**51. Spring Security - Cross Site Request Forgery (CSRF)**

**What is CSRF?**

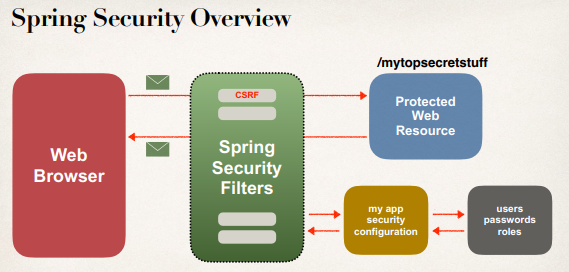
A security attack where an evil website tricks user into executing an action on a web application that user is currently logged in.

**Example**:

* You are logged into your banking app
  + tricked into sending money to another person
* You are logged into an e-commerce app
  + tricked into purchasing unwanted items

**CSRF Protection**:

* To protect against CSRF attacks
* Embed additional authentication data/token into all HTML forms
* On subsequent requests, web app will verify token before processing



Spring Security support for CSRF and it is the part of Spring Security Filters. This filter can actually generate tokens to send back to the browser. We can use this in our HTML forms, and then we can send the data over accordingly.

The Spring Security Filters will verify the tokens to make sure they are valid for this given user session before actually processing the request.

**Spring Security’s CSRF Protection**:

* CSRF protection is enabled by default in Spring Security
* Spring Security uses the Synchronizer Token Pattern
  + Each request includes a session cookie and randomly generated token
* For request processing, Spring Security verifies token before processing
* All of this is handled by Spring Security Filters

**When to use CSRF Protection?**

* The Spring Security team recommends
* Use CSRF protection for any normal browser web requests
* If you are building a service for non-browser clients
  + you may want to disable CSRF protection (after careful review)

**Use Spring Security CSRF Protection**:

* For form submissions use **POST** instead of **GET**
* Include CSRF token in form submission
* **<form:form>** automagically adds CSRF token
* If we don’t use **<form:form>**, we must manually add CSRF token

We have to do that for every form request.

**Manually add CSRF token**:

<form action=*"*...*"* method=*"POST"*>

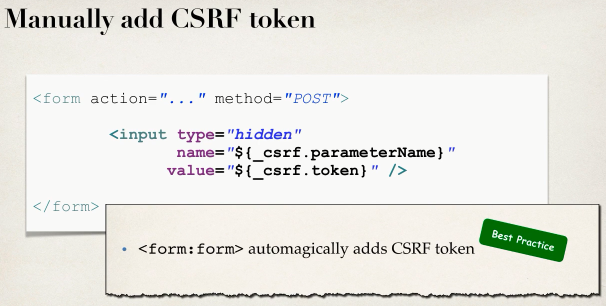
<input type=*"hidden"*

name="${\_csrf.parameterName}"value="${\_csrf.token}" />

</form>

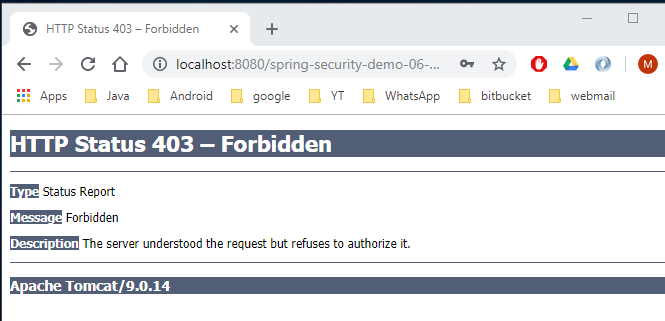
**Note**:

For best practice we don’t write this code manually. As a developer we use the Spring MVC form tags. Because form tag automatically does this for us.



**What happens if you don’t include CSRF token?**

If we don’t use the **CSRF** token then we will get the following error when we submit the form.



**CSRF Resources**:

* CSRF Security Reference

<https://www.owasp.org/index.php/Cross-Site_Request_Forgery_(CSRF)>

* Spring Security CSRF Support

<https://docs.spring.io/spring-security/site/docs/current/reference/htmlsingle/#csrf>

51. Spring Security

- Cross Site Request Forgery (CSRF)