Learn how to use the Option CSRFToken config.txt directive to enable logic to detect and block Cross Site Request Forgery in EZproxy.

Starting with version 6.3, EZproxy allows enabling logic to detect and block Cross Site Request Forgery (CSRF) attempts to log in. Enabling this logic requires the following changes to login.htm, loginbu.htm, and logup.htm files:

1. Locate the form tag and verify that its value is exactly:
   
   ```html
   <form action="/login" method="POST">
   ```

2. Immediately following the form tag, add a new line with just the caret symbol followed by the letter F as:

   ```
   ^F
   ```

With these lines in place, edit config.txt to add the position-independent directive below, then restart EZproxy:

   ```
   Option CSRFToken
   ```

Once this logic is in place, EZproxy will send a cookie named ezproxycsrftoken with a value that must match up to a hidden field that is added to the login form where the `^F` is located. These values must match for authentication to succeed.

This option is incompatible with CAS, CGI, Shibboleth, and Ticket authentication since the actual authentication step occurs external to EZproxy. EZproxy automatically overrides this option for these methods.

If the user blocks the ezproxycsrftoken cookie, the system administrator fails to put `^F` in the log*.htm files, or if an attempt it made to circumvent the CSRF logic, EZproxy will display the message:

   **Login failed due to missing or invalid ezproxycsrftoken cookie**

This message can be overridden by creating the file csrf.htm in the docs directory.

With this option enabled, it is possible to override the requirement for specific lines in user.txt by adding `IgnoreCSRFToken` such as:

   ```
   someuser:somepass:IgnoreCSRFToken
   ```

This option is mainly intended for special lines that override access to the EZproxy administrative interface such as:

   ```
   someadmin:somepass:Admin,IgnoreCSRFToken
   ```

**Syntax**

The complete syntax is:

   ```
   Option CSRFToken
   ```