Red vs Blue

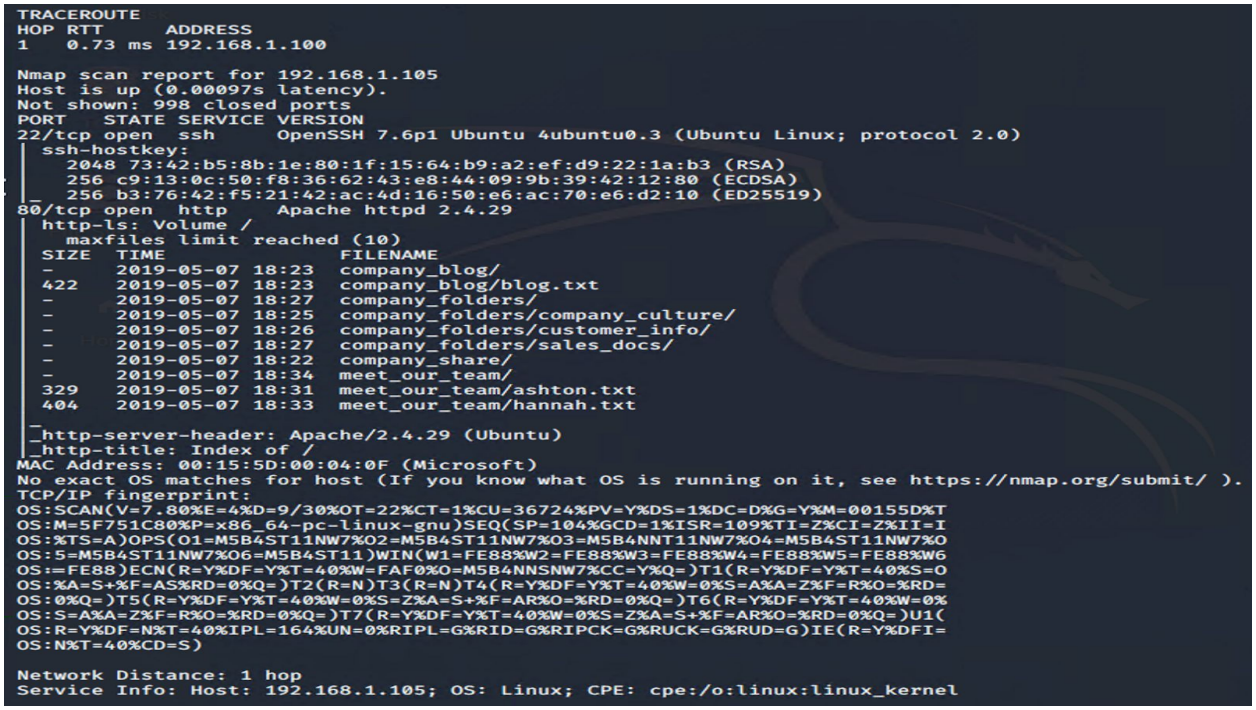
Overview

In Red vs Blue, a Kibana stacked web server was attacked using various penetration testing methods. Upon gaining access to the server, a flag was discovered. After this, a Kibana stack was used to find baselines and traces for the attack.

Attack Summary

- Run nmap scan on the network to find host and open ports
- Use dirbuster or enumeration to find hidden folder
- Enumerate a login ID from the server
- Brute force login using Hydra
- Crack sysadmin password hash
- Load reverse shell payload to the server using sysadmin
- Secure shell into sysadmin account to attain root privileges.

When dropped in the environment, the server ip is unknown, but it is known that is is running on the local network. To find any valuable information, a TCP/SYN scan was selected for being relatively stealthy, and will tell us what open ports and hosts are available. There are 4 hosts on the network, but we're most interested in this apache server with an open http port.



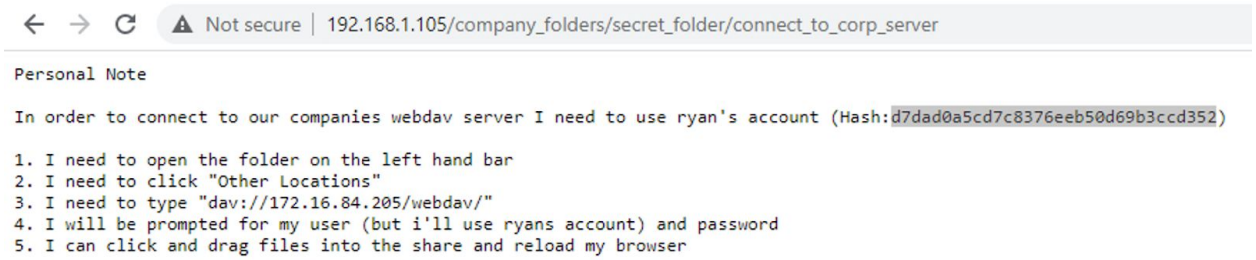
Since port 80 was open, a web browser was used for enumeration.

A hidden directory was discovered through some enumeration of files on the server. Alternatively, a tool such as dirbuster could also find this information.

This folder needs a login, which further enumeration reveals that the admin for the folder is “ashton”. Using Hydra, a brute force was attempted and successful in attaining a login for the secret folder.

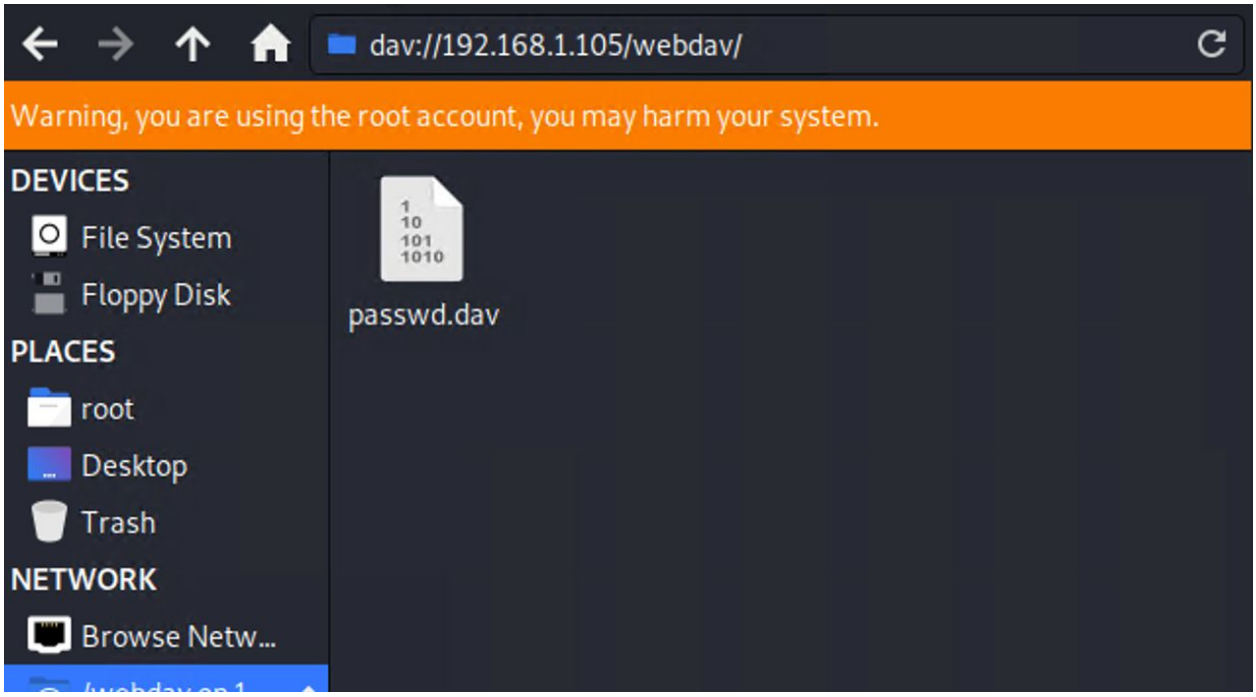
```
[80][http-get] host: 192.168.1.105   login: ashton   password: leopoldo
[STATUS] attack finished for 192.168.1.105 (valid pair found)
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2020-09-30 17:55:21
root@Kali:~# hydra -l ashton -P rockyou.txt -s 80 -f -vV 192.168.1.105 http-get /company_folders/secret_folder/
```

Using this login allows us to access the secret folder and the files inside.

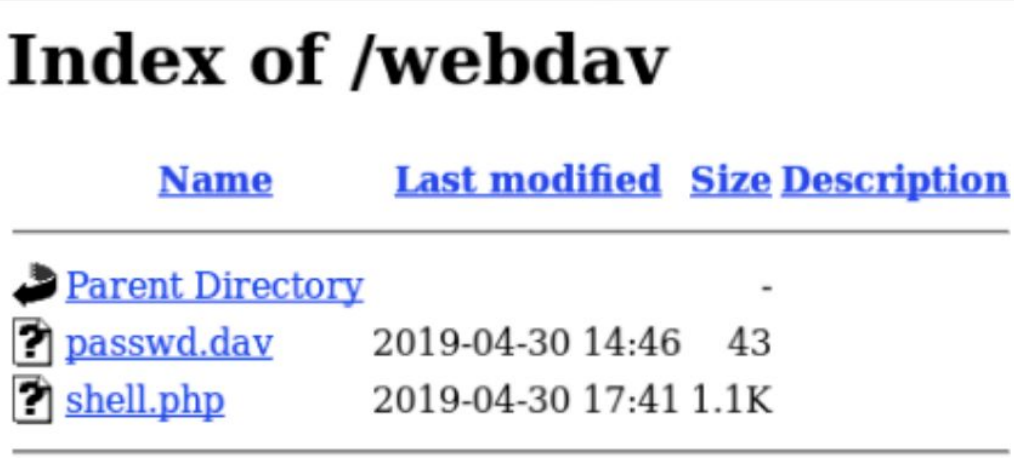


Before moving on, cracking this hash was done using crackstation.

Using this hash and login, a connection can be made to the server, and notably, files can be uploaded:

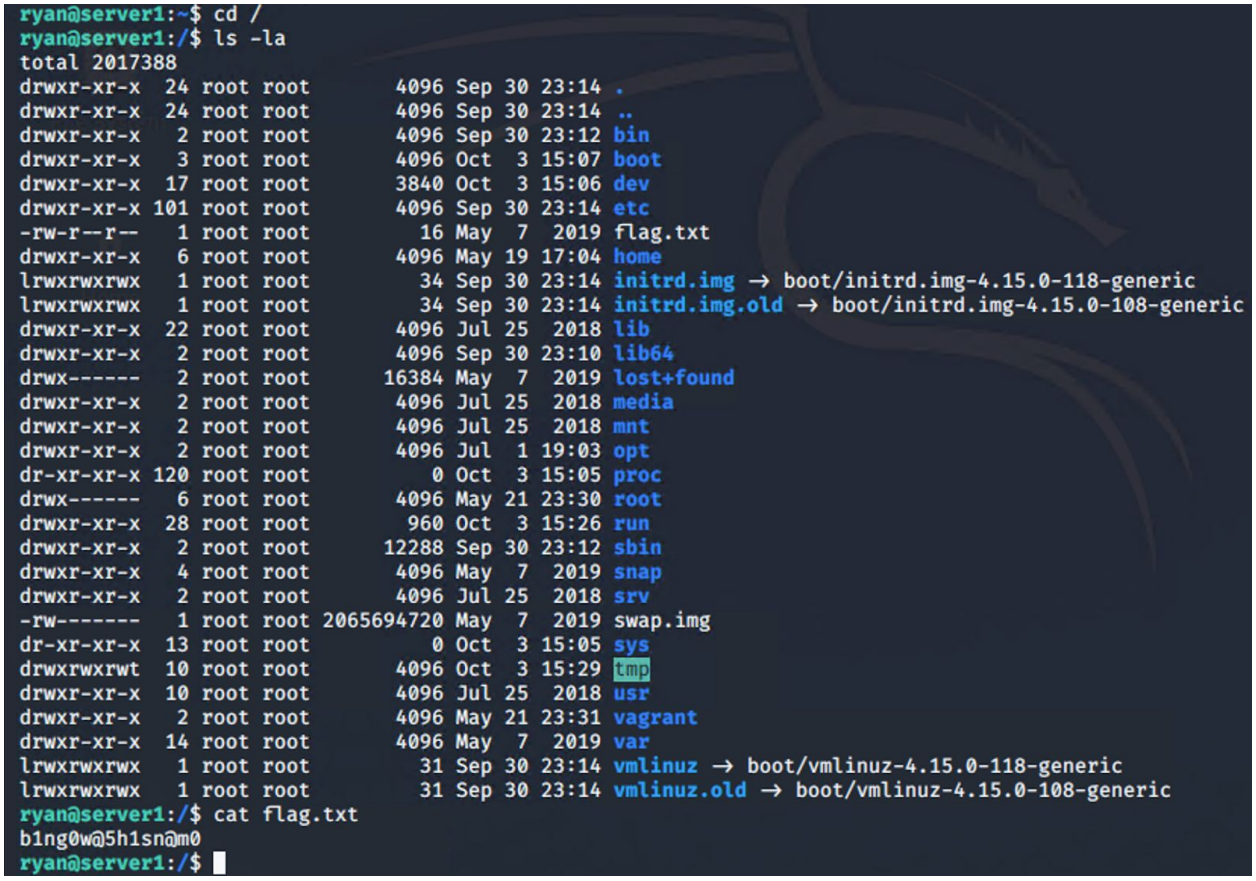


Using msfvenom, a reverse shell payload was developed to upload to the server, and then listening on a meterpreter session allowed us to connect to the server.



Clicking on our shell payload opens a meterpreter session with root privileges.

Alternatively, Ryan's account can be logged in through SSH, and also has root privileges.



This concludes our attack, and we have root access to our server.

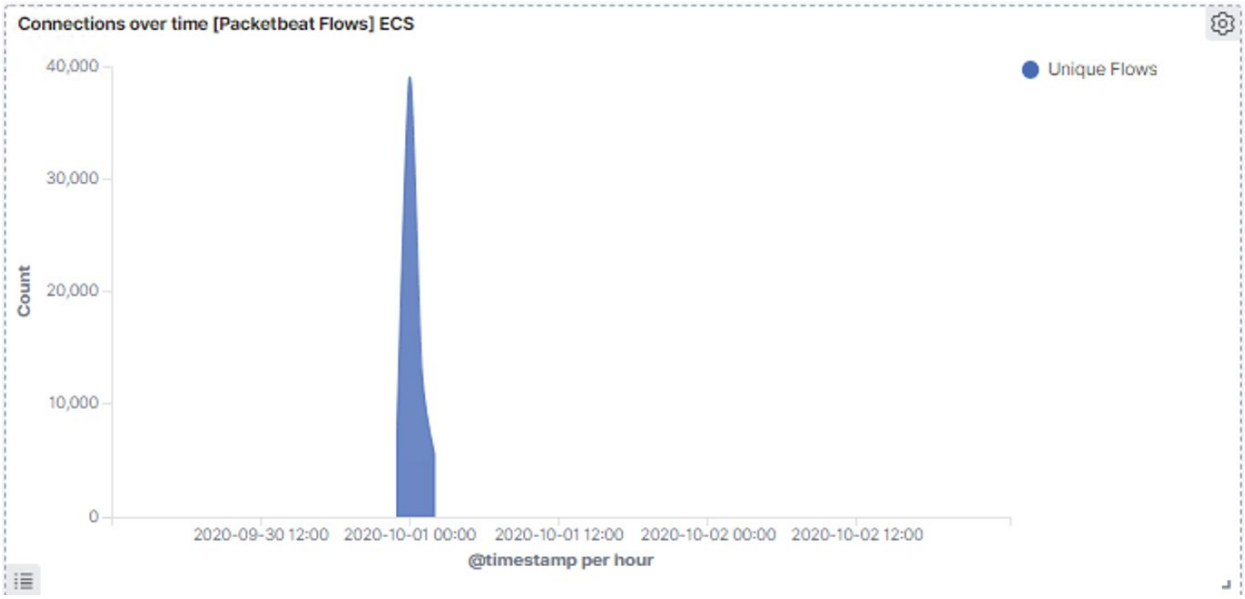
Defense

Identification

Setting up the SIEM dashboard, things to look for on our potential attack would be HTTP status codes, HTTP requests, Top Hosts, and connections over time.

Immediately we can see signs of an attack from our HTTP Status Codes for Top Queries visualization - a very large number of requests ended in a 401 error.

Other notable things to find is that the connection spikes over time



Looking at the top requests we can also see the brute force attack signs:

Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder/	22,148
http://127.0.0.1/server-status?auto=	1,206
http://snnmnkxdhflwgthqismb.com/post.php	196
http://www.gstatic.com/generate_204	98
http://192.168.1.105/	57

server.ip	192.168.1.105
# server.port	80
# source.bytes	163B
source.ip	192.168.1.90
# source.port	42000
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)

Filtering results based on the url.path we find that this user agent was Hydra, our brute force tool.

Based on the data taken from the Kibana stack, we know the secret folder was the target of 22,000 login attempts, most of which were from a single source IP, always sending the same amount of bytes, and using a known brute force tool.

http://192.168.1.105/webdav	28
http://192.168.1.105/webdav/shell.php	24
http://192.168.1.105/webdav/passwd.dav	4
http://192.168.1.105/company_folders/secret_folder/connect_to_corp_server	3

Once inside, the attacker accessed the webdav folder and a file called shell.php

Based on these findings, we can conclude the attacker brute forced a login, uploaded a .php file and attained root access from there.

Prevention:

- Set an alert to trigger any time the secret_folder is accessed, or remove this from the server altogether.
- Set an alert for 401 errors with a reasonable threshold (15 per hour)

- Set an alert if user_agent.original includes “Hydra”
- Set a maximum login attempt number per IP per hour. Auto drop traffic from any requests from a login that exceeds that number.
- Since the webdav folder should only be accessed from one machine, ensure that it only accepts traffic from that one machine.
- Connections to the shared folder should not be accessible via web browser.
- Set an alert for any .php file shared over the server.
- Remove ability to upload files over web browser.