**[My Shitty Code](https://myshittycode.com/" \o "My Shitty Code)**

Embracing the Messiness in Search of Epic Solutions

[Skip to content](https://myshittycode.com/2015/03/30/spring-security-invalid-csrf-token-null-was-found-on-the-request-parameter-_csrf-or-header-x-csrf-token/#content)

Spring Security: Invalid CSRF Token ‘null’ was found on the request parameter ‘\_csrf’ or header ‘X-CSRF-TOKEN’

[2 Replies](https://myshittycode.com/2015/03/30/spring-security-invalid-csrf-token-null-was-found-on-the-request-parameter-_csrf-or-header-x-csrf-token/#comments)

**PROBLEM**

With Spring Security 4.x, the CSRF protection is enabled by default. You may disable it, but to be more aligned with OWASP and the industry security standard, it’s best to leave this setting the way it is. [Learn more about CSRF attack…](https://www.owasp.org/index.php/Cross-Site_Request_Forgery_%28CSRF%29)

To prevent this attack, Spring Security 4.x requires you to attach a server-side generated CSRF token on any POST, PUT or DELETE calls… basically, actions that may modify the request state. Their argument for not attaching this token on GET is to prevent this token value from leaking out.

Further, you will require to call POST /login and POST /logout now. In the past, you can call GET /j\_spring\_security\_logout without problem.

If you invoke POST, PUT or DELETE without this CSRF token, you will get a 403 error with this message: "Invalid CSRF Token 'null' was found on the request parameter '\_csrf' or header 'X-CSRF-TOKEN'.".

**SOLUTION**

To obtain this CSRF token, add this Spring Security custom tag to the JSP file:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | <!DOCTYPE html>  <html>      <head>          <sec:csrfMetaTags/>      </head>      <body>      </body>  </html> |

The rendered HTML looks like this:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | <!DOCTYPE html>  <html class="no-js">      <head>          <meta name="\_csrf\_parameter" content="\_csrf" />          <meta name="\_csrf\_header" content="X-CSRF-TOKEN" />          <meta name="\_csrf" content="e62835df-f1a0-49ea-bce7-bf96f998119c" />      </head>      <body>      </body>  </html> |

Finally, set the request header before making the AJAX call:-

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16 | var header = $("meta[name='\_csrf\_header']").attr("content");  var token = $("meta[name='\_csrf']").attr("content");    $.ajax({      url: '/test',      type: 'POST',      beforeSend: function(xhr){          xhr.setRequestHeader(header, token);      },      success: function(data) {          console.log(data);      },      error: function (xhr, ajaxOptions, thrownError) {          console.log(xhr.status + ": " + thrownError);      }  }); |

Advertisements