Spring Security

Spring Security

- Spring Security is a powerful, flexible security solution for enterprise software, with a particular emphasis on applications that use Spring

What it offers?

- Provides declarative security for Spring-based applications
- Takes full advantage of dependency injection (DI) and aspect-oriented techniques based on the Spring Framework
- Major Features
  - ***Authentication***
  - ***Web URL authorization***
  - ***Method invocation authorization***
  - WS-Security (via Spring Web Services)
  - Flow Authorization (via Spring Web Flow)
  - Human user detection (Captcha)

Authentication & Authorization

- **Authentication** - process of estabilishing a principal (usually a user which can perform an action in application)
- **Authorization** - process of deciding wheather a principal is allowed to perform an action
- Authentication process establish identity of the principal, which is used for authorization decision

Authentication Models

- Spring Security supports various authentication models.
- Spring Security Models
  - HTTP Basic
  - HTTP Digest
  - HTTP X.509 Certificates
  - LDAP
  - Form-based
  - OpenId
  - And many more
- Spring provides implementation of many of these models.

Security Interceptor

- A latch that protects secured resources,
- To get past the latch, users typically provide their credentials
- Implementation depends on resource being secured
  - URLs - Servlet Filter
  - Methods - Aspects

Configuring Web Security

- Configure the **DelegatingFilterProxy** ***filter*** in the web.xml

```
<filter>
        <filter-name>springSecurityFilterChain</filter-name>
        <filter-class>
        org.springframework.web.filter.DelegatingFilterProxy
        </filter-class>
</filter>
<filter-mapping>
        <filter-name>springSecurityFilterChain</filter-name>
        <url-pattern>/*</url-pattern>
</filter-mapping>
```

DelegatingFilterProxy

- This provides a hook into the Spring Security web infrastructure.
- **DelegatingFilterProxy** is a Spring Framework class which delegates to a filter implementation which is defined as a Spring bean in your application context

Configuring Web Security

- Enable web security in the spring-security.xml
  - *<http auto-config='true'>*
    - *<intercept-url pattern="/**" access="ROLE_USER" />*
  - *</http>*
- The <http> element is the parent for all web-related namespace functionality
- The <intercept-url> element defines a pattern which is matched against the URLs of incoming requests.
- auto-config included <form-login /> <http-basic /> <logout />.

Configuring Web Security

- Define the authentication model
  - <authentication-manager>
    - <authentication-provider>
      - <user-service></user-service>
    - </authentication-provider>
  - </authentication-manager>