Using SPNEGO in Geronimo

Replacing default Realm in Geronimo Administering Security

Using the Simple and Protected GSS-API Negotiation Mechanism(SPNEGO) in Geronimo allows HTTP users to log in and authenticate only once in their desktop, then they can receive automatic authentication from the Geronimo server. Note that the feature is only supported in Geronimo 2.2.1 or later versions.

Prerequisite

Using the SPNEGO requires three distinct machines:

- A Microsoft Windows 2000 or Windows 2003 Server running the Active Directory Domain Controller and associated Kerberos Key Distribution Center(KDC)
- A domain member with internet browsers for example, a Microsoft Internet Explorer or Firefox browser
- A server Platform with Geronimo running

Note that the clock on clients, Microsoft Active Directory Domain Controller and Geronimo server must be synchronized to within five minutes, and they must be within the same domain.

Procedure

Setting up the Domain Controller Machine

1. Create a user account in the active directory. Make sure that the user you create is unique and not listed in Computers or domain controllers. This account will be eventually mapped to the Kerberoes service principal name(SPN).
2. Map the user account to the SPN with the command **setspn**. Typically, A SPN looks like *HTTP/<Fully_Qualified_Host_Name>*. Make sure that you do not have the same SPNs mapping to more than one Microsoft user account. If you map the same SPN to more than one user account, the web browser client can send a NT LAN Manager(NTLM) authentication request instead of SPNEGO token to Geronimo server. See Windows 2003 Technical Reference (setspn command) for more usages of the command.

   ```
   setspn -A HTTP/test.xyz.com testuser.
   ```

   Where
   - *testuser* is the user account created in step1
   - *HTTP/test.xyz.com* is the unique SPN mapped with *testuser*, *test.xyz.com* is the host name of Geronimo server.
3. Create the Kereros keytab file(`krb5.keytab`) with the command **ktpass** and make the file available to Geronimo server by copying it from the Domain Controller to the Geronimo server. See Windows 2003 Technical Reference (ktpass command) for more usages of the command.

   ```
   ktpass -out c:\winnt\krb5.keytab -princ HTTP/test.xyz.com@XYZ.COM -mapUser testuser -mapOp set -pass testuser123 -crypto RC4-HMAC-NT -pType KRB5_NT_PRINCIPAL
   ```

   where
   - *HTTP/test.xyz.com@XYZ.COM* is the concatenation of the user logon name, and the realm name which must be in uppercase.
   - *testuser* is the user account for mapping.
   - *testuser123* is the password of the user **testuser**.

Setting up the Client Application Machine

On client machines, the Web browsers are responsible for generating the SPNEGO token for user by the Geronimo server. Perform the following configuration for your browsers. Note that the resources on Geronimo server can only be accessible by the domain name of the Geronimo server, and the client machines must be the members of Domain.

Enable SPNEGO authentication in Microsoft Internet Explorer browser

1. In the Internet Explorer windows, click **Tools>Internet Options>Security** tab.
2. Select the **Local Intranet** icon and click **Sites**.
3. Make sure all check boxes are selected in the **Local Intranet** windows, then click **Advanced** button.

4. Add the URI name of the Geronimo server for example _http://test.xyz.com_ into the list Web sites so that the Single Sign-On (SSO) can be enabled, then click **OK** to complete this step and close the **Local intranet** window.
5. On the **Internet Options** windows, click the **Advanced** tab and go to **Security settings**. Make sure **Enable Integrated Windows Authentication (requires restart)** check box is selected, then click **OK** to close all windows.
6. Restart your Microsoft Internet Explorer to activate the configuration.

Enable SPNEGO authentication in Firefox

1. In the URL address bar of your Firefox browser, type **about:config** and press the Enter key.
2. In the following windows, type **network.nego** in the **Filters**.
3. Double click **network.negotiate-auth.trusted-uris** and add http://,https:// in the pop-up window, then click **OK** to close the window.
4. Double click **network.negotiate-auth.delegation-ruis** and add http://,https:// in the pop-up window, then click **OK** to close the window.
5. Restart your Firefox to activate the configuration.

Setting up the Geronimo server

1. Copy the Keroes keytab file `krb5.keytab` to one of directories of your Geronimo Server. The file was created during Setting up the Domain Controller Machine.
2. Create a basic Kerbeores configuration file named `krb5.ini` in order to use the SPNEGO for the server. The files should be stored on local server and with the following keys list defining the Kerberoes key distribution center(KDC) name and the realm setting for the SPNEGO authentication.

krb5.ini

```
[libdefaults]
  default_realm = XYZ.COM
  default_keytab_name = FILE:c:\winnt\krb5.keytab
  default_tkt_enctypes = rc4-hmac,des-cbc-md4,des-cbc-crc
  default_tgs_enctypes = rc4-hmac,des-cbc-md4,des-cbc-crc
  forwardable=true
[realms]
  XYZ.COM = {
      kdc = domaincontroller.xyz.com:88
      default_domain = xyz.com
  }
[domain_realm]
  xyz.com= XYZ.COM
  .xyz.com = XYZ.COM
```

3. Configure JVM properties with the following key pairs to make sure the JVM read the Kerberoes configurations successfully.

```
set JAVA_OPTS=-Djava.security.krb5.conf=C:\winnt\krb5.ini -Dcom.ibm.security.jgss.debug=all -Dcom.ibm.security.krb5.Krb5Debug=all -Djavax.security.auth.useSubjectCredsOnly=false
```

4. Create a system-scope realm for the Geronimo server as followed. The sample code is a combination of SPNEGO and .properties file realms in order that the authentication will fall back on .Properties realm once the SPNEGO authentication fails. You can remove the .properties file realm if unnecessary.

spnego_properties_realm.xml

```
<module xmlns="http://geronimo.apache.org/xml/ns/deployment-1.2">
    <environment>
        <moduleId>
            <groupId>console.realm</groupId>
            <artifactId>SpnegoTest</artifactId>
            <version>1.0</version>
            <type>car</type>
        </moduleId>
        <dependencies>
            <dependency>
                <groupId>org.apache.geronimo.framework</groupId>
                <artifactId>j2ee-security</artifactId>
                <type>car</type>
            </dependency>
        </dependencies>
    </environment>

    <!--
```

```
      The ConfigEntry and KerberosLoginModule GBeans are not needed on IBM JVM.
     -->

    <gbean name="ConfigEntry" class="org.apache.geronimo.security.jaas.DirectConfigurationEntry"
           xsi:type="dep:gbeanType" xmlns:dep="http://geronimo.apache.org/xml/ns/deployment-1.2"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <attribute name="applicationConfigName">com.sun.security.jgss.accept</attribute>
        <attribute name="controlFlag">REQUIRED</attribute>
        <reference name="Module">
            <name>KerberosLoginModule</name>
        </reference>
    </gbean>

    <gbean name="KerberosLoginModule" class="org.apache.geronimo.security.jaas.LoginModuleGBean"
           xsi:type="dep:gbeanType" xmlns:dep="http://geronimo.apache.org/xml/ns/deployment-1.2"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <attribute name="loginModuleClass">org.apache.geronimo.security.realm.providers.
KerberosLoginModule</attribute>
        <attribute name="loginDomainName">unspecified</attribute>
        <attribute name="options">
          krb5LoginModuleClass=com.sun.security.auth.module.Krb5LoginModule
          krb_debug=true
          krb_useKeyTab=true
          krb_storeKey=true
          krb_doNotPrompt=true
          krb_isInitiator=false
          krb_keyTab=c:/winnt/krb5.keytab
          krb_principal=HTTP/test.xyz.com@XYZ.COM
        </attribute>
    </gbean>

    <gbean name="SpnegoTest" class="org.apache.geronimo.security.realm.GenericSecurityRealm"
           xsi:type="dep:gbeanType" xmlns:dep="http://geronimo.apache.org/xml/ns/deployment-1.2"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        <attribute name="realmName">SpnegoTest</attribute>
        <reference name="ServerInfo">
            <name>ServerInfo</name>
        </reference>
        <xml-reference name="LoginModuleConfiguration">
            <log:login-config xmlns:log="http://geronimo.apache.org/xml/ns/loginconfig-2.0">
                <log:login-module control-flag="SUFFICIENT" wrap-principals="false">
                    <log:login-domain-name>SpnegoTest</log:login-domain-name>
                    <log:login-module-class>org.apache.geronimo.security.realm.providers.
SpnegoLoginModule</log:login-module-class>
                    <log:option name="targetName">HTTP/test.xyz.com</log:option>
                    <log:option name="ldapUrl">ldap://domaincontroller.xyz.com:389</log:option>
                    <log:option name="ldapLoginName">testuser</log:option>
                    <log:option name="ldapLoginPassword">testuser123</log:option>
                    <log:option name="searchBase">DC=xyz,DC=com</log:option>
                </log:login-module>
                <log:login-module control-flag="SUFFICIENT" wrap-principals="false">
                    <log:login-domain-name>demo-properties-realm</log:login-domain-name>
                    <log:login-module-class>org.apache.geronimo.security.realm.providers.
PropertiesFileLoginModule</log:login-module-class>
                    <log:option name="usersURI">var/security/demo_users.properties</log:option>
                    <log:option name="groupsURI">var/security/demo_groups.properties</log:option>
                </log:login-module>
            </log:login-config>
        </xml-reference>
    </gbean>
</module>
```

5. Configure the deployment plan of your application to make sure the SPNEGO realm is invoked properly. See the sample code below for reference.

geronimo-web.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<web:web-app xmlns:app="http://geronimo.apache.org/xml/ns/j2ee/application-2.0"    xmlns:client="
http://geronimo.apache.org/xml/ns/j2ee/application-client-2.0"
        xmlns:conn="http://geronimo.apache.org/xml/ns/j2ee/connector-1.2" xmlns:dep="http://geronimo.
apache.org/xml/ns/deployment-1.2"
        xmlns:ejb="http://openejb.apache.org/xml/ns/openejb-jar-2.2" xmlns:name="http://geronimo.apache.
org/xml/ns/naming-1.2"
        xmlns:pers="http://java.sun.com/xml/ns/persistence" xmlns:pkgen="http://openejb.apache.org/xml/ns
/pkgen-2.1"
        xmlns:sec="http://geronimo.apache.org/xml/ns/security-2.0" xmlns:web="http://geronimo.apache.org
/xml/ns/j2ee/web-2.0.1">
    <dep:environment>
        <dep:moduleId>
            <dep:groupId>com.mycompany.samples</dep:groupId>
            <dep:artifactId>security-demo</dep:artifactId>
            <dep:version>2.2.1</dep:version>
            <dep:type>war</dep:type>
        </dep:moduleId>
        <dep:dependencies/>
        <dep:hidden-classes>
            <dep:filter>
                org.apache.geronimo.security.realm.providers.SpnegoLoginModule
            </dep:filter>
        </dep:hidden-classes>
        <dep:non-overridable-classes/>
    </dep:environment>
    <web:context-root>/demo</web:context-root>
    <web:security-realm-name>SpnegoTest</web:security-realm-name>
    <sec:security>
        <sec:role-mappings>
            <sec:role role-name="content-administrator">
                <sec:principal class="org.apache.geronimo.security.realm.providers.
GeronimoGroupPrincipal" name="Domain Admins"/>
                <sec:principal class="org.apache.geronimo.security.realm.providers.
GeronimoUserPrincipal" name="testuser@TEST.XYZ.COM"/>
            </sec:role>
            <sec:role role-name="Guest-administrator">
           <sec:principal class="org.apache.geronimo.security.realm.providers.GeronimoGroupPrincipal"
name="Domain Admins"/>
             </sec:role>
        </sec:role-mappings>
    </sec:security>
</web:web-app>
```

6. Configure the deployment descriptor to make sure your application uses SPNEGO authentication and the respective realm provider that Geronimo server supports.

excerpt of web.xml

```
<?xml version="1.0" encoding="ISO-8859-1"?>
...
   <login-config>
      <auth-method>SPNEGO</auth-method>
      <realm-name>SpnegoTest</realm-name>
      ...
   </login-config>
```

Few very important points to note

- Make sure that you use Basic as the authentication mechanism in your web application if you want to configure Spnego with geronimo.
- The realm provided is a combination of 2 login modules which can be easily created through geronimo administrative console.

- While you are creating a security realm for Spnego loginmodule you need to just specify one option that will be of the form "targetName=HTTP /<fully_qualified_host_name>". Have a look at the sample realm. This will give you an idea of the option to be used.
- Make sure you choose sufficient as the control-flag while creating the 2 login modules.
- Make sure you map only one user to SPN as defined in #2 of "Setting up the Active Directory Domain Controller".