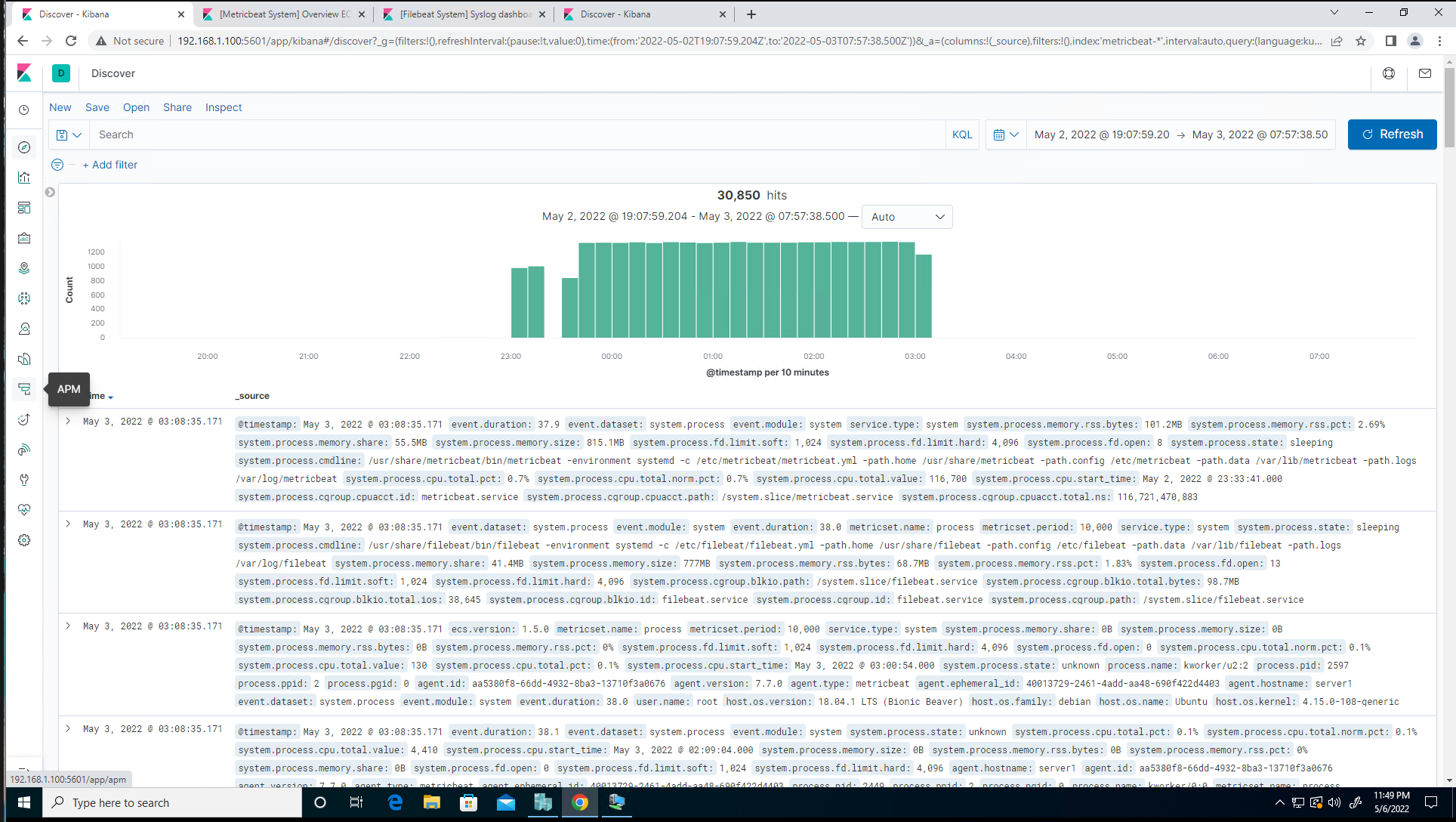
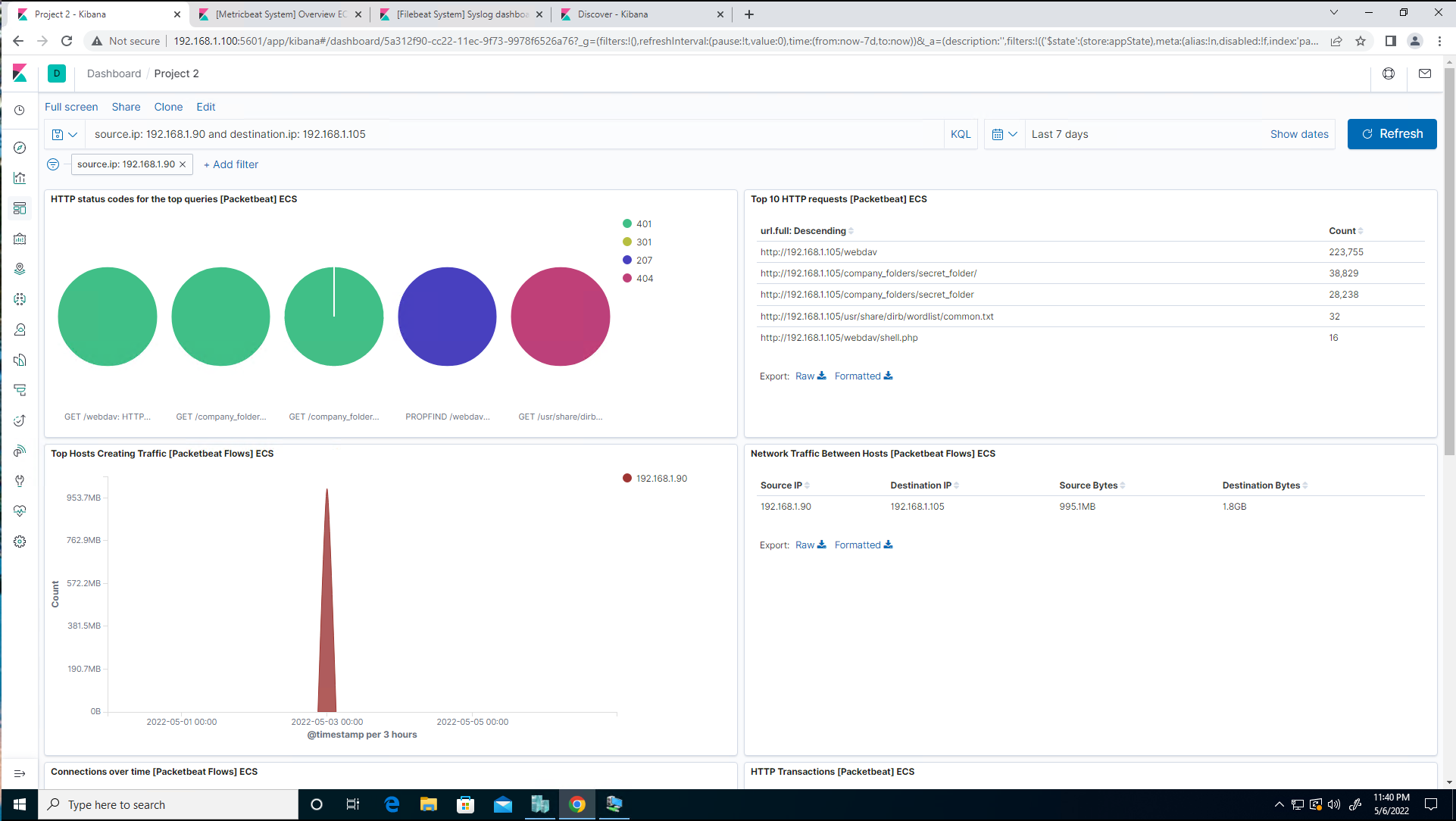
**Week 20 - Activity 2 - Incident Analysis with Kibana**

1. Identify the offensive traffic.
   1. Identify the traffic between your machine and the web machine:
      1. When did the interaction occur?
         1. For all intensive purposes the attack started on **May 2 2022 @ 23:00 and continued through May 5 2022 @ 03:00**

****

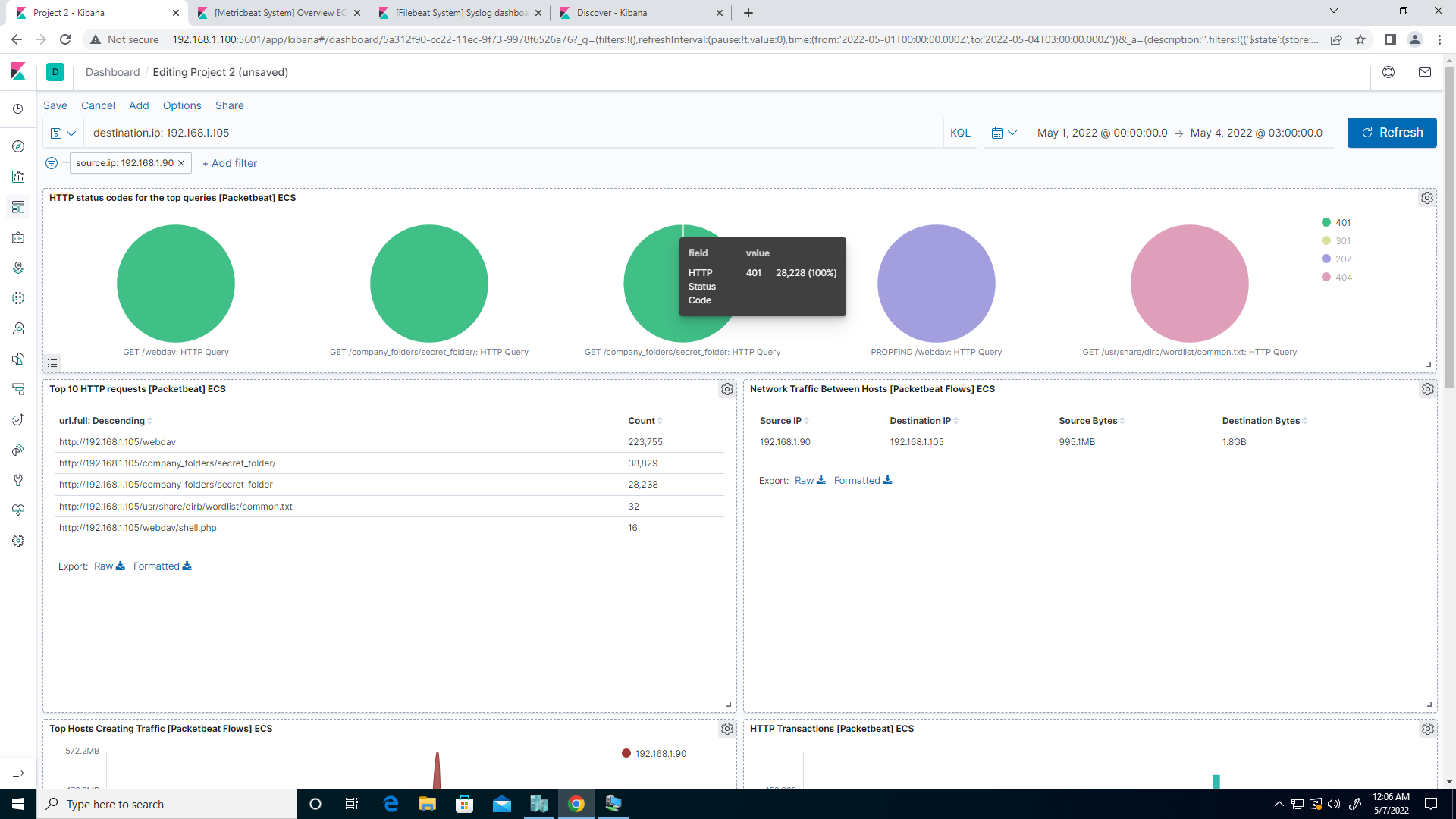
* + 1. What responses did the victim send back?

**http.response.status\_code - 401, 301, 207 and 404**

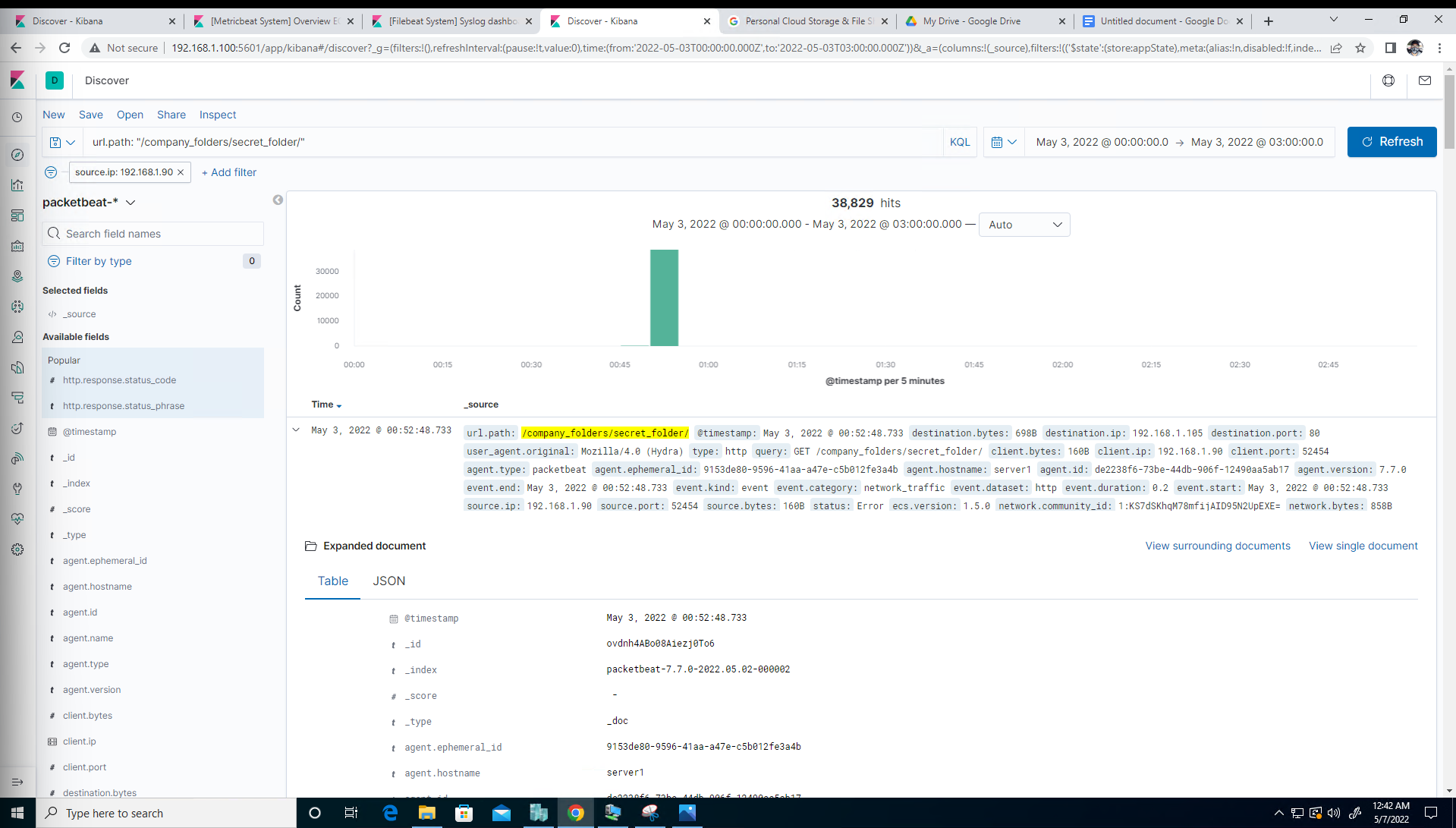
****

* + 1. What data is concerning from the Blue Team perspective?

**There were 28,228 (401) responses indicating that someone was failing to authenticate to the server.**

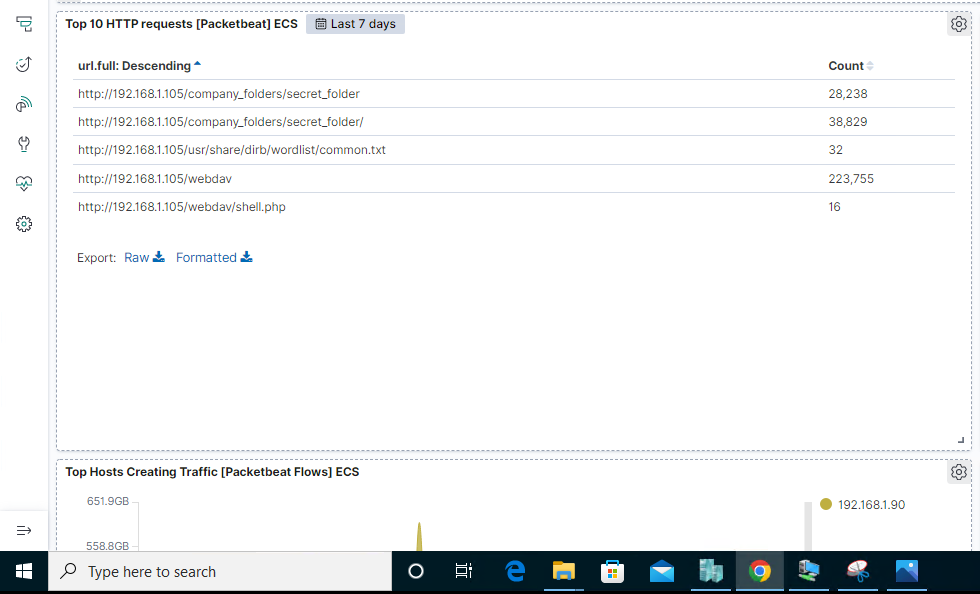
****

1. Find the request for the hidden directory.
   1. In your attack, you found a secret folder. Let's look at that interaction between these two machines.
      1. How many requests were made to this directory? At what time and from which IP address(es)? **38,829 hits to this url May 3, 2022 @ 00:00 from 192.168.1.90.**

****

* + 1. Which files were requested? What information did they contain?

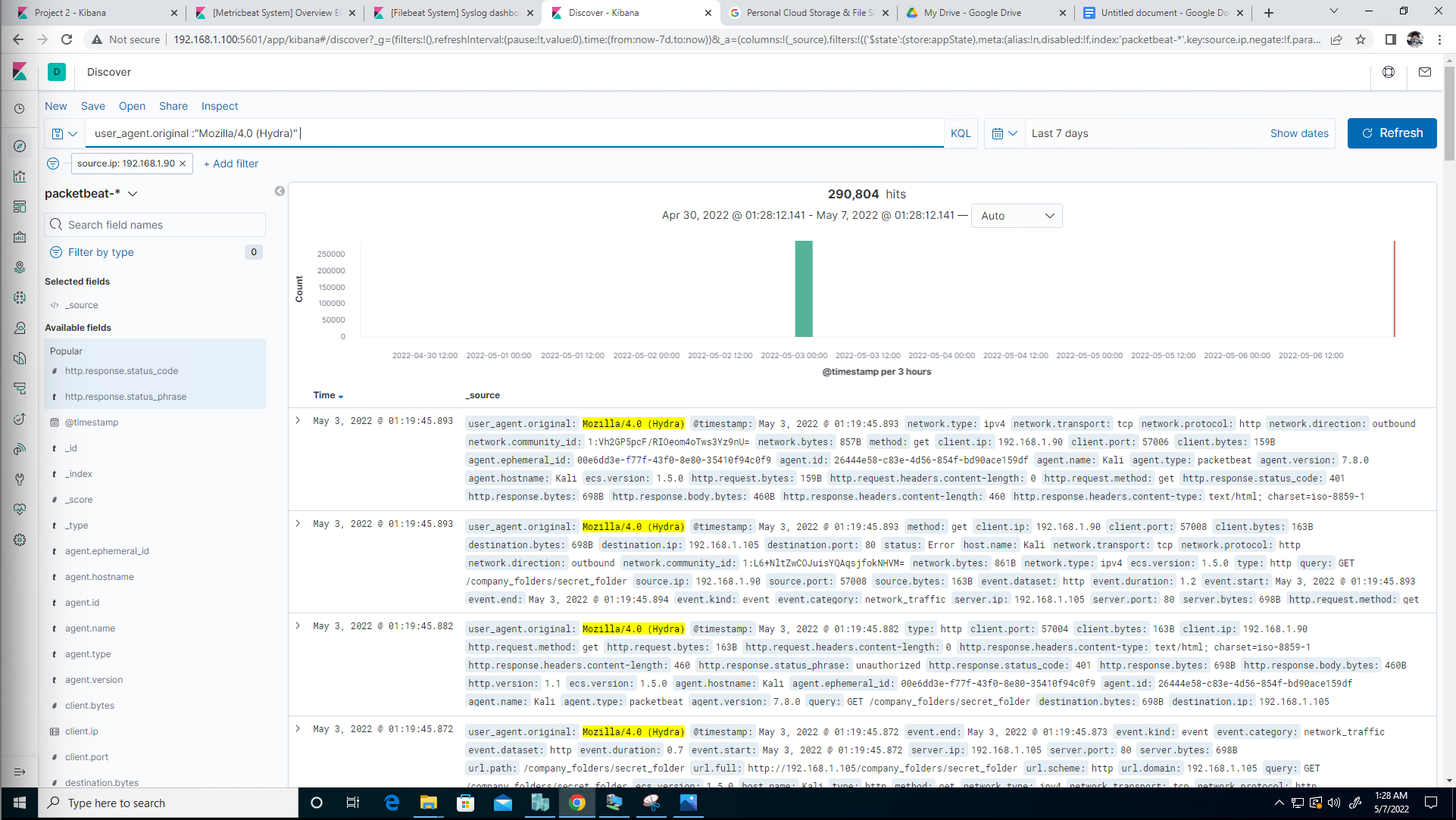
[**http://**](http://snnmnkxdhflwgthqismb.com/post.php)**192.168.1.105/company\_folders/secret\_folder/connect\_to\_corp\_server. However, I couldn't pull up what was shown in the solution file. See screenshot. This was supposed to contain instructions on connecting to corp\_server.**



* + 1. What kind of alarm would you set to detect this behavior in the future?
       1. **Set an alert for a brute force attack after monitoring a baseline.**
       2. **Set an alert whenever this file is accessed to monitor it.**
    2. Identify at least one way to harden the vulnerable machine that would mitigate this attack.
       1. **Disable password authentication and authenticate via ssh key pair to allow only administrator machines to access.**
       2. **Apply an account lock-out feature after so many attempts.**
       3. **Enable MFA to prevent brute forcing attacks**
       4. **Require stronger password policy for administrators**

1. Identify the brute force attack.
   1. After identifying the hidden directory, you used Hydra to brute-force the target server. Answer the following questions:
      1. Can you identify packets specifically from Hydra?

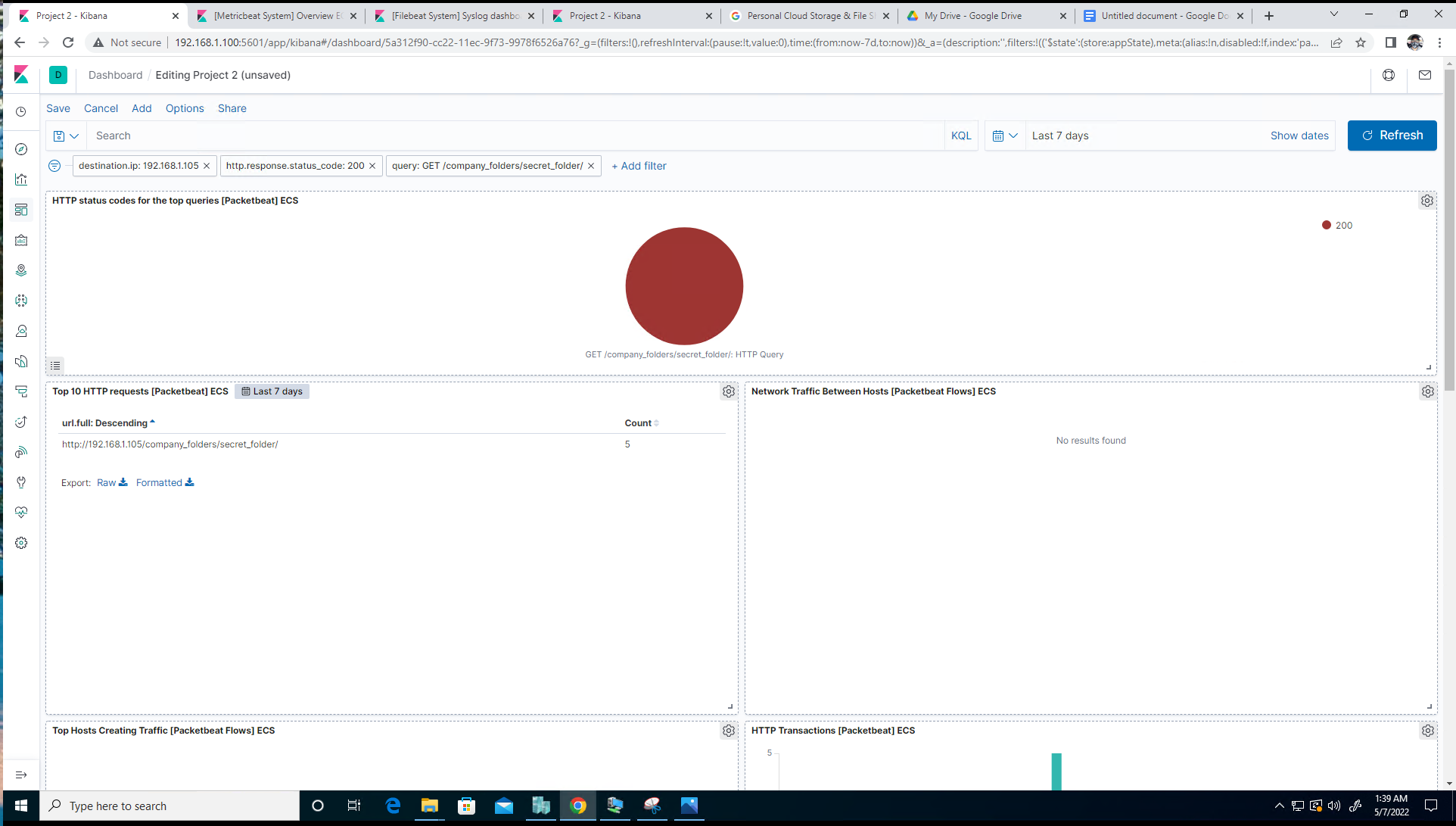
**Yes. user\_agent.original: “Mozilla/4.0 (Hydra)**

****

* + 1. How many requests were made in the brute-force attack?

**290,804**

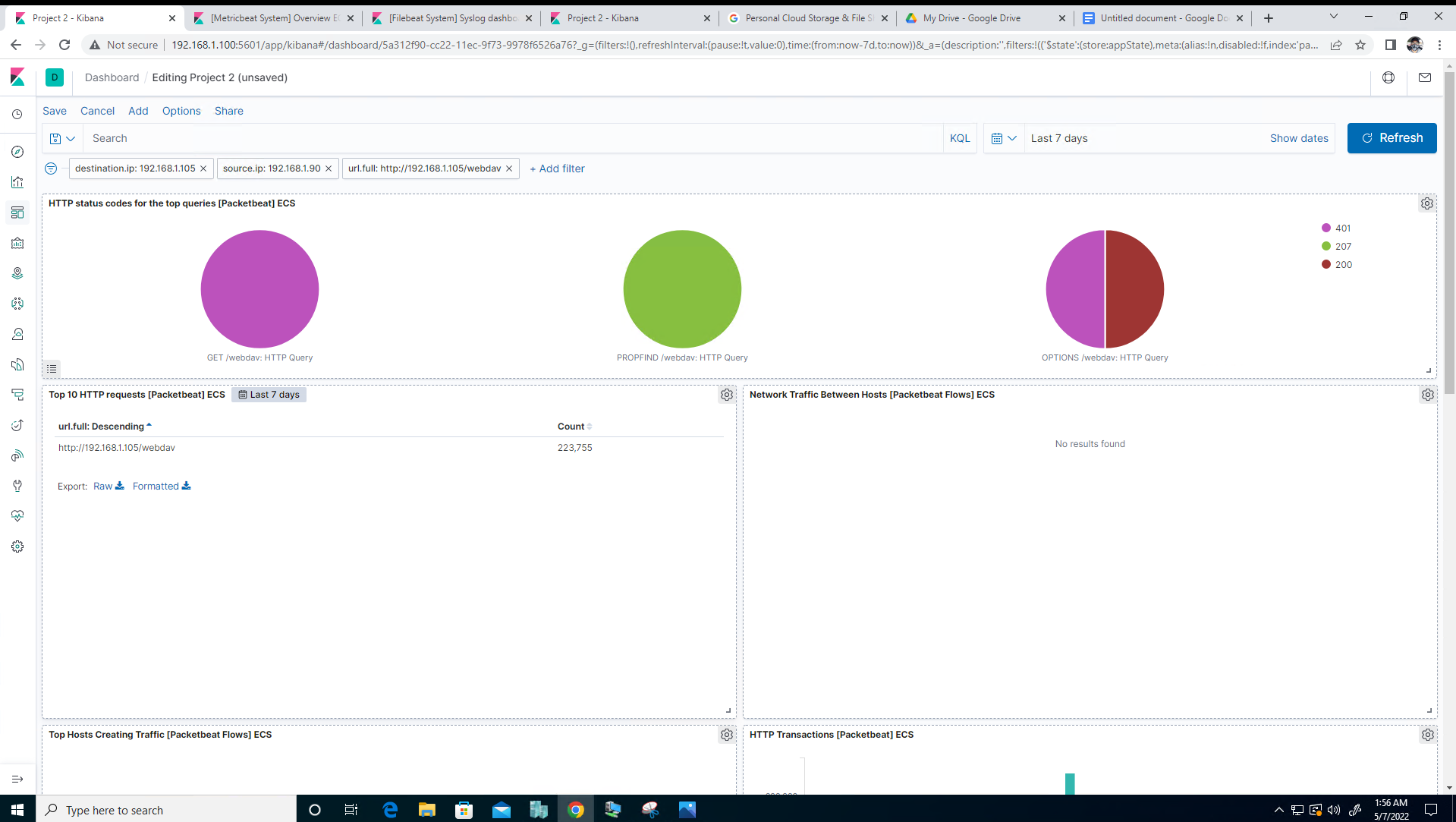
* + 1. How many requests had the attacker made after discovering the correct password in this one?
       1. **5**

****

* + 1. What kind of alarm would you set to detect this behavior in the future and at what threshold(s)?
       1. **I would recommend setting an alert based on a specific threshold of 5 HTTP-GET requests from the same ip address to the same resource that generates a 401 status code.**
       2. **Also, setting an alert when the User-Agent includes the term “Hydra.”**
    2. Identify at least one way to harden the vulnerable machine that would mitigate this attack.
       1. **Implement account lock-out after 5 failed attempts at accessing the webserver.**

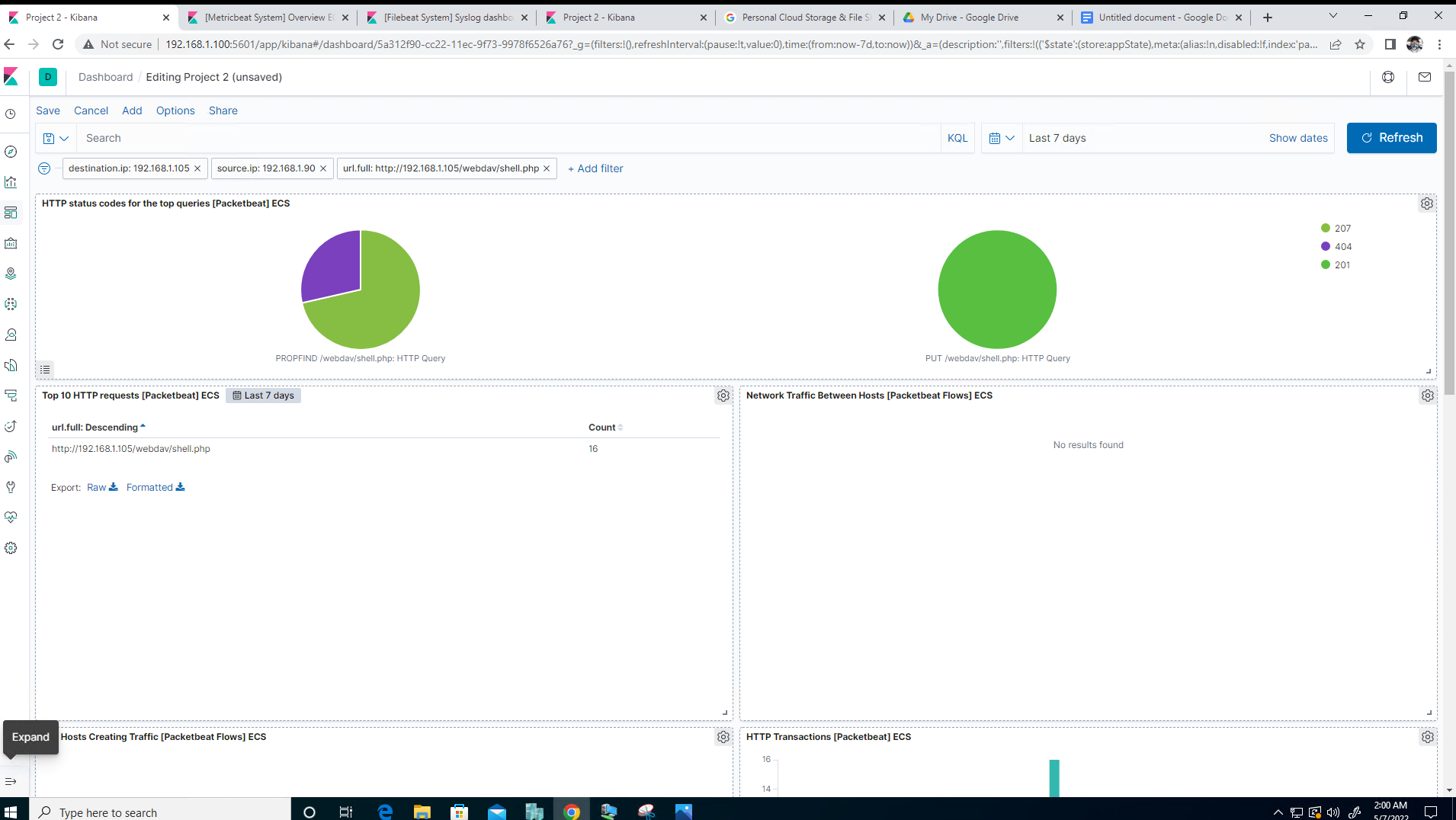
1. Find the WebDav connection.
   1. Use your dashboard to answer the following questions:
      1. How many requests were made to this directory?

**223,755**



* + 1. Which file(s) were requested?

**webdav/shell.php**

****

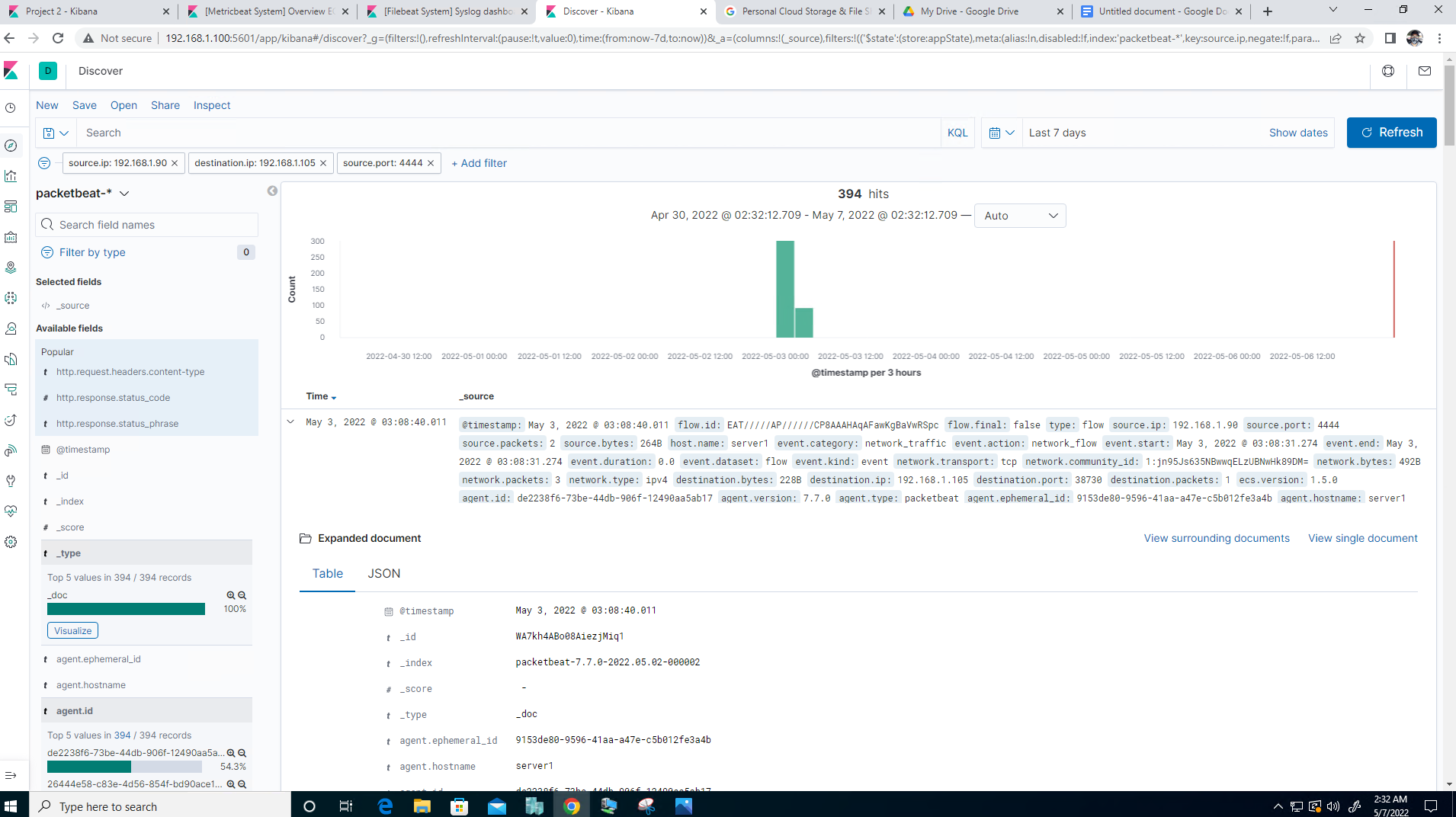
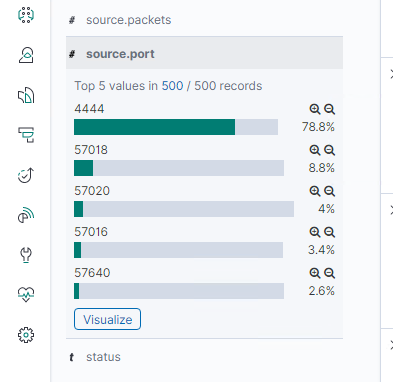
* + 1. What kind of alarm would you set to detect such access in the future?

**Alert if anyone other than those whitelisted access this folder.**

* + 1. Identify at least one way to harden the vulnerable machine that would mitigate this attack.
       1. **File code/signature scan if files are uploaded.**
       2. **Make access to this folder only accessible from the back end.**
       3. **Implement a firewall rule to make this folder accessible by those whitelisted.**

1. Identify the reverse shell and meterpreter traffic.
   1. To finish off the attack, you uploaded a PHP reverse shell and started a meterpreter shell session. Answer the following questions:
      1. Can you identify traffic from the meterpreter session?

**Yes. Searching source port under “Available fields” I see that 4444 was at the top of the list and had a 78.8% usage. Then filtered by this to investigate it further.**

****

* + 1. What kinds of alarms would you set to detect this behavior in the future?
       1. **An alarm if this port is used**
       2. **An alarm if an executable file is uploaded**
    2. Identify at least one way to harden the vulnerable machine that would mitigate this attack.
       1. **Implement firewall rule to block this port number, ip address, and executable file type.**
       2. **File scanning software to scan for malicious code.**
       3. **Remove the ability to upload files if this is possible.**
       4. **Do not allow outbound traffic from unknown ports.**

**Commands Used:**

**source.ip: 192.168.1.90 and destination.ip: 192.168.1.105**

**url.path: /company\_folders/secret\_folder/**

**url.path: /webdav**

**user\_agent.original: “Mozilla/4.0 (Hydra)**

**http.response.status\_code: 401**