

3 Potential Authentication Bypass through "autologin" feature

Share:     

TIMELINE



egix submitted a report to ImpressCMS.

Jan 19th (2 years ago)

Summary:

The vulnerability is located in the `/plugins/preloads/autologin.php` script:

Code 958 Bytes [Wrap lines](#) [Copy](#) [Download](#)

```
1 45.      $uname = $myts->stripSlashesGPC($autologinName);
2 46.      $pass = $myts->stripSlashesGPC($autologinPass);
3 47.      if (empty($uname) || is_numeric($pass)) {
4 48.          $user = false ;
5 49.      } else {
6 50.          // V3
7 51.          $uname4sql = addslashes($uname);
8 52.          $criteria = new icms_db_criteria_Compo(new icms_db_criteria_Item('login_name', $uname4sql));
9 53.          $user_handler = icms::handler('icms_member_user');
10 54.          $users = $user_handler->getObjects($criteria, false);
11 55.          if (empty($users) || count($users) != 1) {
12 56.              $user = false ;
13 57.          } else {
14 58.              // V3.1 begin
15 59.              $user = $users[0] ;
16 60.              $old_limit = time() - (defined('ICMS_AUTOLOGIN_LIFETIME') ? ICMS_AUTOLOGIN_LIFETIME : 604800);
17 61.              list($old_Ynj, $old_encpass) = explode(':', $pass);
18 62.              if (strtotime($old_Ynj) < $old_limit || md5($user->getVar('pass')) .
19 63.                  ICMS_DB_PASS . ICMS_DB_PREFIX . $old_Ynj) != $old_encpass)
20 64.              {
21 65.                  $user = false;
22 66.              }
```

User input passed through the "autologin_uname" and "autologin_pass" cookie values is being used at lines 51-54 to fetch an user object from the database, and then at lines 62-63 to check the correctness of the user's password. The vulnerability exists because of an unsafe way of comparing those parameters, due to comparison operator `!=` is being used instead of `!==` within the "if" statement at lines 62-63. The latter operator returns "true" only if the compared values are equal and the same type, while the first compare the values after "type juggling". This might be exploited to bypass the authentication mechanism and login as any user without the knowledge of the relative password.

ImpressCMS branch :

The vulnerability has been spotted on ImpressCMS version 1.4.2 (the latest at the time of writing).

Steps To Reproduce:

Use the attached Proof of Concept (PoC) script to reproduce this vulnerability. It's a PHP script supposed to be used from the command-line (CLI). You should see an output like the following:

Code 243 Bytes [Wrap lines](#) [Copy](#) [Download](#)

```
1 $ php auth-bypass.php http://localhost/impresscms/ admin
2 [-] Starting authentication bypass attack...
3 [-] 2021-01-20 022141
4 [-] You can autologin with the following cookies:
5 [-] Cookie: autologin_uname=admin; autologin_pass=2021-01-20 022141:0
```

NOTE: the script will try to send multiple requests with incremental dates within the `autologin_pass` cookie (that will be the value of the `$old_Ynj` variable), and this will generate a different MD5 hash for each request, until something like `0e174892301580325162390102935332` will be returned by the `md5()` function. For this reason, the exploitation likelihood is very low, and the script execution might take days, months, or a theoretically infinite time.

Impact

This vulnerability could potentially be exploited to bypass the authentication mechanism and login without valid credentials.

1 attachment:

F1164379: [auth-bypass.php](#)



fiammybe (ImpressCMS staff) changed the status to **Triaged**.

Jan 30th (2 years ago)

Hello, thanks for the information. Its not high risk at the moment, so we will probably have to look into it later (we still have several higher priority fixes to do).



egix posted a comment.

Jan 30th (2 years ago)

Hi @fiammybe, I know this is a low risk issue. However, I thought to report it because the fix is pretty simple and straightforward: just replace `!=` with `!==` at line 63.







fiammybe (ImpressCMS staff) closed the report and changed the status to **Resolved**.

Feb 2nd (2 years ago)

This has been resolved and will be in ImpressCMS 1.4.3. Thank you!

egix posted a comment.

Updated Feb 3rd (2 years ago)

 egix posted a comment. @fiammybe I confirm this has been resolved in 1.4.3. Please feel free to close this report.	Feb 11th (10 months ago)
 egix requested to disclose this report.	Feb 20th (10 months ago)
 fiammybe ImpressCMS staff updated CVE reference to CVE-2021-26600 .	Feb 20th (10 months ago)
 This report has been disclosed.	Mar 22nd (9 months ago)