Principal Providers

Fedora Principal Providers allow a Fedora repository to pull in user security and role designations from other sources (e.g. LDAP). Providers are consulted after the initial container authentication but before finer-grained authentication (such as role resolution) is applied.

The repository configuration file (repository.json) contains the class name of an authentication provider (under "providers") as well as the roles to be used when starting the provider module. By default, the org.fcrepo.auth.common.BypassSecurityServletAuthenticationProvider exists in the configuration file, as it doesn't rely on an external PrincipalProvider and offers the simplest authentication model (the module always gives access privileges to the session).

Different derivatives of the PrincipalProvider class can be initialized differently, either through the repository.json file, other credential files, from information sent via HTTP header or by connecting to external information sources such as LDAP.

PrincipalProvider

The interface that custom providers must implement. Several providers exist in the codebase.

A principal provider must be configured in repo.xml. The following examples shows configuration for a PrincipalProvider class ContainerRolesPrincipalProvider.

repo.xml

```
<bean name="containerRolesPrincipalProvider" class="org.fcrepo.auth.common.ContainerRolesPrincipalProvider"
          p:roleNames="my-new-tomcat-role"/>
<bean name="authenticationProvider" class="org.fcrepo.auth.common.ServletContainerAuthenticationProvider"
          p:fad-ref="fad" p:principalProviders-ref="containerRolesPrincipalProvider"/>
```

Container Roles Principal Provider

ContainerRolesPrincipalProvider is a PrincpalProivder that obtains its set of principals from web.xml.

New roles must be specified in web.xml as shown below. The default role is fedoraUser.

web.xml

```
<auth-constraint>
  <role-name>fedoraUser</role-name>
  <role-name>fedoraAdmin</role-name>
  <role-name>my-new-tomcat-role</role-name>
</auth-constraint>
```

Please refer to the servlet container authentication document for further configuration details.

HTTP Header Principal Provider

HttpHeaderPrincipalProvider is a Principal Provider that obtains its initial set of principals from HTTP header requests.

repo.xml

```
    <!-- Optional PrincipalProvider that will inspect the request header, "some-header", for user role values --
>
    <bean name="headerProvider" class="org.fcrepo.auth.common.HttpHeaderPrincipalProvider">
        <property name="headerName" value="some-header"/>
        <property name="separator" value=","/>
    </bean>
        <bean name="authenticationProvider" class="org.fcrepo.auth.common.
ServletContainerAuthenticationProvider"
          p:fad-ref="fad" p:principalProviders-ref="headerProvider"/>
```

Delegate Header Principal Provider

DelegateHeaderPrincipalProvider is a Principal Provider that uses the `On-Behalf-Of` HTTP header to switch the user principal to the principal given in the header. This switch is only performed if the authenticated user has the fedoraAdmin container role.

repo.xml

```
<bean name="delegatedPrincipalProvider" class="org.fcrepo.auth.common.DelegateHeaderPrincipalProvider"/>
<bean name="authenticationProvider" class="org.fcrepo.auth.common.ServletContainerAuthenticationProvider"
          p:fad-ref="fad" p:principalProviders-ref="delegatedPrincipalProvider"/>
```

Implementation Details

The Fedora class org.fcrepo.auth.common.ServletContainerAuthenticationProvider contains a list of PrincipalProvider derivative instances that are called for every authentication query. The union of the authentication traits of the PrincipalProvider instances will be assigned to the FEDORA_ALL_PRINCIPALS session attribute. In the case that the user is has the fedoraAdmin role, a FedoraAdminSecurityContext is provided as the users SecurityContext. If the user does not have the fedoraAdmin role, an ExecutionContext is provided as the users SecurityContext.