sign on

HP provides comprehensive documentation on how this is achieved, it is summarised below.

Service Manager 7.x and 9.2

For SM 7.x and 9.2, add the following to the smi.ini file:

```
trustedsignon:1
```

And restart Service Manager.

Service Manager 9.3+

For SM 9.3+, trusted sign on will only function when client certificates are enabled and in use by each Web Tier.

HP provide batch files for generating the client certificate but they are not easy to use. We can generate the client certificates for you if provided with the hostnames of the Service Manager and Web Tier hosts. Simply email support@javasystemsolutions.com with this information and we will be pleased to assist.

When you have client certificates, enable trusted sign on by adding the following to the sm.ini file:

```
trustedsignon:1
httpsPort:13443
ssl:1
sslConnector:1
ssl_reqClientAuth:2
keystoreFile:server.keystore
keystorePass:serverkeystore
ssl_trustedClientsJKS:trustedclients.keystore
ssl_trustedClientsPwd:trustedclients
truststoreFile:cacerts
truststorePass:changeit
```

The items in bold represent the client certificate configuration. The files server.keystore, trustedclients.keystore and cacerts should be placed in the SM RUN directory (ie in the same directory as sm.ini). The passwords (ie serverkeystore, clientkeystore, changeit) are the default values set by JSS when generating client certificates; separate values can be generated upon request.

After configuring sm.ini, restart Service Manager.

Configuring the Web Tier

To configure a Web Tier with client certificates, copy the cacerts and client keystore into the WEB-INF directory. If the certificates were generated by JSS then the client keystore is a file with the name hostname-of-webtier.keystore.

Now locate the Web Tier web.xml file, open it in your favourite text editor, locate and make the following changes:

```
<init-param>
  <param-name>ssl</param-name>
  <param-value>true</param-value>
<init-param>
<iinit-param>
    <param-name>cacerts</param-name>
    <param-value>/WEB-INF/cacerts</param-value>
</init-param>
<init-param>
  <param-name>keystore</param-name>
  <param-value>/WEB-INF/hostname-of-webtier.keystore</param-value>
</init-param>
<init-param>
  <param-name>keystorePassword</param-name>
  <param-value>clientkeystore</param-value>
</init-param>
```

The keystorePassword (clientkeystore) is the default provided by JSS and a specific password can be set upon request.

Providing non-SSL access for thick clients

Whilst client certificates provide an additional layer of security during the SSO process, it is time consuming to generate and configure a client certificate for each thick client instance. To avoid having to do this, you may wish to run a separate SM instance with SSL disabled and use this for the thick clients. This can be achieved by editing and adding a line to the sm.cfg file as shown in bold below:

```
sm -httpPort:13099 -ssl:0 -sslConnector:0
sm
sm system.start
```

This assumes the httpPort 13099 is defined as some other value in the sm.ini file, and that port 13099 is not in use by any other process.

After restarting Service Manager, your thick clients should now be able to connect to port 13099.

Disabling Operator LDAP

If you have LDAP enabled with a mapping to the Operator table, delete the mapping from the LDAP mappings. This is because trusted sign on does not appear reliable with the LDAP integration, and SSO Plugin provides the ability for users to authenticate against a Windows domain controller once SSO has been enabled, so the Operator LDAP integration becomes redundant for most deployments.

Please ensure the following is present in sm.ini:

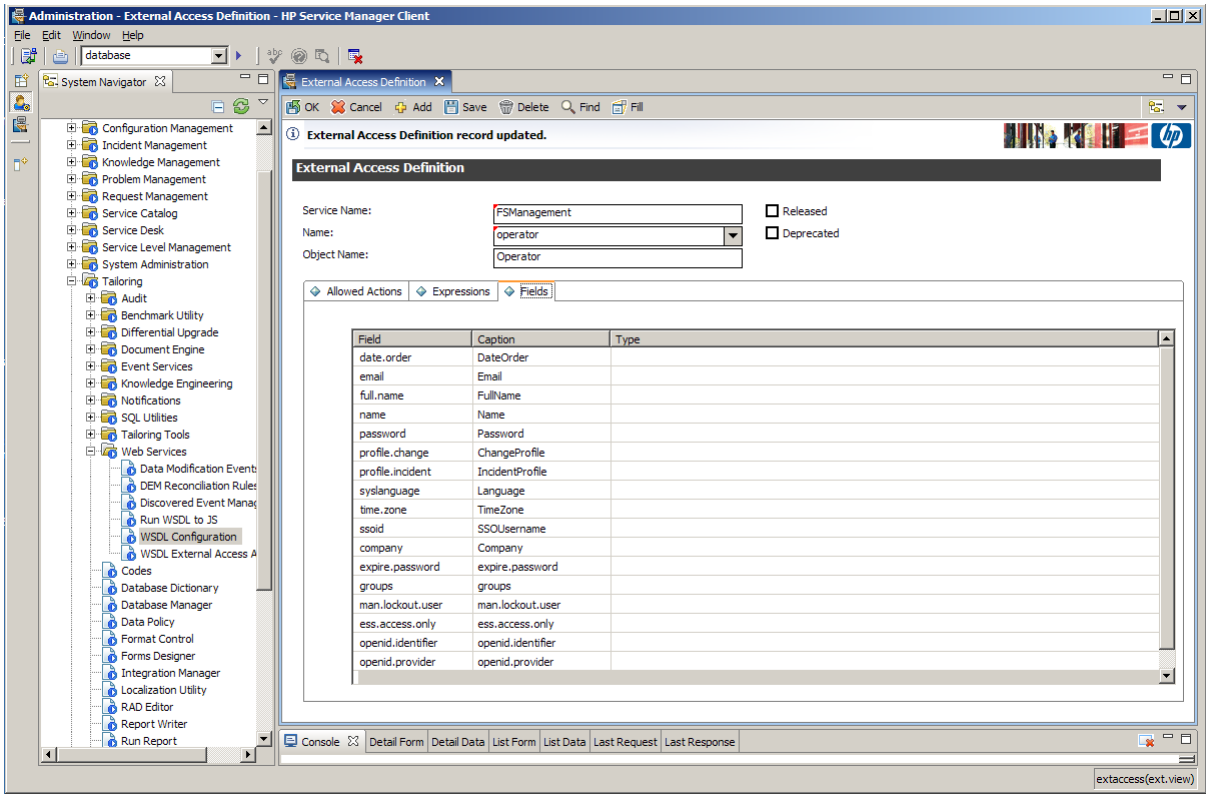
```
ldapdisable:1
```

Modifying the Operator webservice

SSO Plugin uses the webservice interface to perform queries against the Operator table, however the default webservice definition does not expose all the required fields.

1. Locate the Operator web service in the WSDL configuration (Tailoring -> Webservices -> WSDL Configuration), select operator in the name drop down field and press search.
2. Select the Operator (not Operator.700) object.
3. Delete the mapping from cap.exec to CapabilityWords - this has been known to send large lists of capability words to the webservice client causing memory issues.
4. Add mappings for the following fields using captions of the same value:
    1. company
    2. expire.password
    3. groups
    4. man.lockout.user
    5. ess.access.only
    6. login.revoked

The following screenshot shows the operator webservice configuration. Adding extra fields is performed through Tailoring -> Forms designer -> operator.g form.



Please note, there are three extra fields that can be added to the Operator form that may also be added for additional SSO Plugin functionality:

- ssoid: For supporting user aliasing, where the SSO username does not match the Operator login username. More information on this feature is available in the Configuring Web Tier document.
- openid.provider and openid.identifier: For integrating with an OpenID provider such as Google Mail.

These fields are not required for SSO Plugin to perform a standard set of operations that are required by most users. The extra fields are only required for advanced configurations, the most common being user aliasing: mapping an SSO ID to a different Operator login name.

Installing the Web Tier component

A separate highly detailed JSS document ("Configuring Mid Tier and Web Tier") explains how to configure SSO Plugin for Web Tier, with details on the integration process between SSO Plugin and third party SSO providers (such as Active Directory). This section only covers the installation process.

To install the SSO Plugin on the Web Tier, please follow these steps:

1. Copy the contents of the webtier directory from the installation files into the root Web Tier directory. i.e. the contents of webtier into the Web Tier directory that *contains* the WEB-INF directory. If asked to confirm overwriting of files, click yes: SSO Plugin only adds to the files.
2. When deploying Service Manager 9.2+ within Websphere, there is also a requirement to modify the standard Service Manager smlogin.jsp file found in the war file. Open the file in a text editor, search for the following block of code and add the text in bold:

```
else {
  // nothing to remove, carry on.
  urlParams = (null == queryString) ? "" : "?" + queryString;
}
urlParams+= urlParams.length()==0 ? "?" : "&";
urlParams+= "sso=false";
```

3. For Service Manager version 9.21 or greater, a manual change must be made to the Web Tier configuration file to enable third party files to be run.
    1. Locate the Web Tier application-context.xml file in the WEB-INF/classes directory.
    2. Add the text in bold to the file, the non-bold area can be found at approximately line 30:

```
/loginpreload.jsp=#NONE#
/jss-sso/**/*.jsp=#NONE#
/**/*.jsp=resourceFilter
/*.jsp=resourceFilter
/**=httpSessionContextIntegrationFilter,anonymousProcessingFilter
```

4. Restart Web Tier.
5. Go to the SSO Plugin status page by pointing your browser at http://host/webtier/jss-sso/index.jsp. You will be presented with a status page.
6. Locate the document titled Configuring Web Tier to configure the SSO Plugin.
7. If SSO fails then review the troubleshooting document or contact JSS support.

Licensing

The product is licensed by generating a license in the support area of the JSS website.

The licensing tool allows two month trial licenses to be generated, or if you've purchased the product, a permanent license can be generated by entering the name of the Service Manager server (as defined in the web.xml file).

After generating a license, enter it into the SSO Plugin 'Configuration' page.

Upgrades

Assuming SSO Plugin is installed and working, or at least configurable, the steps are as follows:

1. Go to the SSO Plugin status page (ie http://host/webtier/jss-sso/index.jsp), login and disable SSO Plugin on Web Tier.
2. Stop Tomcat.
3. Replace the Web Tier files, ie copy the contents of the webtier directory into the Web Tier directory in the Tomcat webapps directory.
4. Delete the Tomcat 'work' directory, which is a temporary cache directory re-created when Tomcat starts.
5. Start Tomcat.
6. Go to the Web Tier SSO configuration, check it is still correct and press 'set configuration'.

SAP Business Objects and Jasper Reports.

Copy the relevant jar files from the installation files to the third party application.

For example, copy the jar files in businessobjects/WEB-INF/lib (from the installation files) to the relevant location in the Business Objects installation, as per the original deployment.