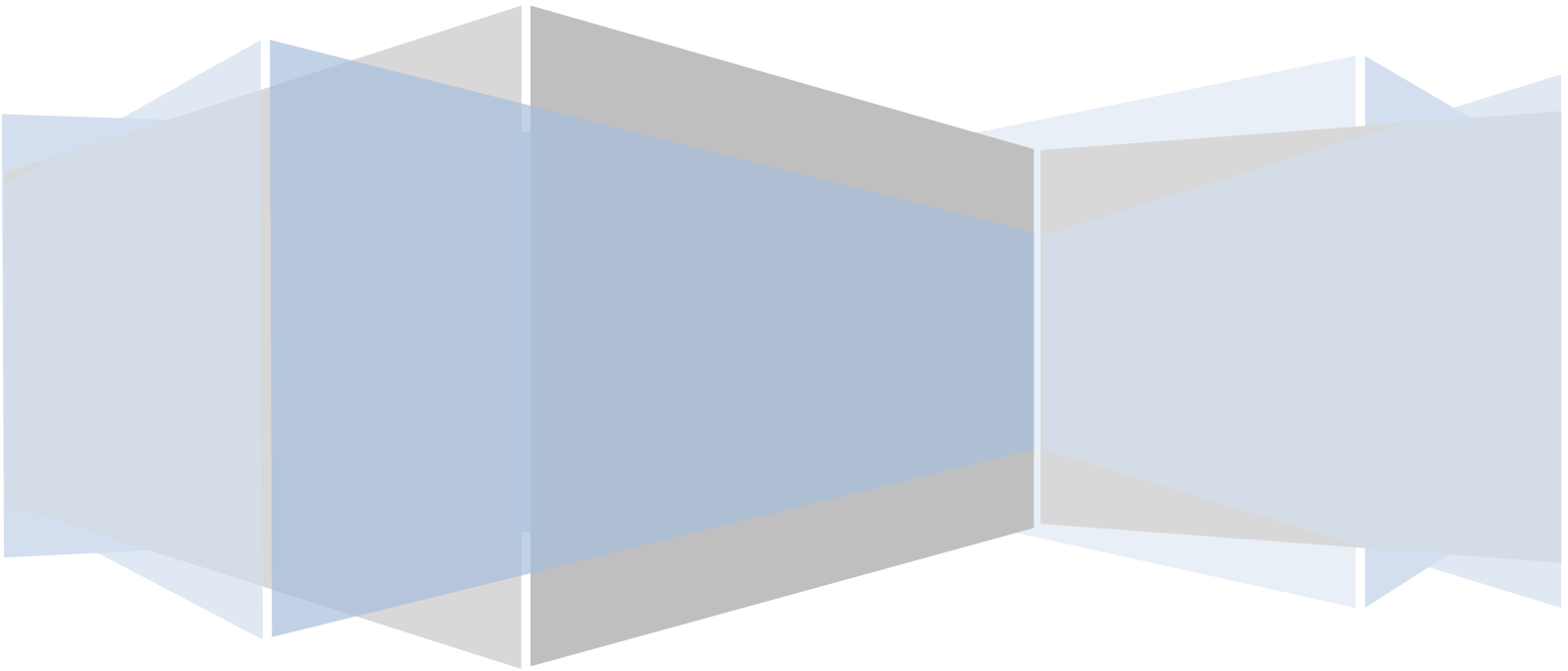
SSO Plugin

Integration with BMC ITBM

J System Solutions

http://www.javasystemsolutions.com

Version 3.5



Introduction.....3
SSO Plugin integration.....3
Supported Java web servers.....3
Configuring SSO Plugin.....4
Installing SSO Plugin for ITBM.....5
JBoss logging.....8
Setting Java system properties.....9
Identity Federation Service URL.....9
Log level.....9
Setting JVM options in JBoss.....9

Introduction

This document covers installation and configuration of SSO Plugin for BMC ITBM. Separate documents are available for other components (ie BMC AR System, BMC Dashboards, BMC Analytics, JasperReports Server).

The JSS support website can be found here:

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

SSO Plugin integration

SSO Plugin runs on the Mid Tier providing corporate SSO, and also extends SSO to ITBM through the JSS *Identity Federation Service*. This allows third party products, BMC or non-BMC, to be SSO enabled with the Mid Tier (and hence, the AR System User form) as a single repository of data.

The integration leaves ITBM available for both SSO and non-SSO usage.

Supported Java web servers

BMC ITBM is a new BMC product and is currently deployed on JBoss 4.3, the Red Hat supported version of JBoss 4.2.3. The integration has been validated against Jboss 4.3 and JSS are happy to evaluate other Java web server as and when requested to do so.

Configuring SSO Plugin

You must first set up SSO Plugin to enable the Identity Federation Service:

1. Login to the Mid Tier SSO Plugin setup page.
2. Tick 'Enable Identity Federation Service'.
3. Enter a unique key or press the button to create one. Take a note of the key.
4. Press 'Set configuration' and ensure the SSO Plugin still functions using the 'Test SSO' link.

Installing SSO Plugin for ITBM

To enable SSO Plugin for Business Objects, the following steps must be followed to copy the plugin to ITBM. Due to the packaging of ITBM, the process involves use of the command line and if you are not comfortable with this approach, please contact JSS for assistance.

1. Ensure the Java jar command is available from the command prompt. To do this, open a command prompt and type jar. If you see “file not found” then add the JDK bin directory to the command prompt path.
2. Stop the Java web server instance running ITBM.
3. Locate the itmsoft-jboss4-form.ear file. This will be found in the JBoss server/production/deploy directory or within the installation files. Please do not use the itmsoft-X-sso.ear file for this process.
4. Go to a temporary directory and copy the ear file. Create two new directories, ear and web.
5. Open a command prompt, change into the ear directory and run the following command:

```
jar xf ..\itmsoft-jboss4-form.ear
```

6. Change into the war directory and run the following command:

```
jar xf ..\ear\itm-web.war
```

7. Locate the war file web.xml file in the WEB-INF directory. It requires patching to allow SSO Plugin to intercept the browser requests. Follow these steps to patch it:
   7.1. Open the web.xml.patch file from installation files directory, select all and copy to clipboard.
   7.2. Open the web.xml file from the war/WEB-INF directory.
   7.3. Locate the last </content-param> element and past the text copied to the clipboard from the patch file, ie.

```
<context-param>
  <param-name>AjaxPartsTaglibConfig</param-name>
  <param-value>/WEB-INF/ajax-portal.xml</param-value>
</context-param>
<!-- Patch for SSO Plugin -->
<filter>
  ...
```

   7.4. Locate the Identity Federation Service key parameter (in the patch, pasted into the web.xml file) and enter the value set in the SSO Plugin web interface when configuring SSO Plugin.

In other SSO Plugin implementations for third party products, the URL of the Identity Federation Service and log level are set in the

web.xml. Given the difficulty in accessing the web.xml file, they are set through Java system properties set on the JVM.

8. Save the web.xml file.
9. Locate the jss-sso-itbm.jar file from the installation files. Copy it to the following directories:
   9.1. The war/WEB-INF/lib directory.
   9.2. The ear directory, so it is in the same directory as the itm-common.jar and itm-ejb.jar files.
   9.3. The JBoss server/production/lib directory.
10. Locate the thirdparty directory in the ear directory. Copy the commons-logging.jar and log4j.jar files to the war/WEB-INF/lib directory.
11. Locate the application.xml file in the ear/META-INF directory. Open the file in a text editor and locate the following:

```
<module>
  <ejb>itm-ejb.jar</ejb>
</module>
```

Add the following module immediately after the one above:

```
<module>
  <java>jss-sso-itbm.jar</java>
</module>
```

12. Locate the original login.jsp page in the login directory. It requires a small change to set no cache headers, to stop browsers (principally IE) caching the page. This is important as we always want the page to load, or go to the SSO service if applicable. Open the file in a text editor and add this line to the top:

```
<% response.setHeader(“Cache-Control”, “no-cache”); %>
```

13. Rebuild the war file by changing into the war directory and running the following command:

```
jar cf ..\itm-ear\itm-web.war *
```

14. Rebuild the ear file by changing into the ear directory and running the following command:

```
jar cf ..\itmsoft-jboss4-form.ear *
```

15. Copy the ear file to the JBoss server/production/deploy directory.
16. Locate the login-config.xml file in the JBoss server/production/config directory. Open it in a text editor and look for the following entry:

```
<application-policy name="itmRealm">
  <authentication>
    <login-module
code=”com.itmsoft.security.auth.module.ITBMLoginModule”
flag=”required”>
```

and change the values highlighted in bold:

```
    <login-module
code="com.javasystemsolutions.sso.integrations.itbm.ITBMLoginModule"
flag="optional">
```

17. Locate the server.xml.xml file in the JBoss server/production/deploy/jboss-web.deployer directory. Open it in a text editor and look for the following entry:

```
    <Valve cachedConnectionManagerObjectName="..." />
```

and add the following line below:

```
    <valve
className="com.javasystemsolutions.sso.integrations.itbm.SSOPluginVa
lve" />
```

18. Locate the JBoss run.bat/run.sh file and add the Identity Federation Service URL and log level as Java options:

```
set JAVA_OPTS=-
Djss.ssoplugin.ifs.url=http://bmcMidTier:8080/arsys/jss-
sso/identityfederationservice -Djss.ssoplugin.ifa.loglevel=INFO
```

This is described in more detail in the setting Java system properties section.

19. Start the JBoss server.

20. Ensure you have SSO access to BMC ITSM, and then navigate to: http://host/itbm/jss-sso. If the user has SSO access, they will be signed in, otherwise they will be sent to a login screen. The standard login page is available by going to http://host/itbm.

JBoss logging

The default JBoss 4.3.0 installation has the logging set to DEBUG. This generates an awful lot of logging in the server/production/logs/server.log file. To reduce the amount of logging, and increase the number of backup log files, follow these steps:

1. Locate the server/production/conf/jboss-log4j.xml file. Open it in a text editor.
2. An appender is defined to create the server.log file and by default it is set to create 500Kb files with one back. Locate this section and modify the values in bold (you can define your own):

```
<appender name="FILE"
class="org.jboss.logging.appender.RollingFileAppender">
  <errorHandler
class="org.jboss.logging.util.OnlyOnceErrorHandler"/>
  <param name="File" value="${jboss.server.log.dir}/server.log"/>
  <param name="Append" value="false"/>
  <param name="MaxFileSize" value="1000KB"/>
  <param name="MaxBackupIndex" value="10"/>
```

3. At the end of the file, locate this section and add the text in bold:

```
<root>
  <priority value=”info” />
  <appender-ref ref="CONSOLE"/>
  <appender-ref ref="FILE"/>
</root>
```

4. Whilst this is only required for debugging purposes, if you wish to modify the log level of the JSS login module (ie the change made to the login-config.xml file during installation), add the following category below any other category in the file:

```
<category name="com.javasystemsolutions">
  <priority value="DEBUG"/>
</category>
```

After modifying this file, JBoss must be restarted for the changes to take affect.

Setting Java system properties

In order to make re-deployment of the ear file easier, without the need to modify the web.xml file (where the Identity Federation Service values are defined), the following Java options are set.

Identity Federation Service URL

Property: jss.ssoplugin.ifs.url

Description: This points to the identity federation service running on the SSO Plugin installation. The identity federation service URL is /jss-sso/identityfederationservice, relative to the Mid Tier installation. Therefore, if the Mid Tier is installed at:

```
http://bmcMidTier:8080/arsys
```

then the value is:

```
http://bmcMidTier:8080/arsys/jss-sso/identityfederationservice
```

Log level

Property: jss.ssoplugin.ifa.loglevel

Description: This tells the JSS code how much logging to produce. The acceptable values are info, debug and trace.

Setting JVM options in JBoss

In JBoss, to set a Java option, modify the run.bat file and use the JAVA_OPTS variable to set an option, ie.

```
export JAVA_OPTS=-Dkey=value -Dkey2=value (Unix)
set JAVA_OPTS=-Dkey=value -Dkey2=value (Windows)
```