Configuring login modules

Where's the login.conf file?

Because of some limitations in the default configuration implementation, Geronimo replaces login.conf entirely with security realms that are configured by GenericSecurityRealm GBeans. You can deploy the security realm that you need with your application and dynamically add the login module classes to the server as needed. You can also distinguish between the same principals when they are created by different login modules or security realms.

So how do I configure a security realm?

A typical security realm GBean looks like this:

```
<gbean name="test-realm" class="org.apache.geronimo.security.realm.GenericSecurityRealm">
    <attribute name="realmName">test-realm</attribute>
    <xml-reference name="LoginModuleConfiguration">
        <lc:login-config xmlns:lc="http://geronimo.apache.org/xml/ns/loginconfig-${geronimoSchemaVersion}">
            <lc:login-module control-flag="REQUIRED" wrap-principals="false">
                <lc:login-domain-name>test-realm</lc:login-domain-name>
                <lc:login-module-class>org.apache.geronimo.itest.TestLoginModule</lc:login-module-class>
                <lc:option name="users">foo,bar</lc:option>
            </lc:login-module>
        </lc:login-config>
    </xml-reference>
    <reference name="ServerInfo">
        <name>ServerInfo</name>
    </reference>
</gbean>
```

Most of the innards here are similar to the login.conf file, converted to xml. You can include as many login-module elements as you need. The control flag is specified as an attribute of this element. Each login module needs a login-domain-name that is unique within the security realm. You can include as many options as you need. Geronimo will supply these options for all login modules so that the login modules can use them without declaring them in the configuration:

- org.apache.geronimo.security.realm.GenericSecurityRealm.KERNEL - the Geronimo kernel
- org.apache.geronimo.security.realm.GenericSecurityRealm.SERVERINFO - the ServerInfo object that lets you find stuff in the Geronimo server file layout
- org.apache.geronimo.security.realm.GenericSecurityRealm.CLASSLOADER - the classloader of the plugin that defines the security realm. Note that this might be different from the classloader of a plugin that is using the security realm.

If you specify wrap-principals as false, your login module will work as usual and only its principals will get into the Subject. However if you specify wrap-principals as true, Geronimo will also add principals that wrap your principals and include the login-domain-name and realm-name of the login module and security realm that created the principal. This enables your role-principal mapping to distinguish between the "same" principal that comes from different sources. For instance, if you had two LDAP servers where the groups had the same names but the meaning was different (perhaps users from different departments), you can wrap the principals yet still distinguish the same group based on the different realms. However, in order to distinguish principals in this way, we supply each login module with its own empty Subject object. Therefore, a later login module cannot access the principals added to a Subject by an earlier login module. If you need to share information between login modules and also wrap principals, you must use the shared state map and not the Subject.