Java EE WildFly Setup Guide

Java EE WildFly Setup Guide: JMS Topics, Queues, and Elytron Security

Part 1: Creating JMS Queues and Topics

Create a JMS Queue

```
/subsystem=messaging-activemq/server=default/jms-
queue=ActivityLogQueue:add(

entries=["java:/jms/queue/ActivityLogQueue","java:jboss/exported/jms/qu
eue/ActivityLogQueue"],
  durable=true
)
```

Create a JMS Topic

```
/subsystem=messaging-activemq/server=default/jms-
topic=NotificationTopic:add(

entries=["java:/jms/topic/NotificationTopic","java:jboss/exported/jms/t
opic/NotificationTopic"]
)
```

Verify the Resources

```
/subsystem=messaging-activemq/server=default/jms-
queue=ActivityLogQueue:read-resource
/subsystem=messaging-activemq/server=default/jms-
topic=NotificationTopic:read-resource
```

Delete JMS Resources

```
/subsystem=messaging-activemq/server=default/jms-
queue=ActivityLogQueue:remove
/subsystem=messaging-activemq/server=default/jms-
topic=NotificationTopic:remove
```

Part 2: JAAS User and Role Configuration

application-users.properties

```
admin=admin123
bob=bob123
alice=alice123
```

application-roles.properties

```
admin=Admin
bob=User
alice=User
```

Add Users via Script

```
./add-user.sh
```

Follow the prompts:

```
What type of user do you wish to add?
 a) Management User (mgmt-users.properties)
 b) Application User (application-users.properties)
```

Choose: **b)**

Part 3: Elytron Security Setup (Replacing JAAS)

1. Define Properties Realm

```
/subsystem=elytron/properties-realm=AppRealm:add(
  users-properties={path=application-users.properties, relative-
to=jboss.server.config.dir},
  groups-properties={path=application-roles.properties, relative-
to=jboss.server.config.dir}
)
```

2. Add a Role Decoder

```
/subsystem=elytron/simple-role-decoder=groups:add(attribute=groups)
```

3. Create the Security Domain

```
/subsystem=elytron/security-domain=AppDomain:add(
  realms=[{realm=AppRealm, role-decoder=groups}],
  default-realm=AppRealm,
  permission-mapper=default-permission-mapper
)
```

4. Configure HTTP Authentication Factory

```
/subsystem=elytron/http-authentication-factory=AppHttpAuth:add(
  security-domain=AppDomain,
  http-server-mechanism-factory=global,
  mechanism-configurations=[{mechanism-name=BASIC}]
)
```

5. Link Security Domain to Web App

```
/subsystem=undertow/application-security-domain=AppWebSecurity:add(
  http-authentication-factory=AppHttpAuth
)
```

Part 4: Secure Web App (`web.xml`)

```
<?xml version="1.0" encoding="UTF-8"?>
<web-app xmlns="http://xmlns.jcp.org/xml/ns/javaee"
         xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
         xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd"
         version="4.0">

  <display-name>Minisocial</display-name>
```

```
  <!-- Welcome Files -->
  <welcome-file-list>
    <welcome-file>index.html</welcome-file>
    <welcome-file>index.jsp</welcome-file>
    <welcome-file>index.htm</welcome-file>
    <welcome-file>default.html</welcome-file>
    <welcome-file>default.jsp</welcome-file>
    <welcome-file>default.htm</welcome-file>
  </welcome-file-list>

  <!-- Secure Admin Endpoints -->
  <security-constraint>
    <web-resource-collection>
      <web-resource-name>Admin Area</web-resource-name>
      <url-pattern>/admin/*</url-pattern>
      <http-method>GET</http-method>
      <http-method>POST</http-method>
      <http-method>PUT</http-method>
      <http-method>DELETE</http-method>
    </web-resource-collection>
    <auth-constraint>
      <role-name>Admin</role-name>
    </auth-constraint>
  </security-constraint>

  <!-- Optional: Secure User Endpoints -->
  <security-constraint>
    <web-resource-collection>
      <web-resource-name>User Area</web-resource-name>
      <url-pattern>/users/*</url-pattern>
      <http-method>GET</http-method>
      <http-method>POST</http-method>
      <http-method>PUT</http-method>
    </web-resource-collection>
    <auth-constraint>
      <role-name>User</role-name>
      <role-name>Admin</role-name>
    </auth-constraint>
  </security-constraint>

  <!-- Authentication Config -->
  <login-config>
    <auth-method>BASIC</auth-method>
    <realm-name>AppRealm</realm-name>
  </login-config>

  <!-- Declared Roles -->
  <security-role>
    <role-name>Admin</role-name>
  </security-role>

  <security-role>
    <role-name>User</role-name>
  </security-role>

</web-app>
```

With this setup, you've:

- Enabled secure role-based access via Elytron
- Created durable JMS queues and topics
- Configured `web.xml` to enforce authentication and restrict access by role

You’re now ready to deploy a fully secure and message-enabled Java EE application on WildFly!