Week 8 - Networking Fundamentals

Phase 1: "I'd like to Teach the World to Ping"

Determine the IPs for the Hollywood office and run **fping** against the IP ranges in order to determine which IP is accepting connections.

Determined the IP ranges to scan were 15.199.95.91/28, 15.199.94.91/28, 11.199.158.91/28, 167.172.144.11/32, and 11.199.141.91/28

- Ran **fping** against those addresses using the following command:

```
cristina@kali:~$ sudo fping -g 15.199.95.91/28
[sudo] password for cristina:
15.199.95.81 is unreachable
15.199.95.82 is unreachable
15.199.95.83 is unreachable
15.199.95.84 is unreachable
15.199.95.85 is unreachable
15.199.95.86 is unreachable
15.199.95.87 is unreachable
15.199.95.88 is unreachable
15.199.95.89 is unreachable
15.199.95.90 is unreachable
15.199.95.91 is unreachable
15.199.95.92 is unreachable
15.199.95.93 is unreachable
15.199.95.94 is unreachable
cristina@kali:~$ sudo fping -g 15.199.94.91/28
15.199.94.81 is unreachable
15.199.94.82 is unreachable
15.199.94.83 is unreachable
15.199.94.84 is unreachable
15.199.94.85 is unreachable
```

```
15.199.94.86 is unreachable
15.199.94.87 is unreachable
15.199.94.88 is unreachable
15.199.94.89 is unreachable
15.199.94.90 is unreachable
15.199.94.91 is unreachable
15.199.94.92 is unreachable
15.199.94.93 is unreachable
15.199.94.94 is unreachable
cristina@kali:~$ sudo fping -g 167.172.144.11/32
167.172.144.11 is alive
cristina@kali:~$ sudo fping -g 11.199.158.91/28
11.199.158.81 is unreachable
11.199.158.82 is unreachable
11.199.158.83 is unreachable
11.199.158.84 is unreachable
11.199.158.85 is unreachable
11.199.158.86 is unreachable
11.199.158.87 is unreachable
11.199.158.88 is unreachable
11.199.158.89 is unreachable
11.199.158.90 is unreachable
11.199.158.91 is unreachable
11.199.158.92 is unreachable
11.199.158.93 is unreachable
11.199.158.94 is unreachable
cristina@kali:~$ sudo fping -g 11.199.141.91/28
11.199.141.81 is unreachable
11.199.141.82 is unreachable
11.199.141.83 is unreachable
11.199.141.84 is unreachable
11.199.141.85 is unreachable
```

```
11.199.141.86 is unreachable
11.199.141.87 is unreachable
11.199.141.88 is unreachable
11.199.141.89 is unreachable
11.199.141.90 is unreachable
11.199.141.91 is unreachable
11.199.141.92 is unreachable
11.199.141.93 is unreachable
11.199.141.94 is unreachable
```

After scanning, found that 167.172.144.11 is responding, which is a **vulnerability**.

Recommend that IP 167.172.144.11 restricts allowing their ICMP echo requests in order to prevent successful responses from ping requests.

OSI Layer: **Layer 3 - Network Layer**

Phase 2: "Some Syn for Nothin`"

Ran the nmap command to determine which ports were open. **Port 22** is showing that it's open.

```
cristina@kali:~$ sudo nmap -sS 167.172.144.11
Starting Nmap 7.80 ( https://nmap.org ) at 2020-06-13 10:33 PDT
Nmap scan report for 167.172.144.11
Host is up (1.8s latency).
Not shown: 999 closed ports
PORT   STATE SERVICE
22/tcp open  ssh

Nmap done: 1 IP address (1 host up) scanned in 3.12 seconds
```

Port 22 is open, meaning that a hacker can ssh into the server, that presents a **vulnerability**.

Recommend closing Port 22 to prevent unauthorized entry.

OSI Layer: **Layer 4 - Transport Refresher**

Phase 3: "I Feel a DNS Change Comin' On"

Used the command below to ssh into the server, using **jimi** as the username and **hendrix** as the password:

```
cristina@kali:~$ sudo ssh jimi@167.172.144.11 -p 22
[sudo] password for cristina:
jimi@167.172.144.11's password:
Linux GTscavengerHunt 4.9.0-11-amd64 #1 SMP Debian 4.9.189-3+deb9u1 (2019-09-
20) x86_64

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Mon Jun 15 17:17:19 2020 from 173.79.148.90
Could not chdir to home directory /home/jimi: No such file or directory
```

Used **ls** to to view the directories and used the command:

```
cd etc/
```

Once in the **etc** folder, used the command below to see where rollingstone.com was being redirected to:

```
$ cat hosts
# Your system has configured 'manage_etc_hosts' as True.
# As a result, if you wish for changes to this file to persist
```

```
# then you will need to either
# a.) make changes to the master file in /etc/cloud/templates/hosts.tmpl
# b.) change or remove the value of 'manage_etc_hosts' in
#     /etc/cloud/cloud.cfg or cloud-config from user-data
#
127.0.1.1 GTscavengerHunt.localdomain GTscavengerHunt
127.0.0.1 localhost
98.137.246.8 rollingstone.com

ooooooooollowing lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts

$ pwd
/etc
```

rollingstone.com is redirecting to 98.137.246.8.

Using my local Apple computer, used the command below to determine the real domain of the IP address found for rollingstone.com redirect:

```
cristina@Cristinas-Mac-mini ~ % nslookup 98.137.246.8
Server:     209.18.47.62
Address:    209.18.47.62#53

Non-authoritative answer:
8.246.137.98.in-addr.arpa   name = media-router-
fp2.prod1.media.vip.gq1.yahoo.com.
```

```
Authoritative answers can be found from:
```

The domain name found was: **media-router-fp2.prod1.media.vip.gq1.yahoo.com**.

The **vulnerability** is that the permissions are not restricted enough, thereby allowing website address redirects.

Recommendation includes changing the redirect for the rollingstone.com to the actual website as well as change permissions so that only sudo users can access.

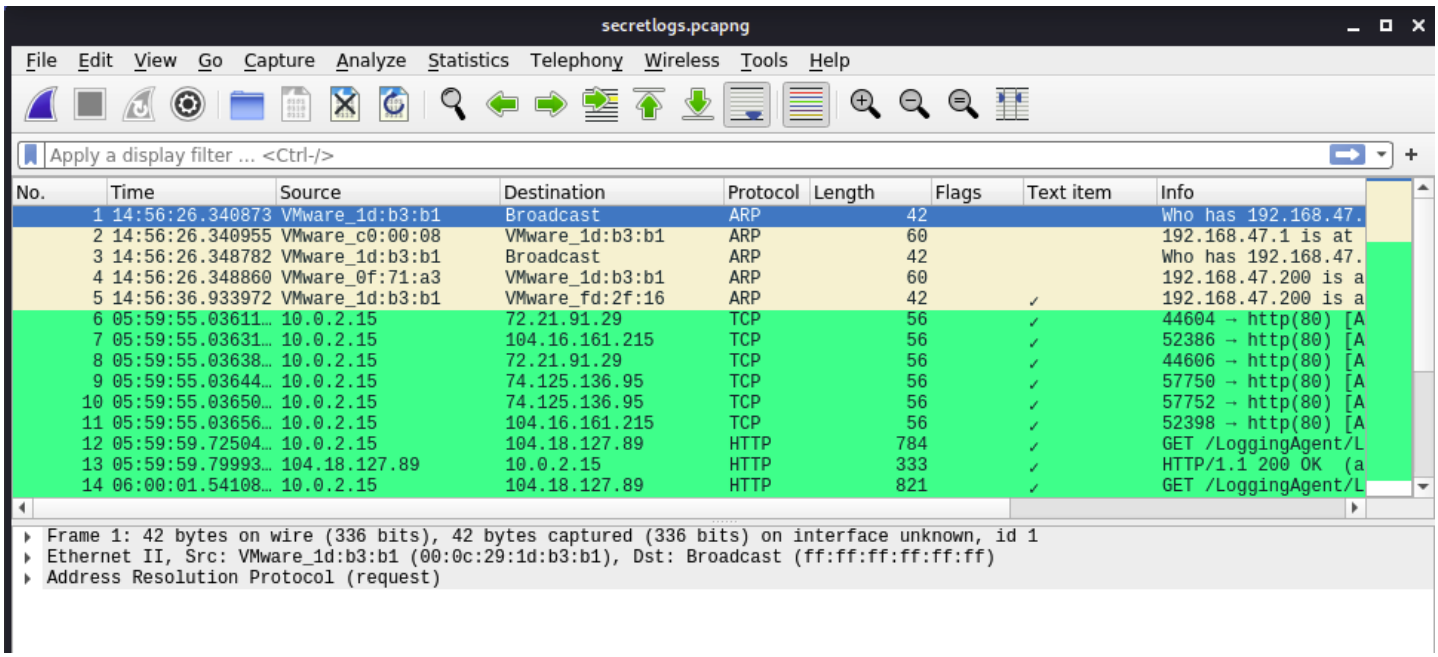
OSI Layer**: Layer 7 - Application - HTTP/FTP, working through Layer 3 - Network Layer**

Phase 4: "ShARP Dressed Man"

Used the **cat** command to view the document that the hacker left behind:

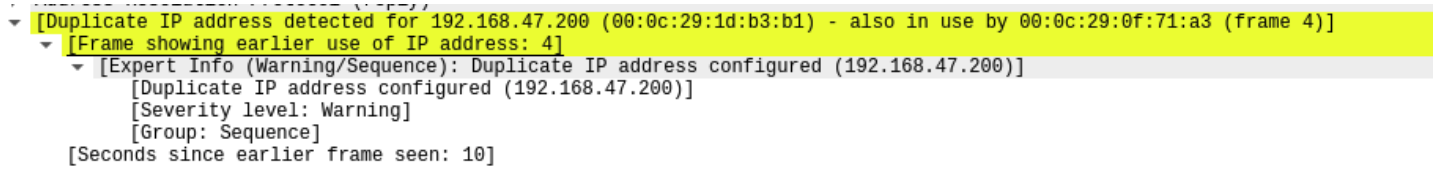
```
$ cat packetcaptureinfo.txt
 Captured Packets are here:
 https://drive.google.com/file/d/1ic-CFFGrbruloYrWaw3PvT71elTkh3eF/view?
usp=sharing
```

The link led to a pcap file called secretlogs. Opened the file using Wireshark:



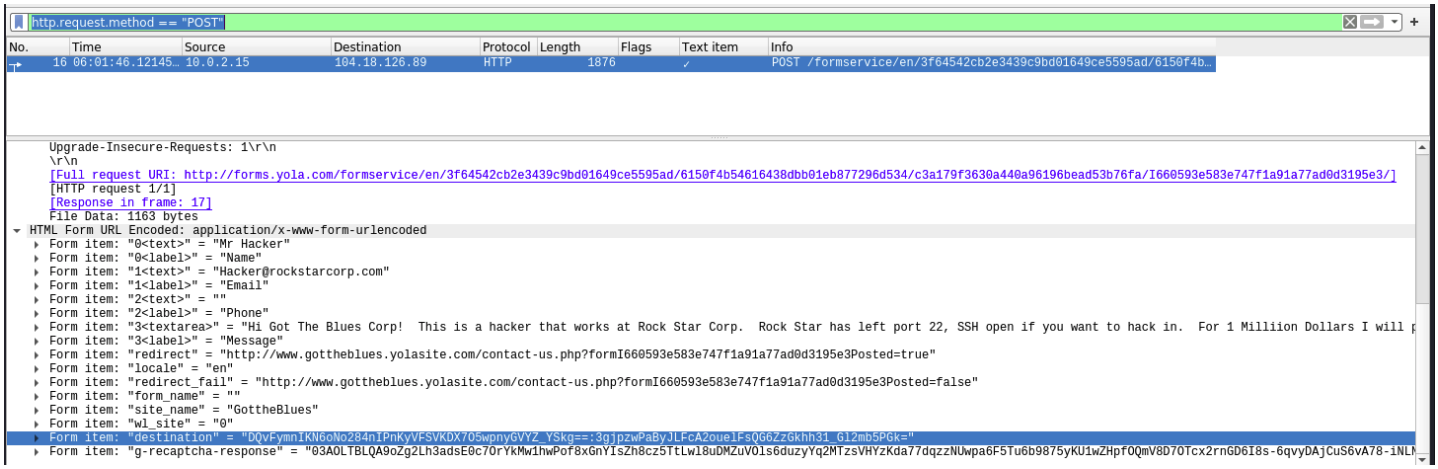
Based on my analysis of the secretlog pcap on Wireshark, I have found the following:

- Duplicate IP address detected, meaning a potential ARP spoofing attack - **Vulnerability**



ISO Layer - **Layer 2 - ARP**

Used the following command to show any POST requests: **http.request.method == "POST"**



The results revealed an email from the hacker. The email shows that the hacker is aware of the open port and is using the login information to scam for money.

Recommendation to do all the recommendations listed in the previous phases and if possible, to list this email as a phishing threat.

ISO Layer - Layer 7 - Application - HTTP