Access control
Web.xml part for access control

```xml
<security-constraint>
    <web-resource-collection> <url-pattern> /protected/* </url-pattern> </web-resource-collection>
    <auth-constraint> <role-name>role1</role-name> </auth-constraint>
</security-constraint>

<login-config>
    <auth-method>FORM</auth-method>
    <form-login-config>
        <form-login-page>/login_form.html</form-login-page>
        <form-error-page>/login_error.html</form-error-page>
    </form-login-config>
</login-config>
```
Auth-method

- BASIC
  - pop up a dialog box

- FORM:
  - Must contain a `j_username` field for the username and a `j_password` field for the password. The form must submit to `j_security_check`. 
Auth-method (cont.)

- **DIGEST**
  - Similar to BASIC
  - Username and password are encrypted into a message digest value

- **CLIENT-CERT**
  - Require to provide a digital certificate for authentication
  - Most secure, but costly
Use “role” after login

- Restricted access to Actions
  - `<action path="……" type= …… name=…… scope="request" roles="administrator">`</action>
- In JSP
  - `<logic:present role="administrator"/>`……
    `<logic:present>`
- Two methods of HTTP request
  - `getUserPrincipal()` & `isUserInRole()`
Limitation

- Accessing protected pages is the ONLY mechanism to trigger the login page
  - If the user goes directly to the login page ……

- Alternative: application-managed security (vs. container-managed security)
logout

- BASIC: have to exit browser to kill the session

- FORM
  - `Session.invalidate();`