Configuring Integrated Windows Authentication for JBoss with SAS® 9.2 Web Applications

Copyright Notice

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *Configuring Integrated Windows Authentication for JBoss with SAS 9.2 Web Applications*, Cary, NC: SAS Institute Inc., 2010.

Configuring Integrated Windows Authentication for JBoss with SAS 9.2 Web Applications

Copyright © 2010, SAS Institute Inc., Cary, NC, USA.

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, by any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc. Limited permission is granted to store the copyrighted material in your system and display it on terminals, print only the number of copies required for use by those persons responsible for installing and supporting the SAS programming and licensed programs for which this material has been provided, and to modify the material to meet specific installation requirements. The SAS Institute copyright notice must appear on all printed versions of this material or extracts thereof and on the display medium when the material is displayed. Permission is not granted to reproduce or distribute the material except as stated above.

U.S. Government Restricted Rights Notice. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987).

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries.

® indicates USA registration.

Other brand and product names are trademarks of their respective companies.

Table of Contents

Integrated Windows Authentication for JBoss 3

Overview of Integrated Windows Authentication 3

Integrated Windows Authentication for JBoss 3

- Configuring JBoss for Integrated Windows Authentication 4
- Configuring Web Authentication 5
- Adding JAR Files to the SAS Web Infrastructure Platform Services 5
- Configuration Tasks on the Active Directory Domain Controller Machine 5
  - Create a Group in the Microsoft Active Directory 5
  - Create a User Account in the Microsoft Active Directory 5
  - Configure Kerberos SPN for JBoss Application Server 6
  - Create the Kerberos Keytab File Used by SPNEGO 7
- Configuration Tasks on JBoss 8
  - Copy the Keytab File to the JBoss Application Server 8
  - Create the Kerberos Configuration Files 8
  - Verify Kerberos Authentication 9
- Modifying the login-config.xml File 9
- Modifying SAS Logon Manager 10
- Configuring the Files for the SPNEGO User Properties and Role Properties 11
- Configuring the Client Browser to Use SPNEGO 11
  - Configure Local Intranet Domains 11
  - Configure Intranet Authentication 11
  - Verify the Proxy Settings 12
  - Specify Integrated Authentication for Internet Explorer 12
- Verifying IWA 12
- Troubleshooting SPNEGO Support 12

Recommended Reading 12

Integrated Windows Authentication for JBoss

Overview of Integrated Windows Authentication

Integrated Windows Authentication (IWA) is a Microsoft technology that is used in an intranet environment where users have Windows domain accounts. With IWA, the credentials (user name and password) are hashed before being sent across the network. The client browser proves its knowledge of the password through a cryptographic exchange with your Web application server.

The key components of IWA include an Active Directory Controller machine (Windows 2000 Server or higher), Kerberos Key Distribution Center (KDC) in a Domain Controller machine, a machine with a client browser, and a Web application server.

When used in conjunction with Kerberos, IWA enables the delegation of security credentials. Kerberos is an industry-standard authentication protocol that is used to verify user or host identity. The Kerberos protocol uses strong cryptography so that a client can prove its identity to a server (and vice versa) across an insecure network connection. After a client and server have used Kerberos to provide their identity, they can also encrypt all of their communications to assure privacy and data integrity.

If Active Directory is installed on a Domain Controller running Windows 2000 Server (or higher) and the client browser supports the Kerberos authentication protocol, Kerberos authentication is used.

Use of the Kerberos protocol is guided by the following requirements:

- The client must have a direct connection to Active Directory
- Both the client and the server must have a trusted connection to a Key Distribution Center (KDC) and be Active Directory-compatible
- Service Principal Names (SPNs) are required for multiple worker processes.

Integrated Windows Authentication for JBoss

When IWA is configured, HTTP clients use a Windows login user name to access the SAS Web applications deployed in the WebSphere application server without any authentication challenge.

To configure IWA for JBoss 4.2 and create a single sign-on for HTTP requests using the Simple and Protected GSS-API Negotiation Mechanism (SPNEGO), the following requirements should be met:

- SAS 9.2 or the October 2009 Maintenance Release of SAS 9.2 is installed and configured on JBoss 4.2
- The JBoss SPNEGO 2.0.3GA or 2.0.3.sp1 module is installed

- Web authentication. Complete the configuration of Web authentication. For instructions on configuring Web authentication for JBoss, see "Configuring JBoss Application Server 4.2.0 for Web Authentication with SAS 9.2 Web Applications".
- Modifications to Web authentication. Complete the modifications to Web authentication.
- An Active Directory Domain Controller running Windows 2000 Server or higher
- Machine with a client browser. This is a Microsoft Windows 2000 (or higher) domain member that has a browser client and supports the SPNEGO authentication mechanism. Microsoft Internet Explorer Version 7.0 or later qualifies as the client.
- JBoss Web application server that is running. Users on the active directory must have access to JBoss.
- The domain controller and the JBoss application server have the same local time
- The clock on all three machines are synchronized to within five minutes

Configuring JBoss for Integrated Windows Authentication

The JBoss Negotiation 2.0.3 module is available for download. When the downloaded module is unzipped, you will have a copy of the *User Guide for JBoss Negotiation – A Guide for Administrators*. Follow the instructions in the user guide to complete the following tasks:

1. Download and install the JBoss Authenticator module. This is also known as the SPNEGO Authenticator and Login Module. This module is available in the **jboss-negotiation-2.0.3.GA.jar** file and should be placed in the **`JBOSS_HOME/server/SASServer1/lib`** directory.
2. In the **jboss-service.xml** file located in the **`JBOSS_HOME/server/configuration/deploy/jboss-web.deployer/META-INF`** directory, define the SPNEGO Authenticator class. For instructions, see the JBoss documentation.
3. Set up the following properties for the realm: **java.security.krb5.realm** (Kerberos realm) and **java.security.krb5.kdc** (hostname of the KDC). This step ensures that the JBoss machine can authentication against a Kerberos KDC. If you start JBoss as a service, add the KDC realm properties as the JVM properties to the **wrapper.conf** file. Your configuration should resemble the content of this example:

```
wrapper.java.additional.33=-Djava.security.krb5.realm=ABC.SAS.COM
wrapper.java.additional.34=-Djava.security.krb5.kdc=redwood2.abc.sas.com
```

4. After the modifications to the **login-config.xml** file, create **spnego-users.properties**.
5. As an alternative to adding the JVM arguments, update the system properties service in the **properties-service.xml** file by defining properties in a descriptor.

These properties go into effect when JBoss is started. Following is an example:

```
<attribute name=”Properties”>
java.security.krb5.kdc=Kerberos.security.jboss.org
java.security.krb5.realm=KERBEROS.JBOSS.ORG
</attribute>
```

6. Add Multiple KDCs or Java options for each SASServer definition.

Configuring Web Authentication

An initial SAS 9.2 installation uses the Metadata Server to authenticate users who log into SAS Web applications. You must configure the JBoss application server to use Web authentication. The credentials supplied by the users are authenticated in the Active Directory by the Domain controller.

For instructions on configuring Web authentication for JBoss, see “Configuring JBoss Application Server 4.2.0 for Web Authentication with SAS 9.2 Web Applications.”

Adding JAR Files to the SAS Web Infrastructure Platform Services

If you are using the second Maintenance Release for SAS 9.2 or a previous release, add the following JAR files into the **sas.wip.services.war** file:

sas.svc.sec.login.jboss.jar

sas.svc.sec.login.jboss.nls.jar

Configuration Tasks on the Active Directory Domain Controller Machine

To perform tasks on the Microsoft Active Directory domain controller machine, you should be familiar with Active Directory Users and Computer on a Windows server. This task is required to process single sign on browser requests to the JBoss application server and SPNEGO.

For instructions on how to use the Active Directory Users and Directory, refer to the product’s online Help.

Complete the following tasks on the Microsoft Active Directory domain controller machine.

Create a Group in the Microsoft Active Directory

Create an organizational unit or group for user accounts, for example, SASIWAUsers, in the Active Directory on the Windows server. Active Directory users who will be allowed to access SAS Web applications will require membership in this group. Later in the configuration, it will be mapped to a JAAS authorization role, which, in turn, is used by the Web application server for determining authorization to the SAS Web applications.

Create a User Account in the Microsoft Active Directory

The IWA configuration requires a user account within the Active Directory to represent the Web Application Server process. This account does not represent the host; it represents the individual process running on the host.

1. (Optional). On the domain controller machine, run the following command to find the principals for all users:

   ```
   dsquery user
   ```

2. Create a user account (for example, iwauser) within the Active Directory Users and Directory window. This user account will eventually be mapped to the Kerberos service principal name (SPN). Make sure that the following options are selected when you create the user: **User cannot change password** and **Password never expires**. Note the password you defined when creating the user account. You will need it later.
3. Configure the new user account to comply with the Kerberos protocol.
   a) Right-click the name of the user account in the Users tree in the left pane and select **Properties**.
   b) In the Properties dialog box for the user, click **Account** tab.
   c) Under Account Options, select the following:

      Password never expires

      Use DES Encryption types for this Account (Do not select this option if running Windows 2008.)

      Do not require Kerberos preauthentication

      Selection **"Do not require Kerberos preauthentication"** is optional.

   d) Setting the encryption type might corrupt the password. Therefore, reset the user password by right-clicking the name of the user account, selecting **Reset Password**, and re-entering the same password specified earlier.
4. Add the user to the organizational unit or group that you created.

Configure Kerberos SPN for JBoss Application Server

The Microsoft Active Directory provides support for service principal names (SPN), which are a key component in Kerberos authentication. SPNs are unique identifiers for services running on servers. Every service that uses Kerberos authentication needs to have an SPN set for it so that clients can identify the service on the network. An SPN usually looks something like name@YOUR.REALM. You need to define an SPN to represent your JBoss Server in the Kerberos realm. If an SPN is not set for a service, clients have no way of locating that service. Without correctly set SPNs, Kerberos authentication is not possible.

1. On the Active Directory Controller ***or any other machine where you have the correct domain permissions***, access the command prompt window to use the **setspn** commands.

2. Before executing the **setspn** commands, verify that there are no additional mappings already configured for the users:

   ```
   setspn -l HTTP/fully-qualified-host-name
   ```

   (Note that `-l` is a lower case L.)

   No Service Principal Names should be presented.

3. Enter the following commands for SPNs by using correct capitalization of letters and substituting the host name and user name that you created earlier:

   ```
   setspn -a HTTP/hostname username
   setspn -a HTTP/fully-qualified-host-name username
   ```

   Here is an example of the use of the **setspn** commands:

   ```
   setspn -a HTTP/redwood2.abc.sas.com iwauser
   setspn -a HTTP/redwood2 iwauser
   ```

4. Run the **setspn** command to view the SPN that you created:

   ```
   setspn -l username
   ```

   (Note that `-l` is a lower case L.)

   This is an important step. If the same service is linked to different accounts in the Active Directory server, the client will not send a Kerberos ticket to the server.

Create the Kerberos Keytab File Used by SPNEGO

A *keytab* is a file containing pairs of Kerberos principals and encrypted keys (these are derived from the Kerberos password). The keytab file contains the requisite information for the WebLogic Server to authenticate to the Key Distribution Center (KDC). Keytab files are copied to the WebLogic Server and must be readable by the user account running the WebLogic Server.

1. Create the Kerberos keytab file and make it available to the JBoss application server. Use the **ktpass** command to create a user mapping and the Kerberos keytab file:

   ```
   ktpass -out C:\hostname.host.keytab -mapuser username -crypto DES-CBC-MD5 -princ
   HTTP/fully-qualified-domain-name@URL address -pass password -ptype
   KRB5_NT_PRINCIPAL
   ```

   The **ktpass** command creates the **hostname.host.keytab** file. Note that the input for the **–crypto** parameter depends on type of Windows server used in your environment.

   Here is an example of the use of the **ktpass** command and the options which create the **redwood2.host.keytab** file:

   ```
   ktpass -out C:\keytab\redwood2.host.keytab -mapuser iwauser -crypto DES-CBC-MD5 -
   princ HTTP/redwood2.abc.sas.com@ABC.SAS.COM -pass password -ptype KRB5_NT_PRINCIPAL
   ```

The following table explains the options used with the **ktpass** command.

Option	Explanation
-out	The key is written to this output file.
-mapuser	The key is mapped to this user.
-crypto DES-CBC-MD5	This option uses the single DES encryption key.
-princ	Principal name.
-pass	This option denotes the password for the user ID.
-ptype KRB5_NT_PRINCIPAL	This option specifies the KRB5_NT_PRINCIPAL principal value. Specify this option to avoid warning messages.

Configuration Tasks on JBoss

To enable the use of SPNEGO for JBoss, the Kerberos configuration must be completed. Configuration tasks on JBoss include copying the keytab file to the appropriate directory, and creating the Kerberos configuration file, **krb5.ini** on Windows.

Copy the Keytab File to the JBoss Application Server

On Windows, copy the Keytab file from the Active Directory Controller machine to this directory: **`C:\WINNT\`*`keytab filename`*** on the JBoss application server.

Create the Kerberos Configuration Files

1. On Windows, create a directory: **`C:\WINNT`**.
2. On Windows, create the **krb5.ini** file and save it in the **`C:\WINNT`** directory.

   The content in the **krb5.ini** file should resemble the following example:

```
[libdefaults]
    default_realm = ABC.SAS.COM
    default_keytab_name = FILE:C:\keytab\redwood2.host.keytab
    default_tkt_enctypes = des-cbc-md5
    default_tgs_enctypes = des-cbc-md5
    kdc_default_options = 0x54800000
            ticket_lifetime = 600

[realms]
    ABC.SAS.COM = {
            kdc = redwood1.abc.sas.com:88
            admin_server= redwood1.abc.sas.com
            default_domain = abc.sas.com
    }
```

```
[domain_realm]
     abc.sas.com = ABC.SAS.COM
            abc.sas.com = ABC.SAS.COM

[appdefaults]
    autologin = true
    forward = true
           forwardable= true
    encrypt = true
```

Substitute your hostname for the **default_keytab_name** command. Make sure that the value specified for the **default _tkt_enctypes** variable matches the value specified for **–crypto** option in the **ktpass** command that you used on the Active Controller Directory machine.

Verify Kerberos Authentication

A Ticket Granting Ticket (TGT) could expire or get lost from the cache. To ensure that a valid TGT is available in the system, use the kinit command. The kinit command obtains and caches the Kerberos ticket-granting tickets.

1. Bring up a command prompt window, and go to the Java directory where the kinit utility resides (for example**, C:\jdk1.5.0.19\bin** directory).
2. On Windows, run the kinit utility to make a Kerberos request. Substitute the name of the keytab filename, URL address and domain name:

   ```
   kinit -k -t C:\krb5.keytab\redwood2.host.keytab
   HTTP/redwood2.abc.sas.com@ABC.SAS.COM
   ```

   It is important that the following message displays at the end of the output:

   "New ticket is stored in cache file C:\Documents and settings…"

Modifying the login-config.xml File

The application server requires a security domain that it can use to authenticate against the KDC.

On Windows, the **login-config.xml** file is typically located in the ***JBOSS_HOME*/server/SASServer1/conf** directory.

The contents of the **login-config.xml** file should contain an application policy for the host security domain that is required by the SPNEGO support. The file should also specify a keytab that is required for the principal that represents JBoss. As a result, JBoss can authenticate against the Kerberos configuration. The SPNEGO log in module specified in this file must match the security-domain name specified in the **jboss-web.xml** file for the SAS Logon Manager application.

Your content should resemble the following example.

```
<application-policy name="host">
  <authentication>
    <login-module code="com.sun.security.auth.module.Krb5LoginModule" flag="required">
     <module-option name="storeKey">true</module-option>
     <module-option name="useKeyTab">true</module-option>
     <module-option name="principal">HTTP/testserver@KERBEROS.JBOSS.ORG</moduleoption>
     <module-option name="keyTab">/jboss_user/testserver.keytab</module-option>
     <module-option name="doNotPrompt">true</module-option>
     <module-option name="debug">true</module-option>
    </login-module>
  </authentication>
</application-policy>
<application-policy name="SASApplicationLogin">
  <authentication>
    <login-module code="org.jboss.security.negotiation.spnego.SPNEGOLoginModule"
                  flag="requisite">
      <module-option name="password-stacking">useFirstPass</module-option>
      <module-option name="serverSecurityDomain">host</module-option>
    </login-module>

    <login-module code="org.jboss.security.auth.spi.UsersRolesLoginModule"
                  flag="required">
      <module-option name="password-stacking">useFirstPass</module-option>
      <module-option name="usersProperties">props/spnego-users.properties</module-option>
      <module-option name="rolesProperties">props/spnego-roles.properties</module-option>
    </login-module>

    <login-module code="com.sas.services.security.login.jboss.JBossTrustedLoginModule"
                flag="required">
      <module-option name="host">metadata-server-host</module-option>
      <module-option name="port">8561</module-option>
      <module-option name="repository">Foundation</module-option>
       <module-option name="domain">web</module-option>
      <module-option name="trusteduser">sastrust@saspw</module-option>
      <module-option name="trustedpw">encoded-password</module-option>
    </login-module>
  </authentication>
</application-policy>
</authentication> </application-policy>
```

Modifying SAS Logon Manager

Edit the **web.xml** file and the **jboss-web.xml** files. Both files are located in the WEB-INF application directory. For instructions on extracting and editing these files, see "Configuring JBoss Application Server 4.2.0 for Web Authentication with SAS 9.2 Web Applications."

The file contents of the **web.xml** file and the **jboss-web.xml** file should resemble the following examples.

Example of the **web.xml** file with SPNEGO specified for the auth-method AND realm-name parameters.

```
<security-constraint>
  <web-resource-collection>
    <web-resource-name>All resources</web-resource-name>
    <url-pattern>/*</url-pattern>
    <http-method>GET</http-method>
    <http-method>POST</http-method>
  </web-resource-collection>

  <auth-constraint>
    <role-name>SASWebUser</role-name>
  </auth-constraint>
</security-constraint>
```

```
<login-config>
  <auth-method>SPNEGO</auth-method>
  <realm-name>SPNEGO</realm-name>
</login-config>

<security-role>
  <role-name>SASWebUser</role-name>
</security-role>
```

Example of the **jboss-web.xml** file with the security domain that JBoss uses when a user tries to log in to Logon Manager. The security domain specified in the **jboss-web.xml** file must match the security domain specified in the **login-config.xml** file. Add the line shown in bold:

```
<!-- File containing settings specific to the JBoss application server -->
<jboss-web>
  <context-root>SASLogon</context-root>
  <security-domain>java:/jaas/SASApplicationLogin</security-domain>
</jboss-web>
```

Configuring the Files for the SPNEGO User Properties and Role Properties

Create the **spnego-users.properties** and **spnego-roles.properties** files, and place them in the **`JBOSS_HOME/server/`*`SASServer1`*`/conf/props`** directory. The **spnego-users.properties** file should be an empty file. The **spnego-roles.properties** contains usernames and their role mappings. Here is an example:

```
username@domain.com=SASWebUser
```

Configuring the Client Browser to Use SPNEGO

Complete the following steps on the machine with the client browser application to ensure that your Microsoft Internet Explorer browser is enabled to perform SPNEGO authentication.

Configure Local Intranet Domains

1. In the Internet Explorer window, select **Tools ► Internet Options ► Security**.
2. Under Local Intranet, click **Sites**.
3. Verify that the checkboxes are selected for the following options:
   Include all local (Intranet) sites not listed in other zones
   Include all sites that bypass the proxy server
4. Add your domain name to the list of Web sites to ensure that Internet Explorer recognizes any site with your domain name as the intranet.

Configure Intranet Authentication

1. In the Internet Explorer window, select **Tools ► Internet Options ► Security**.
2. Under Local Intranet, click **Sites**.
3. On the **Security** tab, select **Local Intranet** and click **Custom Level**.
4. In the Security Settings – Local Intranet Zone, under **User Authentication**, select **Automatic Logon only in Intranet Zone** and click **OK**.

Verify the Proxy Settings

1. In the Internet Explorer window, select **Tools ► Internet Options ► Connections** .
2. Click **LAN Settings**.
3. Verify that the proxy server address and port number are correct.
4. Click **Advanced**.
5. In the Proxy Settings dialog box, ensure that all desired domain names are entered in the **Exceptions** field.
6. Click **OK** to close the Proxy Settings dialog box.

Specify Integrated Authentication for Internet Explorer

1. On the Internet Options window, click the **Advanced** tab and scroll to **Security settings**. Verify that the checkbox is selected for **Enable Integrated Windows Authentication**.
2. Click **OK**. Restart your Microsoft Internet Explorer to activate this configuration.

Verifying IWA

Log on to SAS Web applications to confirm that no prompt is presented for logon credentials, and that the applications load with the current Windows user logged into the application.

Do NOT test from a browser on the middle-tier machine itself (that is, the machine where the application server is installed). This will not work. Testing must be performed on a separate client machine within the Windows domain.

Troubleshooting SPNEGO Support

To troubleshoot SPNEGO support within JBoss, download and use the Negotiation Toolkit provided for JBoss.

Recommended Reading

Massachussetts Institute of Technology. *Kerberos: The Network Authentication Protocol*. Available at web.mit.edu/Kerberos.

RedHat Inc. 2009. *User Guide for JBoss Negotiation – A Guide for Administrators*. Available at issues.jboss.org/secure/attachment/12341757/Negotiation_User_Guide_(en-US).pdf?_sscc=t.

SAS Institute, Inc. 2009. *Configuring JBoss Application Server 4.2.0 for Web Authentication with SAS 9.2 Web Applications*. Available at support.sas.com/resources/thirdpartysupport/v92m2/appservers/ConfiguringJBossWebAuth.pdf.

SAS Institute, Inc. 2009. *SAS 9.2 Intelligence Platform: Security Administration Guide*. Cary, NC. SAS Institute, Inc. Available at

support.sas.com/documentation/cdl/en/bisecag/61133/PDF/default/bisecag.pdf.

support.sas.com

SAS is the world leader in providing software and services that enable customers to transform data from all areas of their business into intelligence. SAS solutions help organizations make better, more informed decisions and maximize customer, supplier, and organizational relationships. For more than 30 years, SAS has been giving customers around the world The Power to Know®. Visit us at **www.sas.com**.