Spring Security

Intro

About Spring Security

- Began in 2003 as "The Acegi Security System for Spring"
- Spring subproject 1.0.0 released in 2006
- Rebranded as Spring Security in 2007
- Current version 4.1.0 (ongoing)

Introduction

- Comprehensive security services for Java EE-based enterprise software applications
- Portability between server environments
- Authentication & authorization
- HTTP BASIC, HTTP Digest, HTTP X.509, LDAP, Form-based, OpenID, JA-SIG CAS, Automatic "remember-me", Anonymous, Run-as, JAAS, JEE container, Kerberos, JOSSO, OpenNMS, AppFuse, AndroMDA, Mule ESB, Direct Web Request, Grails, Tapestry, Jtrac, Jasypt, Roller, Elastic Path, Atlassian Crowd

Spring Security – basic configuration XML

https://github.com/zbynekvavros/spring-security-intro.git

```xml
<web-app version="3.0" xmlns="http://java.sun.com/xml/ns/javaee"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    xsi:schemaLocation="http://java.sun.com/xml/ns/javaee
                        http://java.sun.com/xml/ns/javaee/web-app_3_0.xsd">

    <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>
</web-app>
```

```xml
<http>
    <intercept-url pattern="/**" access="hasRole('USER')" />
    <form-login />
    <logout />
</http>

<authentication-manager>
    <authentication-provider>
        <user-service>
            <user name="user" password="password" authorities="USER" />
        </user-service>
    </authentication-provider>
</authentication-manager>
```

Spring Security – basic configuration JavaConfig

```java
@EnableWebSecurity
public class SecurityConfig extends WebSecurityConfigurerAdapter {

    @Override
    protected void configure(HttpSecurity http) throws Exception {
        http.authorizeRequests().antMatchers("/**").hasRole("USER").and().formLogin().and().logout();
    }

    @Override
    protected void configure(AuthenticationManagerBuilder auth) throws Exception {
        auth.inMemoryAuthentication().withUser("user").password("password").roles("USER");
    }
}
```

Spring Security Filter Ordering

- http://docs.spring.io/spring-security/site/docs/4.1.0.
-

Workflow

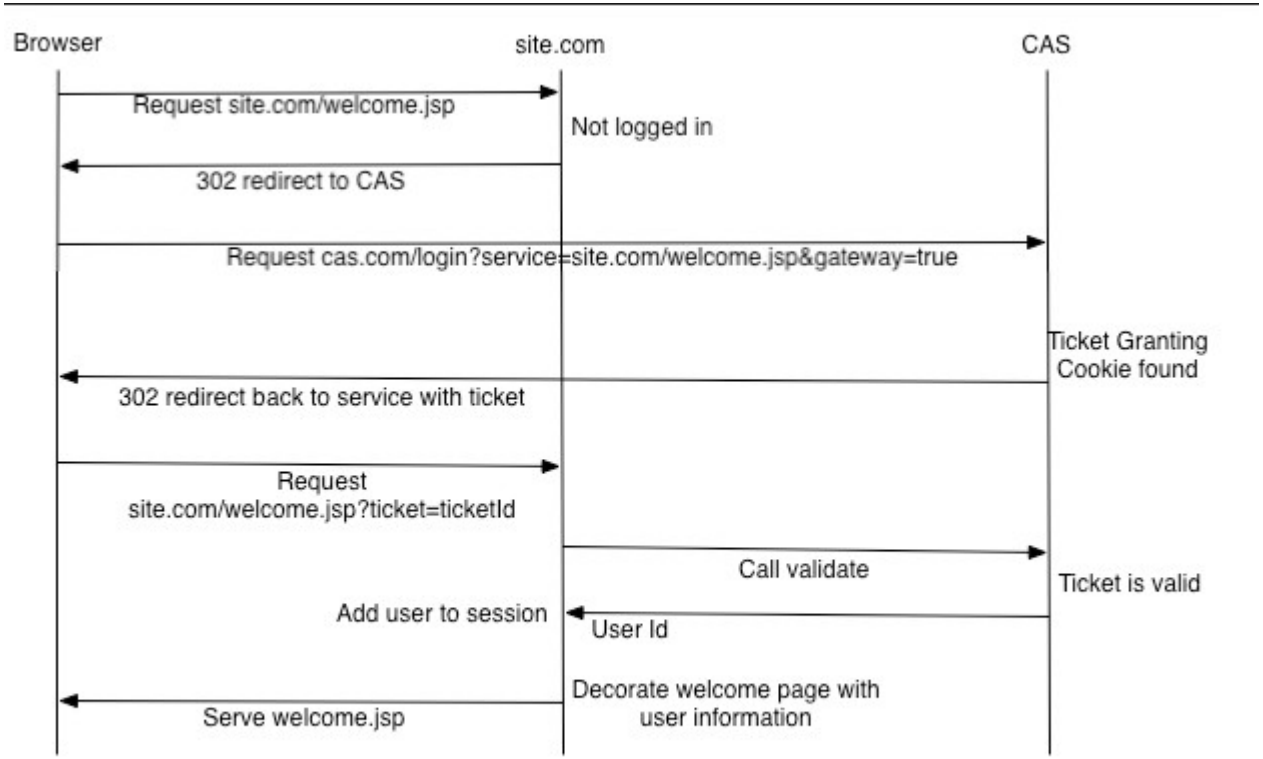
- http://docs.spring.io/spring-security/site/docs/4.1.0.
- FilterSecurityInterceptor → AccessDecisionManager
-
- Authentication manager configured manually
-
- security=”none”

Additional log in methods

- Custom form
- HTTP Basic
- HTTP Basic for REST

Additional authentication methods

- JDBC
  - http://docs.spring.io/spring-security/site/docs/4.1.0.REL
- LDAP
- CAS (Jasig)



Method security

- <global-method-security pre-post-annotations="enabled" />
- @PreAuthorize(expression) on interface

Expressions

- <http use-expressions="true">
- http://docs.spring.io/spring-security/site/docs/3.0.x/r
-

Additional features

- Remember Me
- CSFR - Cross-site request forgery
- HTTP/HTTPS Channel Security (pattern/redirect)
- Session management (AJAX !)
- Concurrent Session Control
- Session Fixation Attack Protection
- Custom filter !