NETWORK SECURITY OPERATIONS LAB 2

In this lab i analyze real world PCAP file using Wireshark to uncover the followings!

- Identifying the Geographical Origin of the Attack:

From which city did the attack originate

The attacker's IP address is identified as **117.11.88.124**

City: Tianjin, China as check from maxmind.com

IP Address	Location	Network	Postal Code	Approximate Latitude / Longitude*, and Accuracy Radius	ISP / Organization	Domain	Connection Type
117.11.88.124	Tianjin, Tianjin, China (CN), Asia	117.11.80.0/20	-	39.1424, 117.1727 (10 km)	China Unicom	online.tj.cn	Cable/DSL

- Determining the Attacker's User-Agent:

```
GET / HTTP/1.1
Host: shoporoma.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
```

User-Agent String**:** Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0.

The attacker is using FIREFOX browser and LINUX Operating System

- Identifying the Malicious Web Shell:

The name of the malicious web shell that was successfully uploaded

```
GET /reviews/uploads/image.jpg.php HTTP/1.1
Host: shoporoma.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://shoporoma.com/reviews/uploads/
Upgrade-Insecure-Requests: 1
```

IMAGE.JPG.PHP is the name of the file which a php file and it might consist of scripts that can be executed on the website.

- Discovering the Directory Used for File Uploads:

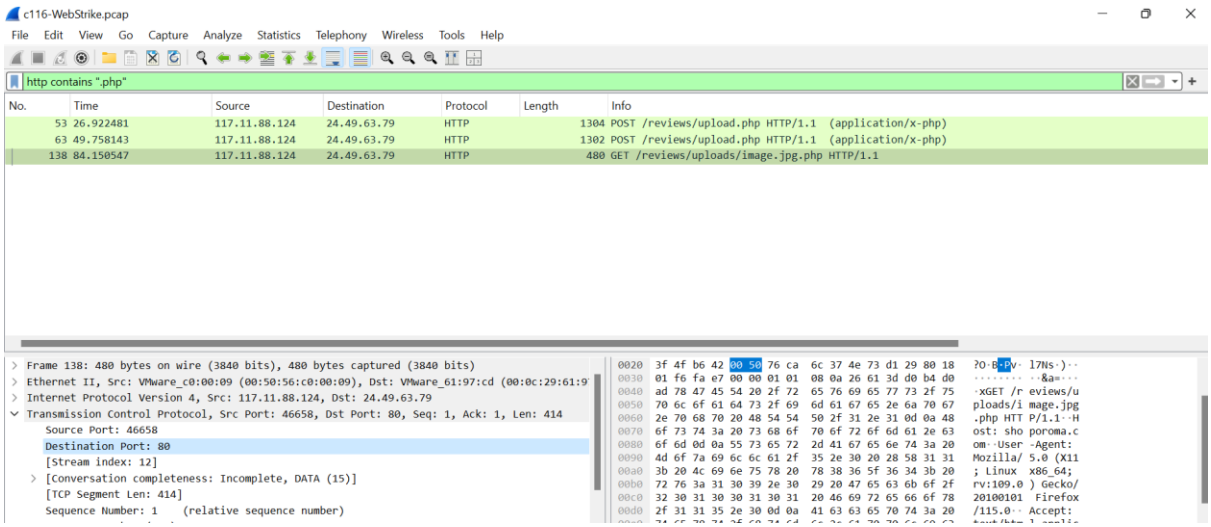
The directory on the server that was used to store uploaded files

```
GET /reviews/uploads/image.jpg.php HTTP/1.1
Host: shoporoma.com
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://shoporoma.com/reviews/uploads/
Upgrade-Insecure-Requests: 1
```

The directory is /reviews/uploads/ extracted from the website
http://shoporoma.com/reviews/uploads/

- Determining the Port Used for Outbound Communication

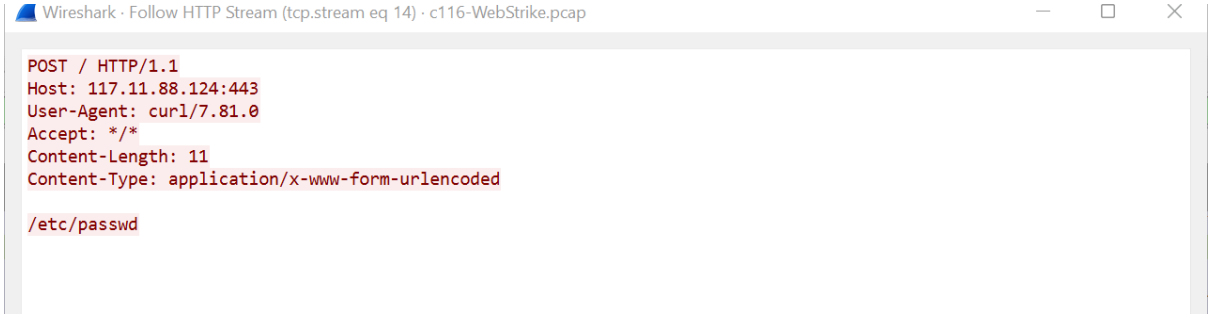
The outbound port that the malicious web shell was use to contact the attacker's machine



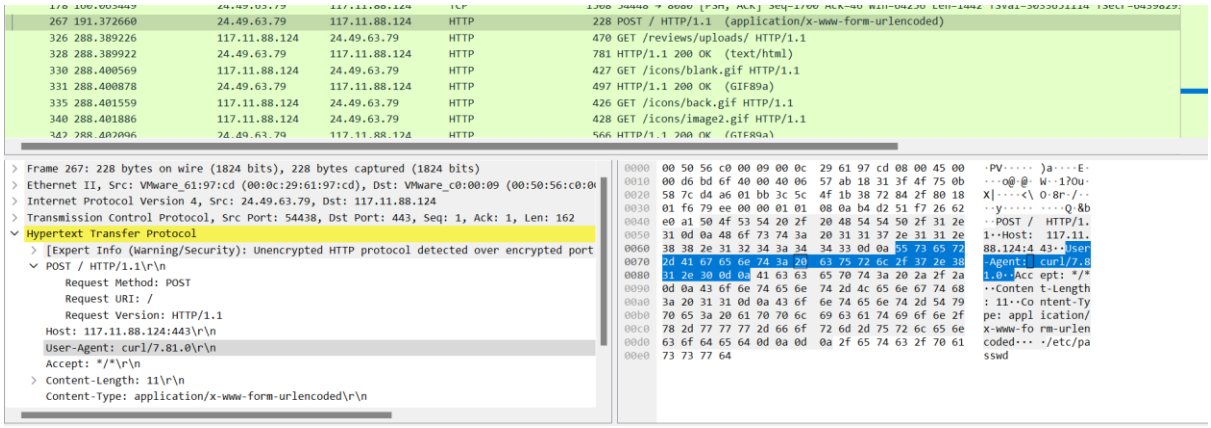
The Port is**:** 80 which is the default port for http

- Identifying the File Targeted for Exfiltration:

The file the attacker attempt to extract from the server



The file targeted is: /etc/passwd and the attacker executed the command cat /etc/passwd to display the contents of the system's password file. They then attempted to exfiltrate this specific data using a curl POST request as shown from the screenshot below



Attack Summary

The investigation reveals a targeted web-based attack originating from Tianjin, China with an IP address: 117.11.88.124. The attacker used a Linux-based system running Firefox browser to exploit a file upload vulnerability on the shoporoma.com website. By uploading a malicious PHP script disguised as an image, the attacker gained the ability to execute commands on the server and attempted to extract sensitive system configuration files.

Attack Timeline

The incident unfolded in several distinct stages:

- Initial Connection: The attacker established contact from the Tianjin region using a standard web browser.
- Web Shell Upload: The attacker successfully uploaded a malicious file named image.jpg.php to the server.
- Establishing Persistence: The file was stored in the /reviews/uploads/ directory, allowing the attacker a permanent access to run scripts.
- Command Execution: The attacker used the web shell to contact their home machine via Port 80.
- Attempted Exfiltration: Using a curl command, the attacker tried to extract the /etc/passwd file, a system file containing user account informations and send it back to their own server.

Affected System

- Primary Target: The web server hosting shoporoma.com.
- Directory Compromised: The /reviews/uploads/ folder was used as a staging area for the attack.
- Data at Risk: The system's password file (/etc/passwd) was targeted for theft. While this file doesn't usually contain plain-text passwords in modern Linux, it provides a "map" of all user accounts on the system, which is a major stepping stone for deeper hacking.

Security Recommendations

1. Tighten File Upload Rules: Never trust a file just because it has an image extension like .jpg. The server should be configured to strictly block any file containing .php or other executable code from being uploaded.
2. Move Uploads Out of Reach: Store uploaded files in a directory that is not allowed to execute scripts. If a file can’t "run," then it can’t do any damage.
3. Rename Uploaded Files policy: Automatically rename every file a user uploads to a random string of numbers or letters. By doing this makes it much harder for an attacker to find and "trigger" any malicious script.
4. Limit Outbound Traffic: Servers should generally not be allowed to initiate new connections to random IP addresses on the internet. Blocking unauthorized "outbound" requests would have stopped the curl command from sending the stolen data.