Configuring SPNEGO based SSO with Websphere and Active Directory

Table of contents

1.Introduction

Websphere allows third party programs to authenticate a user using their own ways and can be configured to trust these programs. This kind of interaction is done using a mechanism named “Trust Association Interceptor” (or TAI). A TAI is simply a Java class which implements a particular interface to negotiate user authentication itself. Jaaslounge provides you a TAI that can speak with Active Directory to automatically authenticate a user who previously logged in on a windows workstation.

2.Compilation

The SPNEGO Trusted Association Interceptor provided consists in one java class : SpnegoTAI.java. To compile this class you need several libraries :

- jCifs-ext : version 0.9.4
- jCifs : version 1.1.11 , later versions must not be used as jCifs-ext does not work with.
- wssec.jar : found in directory “lib” of Websphere application server
- j2ee.jar : found in directory “lib” of Websphere application server

3.Installation and Configuration

1. General

First of all, ensure that your active directory is properly configured. Steps required to configure Active Directory are detailed in the document “AD_Spnego_Configuration”.

Once you have generated a jar containing the compiled TAI class. Copy it to the lib\ext directory of your Websphere installation. You then need to configure Websphere. Open and log in the admin console in your browser.

2. Writing Kerberos configuration

To make use of Kerberos, you need to write an configuration file. This file specifies settings like the Kerberos Realm, address of the server... Here is a sample of such a file, considering that our realm is named MY.DOMAIN.COM, and that our Active Directory server is named adserver.my.domain.com :

```
[libdefaults]
default_realm = MY.DOMAIN.COM
default_tkt_enctypes = rc4-hmac des-cbc-md5 des-cbc-crc des3-cbc-sha1
default_tgs_enctypes = rc4-hmac des-cbc-md5 des-cbc-crc des3-cbc-sha1

[realms]
 MY.DOMAIN.COM = {
  kdc = adserver.my.domain.com:88
  admin_server = adserver.my.domain.com
  default_domain = MY.DOMAIN.COM
 }

[domain_realm]
 .my.domain.com = MY.DOMAIN.COM
 MY.DOMAIN.COM = MY.DOMAIN.COM
```

3. Declaring Kerberos configuration file to Websphere

In the left navigation pane, expand “**Servers**” and select “**Application Servers**”. In the list, select the server you want to configure. In the “**Server Infrastructure**” section on the right side of the main panel, expand “**Processes and Java Management**” and choose “**Process Definition**”.

On the right side of the panel, under “**Additional Properties**”, select “**Java Virtual Machine**”. Then, in “**Generic JVM Arguments**”, add “-Djava.security.krb5.conf=*path_to_krb5_config*”. Where “*path_to_krb5_config*” points to your Kerberos configuration file. Save your changes.

4. Configuring Active Directory as the user registry

In the left navigation pane, expand “**Security**” and select “**Global Security**”. In the “**User Registries**” section on the right side of the main panel, select “**LDAP**” to specify a LDAP user registry (Active Directory is a LDAP registry secured by Kerberos).

In the “**General Properties**”, fill in the different options :

- For the **Server user ID**, specify an existing account name in Active Directory (for example : wasadmin@MY.DOMAIN.COM).
- For the **Server user password**, specify the real password of this account.
- Select “**Active Directory**” in the **Type** combo box.
- For the **Host**, specify the address of the Active Directory host (adserver.my.domain.com).
- The **Base distinguished name (DN)** entry specifies the root of the portion of LDAP directory that Websphere must process to search for users and groups (for example : dc=my,dc=domain,dc=com).
- The **Bind distinguished name (DN)** entry specifies the principal used by Websphere to connect to the LDAP directory. We will use the service principal name configured for our application server service in active directory (for example : HTTP/webserver.my.domain.com).

- The **Bind password** entry specifies the password used by Websphere to connect to the LDAP directory.

Others entries can be leaved with their default values.

Under “**Additional Properties**”, select “**Advanced Lightweight Directory Access Protocol (LDAP) user registry settings**”. We will change the entries **User Filter** and **User ID map** to search for Active Directory names (by default, the Active Directory settings makes Websphere search for NTLM names).

- In the **User Filter** entry, replace “sAMAccountName” by “userPrincipalName”
- In the **User ID map** entry, replace “user:sAMAccountName” by “user:userPrincipalName”

Apply and save your changes.

5. Configuring a JAAS Login Module for Kerberos Authentication

Return to the **Global Security** panel (In the left navigation pane, expand “**Security**” and select “**Global Security**”). On the right, under “**Authentication**”, expand “**JAAS Configuration**” and select “**Applications connections**”. We will create 2 Login Module aliases to be used by jCifs-ext : jcifs.spnego.initiate and jcifs.spnego.accept.

To create such an alias, click “**New**” on the list, give an alias name for your Login Module configuration for the entry “**Alias**” (jcifs.spnego.accept for example). Click “**Apply**” and select “**JAAS Connection Modules**” under “**Additional Properties**”.

Click “**New**” on the presented list. Under “**General Properties**” specify “*com.ibm.security.auth.module.Krb5LoginModule*” for the entry “**Module class name**”. You can fine tune the Login Module under the entry **Additional Properties**.

Apply and save your changes.

6. Configuring Security in Websphere

Return to the **Global Security** panel (In the left navigation pane, expand “**Security**” and select “**Global Security**”). On the right, under “**Authentication**”, expand “**Authentication mechanisms**” and select “**LTPA**”.

Specify and confirm a password, this password will be used to encrypt and decrypt LTPA keys for the cell node. Apply and save your settings.

If desired, select **Single Sign-On (SSO)** under “**Additional Properties**” to configure the application server SSO. This SSO is for authentication between applications on the application server, once you have successfully authenticated yourself on one of the running applications. This is not SSO using your windows user id.

Under “**Additional Properties**”, select **Trust Relationship** and check the check box “**Enable trust relationship**”. Apply and select “**Interceptors**” under “**Additional Properties**” on the right of the panel. Click “**New**” on top of the list.

In “**General Properties**”, specify the fully qualified name of the Trust Association Interceptor class in the entry **Interceptor Class Name**. To use jaaslounge SpnegoTAI, enter the value *org.jaaslounge.sso.websphere.spnego.SpnegoTAI.*

Once applied, select **Personalized Properties** under “**Additional Properties**”. This is were you will configure the jaaslounge's Trust Association Interceptor. You need to add 4 properties :

- *domainController* : specifies the address of the Active Directory domain controller
- *domainName* : specifies the name of the domain
- *servicePassword* : specifies the password to use to connect on Active Directory
- *servicePrincipalName* : specifies the principal name to use to connect on Active Directory

For example, we can set *domainController*=adserver.my.domain.com, *domainName*=my.domain.com, and *servicePrincipalName*=HTTP/webserver.my.domain.com.

Apply and save your changes.

7. Activating security settings

Return to the **Global Security** panel (In the left navigation pane, expand "**Security**" and select "**Global Security**"). Under "**General Settings**", check "**Enable Global Security**", uncheck "**Enforce java 2 security**". Select "**Lightweight Third Party Authentication (LTPA)**" in the combo box "**Active authentication mechanism**" and select "**LDAP**" in the combo box "**Active User Registry**".

Apply and save your changes.

4. Mapping Active Directory groups to application roles

Now that you have configured the security in Websphere, you need to map Active Directory groups names to application roles names. For example, you can add your Active Directory users in some groups and want to use this group as authorization for Websphere admin console access. Be aware that configuring access to Websphere admin console and to other applications is slightly different.

1. Configuring accesses to admin Console

In the left navigation pane, expand "**System Administration**", expand the sub entry "**Console parameters**" and select "**Console groups**". In the list, click "**Add**". In the **Group** zone, specify the full Active Directory name of the group and then select associated roles in the **Roles** list.

Apply and save your changes.

2. Configuring accesses to applications

In the left navigation pane, expand "**Applications**" and select "**Enterprise Applications**". Select the desired application. Under "**Additional Properties**", select "**Mapping users to roles**". Select the desired role, and click "**Search for groups**". Search for the wanted groups names, add them to the **Selected** list and apply your changes.