chamilo / **chamilo-lms** Public

Code | Issues 415 | Pull requests 12 | Discussions | Actions | Projects

Commit 750a453

ywarnier committed 3 weeks ago

```
Security: Fix weak pass recovery - refs GHSA-f27g-66gq-g7v2
```

1.11.x · v1.11.38

1 parent 46c18c4 commit 750a453

1 file changed +74 -11 lines changed

Top

Filter files...

main/inc/lib

login.lib.php

1 file changed +74 -11 lines changed

Search within code

main/inc/lib/login.lib.php

```
@@ -37,9 +37,9 @@ public static function get_user_account_list($user, $reset = false, $by_username
37 37
38 38             if ($reset) {
39 39                 if ($by_username) {
40    -                 $secret_word = self::get_secret_word($user['email']);
   40 +                 $token = self::generate_reset_token($user['uid']);
41 41                  if ($reset) {
42    -                     $reset_link = $portal_url."main/auth/lostPassword.php?reset=".$secret_word."&id=".$user['uid'];
   42 +                     $reset_link = $portal_url."main/auth/lostPassword.php?reset=".$token."&id=".$user['uid'];
43 43                      $reset_link = Display::url($reset_link, $reset_link);
44 44                  } else {
```

```
45  45                          $reset_link = get_lang('Pass')." : $user[password]";
    @@ -53,9 +53,9 @@ public static function get_user_account_list($user, $reset
    = false, $by_username
53  53                      }
54  54                  } else {
55  55                      foreach ($user as $this_user) {
56      -                       $secret_word = self::get_secret_word($this_user['email']);
    56  +                       $token = self::generate_reset_token($this_user['uid']);
57  57                          if ($reset) {
58      -                           $reset_link = $portal_url."main/auth/lostPassword.php?
    reset=".$secret_word."&id=".$this_user['uid'];
    58  +                           $reset_link = $portal_url."main/auth/lostPassword.php?
    reset=".$token."&id=".$this_user['uid'];
59  59                              $reset_link = Display::url($reset_link, $reset_link);
60  60                          } else {
61  61                              $reset_link = get_lang('Pass')." :
    $this_user[password]";
    @@ -248,9 +248,36 @@ public static function sendResetEmail(User $user)
248 248          *
249 249          * @author Olivier Cauberghe <olivier.cauberghe@UGent.be>, Ghent University
250 250          */
    251 +      /**
    252 +       * Generate a cryptographically random reset token for the given user,
    253 +       * store it (with a timestamp) in the database, and return it.
    254 +       *
    255 +       * @param int $userId
    256 +       *
    257 +       * @return string The hex token
    258 +       */
    259 +      public static function generate_reset_token($userId)
    260 +      {
    261 +          $token = bin2hex(random_bytes(32));
    262 +          $em = Database::getManager();
    263 +          /** @var User $user */
    264 +          $user = $em->find('ChamiloUserBundle:User', (int) $userId);
    265 +          if ($user) {
    266 +              $user->setConfirmationToken($token);
    267 +              $user->setPasswordRequestedAt(new \DateTime());
    268 +              $em->persist($user);
```

```
                $em->flush();
            }

            return $token;
        }

        /**
         * @deprecated Use generate_reset_token() instead.
         */
    public static function get_secret_word($add)
    {
-           return $secret_word = sha1($add);
+           return sha1($add);
    }

    /**
@@ -285,15 +312,51 @@ public static function reset_password($secret, $id,
$by_username = false)
                return get_lang('CouldNotResetPassword');
            }

-           if (self::get_secret_word($user['email']) == $secret) {
-               // OK, secret word is good. Now change password and mail it.
-               $user['password'] = api_generate_password();
-               UserManager::updatePassword($id, $user['password']);
+           // Validate token against the stored confirmation_token.
+           $em = Database::getManager();
+           /** @var User $dbUser */
+           $dbUser = $em->find('ChamiloUserBundle:User', $id);

-               return self::send_password_to_user($user, $by_username);
+           if (!$dbUser) {
+               return get_lang('CouldNotResetPassword');
+           }

+           $storedToken = $dbUser->getConfirmationToken();
+           $requestedAt = $dbUser->getPasswordRequestedAt();

+           // Token must exist (a reset must have been requested first).
+           if (empty($storedToken) || empty($requestedAt)) {
```

```
                +             return get_lang('NotAllowed');
                +         }
                +
                +         // Token expires after 1 hour.
                +         $expiresAt = clone $requestedAt;
                +         $expiresAt->modify('+1 hour');
                +         if (new \DateTime() > $expiresAt) {
                +             // Clear expired token.
                +             $dbUser->setConfirmationToken(null);
                +             $dbUser->setPasswordRequestedAt(null);
                +             $em->persist($dbUser);
                +             $em->flush();
                +
                +             return get_lang('NotAllowed');
                +         }
                +
                +         // Timing-safe comparison.
                +         if (!hash_equals($storedToken, $secret)) {
                +             return get_lang('NotAllowed');
294             348           }
295             349
296             -         return get_lang('NotAllowed');
                350       +         // Token is valid — change the password and clear the token.
                351       +         $user['password'] = api_generate_password();
                352       +         UserManager::updatePassword($id, $user['password']);
                353       +
                354       +         $dbUser->setConfirmationToken(null);
                355       +         $dbUser->setPasswordRequestedAt(null);
                356       +         $em->persist($dbUser);
                357       +         $em->flush();
                358       +
                359       +         return self::send_password_to_user($user, $by_username);
297             360       }
298             361
299             362       /**
```

Comments 0

Please sign in to comment.