Table of Contents

Public Services

The REST API of JVx offers a great solution for generic services accessible via REST. However, the standard access is protected by BASIC authentication, and sometimes it's important to offer public services without authentication. This isn't possible with standard JVx authentication implementations. Our application framework enables you to do really cool things with JVx' REST API.

To enable public REST services, first configure your REST zone to use the forwarding authentication mode. Simply add the following to your deployment descriptor (web.xml):

```
<init-param>
  <!-- Authentication type -->
  <param-name>authtype</param-name>
  <param-value>forward</param-value>
</init-param>
```

The full definition:

web.xml

```
<!-- Restlet adapter -->
<servlet>
  <servlet-name>RestletServlet</servlet-name>
  <servlet-class>org.restlet.ext.servlet.ServerServlet</servlet-class>

  <init-param>
    <!-- Application class name -->
    <param-name>org.restlet.application</param-name>
    <param-value>com.sibvisions.rad.server.http.rest.RESTAdapter</param-value>
  </init-param>

  <init-param>
    <!-- Authentication type -->
    <param-name>authtype</param-name>
    <param-value>forward</param-value>
  </init-param>
</servlet>

<servlet-mapping>
  <servlet-name>RestletServlet</servlet-name>
  <url-pattern>/services/rest/*</url-pattern>
</servlet-mapping>
```

The authentication type defines that the configured SecurityManager will be used for authentication without pre-authentication with BASIC authentication. Usually, the BASIC authentication will be done before using the SecurityManager (e.g., authtype set to basic). The forward mode can be used to implement Single-Sign-On or no-authentication at all.

In our case, we performa an anonymous authentication with a pre-configured user. This enables you to configure the user with roles and offer public REST services without changing the application.

To enable anonymous authentication, use the AnonymousDBSecurityManager in your config.xml:

```
<securitymanager>
<class>com.sibvisions.apps.server.security.AnonymousDBSecurityManager</class
>
<accesscontroller>com.sibvisions.apps.server.object.DBWorkScreenAccess</acce
sscontroller>
  ...
</securitymanager>
```

To finish the configuration, set one user in the USERS database table as anonymous user. Simply set the column ANONYMOUS to 'Y'. That's all you need.

Now it's possible to test you REST services, e.g.

```
http://localhost:8080/webapp/services/rest/myapp/Session/action/getInternalN
ame?client.login.anonymous=true
```

In the above URL, the application is available in the context **webapp**. The application name is **myapp** and the **Session** LCO contains the method:

```
public String getInternalName()
{
    return "Session";
}
```

In our current configuration, the anonymous user will use the same LCO (Session) like the authenticated user. This might be good but can be a risk if the anonymous user shouldn't be able to use the same services. To change the LCO, it's possible to configure a custom LCO, via config.xml:

```
<lifecycle>
  <mastersession>com.sibvisions.apps.myapp.Session</mastersession>
  <application>com.sibvisions.apps.myapp.Application</application>

  <anonymoussession>com.sibvisions.apps.myapp.Anonymous</anonymoussession>
</lifecycle>
```

And the LCO code:

Anonymous.java

```
@StrictIsolation
@Replacement(name = "Public")
public class Anonymous extends GenericBean
{
  public String getInternalName()
  {
    return "Anonymous";
    }
}
```

This class doesn't extend the Session LCO, and it's an Isolated LCO. Additionally, the class name got a Replacement with the name Public.

With this configuration, our REST call will be changed to:

```
http://localhost:8080/webapp/services/rest/myapp/Public/action/getInternalNa
me?client.login.anonymous=true
```

The LCO name is now **Public**. The call with the original name, **Anonymous**, is also possible

```
http://localhost:8080/webapp/services/rest/myapp/Anonymous/action/getInterna
lName?client.login.anonymous=true
```

From:
http://doc.sibvisions.com/ - **Documentation**

Permanent link:
http://doc.sibvisions.com/applications/rest_extensions

Last update: **2020/07/08 12:55**