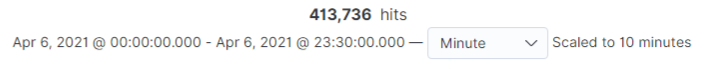
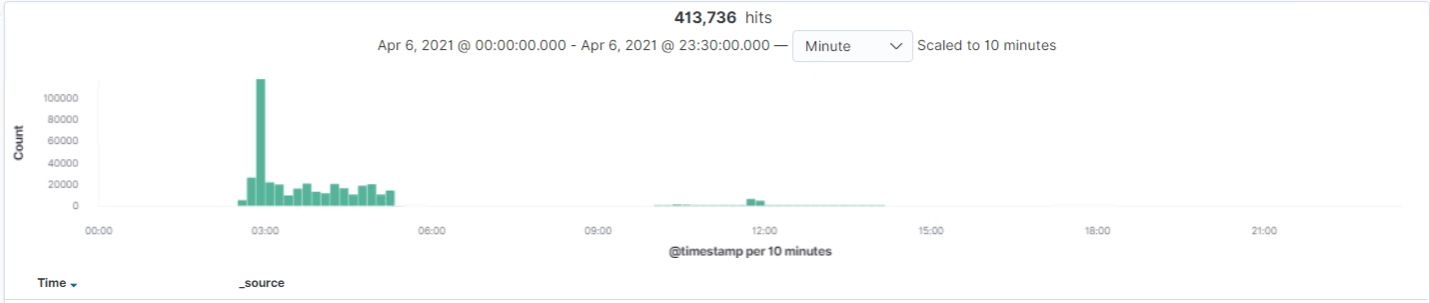
**Day 2 Activity File: Incident Analysis with Kibana**

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

1. Identify the offensive traffic.
   * Identify the traffic between your machine and the web machine:
     + When did the interaction occur?

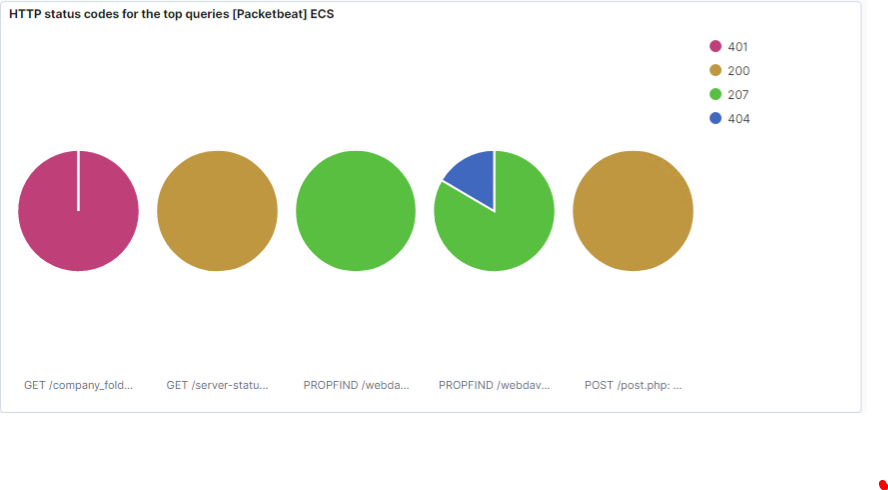
Starts





* + - What responses did the victim send back?

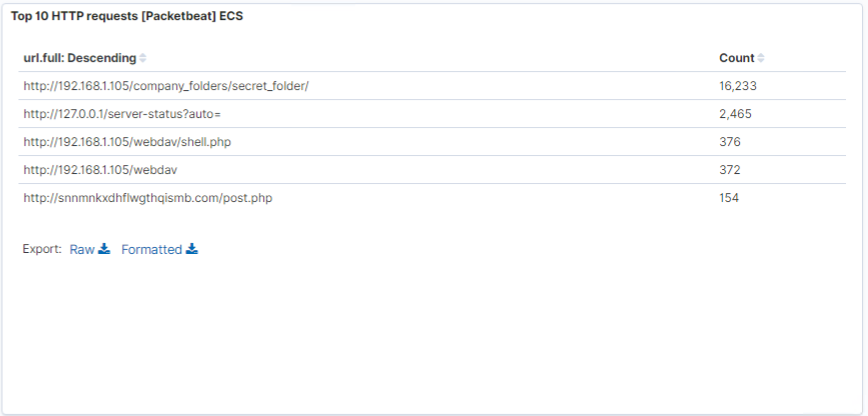
Response codes: 200, 207, 401, 404



* + - What data is concerning from the Blue Team perspective?

Blue team needs to be concerned with code 401; this status response code indicates that the request has not been applied because it lacks valid authentication credentials for the target resource.

1. 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)?



16,233 requests were made to the secret folder



From IP Address: 139.168.1.105

* + - Which files were requested? What information did they contain?

http://192.168.1.105/company\_folder/secret\_folder

* + - What kind of alarm would you set to detect this behavior in the future?

set alert, when //192.168.1.105/company\_folder/secret\_folder is being requested

* + - Identify at least one way to harden the vulnerable machine that would mitigate this attack.

Disable access to Apache folders

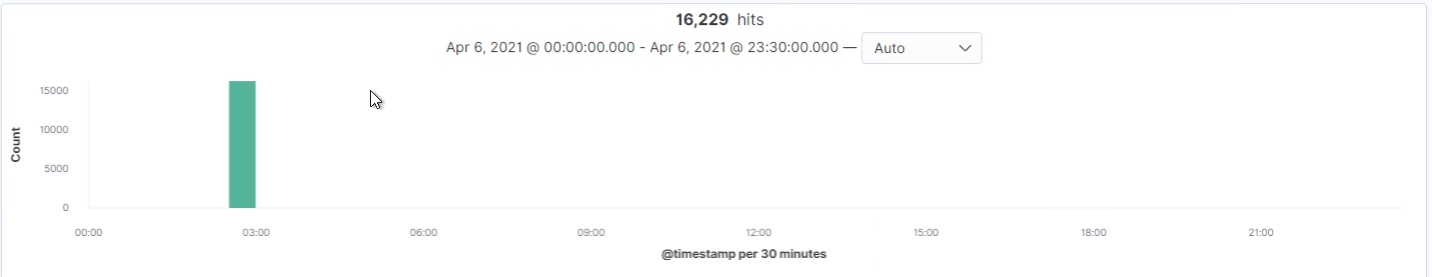
*Command* nano/etc/apache2/apache2.config

*Set* Options Indexes FollowSymLinks  to Options -Indexes + FollowSymLinks

1. 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?

yes

* + - How many requests were made in the brute-force attack?



* + - How many requests had the attacker made before discovering the correct password in this one?

16229 requests were made

* + - What kind of alarm would you set to detect this behavior in the future and at what threshold(s)?

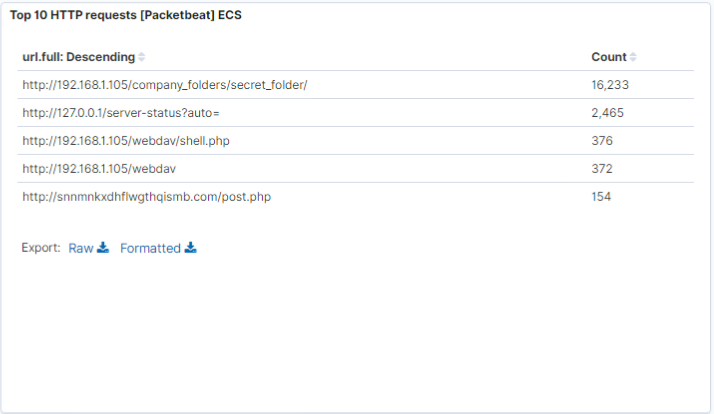
Set Alert if encounter a threshold of login 10 failures within 1 minutes from the same IP address

* + - Identify at least one way to harden the vulnerable machine that would mitigate this attack.

Stronger password policy. For example: Password make up at least 10 character long with upper case letter, lower case letter, numbers and special characters.

Use PBKDF2-SHA256 Hashing

1. Find the WebDav connection.
   * Use your dashboard to answer the following questions:
     + How many requests were made to this directory? 372



* + - Which file(s) were requested?

//192.168.1.105/webdav

* + - What kind of alarm would you set to detect such access in the future?

Trigger alarm when attempt to read //192.168.1.105/webdav

* + - Identify at least one way to harden the vulnerable machine that would mitigate this attack.

Disable access to Apache folders

*Command* nano/etc/apache2/apache2.config

*Set* Options Indexes FollowSymLinks to Options -Indexes + FollowSymLinks

1. 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?

Yes, traffic of metrepreter session can be identified with the following search conditions:

Source IP is 192.168.1.105

POST

Response code of 200

Note: As for the shell session, was not possible to pinpoint the pacific packet.

* + - What kinds of alarms would you set to detect this behavior in the future?

Shell session packets is difficult to detect, setting detection is challenging. There are other requested with a POST and response code of 200.

* + - Identify at least one way to harden the vulnerable machine that would mitigate this attack.

Strong password policy is a must to ensure credential compromise are under the control

Limited login failures for user is a must to prevent brute force attacks.

Follows best practice, using OWASP Top 10 as best practice.