

Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Somayeh Izad, Ireedui Batsaikhan, Mukhlissa Khojayeva,
Ardan Goin, Alissa Perri, Mark Hurley

Table of Contents

This document contains the following resources:



Network Topology & Critical Vulnerabilities



Exploits Used



Avoiding Detect

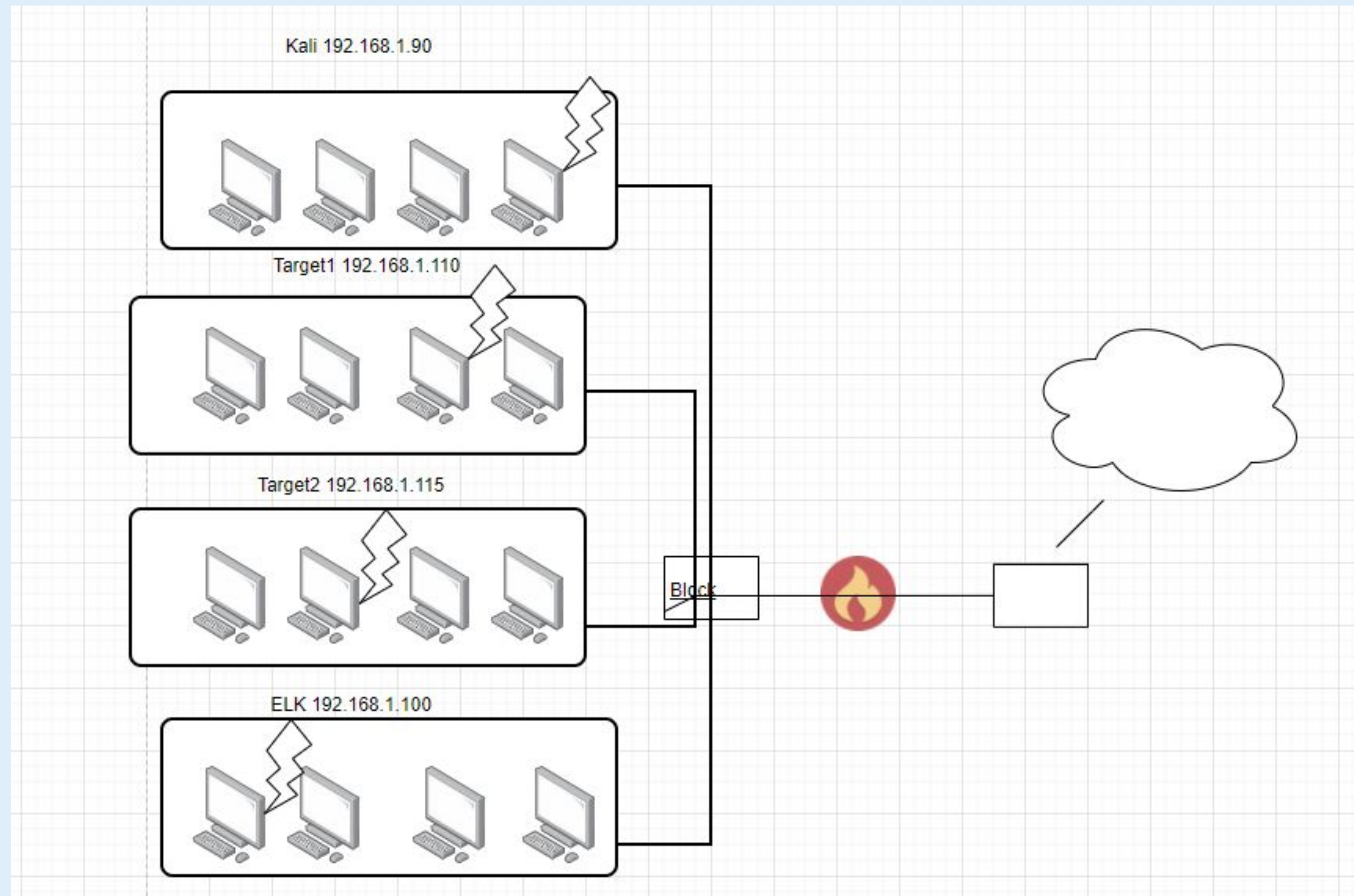


Maintaining Access



Network Topology & Critical Vulnerabilities

Network Topology



Network

Address Range:
192.168.1.0/24
Netmask:
255.255.255.0=24
Gateway: 192.168.1.1

Machines

IPv4:192.168.1.100
OS: Unbutu
Hostname: ELK

IPv4: 192.168.1.110
OS: Unbutu
Hostname: Target 1

IPv4:192.168.1.90
OS: Unbutu
Hostname: Kali

IPv4: 192.168.1.115
OS: Unbutu
Hostname: Target 2

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
Weak Password Policy	Password strength is the measure of effectiveness a password resists guessing and brute force attacks.	Weak passwords are an easy target for brute force attacks which will lead to compromise system security.
Privilege Escalation Vulnerability	Able to escalate sudo privileges for user Steven	With escalated privileges one is able to gain access to more systems and spread throughout the network
SSH	22/TCP	Open SSH
HTTP	80/TCP	Apache http 2.4.10

Exploits Used

Exploitation: [SSH]

Summarize the following:

- How did you exploit the vulnerability? E.g., which tool (Nmap, etc.) or technique (XSS, etc.)?

Weak password requirements allowed to SSH in as Michael

- What did the exploit achieve? E.g., did it grant you a user shell, root access, etc.?

Gaining user access and to their shell

- Include a screenshot or command output illustrating the exploit.

```
root@Kali:~# ssh michael@192.168.1.110
michael@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.
You have new mail.
Last login: Mon Oct  5 12:18:01 2020
michael@target1:~$
```

Exploitation: [HTTP]

Summarize the following:

- How did you exploit the vulnerability? E.g., which tool (Nmap, etc.) or technique (XSS, etc.)?

Nmap and wpscan

- What did the exploit achieve? E.g., did it grant you a user shell, root access, etc.?

Enumerating users and vulnerabilities plugins from wordpress website

- Include a screenshot or command output illustrating the exploit.

wpscan --url <http://192.168.1.110/wordpress> --wp-content-dir-eu

Exploitation: [MySQL 5.5]

Summarize the following:

- How did you exploit the vulnerability? E.g., which tool (Nmap, etc.) or technique (XSS, etc.)?

Hosting the file with Python's SimpleHTTPServer module

- What did the exploit achieve? E.g., did it grant you a user shell, root access, etc.?

Log in to the MySQL database mysql

- Include a screenshot or command output illustrating the exploit.

python -m SimpleHTTPServer 80

Avoiding Detection

Stealth Exploitation of [HTTP Errors]

Monitoring Overview

- Which alerts detect this exploit? Excessive HTTP Errors
- Which metrics do they measure? `http.response.status_code`
- Which thresholds do they fire at? 400

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
 - if we have possession of the Hash password, you can brute force them offsite.
- Are there alternative exploits that may perform better?
 - It is possible to use phishing in conjunction with metasploit to target the users

Stealth Exploitation of [HTTP Request Size]

Monitoring Overview

- Which alerts detect this exploit? HTTP Request Size Monitor
- Which metrics do they measure? sum of http request bytes
- Which thresholds do they fire at? 3500

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
by keeping cookies and request headers as small as possible to ensure that it will fit in a single packet.
- Are there alternative exploits that may perform better?
By using scp to extract or upload files

Stealth Exploitation of [CPU Usage Monitor]

Monitoring Overview

- Which alerts detect this exploit? CPU Usage Monitor
- Which metrics do they measure? max of system process cpu total pct
- Which thresholds do they fire at? 0.5 pct

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
- Are there alternative exploits that may perform better?
- If possible, include a screenshot of your stealth technique.

Maintaining Access

Backdooring the Target

Backdoor Overview

- What kind of backdoor did you install (reverse shell, shadow user, etc.)?

Reverse shell/backdoor.php with netcat listener

- How did you drop it (via Metasploit, phishing, etc.)?
by using an Ncat connection to run a script that would upload the file.

Command= ncat 192.168.1.115 80

- How do you connect to it?
http://192.168.1.115/contact.php?cmd=id

[Network Topology & Critical Vulnerabilities]

Table of Contents

This document contains the following resources:



Network Topology & Critical Vulnerabilities



Alerts Implemented



Hardening



Implementing Patches

Critical Vulnerabilities: Target 1

Our assessment uncovered the following critical vulnerabilities in **Target 1**.

Vulnerability	Description	Impact
SSH	22/TCP Open SSH	Medium
HTTP	80/TCP Apache httpd 2.4.10	High
rpcbind	111/TCP 2-4	Medium
netbios-ssn	139/TCP Samba smbd 3.X-4.X	Medium

Alerts Implemented

Watcher

[Watcher docs](#)

Watch for changes or anomalies in your data and take action if needed.

Create ▾

<input type="checkbox"/> ID	Name	State	Last fired	Last triggered	Comment	Actions
<input type="checkbox"/> 7154abad-eda0-4026-aa6b-7594bfe50f21	HTTP Request Size Monitor	✓ OK		a few seconds ago		✎ 🗑
<input type="checkbox"/> 4549308a-643d-4934-b3e0-b248be58f7b2	Excessive HTTP Errors	✓ OK		a few seconds ago		✎ 🗑
<input type="checkbox"/> f2f0e141-e329-4f4c-b495-951b1ae0be51	CPU Usage Monitor	✓ OK		a few seconds ago		✎ 🗑

Rows per page: 10 ▾

< [1](#) >

[Excessive HTTP Errors]

- Monitors the HTTP errors using filebeat
- The **threshold** is above 400 for the last 5 mins
- Helps to mitigate Brute Force Attacks

Current status for 'Excessive HTTP Errors filebeat'

Deactivate

Execution history

Action statuses

Last 7 days

Trigger time	State ↑	Comment
2020-10-01T00:38:06+00:00	▶ Firing	
2020-10-01T00:37:06+00:00	▶ Firing	
2020-10-01T00:36:06+00:00	▶ Firing	
2020-10-01T00:35:05+00:00	▶ Firing	
2020-10-01T00:34:05+00:00	▶ Firing	
2020-10-06T22:39:45+00:00	✓ OK	

[HTTP Request Size Monitor]

- Monitors http.request.bytes using filebeat and packetbeat
- The threshold is above 3500 for the last minute
- Helps to mitigate Denial of Service Attacks

Current status for 'HTTP Request Size Monitor Packetbeat'

Execution history Action statuses

Last 7 days ▾

Trigger time	State	Comment
2020-09-30T23:34:55+00:00	▶ Firing	
2020-09-30T23:33:55+00:00	▶ Firing	
2020-09-30T23:32:55+00:00	▶ Firing	
2020-09-30T23:31:55+00:00	▶ Firing	
2020-09-30T23:30:55+00:00	▶ Firing	
2020-09-30T23:29:55+00:00	▶ Firing	
2020-09-30T23:28:54+00:00	▶ Firing	
2020-09-30T23:27:55+00:00	▶ Firing	
2020-09-30T23:26:55+00:00	▶ Firing	
2020-09-30T23:25:54+00:00	▶ Firing	

[CPU Usage Monitor]

- Monitors the `system.process.cpu.total.pct` using `metricbeat`
- Threshold is above 0.5 for the last 5 minutes
- Helps to mitigate the Excessive CPU usage
- The Alert did not run (Most likely, it didn't exceed the threshold)

Hardening

Hardening Against Weak Password Requirements on Target 1

- Patch:
 - Require stronger passwords.
 - Include upper and lowercase letters, symbols and numbers. Passwords should be greater than 12 characters and should be changed every 60-90 days
 - Implement 2 factor authentication
- Why It Works:
 - Stronger passwords will be more difficult to crack by hackers
 - 2 factor authentication provides a second line of authentication

Hardening Against Apache/2.4.10 (Debian) CVE-2019-10098 on Target 1

- Patch:
 - Patch [oval:com.redhat.rhsa:def:20203958](https://oval.com.redhat.rhsa:def:20203958)
- Why It Works:
 - The patch fixes CVE-2019-10098. httpd: mod_rewrite potential open redirect (CVE-2019-10098). It stops the possibility of rewriting, changing and redirecting the URL.

Hardening Against Rpcbind CVE-2017-8779 on Target 1

- Patch:
 - Patch [oval:com.redhat.rhsa:def:2017126](https://oval.com.redhat.rhsa:def:2017126)
- Why It Works:
 - It fixes the way rpcbind uses libtirpc (libntirpc), a memory leak can occur when parsing specially crafted XDR messages. An attacker sending thousands of messages to rpcbind could cause its memory usage to grow without bound, eventually causing it to be terminated by the OOM killer. (CVE-2017-8779)

Implementing Patches

Implementing Patches with Ansible

Playbook Overview

Ansible is a popular open-source tool that provides automation, configuration management, and orchestration all in one. The patching is customizable via role's variables definition.

Run:

ansible-playbook orapatch.yml -k

The -k option will prompt you to enter the SSH password.

if you are using SSH keys then -k option dont have to be used.



[Start of Network Analysis]

Traffic Profile

Traffic Profile

Our analysis identified the following characteristics of the traffic on the network:

Feature	Value	Description
Top Talkers (IP Addresses)	172.16.4.205(26m Bytes) 166.62.111.64(16m Bytes)	Machines that sent the most traffic.
Most Common Protocols	UDP TCP TLSv.1.2 and 1.3	Three most common protocols on the network.
# of Unique IP Addresses	808	Count of observed IP addresses.
Subnets	255.255.255.0 is the only range observed in the private ip's	Observed subnet ranges.
# of Malware Species	68	Number of malware binaries identified in traffic.

Behavioral Analysis

Purpose of Traffic on the Network

Users were observed engaging in the following kinds of activity.

“Normal” Activity

- Downloading and installing desktop backgrounds

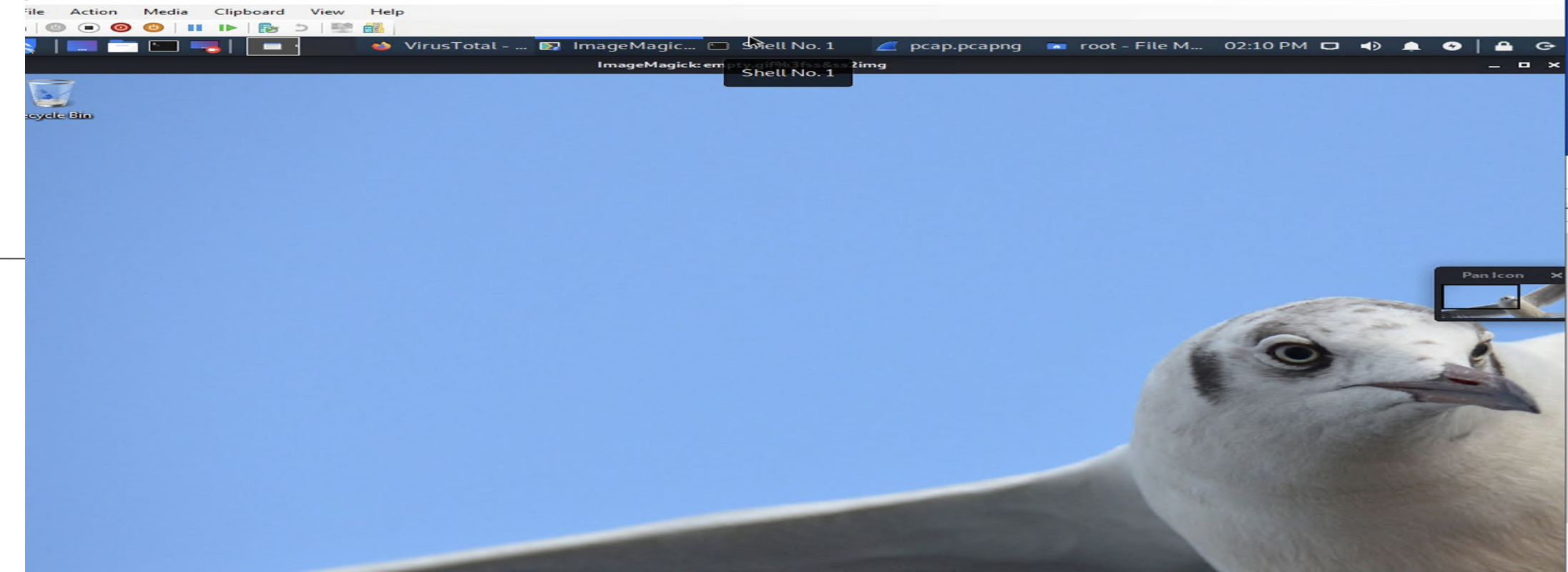
Suspicious Activity

- Set up AD network and domain controller
- Downloading malware

Normal and Malicious Activity

[Installing Desktop Backgrounds]

- What kind of traffic did you observe? HTTP Protocol, POST
- What, specifically, was the user doing? Which site were they browsing? Etc. Installing Desktop Background. b5689023.green.mattingsolutions.co



pcap.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http.request.method=="POST"

No.	Time	Source	Destination	Protocol	Length	Info
42339	2020-10-03 07:38:01.005848800	192.168.1.90	192.168.1.100	HTTP	3726	POST /_bulk HTTP/1.1 (appl
42962	2020-10-03 07:38:11.006079000	192.168.1.90	192.168.1.100	HTTP	3726	POST /_bulk HTTP/1.1 (appl
43598	2020-10-03 07:38:21.005896500	192.168.1.90	192.168.1.100	HTTP	3725	POST /_bulk HTTP/1.1 (appl
44241	2020-10-03 07:38:31.005907000	192.168.1.90	192.168.1.100	HTTP	2044	POST /_bulk HTTP/1.1 (appl
44251	2020-10-03 07:38:31.129427000	Rotterdam-PC.mind-h...	31.7.62.214	HTTP	282	POST http://31.7.62.214/fak
44643	2020-10-03 07:38:37.184590000	Rotterdam-PC.mind-h...	b5689023.green.matt...	HTTP	1366	POST /empty.gif?ss&ss2img H
44654	2020-10-03 07:38:37.202489500	Rotterdam-PC.mind-h...	31.7.62.214	HTTP	282	POST http://31.7.62.214/fak
44656	2020-10-03 07:38:37.207868400	Rotterdam-PC.mind-h...	31.7.62.214	HTTP	282	POST http://31.7.62.214/fak
44658	2020-10-03 07:38:37.213250900	Rotterdam-PC.mind-h...	31.7.62.214	HTTP	282	POST http://31.7.62.214/fak

[Frame: 44637, payload: 3585938-3587294 (1357 bytes)]
[Frame: 44638, payload: 3587295-3588651 (1357 bytes)]
[Frame: 44639, payload: 3588652-3590008 (1357 bytes)]
[Frame: 44642, payload: 3590009-3591365 (1357 bytes)]
[Frame: 44643, payload: 3591366-3592677 (1312 bytes)]
[Segment count: 2652]
[Reassembled TCP length: 3592678]
[Reassembled TCP Data: 504f5354202f656d7074792e67696663f737326737332696d...]
Hypertext Transfer Protocol
POST /empty.gif?ss&ss2img HTTP/1.1\r\n

Wireshark - Follow TCP Stream (tcp.stream eq 278) - pcap.pcapng

POST /empty.gif HTTP/1.1
Accept: */*
Accept-Language: en-US
Age: 911068f789126eb9
Content-Type: application/x-www-form-urlencoded
UA-CPU: AMD64
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 6.1; Win64; x64; Trident/7.0; .NET CLR 2.0.50727; SLCC2; .NET CLR 3.5.30729; .NET CLR 3.0.30729; Media Center PC 6.0; .NET4.0C; .NET4.0E)
Host: b5689023.green.mattingsolutions.co
Content-Length: 72
Connection: Keep-Alive
Cache-Control: no-cache

a=4f54646966376d606360653572656961646172666965616267616266676c6c67606672HTTP/1.1 200 OK
Server: nginx/1.10.3 (Ubuntu)
Date: Fri, 19 Jul 2019 18:53:12 GMT
Content-Type: text/html; charset=UTF-8
Transfer-Encoding: chunked
Connection: keep-alive
X-Powered-By: PHP/7.2.19
Access-Control-Allow-Origin: *
Access-Control-Allow-Methods: GET, POST, OPTIONS, DELETE, PUT

f09
b1001c3a578453a43c22f1cf4f3d0280b3a3d2aa3ed38096c42bea4787be5ab053d758dfcb3a72167634b8
a46d9d1aee46eb3914ab3021247509e3cd3eea598db3a5f5f9a0f113079c20a5d4848078e1b1956b606642
b736ffb94a2ae226ca5f5698a0211d71f4d6d3e08217a8f6f7b60b9922ebd814d13eb265f8ec0101731a52
f6c500f827b915884b7ddb440f77167b8abd4aac2ce1dfb9494c5ca6566ee00c6bbeaf41d8fc5a7564741
79c1578d992945ea27c15256c2f2681a39f6d2d1f2c951bdf4f3aa5fde7aa09961d626b93baba2551c3617
06d0c115c26f802f885670ce185250072098bb5f9879ecc1d7c8a2fdca7568ee08d3b5f6a011dc81b9075d
1536c51893dc244df03b9f551e9df4261629a9c8d5b3d81ebbe5eef909da77b78e789b7b9921b2a95f003e
0540ca8a0ef57b9c05845d71ea44094d5c20cdb951b92dfdacb92d191c73a2ea748c0acb9e81d999fe61e
021536c51e97dc3204e836ca5b5699e27b1439e1c1d1fcda4ba0e8beac4e926ebcc328d527a22ab1eb5b00

5,320 client pkts, 6,064 server pkts, 9 turns.

Entire conversation (13 MB) Show and save data as ASCII Stream 278

Find: Find Next

Filter Out This Stream Print Save as... Back Close

[Set up AD Network and Domain Controller]

- What kind of traffic did you observe? Which protocol(s)?
We observed the client traffic and the server traffic which included DHCP, TCP ,DNS, HTTP,LDAP protocols.The user was getting authentication for the Frank-n-ted.com domain.
- Domain name: Frank-n-Ted-DC.frank-n-ted.com and User: DESKTOP-86J4BX.frank-n-ted.com - subnet mask: 255.255.255.0

Domain Controller (DC) of the
AD Network is 10.6.12.12

ip.src==10.6.12.0/24							
No.	Time	Source	Destination	Protocol	Length	Info	
71637	2020-10-03 07:41:37...	Frank-n-Ted-DC.frank-n-ted.com	255.255.255.255	DHCP	351	DHCP	ACK
71638	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membersh	
71639	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membersh	
71640	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membersh	
71641	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	54	Membersh	
71642	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	224.0.0.251	MDNS	80	Standard	
71643	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	224.0.0.251	MDNS	90	Standard	
71644	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	224.0.0.252	LLMNR	74	Standard	
71645	2020-10-03 07:41:37...	DESKTOP-86J4BX.frank-n-ted.com	igmp.mcast.net	IGMPv3	62	Membersh	

[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)

Message type: Boot Reply (2)

Hardware type: Ethernet (0x01)

Hardware address length: 6

Hops: 0

Transaction ID: 0xba8bd7f0

Seconds elapsed: 0

Bootp flags: 0x0000 (Unicast)

Client IP address: 0.0.0.0 (0.0.0.0)

Your (client) IP address: DESKTOP-86J4BX.frank-n-ted.com (10.6.12.157)

Next server IP address: 0.0.0.0 (0.0.0.0)

Relay agent IP address: 0.0.0.0 (0.0.0.0)

[Downloading Malware]

- What kind of traffic did you observe? Which protocol(s)?

We observed the HTTP traffic being downloaded and there was malicious traffic.

- What, specifically, was the user doing? Which site were they browsing? Etc.

The user downloaded malware. Host: <http://205.185.125.104/files/june11.dll>

- Include a description of any interesting files.

The file contained malware binaries, including trojan.

The screenshot shows the Wireshark network protocol analyzer interface. At the top, the title bar reads "pcap.pcapng". Below it is a menu bar with options: File, Edit, View, Go, Capture, Analyze, Statistics, Telephony, Wireless, Tools, Help. Underneath the menu is a toolbar with various icons for file operations, capture control, and navigation. A green filter bar contains the expression "ip.src==10.6.12.0/24 && http".

The main window is divided into three panes:

- Packet List Pane:** Displays a list of captured packets. Several packets are visible, all originating from "LAPTOP-5WKHX9YG.fra..." and destined for "cardboardspaceshipt...". Packet 75293 is highlighted.
- Packet Details Pane:** Shows the hierarchical structure of the selected packet (75293). It identifies it as an HTTP GET request for "/files/june11.dll".
- Packet Bytes Pane:** Displays the raw hexadecimal and ASCII data of the selected packet's payload, starting with "GET /files/june11.dll HTTP/1.1".

VirusTotal

https://www.virustotal.com/gui/file/d3636666b407fe5527b96696377ee7ba9b609c8ef4561fa76af218ddd764dec

54 / 67

54 engines detected this file

d3636666b407fe5527b96696377ee7ba9b609c8ef4561fa76af218ddd764dec

Google update

549.84 KB Size

2020-09-06 05:48:38 UTC 27 days ago

invalid-signature overlay pedll signed

DETECTION DETAILS RELATIONS BEHAVIOR COMMUNITY 2

Ad-Aware	Trojan.GenericKD.34007934	AegisLab	Trojan.Multi.Generic.4!c
AhnLab-V3	Malware/Win32.RL_Generic.R346613	Alibaba	TrojanSpy:Win32/Yakes.56555
ALYac	Trojan.GenericKD.34007934	Antiy-AVL	GrayWare/Win32.Kryptik.ehls
SecureAge APEX	Malicious	Arcabit	Trojan.Generic.D206EB7E
Avast	Win32:DangerousSig [Trj]	AVG	Win32:DangerousSig [Trj]
Avira (no cloud)	TR/AD.ZLoader.ladbd	BitDefender	Trojan.GenericKD.34007934
BitDefenderTheta	Gen:NN.ZedlaF.34216.lu9@aul7OQgi	Bkav	W32.AIDetectVM.malware2
Cylance	Unsafe	Cynet	Malicious (score: 100)



The End