Java Enterprise Edition Security Explained - Lab

CTU Prague, Nov 2, 2018

Peter Škopek, pskopek@redhat.com, twitter: @pskopek

Abstract

This lab will let you play with demo applications and Keycloak server and its setup.

Downloads

Get Keycloak Standalone server distribution 3.3.0.Final

Get Keycloak Client Adapter for WildFly 14

> Do not use the one from standard download page (not working at the moment).

WildFly 14 Server

> https://downloads.jboss.org/keycloak/4.5.0.Final/keycloak-4.5.0.Final.zip

Lab git repository

> https://github.com/qa/pv243-a4m36jee-2018-security-seminar

Setup

Install and setup Keycloak server as instructed in lecture slides.

Install WildFly Server and setup Keycloak Client Adapter (lecture slides).

Useful tips

Start Keycloak

```
./bin/standalone.sh -Djboss.socket.binding.port-offset=100
```

Start service provider (WildFly Server)

```
./bin/standalone.sh
```

Deploy/redeploy web application

```
mvn clean wildfly:deploy
```

Task 1: Secure access to servlet using annotations

Start branch: “task-01” | Solution branch: “task-02”

- Secure SecuredServlet using annotations and test with our Keycloak setup
- Only user with role “gooduser” can have access to the servlet using all HTTP methods

Task 2: Secure static content of the web application

Start branch: “task-02” | Solution branch: “task-03”

- Secure static content of the web-application at /static/secured/. All pages there must be readable for “superuser” only.
- All pages located at /static/ should be readable by any authenticated user.
- SecuredServlet from previous task has to stay secured as it was.
- **Hint:** For details see Java Servlet 3.1 specification.

Task 3: Override security annotations using deployment descriptor

Start branch: “task-03” | Solution branch: “task-03-solution”

- Modify security constraint attached to the SecuredServlet so that only members of “superuser” group can run it.
- Act as application assembler, therefore you are not allowed to change code of SecuredServlet.java.
- **Hint:** the hit is already show at title of this task. You have to define servlet in web.xml.

Task 4: Identity Prop. & Programmatic Security

Start branch: “task-04” | Solution branch: “task-04-solution”

- At the beginning of doGet SecuredServlet method display following information:
  - remote user name
  - user principal
  - information if the user is in “superuser” role
- Create new method in TestBean which displays following information:
  - user principal
  - information if the user is in “gooduser” role
- Call the method at the end of SecuredServlet doGet method.
- **Hint:** Use @Resource annotation and HttpServlerRequest.