

New issue

[Jump to bottom](#)

Fix CSRF attack that can cause an authenticated user to be logged out #7302

🔒 Closed xudongzheng wants to merge 1 commit into `roundcube:master` from `xudongzheng:master`

Conversation 7 Commits 1 Checks 0 Files changed 1



xudongzheng commented on Mar 29, 2020 • edited

A login POST request without a valid token deleted the active session. The login code should not run if the session is already authenticated.

A logout POST request succeeded without a valid token as the token checking code only considers GET. Logging out should therefore be restricted to GET.

A simple proof of concept:

```
<div>
  <form method="post" action="https://demo.roundcubeplus.com/">
    <input type="hidden" name="_task" value="logout">
    <button type="submit">Logout</button>
  </form>
  <form method="post" action="https://demo.roundcubeplus.com/">
    <input type="hidden" name="_task" value="login">
    <input type="hidden" name="_action" value="login">
    <input type="hidden" name="_user" value="example">
    <input type="hidden" name="_pass" value="example">
    <button type="submit">Login</button>
  </form>
</div>
```

- Fix CSRF attack that can cause an authenticated user to be logged out

7156ba7

alecpl added bug C: Security labels on Mar 29, 2020

alecpl added this to the 1.4.4 milestone on Mar 29, 2020

alecpl commented on Apr 16, 2020

Member

I have no objections to the second part of the patch, but I'm not sure about the first. I have unfortunately no better idea. I'd say that we should check the token instead, but some plugins may already implement their own validation in 'authenticate' hook. Moving `kill_session()` call after the hook may also make some problems, or maybe not?

@thomascube I need a second pair of eyes on this.

thomascube commented on Apr 19, 2020

Member

I hesitate to accept the `!isset($_SESSION['user_id'])` part because it significantly changes the current behavior. Roundcube's CSRF check should fail if the `_token` parameter (GET or POST) is missing. In the login case, this is only checked if `$_SESSION['temp']` is set. I think we should consider to move the `kill_session()` command after the `authenticate` hook and make it conditional with `$auth['valid']`.

Regarding the logout GET request: the proper solution would be to use POST requests for this action. But for now it might already be enough to consider the actual request method in the call to `request_security_check()`.

alecpl commented on Apr 22, 2020 • edited

Member

That first part is a real PITA. Killing the session after authenticate hook will break some plugins (e.g. kolab_auth). Adding `!isset($_SESSION['user_id'])` will break authentication plugins that do request validation by themselves. How about this?:

```
--- a/index.php
+++ b/index.php
@@ -106,7 +106,9 @@ if ($RCMAIL->task == 'login' && $RCMAIL->action == 'login') {
     $pass_charset = $RCMAIL->config->get('password_charset', 'UTF-8');

    // purge the session in case of new login when a session already exists
-   $RCMAIL->kill_session();
+   if ($request_valid) {
+       $RCMAIL->kill_session();
+   }

    $auth = $RCMAIL->plugins->exec_hook('authenticate', array(
        'host' => $RCMAIL->autoselect_host(),
@@ -154,7 +156,7 @@ if ($RCMAIL->task == 'login' && $RCMAIL->action == 'login') {
    // send redirect
    $OUTPUT->redirect($redir, 0, true);
}
- else {
+ else if (!isset($_SESSION['user_id'])) {
```

-- correction: I would move that `if` to only skip the `kill_session()` below in the `else` block.

```
if (!$auth['valid']) {  
    $error_code = rcmail::ERROR_INVALID_REQUEST;  
}
```

It still might be a regression for plugins that use the 'authenticate' hook, and require to kill the session on their side, but it might be still the best approach. @thomascube ?

thomascube commented on Apr 22, 2020

Member

@alecpl I'd suggest to move the `$RCMAIL->kill_session();` after the plugin hook and use `$auth['valid']` as a condition for it.

alecpl commented on Apr 22, 2020

Member

@thomascube, as I said, this will break kolab_auth plugin which sets some session vars. Of course, I can fix that plugin, but we're considering BC breaks here. You think this would be the best approach?
And I just checked, the second part of my patch would still be needed.

thomascube commented on Apr 22, 2020

Member

OK, I agree to your suggestion.

alecpl added a commit that referenced this pull request on Apr 26, 2020

 Fix CSRF bypass that could be used to log out an authenticated user (#_ ...

✖ 8344f07

alecpl added a commit that referenced this pull request on Apr 26, 2020


 Fix CSRF bypass that could be used to log out an authenticated user (#_ ...

✔ 9bbda42


alecpl commented on Apr 26, 2020

Member

Fixed with [8344f07](#).


 alecpl closed this on Apr 26, 2020

thomascube pushed a commit that referenced this pull request on Apr 26, 2020

 Fix CSRF bypass that could be used to log out an authenticated user (#_ ...

1e7bec9

thomascube pushed a commit that referenced this pull request on Apr 28, 2020

 Fix CSRF bypass that could be used to log out an authenticated user (#_ ...

cceeff2

Reviewers

No reviews

Assignees

No one assigned

Labels

bug C: Security

Projects

None yet

Milestone

1.4.4

Development

Successfully merging this pull request may close these issues.

None yet

3 participants

