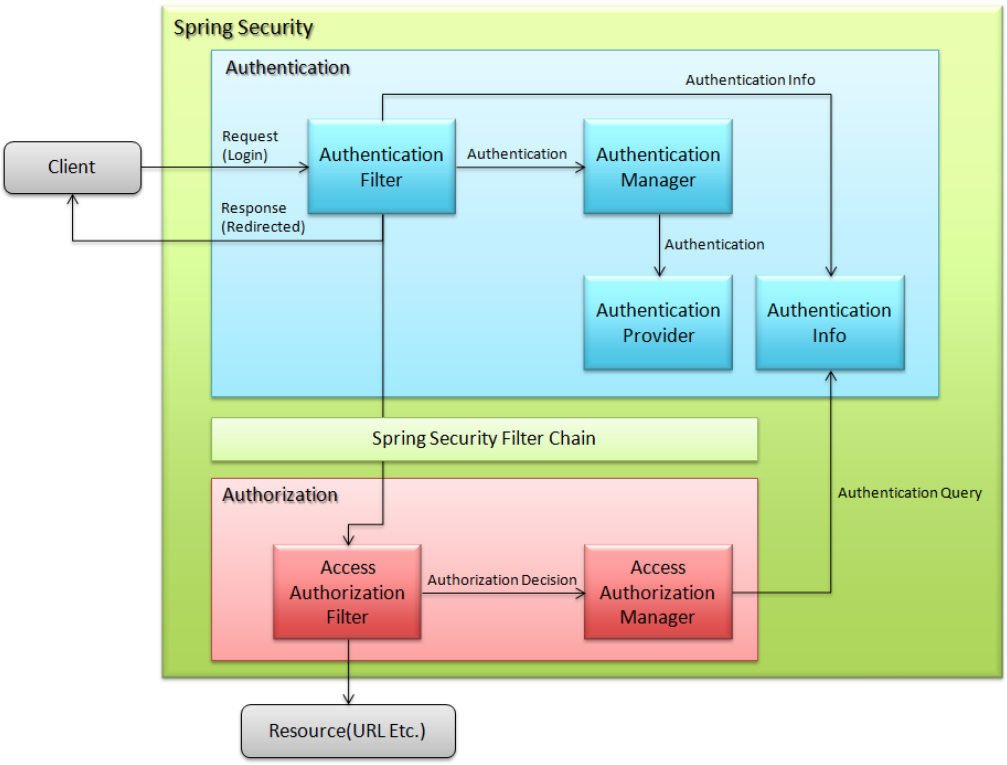
6.1. Spring Security Overview

Table of contents

6.1.1. Overview ¶

Spring Security and is responsible for the security of the application "authentication", and two of the "Authorization"
It is provided as a main function.
And authentication functions, to combat unauthorized access by spoofing is a function that identifies a user.
The authorized function, authenticated (logged in), depending on the permissions of the user,
This is a function controls access to the resources of the system.
Also has a function of imparting an HTTP header.

The schematic diagram of Spring Security, I show below.



Picture - Spring Security Overview

Spring Security authentication, continuing for many layers of process of approval
I have realized a collection of ServletFilter.
In addition, and password hash function, also such approval tag library of JSP has to offer.

6.1.1.1. Authentication

Authentication is the act of confirming validity, when connecting to a network or server
By using a combination of user name and password, and whether there is the right to use user,
it is that the person to determine whether a user's own use.
How they are used in Spring Security, the *authentication* see.

6.1.1.2. Password hash

The clear-text password, the calculated hash value using a hash function is to replace the original password.
How they are used in Spring Security, the *password hashed* see.

6.1.1.3. Approval

Authorization is, when the authenticated user is trying to access the resource,
And is to check that the user in the access control process is allowed to use the resource.
How they are used in Spring Security is *authorized* see.

6.1.2. How to use

To use Spring Security, it is necessary to define the following settings.

6.1.2.1. Pom.xml configuration of

When using Spring Security, the following dependency, it is necessary to add to the pom.xml.

```
<Dependency>
    <groupId> Org.Terasoluna.Gfw </ groupId>
    <artifactId> terasoluna-gfw-Security-Core </ artifactId>   <-! (1) ->
</ dependency>

<Dependency>
    <groupId> Org.Terasoluna.Gfw </ groupId>
    <artifactId> terasoluna-gfw-Security-web </ artifactId>   <-! (2) ->
</ dependency>
```

No.	Description
(1)	terasoluna-gfw-security-core, because that does not depend on the web, if you want to use from the project of the domain layer, only to be added to the dependency terasoluna-gfw-security-core.
(2)	terasoluan-gfw-web to provide the functions related to the web. because it is also dependent on the terasoluna-gfw-security-core, Web project, adding only the dependency terasoluna-gfw-security-web.

6.1.2.2. Web.xml set of

```
<Context-param>
  <param-name> ContextConfigLocation </ param-name>
  <param-value>   <-! (1) ->
      classpath *: META-INF / spring / applicationContext.xml
      classpath *: META-INF / spring / spring-security.xml
```

```
    </ param-value>
  </ context-param>
  <listener>
    <listener-class>
      org.springframework.web.context.ContextLoaderListener
    </ Listener-class>
  </ listener>
  <filter>
    <filter-name> SpringSecurityFilterChain </ filter-name>   <-! (2) ->
    <filter-class> Org.Springframework.Web.Filter.DelegatingFilterProxy < / filter-class>
<-! (3) ->
  </ filter>
  <filter-mapping>
    <filter-name> SpringSecurityFilterChain </ filter-name>
    <url-pattern> / * </ url-pattern>   < ! - (4) ->
  </ filter-mapping>
```

No.	Description
(1)	The contextConfigLocation, in addition to the applicationContext.xml, I want to add the Spring Security configuration file to the class path. In this guideline, I will be a "spring-security.xml".
(2)	The filter-name, Bean name used in the interior of the Spring Security, and be defined in the "springSecurityFilterChain".
(3)	To enable a variety of functions, Spring Security filter settings.
(4)	I want to enable the setting for all of the requests.

6.1.2.3. Spring-security.xml set of

in the path specified in web.xml, to place the spring-security.xml.
Usually set to src / main / resources / META-INF / spring / spring-security.xml.
The following example, because it is only template, detailed description, see the following sections.

- spring-mvc.xml

```
<Beans  xmlns = "Http://Www.Springframework.Org/schema/beans"
    xmlns: xsi = "http://www.w3.org/2001/XMLSchema-instance"
    xmlns: sec = "http: // www .Springframework.Org / schema / Security "
    xmlns: context = "Http://Www.Springframework.Org/schema/context"
    xsi: schemaLocation = "Http://Www.Springframework.Org/schema/security
        http: // Www.Springframework.Org/schema/security/spring-security.Xsd
        Http://Www.Springframework.Org/schema/beans
        Http://Www.Springframework.Org/schema/beans/spring-beans.Xsd
        http: / /Www.Springframework.Org/schema/context
        Http://Www.Springframework.Org/schema/context/spring-context.Xsd " >
    <sec: http   use-expressions = "true" >   ! <- (1) ->
    <- OMITTED ->!
    </ sec: http>
</ beans>
```

No.	Description
(1)	By referred to as use-expressions = "true", it is possible to enable Spring EL type of access attributes.

Note

use-expressions = Enable to become Spring EL expression in the "true" is, see below.

Expression-Based Access Control

6.1.3. Appendix

6.1.3.1. Secure set of HTTP headers grant

The spring-security.xml as follows `<sec: http>` of the `<sec: headers>` By setting the elements, it is possible to set the header security automatically in the HTTP response. By putting these HTTP response header, Web browser can be addressed to detect the attack. Is not a mandatory setting, but it is recommended that you set for enhanced security.

```
<Sec: http  use-expressions = "true" >
  <-! OMITTED ->
  <sec: headers  />
  <-! OMITTED ->
</ sec: http>
```

In this setting, HTTP response header is set for the following items.

- Cache-Control
- X-Content-Type-Options
- Strict-Transport-Security
- X-Frame-Options
- X-XSS-Protection

HTTP header name	Problem of If the setting is inappropriate (including not set)	Behavior when you have properly configured
`Cache-Control`	One user content that can be viewed by logging is cached, there is a case in which would another user also can view after logout.	By an instruction to not cache content, the browser so as to always get the information of the server.
`X-Content-Type-Options`	Browser, would be to determine what to operate by examining the contents of the content without deciding the content in Content-Type, it may Script it does not assume from being executed.	Browser, so as not to determine the content to be operated by examining the contents of content without deciding the contents in the Content-Type. If the MIME type does not match, I will limit that the Script is executed.
`Strict-Transport-Security`	In spite hopes to be accessed by HTTPS secure page, when it is accessed by HTTP, there is a possibility to receive an HTTP from attack. (Example: the middle attacker intercepts the user's HTTP request, and to redirect to a malicious site.)	Once you access the HTTPS to legitimate Web site, the browser automatically to understand as to use only HTTPS, to prevent the execution of the man-in-the-middle attack that is induced to a malicious site.
`X-Frame-Options`	The screen of malicious Web site A was invisible in the transmission process, instead `<iframe>` If you embed	Own-created Web site (= site B) is to other Web Sites (= site A)

	the other normal site B in the tag, the attacker can be access to the site A with the intention of site B to the user. In this situation, when overlaying the position of the link the submit button and site B of site A, the attacker to the user, can be sent a malicious request by site A with the intention that you click on the link of the normal site B . (ClickJacking)	<iframe> to avoid loaded by using the tag.
X-XSS-Protection	Determination of harmful script by XSS filter that has been implemented in the browser is disabled.	XSS filter is implemented in the browser, it is determined the harmful script queries the user whether to perform or disabled (behavior varies by the browser).

The above configuration is also separately configured as the following (1) to (5). It should be sift as necessary.

```
<Sec: http  use-expressions = "true" >
  <-! OMITTED ->
  <sec: headers>
    <sec: Cache-Control  />   <-! (1) ->
    <sec: content-Type- options  />   <- (2) ->!
    <sec: HSTS  />   <- (3) ->!
    <sec: frame-options  />   ! <- (4) ->
    <sec: XSS-Protection  />   <- (5) ->!
  </ sec: headers>
  ! <- OMITTED ->
</ sec: http>
```

HTTP header grants by Spring Security

No.	Description	HTTP response header that is output by default	Attribute the presence or absence
(1)	And instructs it to not cache the data to the client.	Cache-Control: no-Cache, no-store, max-age = 0, MUST-revalidate Pragma: no-Cache Expires: 0	Without
(2)	Ignoring the content type, the client side by the contents content and instructed not decide automatically processing method.	X-Content-Type-Options: nosniff	Without
(3)	The site that you access with HTTPS, and instructed to continue the HTTPS connection. (In the case of a site in the HTTP, it will be ignored and will not be granted as a header item.)	Strict-Transport-Security: max-age = 31536000 ; IncludeSubDomains	There
(4)	The content inside iframe I instruct the display of the information contained herein.	X-Frame-Options: DENY	There
(5)	XSS attacks against browser filter that can detect	X-XSS-Protection: 1; mode = block	There

is implemented, I will be an instruction to enable
the XSS filter function.

It is possible to set the attribute if it were individually set. Some I will describe the configurable attributes.

Configurable attributes

No.	Options	Description	Specification example	HTTP response header to be output
(3)	`max-age-seconds`	The number of seconds to remember that only access over HTTPS for the relevant site (the default is 365 days)	`<Sec: HSTS max-age-seconds = "1000" />`	`Strict-Transport-Security: max-age = 1000 ; IncludeSubDomains`
(3)	`include-subdomains`	Apply instructions to the subdomain. The default value is `true` is. `false` and will not be output when you specify a.	`<Sec: HSTS include-subdomains = "false" />`	`Strict-Transport-Security: max-age = 31536000`
(4)	`policy`	I instruct the permit how to display the contents inside iframe. The default value `DENY` is (ban from being displayed in the frame). `SAMEORIGIN` also can be changed to (to allow reading to frame only the site within the page).	`<Sec: frame-options policy = "SAMEORIGIN" />`	`X-Frame-Options: SAMEORIGIN`
(5)	`enabled, block`	`false` by specifying, it becomes possible to disable XSS filter is recommended enabled.	`<Sec: XSS-Protection enabled = "false" block = "false" />`	`X-XSS-Protection: 0`

Note

Processing for these header is not supported in some browsers. See the official website or the following pages of the browser.

- Https://Www.Owasp.Org/index.Php/HTTP_Strict_Transport_Security (Strict-Transport-Security)
- Https://Www.Owasp.Org/index.Php/Clickjacking_Defense_Cheat_Sheet (X-Frame-Options)
- Https://Www.Owasp.Org/index.Php/List_of_useful_HTTP_headers (X-Content-Type-Options, X-XSS-Protection)

For more information about the official reference I see.