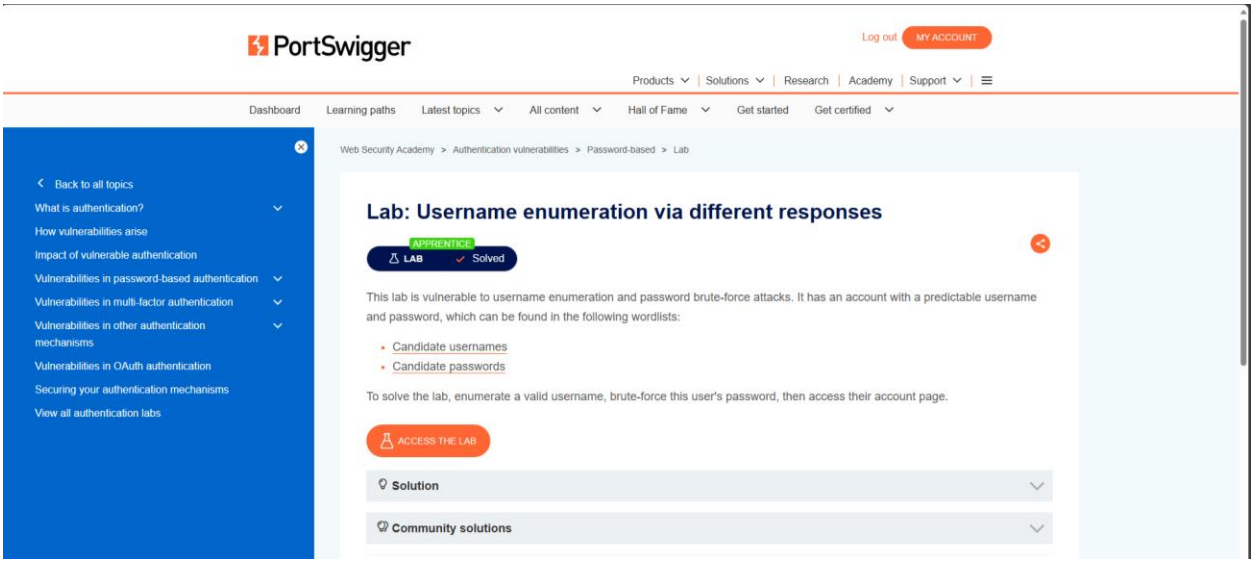
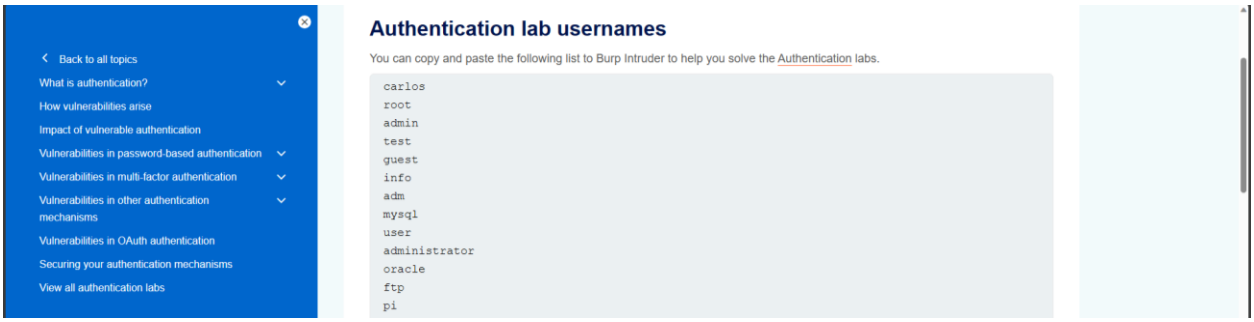
Lab 3: Authentication Vulnerabilities (password-based)	
Name	Dang Hoang Nguyn
Student ID	SE171946

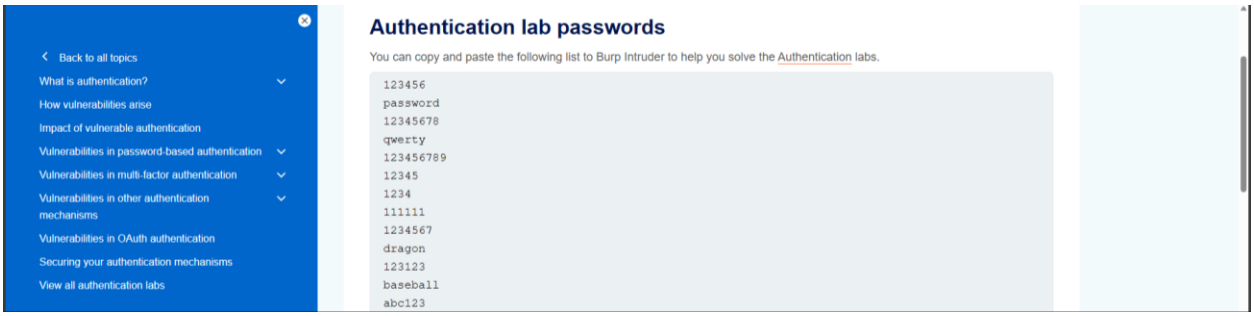
Giới Thiêu: Trong bài lab này ta sẽ tìm hiểu về lỗ hổng Authentication Vulnerabilities (password-based)

I. Username enumeration via different responses <Here>

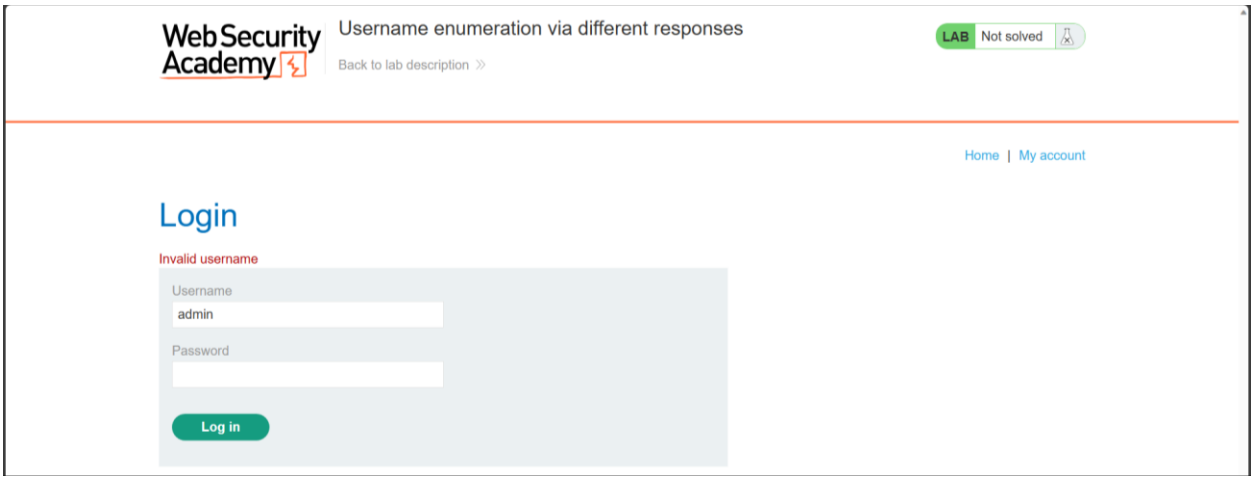


- ⇨ Mục tiêu của bài lab này là tìm được username và password để login vào được tài khoản đó
- ⇨ Theo bài lab thì ta sẽ sử dụng wordlists username và password mà PortSwigger cung cấp: wordlist usernames <link>, wordlist passwords <link>





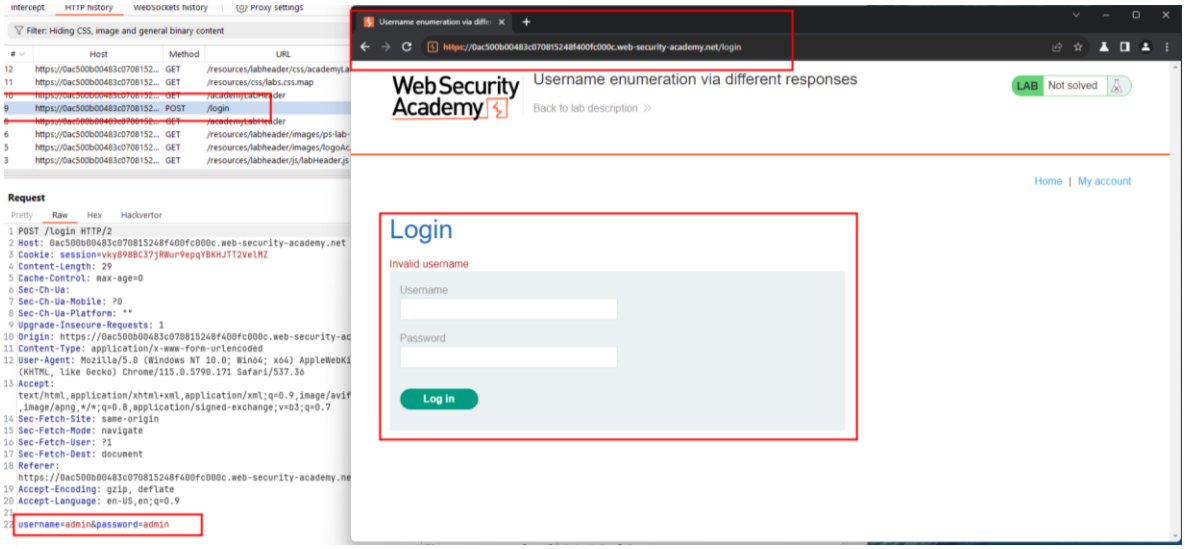
1. Đầu tiên ta ta sẽ thử truy cập vào link bài lab và thử đăng nhập bằng một tài khoản bất kì



⇨ Có thể thấy được là khi nhập thử tài khoản admin:admin thì có xuất hiện lỗi "**Invalid username**" => ta có thể đặt giả thiết rằng có phải ta nhập đúng tên username thì nó sẽ không báo lỗi không.

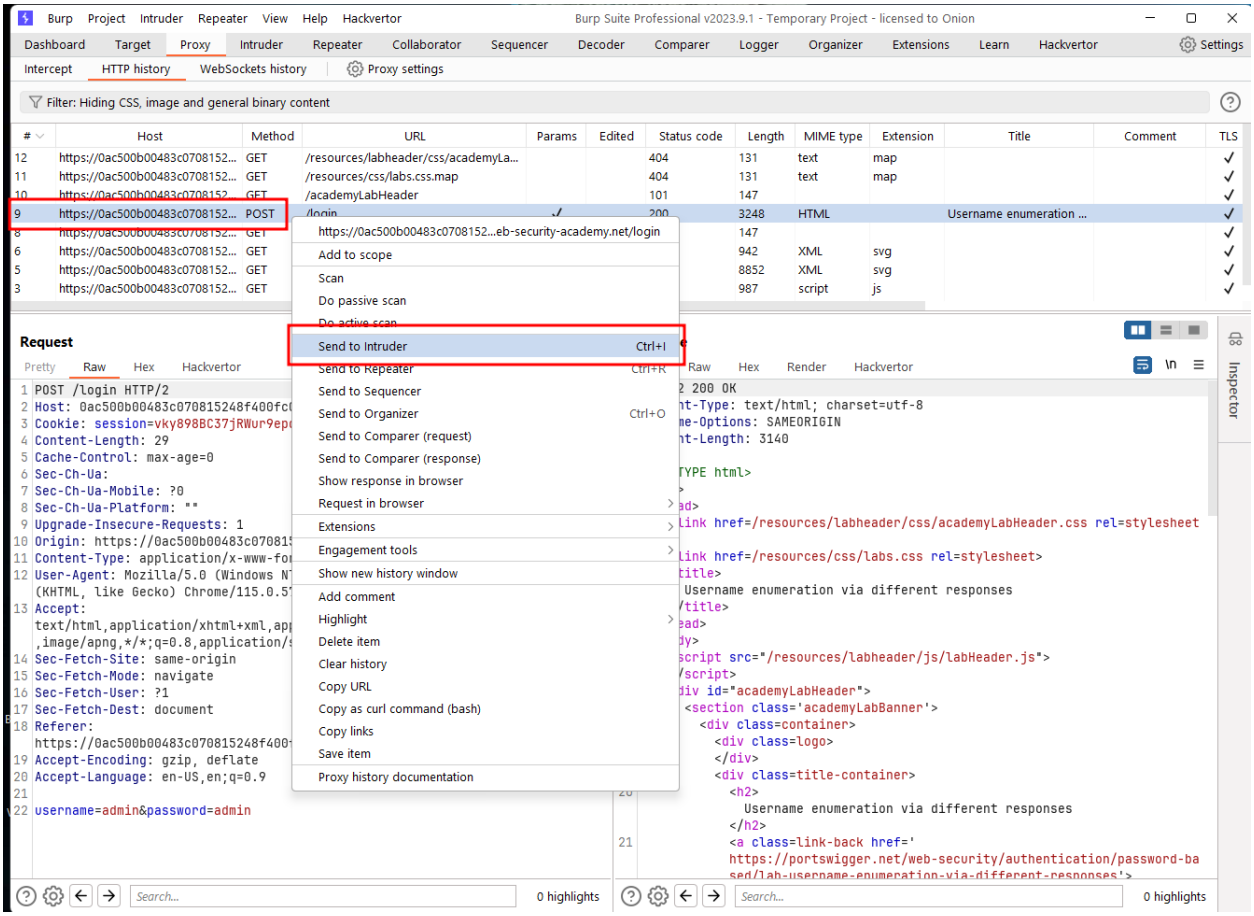
2. Giờ ta sẽ kiểm chứng giả thiết ở trên bằng tính năng **Intruder** công cụ **Burp Suite**.

   Mở **Burp Buite > Proxy > HTTP history > Open browser**. Sau khi browser mở lên thì nhập link bài lab vào để bắt gói tin sau khi gửi gói tin login

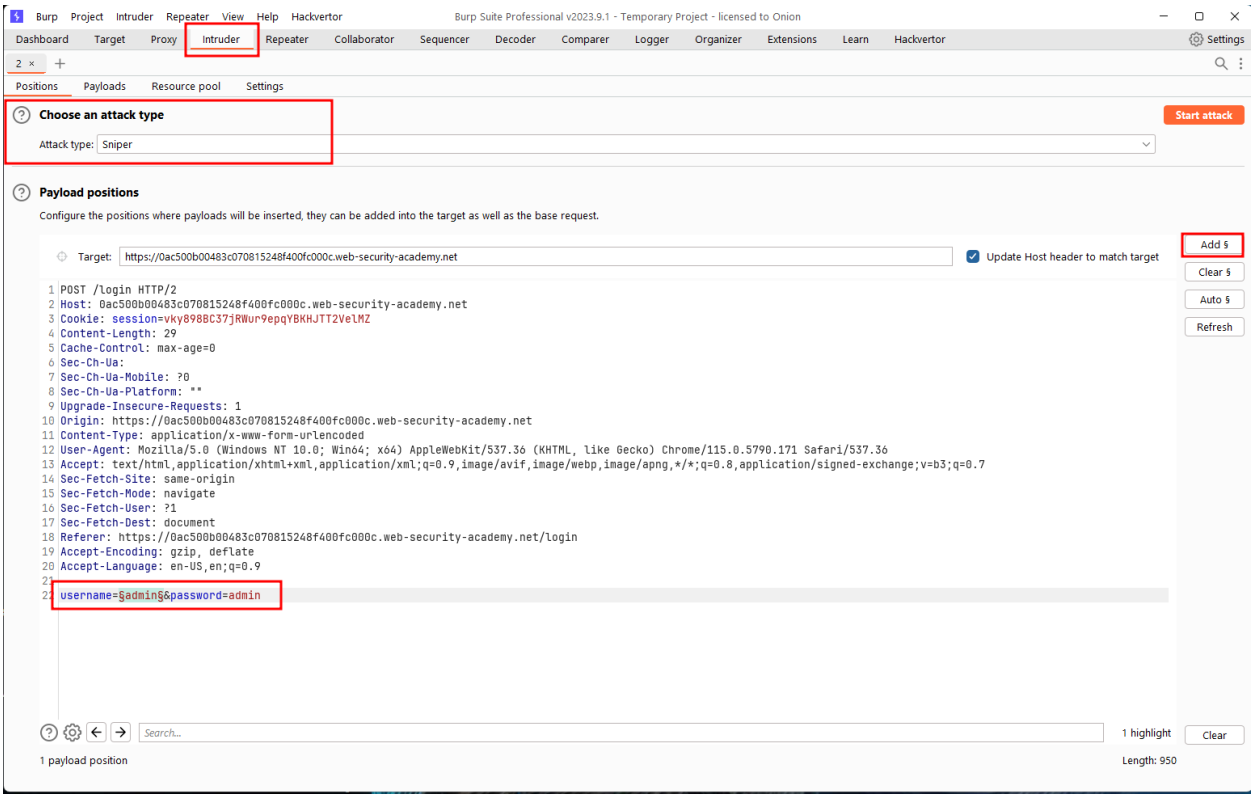


⇨ Có thể thấy là ta đã bắt được gói HTTP khi mà ta login vào

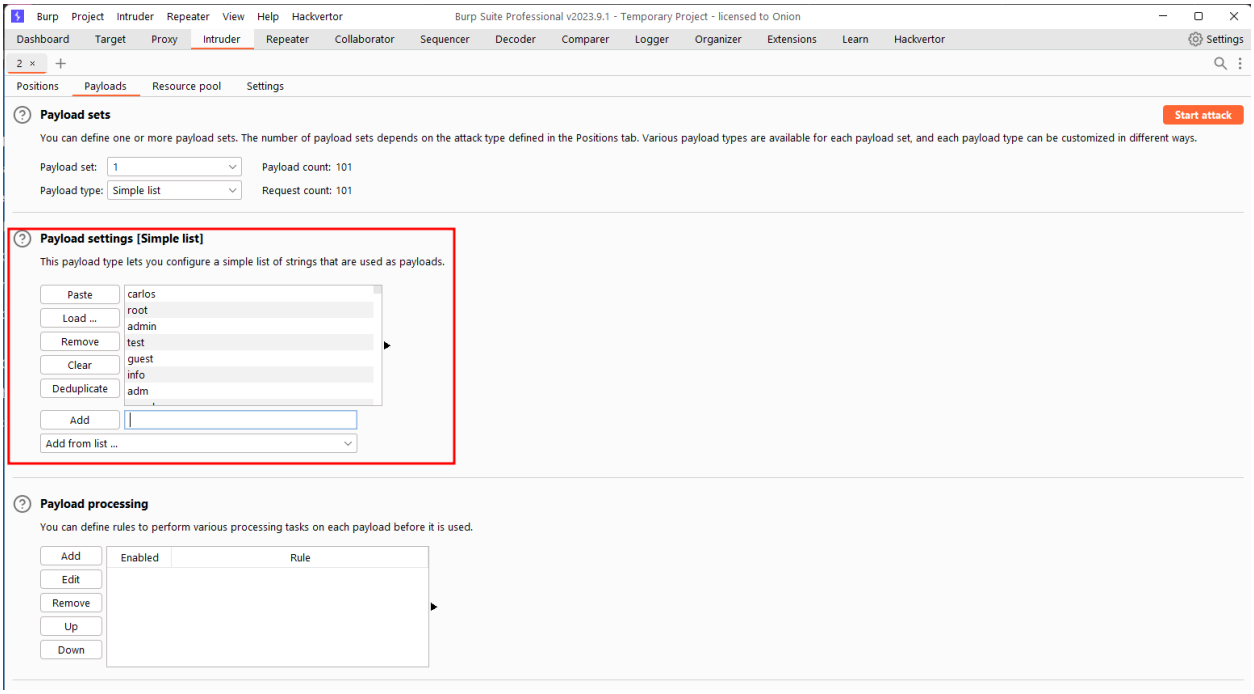
3. Tiếp theo ta sẽ gửi gói tin đó đến Intruder bằng cách **Right-click > Send to Intruder**



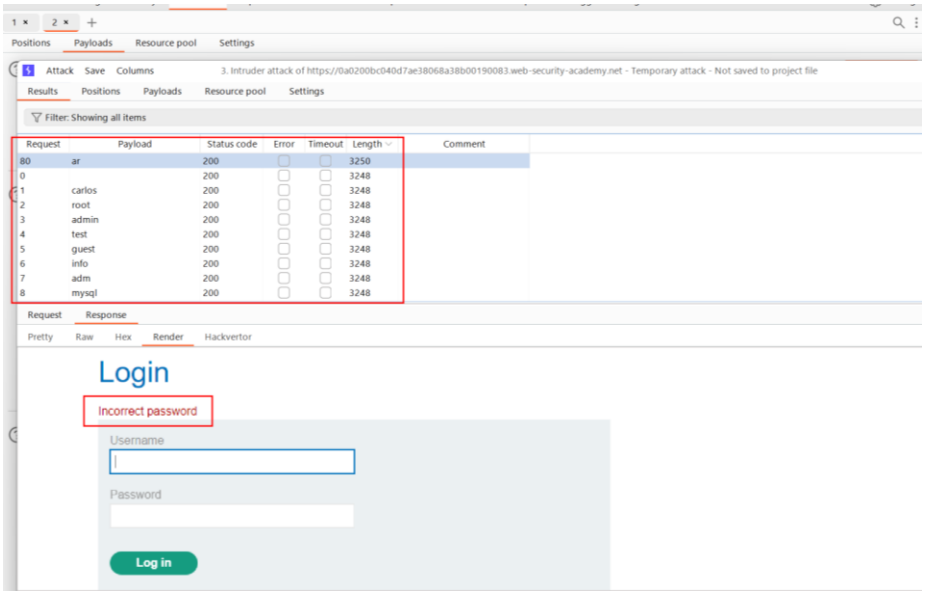
4. Tại tab Intruder, ở phần Positions ta sẽ chọn Attack type: Sniper, Payload positions thì ta sẽ them dấu **$** vào **admin**:



5. Tiếp theo đến phần Payloads, ta sẽ dán wordlist mà bài lab đã cho sẵn vào phần Payload settings.

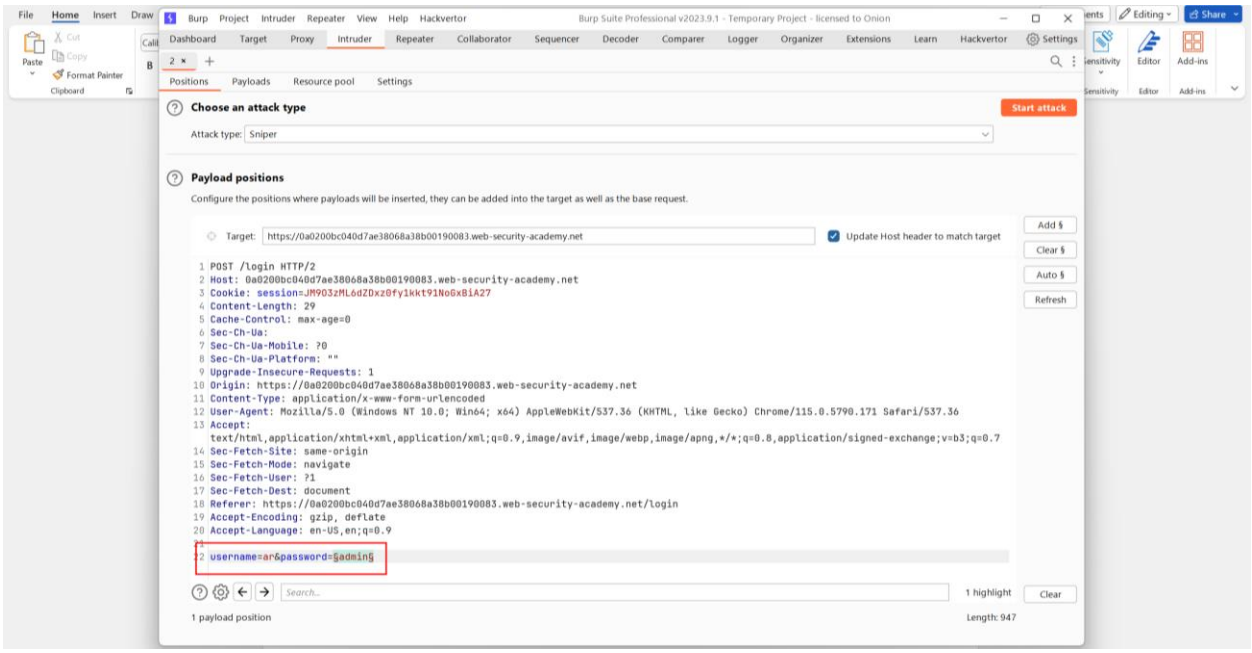


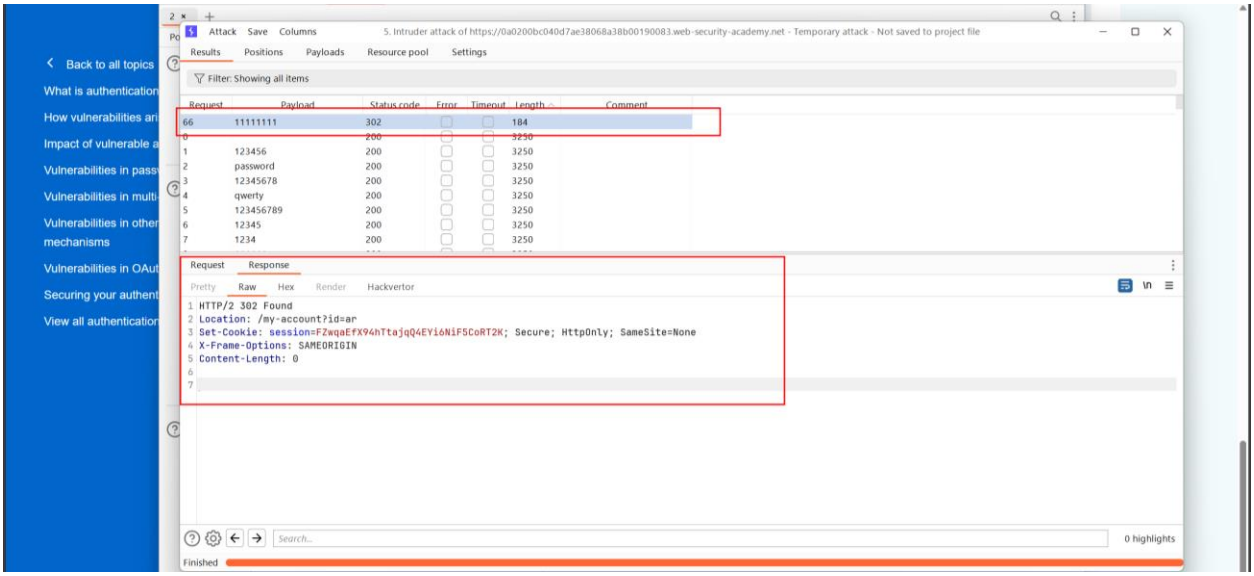
6. Sau chọn **Start attack.**



- ⇨ Có thể thấy được sau khi sắp xếp theo Length thì có thể thấy được có một request khác với các request còn lại. Sau khi render response thì có thể thấy được lỗi "**Incorrect password**"
- ⇨ Vậy có thể thấy được là giả thiết đặt ra đã đúng với username là "**ar**"

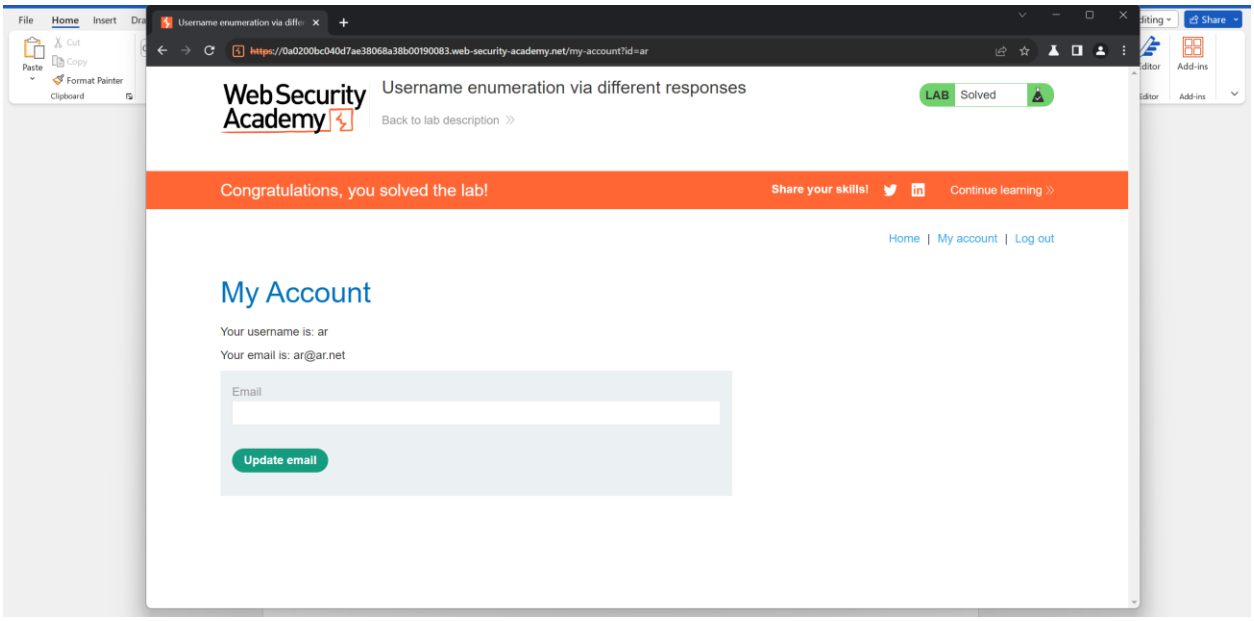
7. Làm tương tự với password, tuy nhiên ta thay username thành "**ar**"





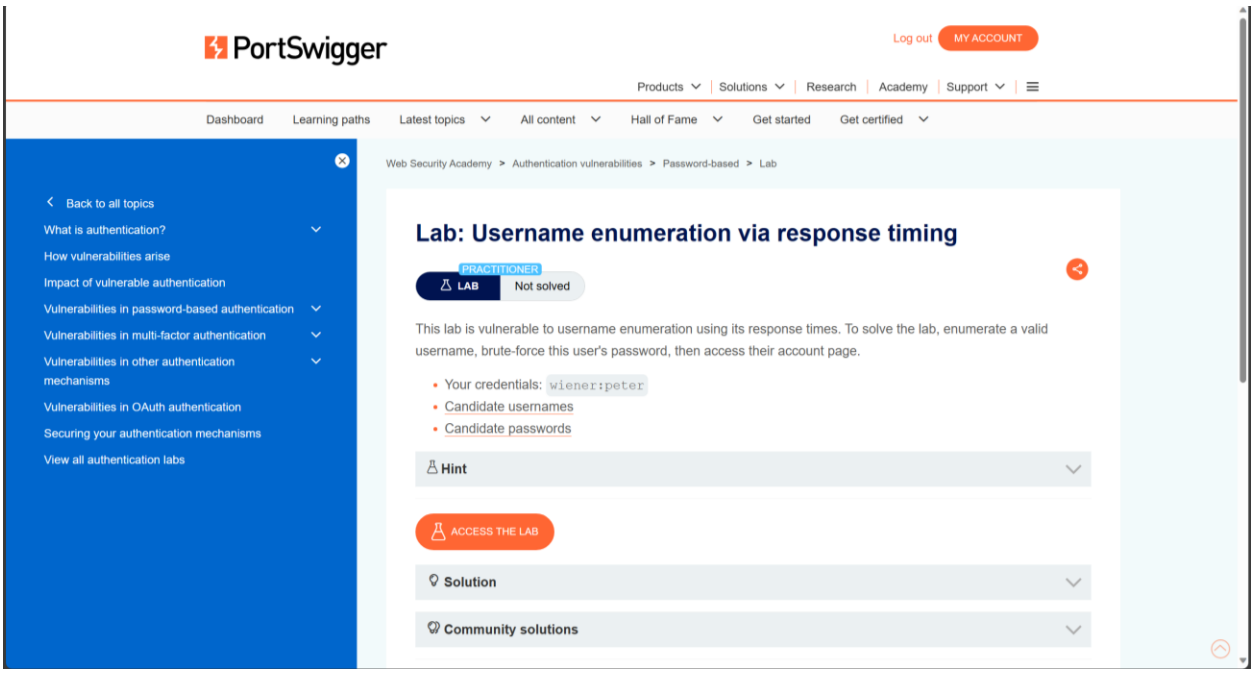
⇨ Có thể thấy được là với password là “**11111111**” thì ta sẽ nhận được response với status code là 302. Tức là ta login thành công thì nó sẽ redirect đến account page

8. Giờ ta sẽ thử login vào account với username là “**ar**” và password là “**11111111**”



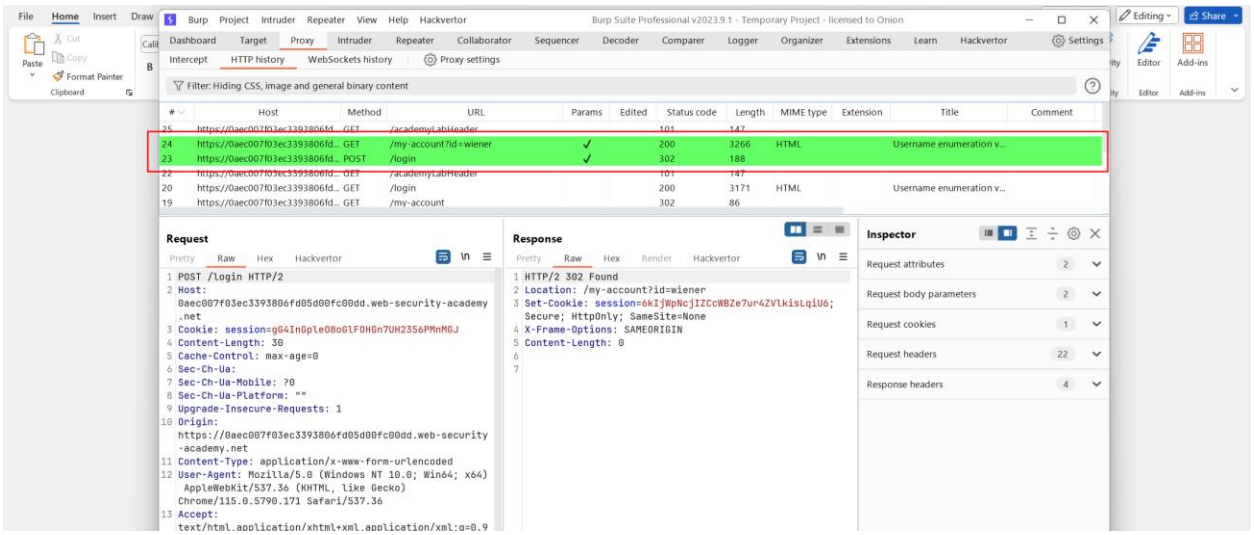
⇨ Có thể thấy được là ta đã login thành công

II. Username enumeration via response timing <Here>



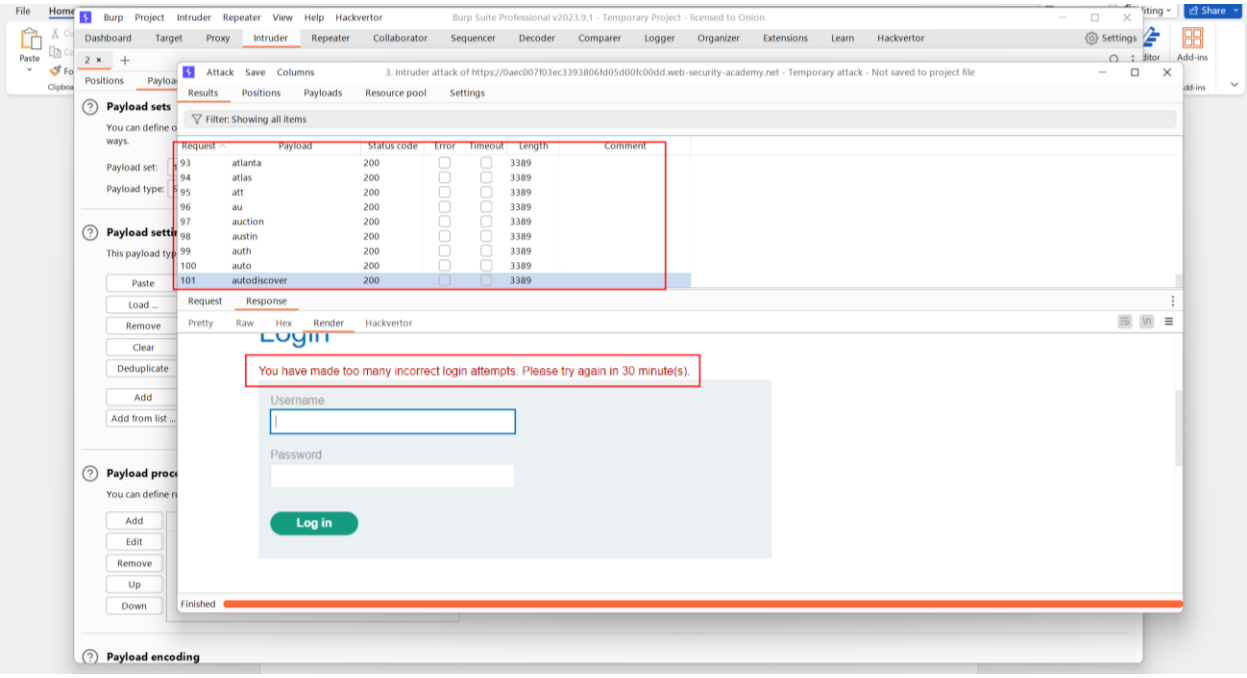
- ⇨ Để solve được bài lab này thì ta sẽ tìm được account hợp lệ đề login vào
- ⇨ Đề bài có cho ta một số thông tin như:
  - Account để thử: **wiener: peter**
  - Wordlist username: <link>
  - Wordlist password: <link>

1. Đầu tiên thử dùng account cho sẵn để login vào thử



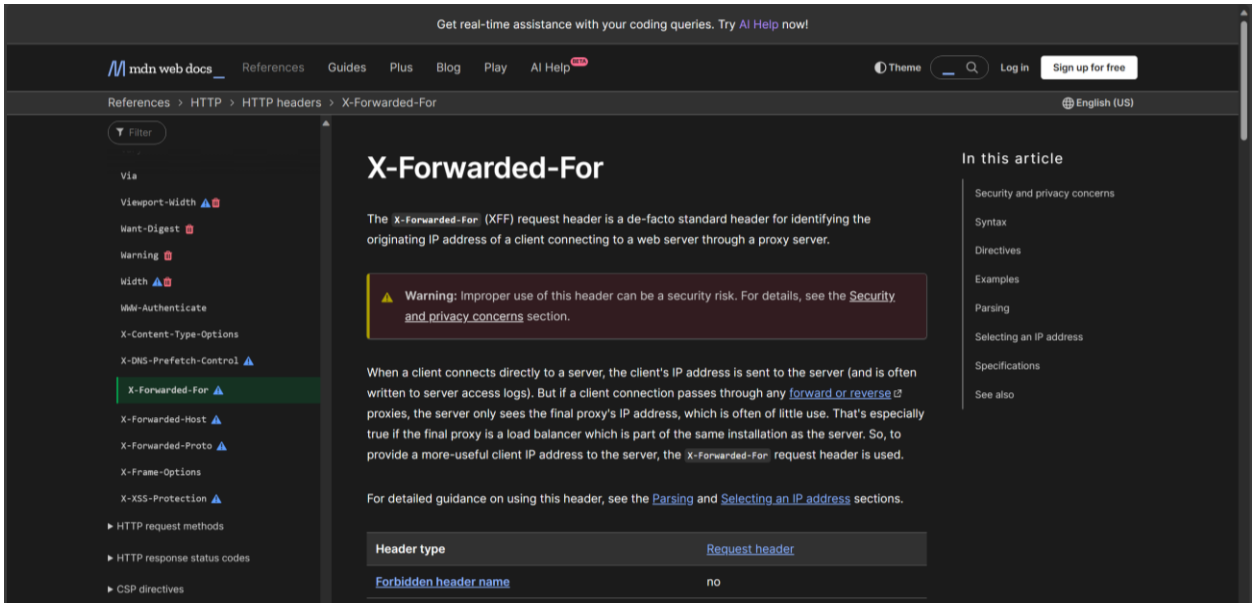
- ⇨ Có thể thấy là tài khoản mà bài lab cho sẵn này đã có thể login vào được

2. Giờ ta sẽ thử brute-force username và password để kiểm tra xem có điều gì bất thường không

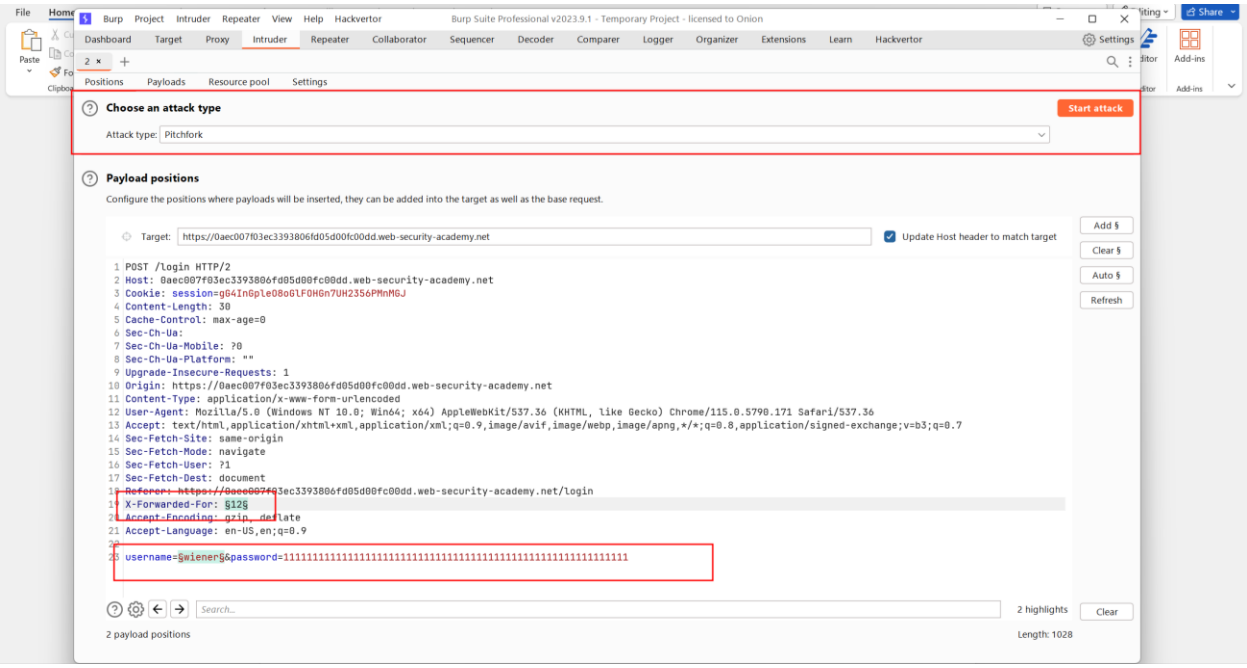


- ⇨ Có thể thấy được rằng khi ta brute-force thì có lỗi thông báo "**You have made too many incorrect login attempts. Please try again in 30 minute(s).**"
- ⇨ Có thể đoán rằng có thể IP mà mình đã dùng để brute-force đã bị block nên không thể tiếp tục login vào được

3. Giờ ta đặt ra câu hỏi rằng: "**Liệu có HTTP header nào hỗ trợ thay đổi IP để tránh bị block không?**"
   Sau khi search thì có một header có thể giúp ta làm điều đó. Đó là **X-Forwarded-For** header.



4. Giờ đã có ý tưởng thì ta sẽ thêm **X-Forwarded-For** header vào request. Đồng thời thêm kí tự **$** vào giá trị của header.



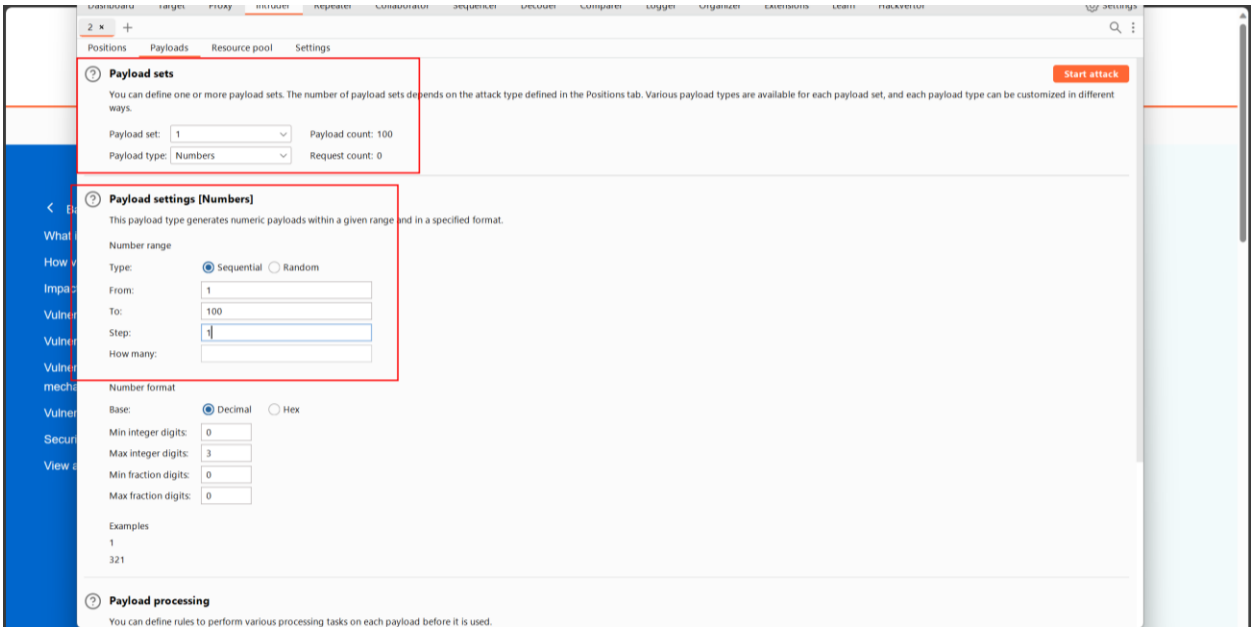
⇨ Chọn attack type là **Pitchfork** – kiểu attack này lặp lại thông qua một bộ payload khác nhau cho từng vị trí được xác định. Payload sẽ được đặt đồng thời vào từng vị trí.

5. Về phần Payloads ta sẽ cấu hình như sau:

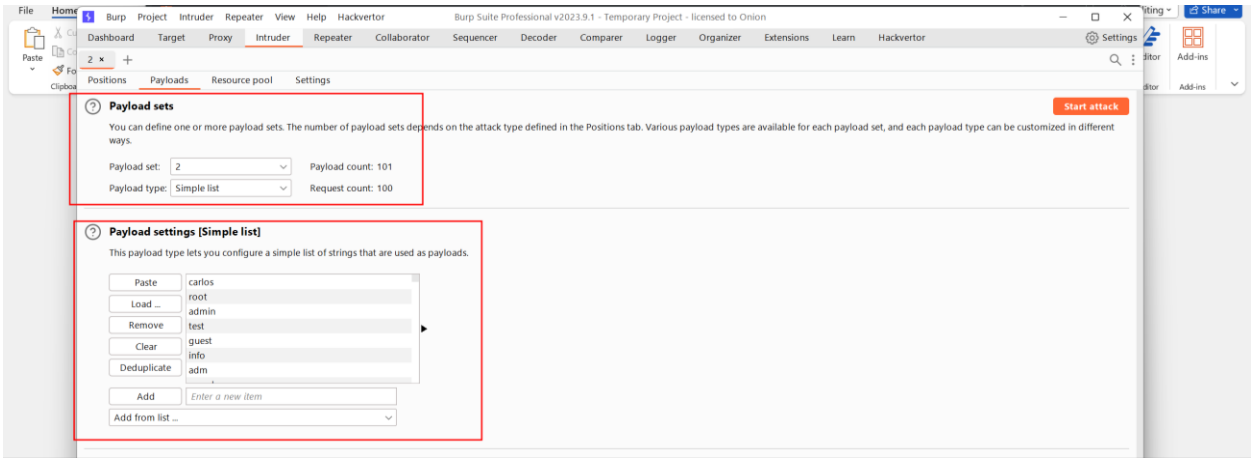
- Payload set 1 được cấu hình như sau:

Payload type: Numbers

Payload settings: Number range 1->100



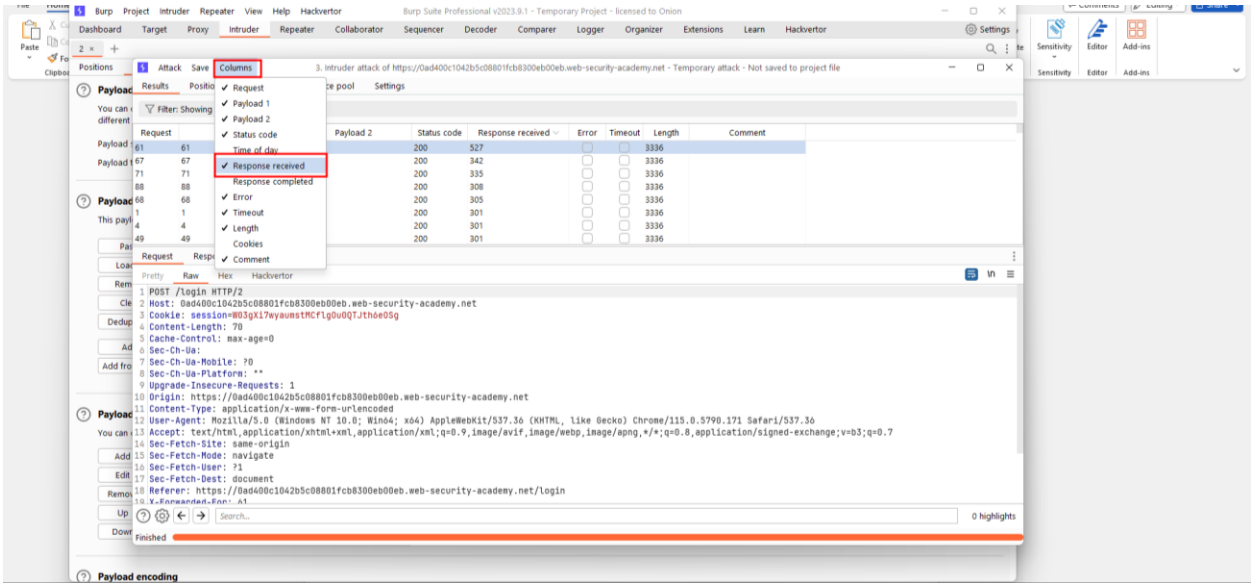
- Payload set 1 được cấu hình như sau:

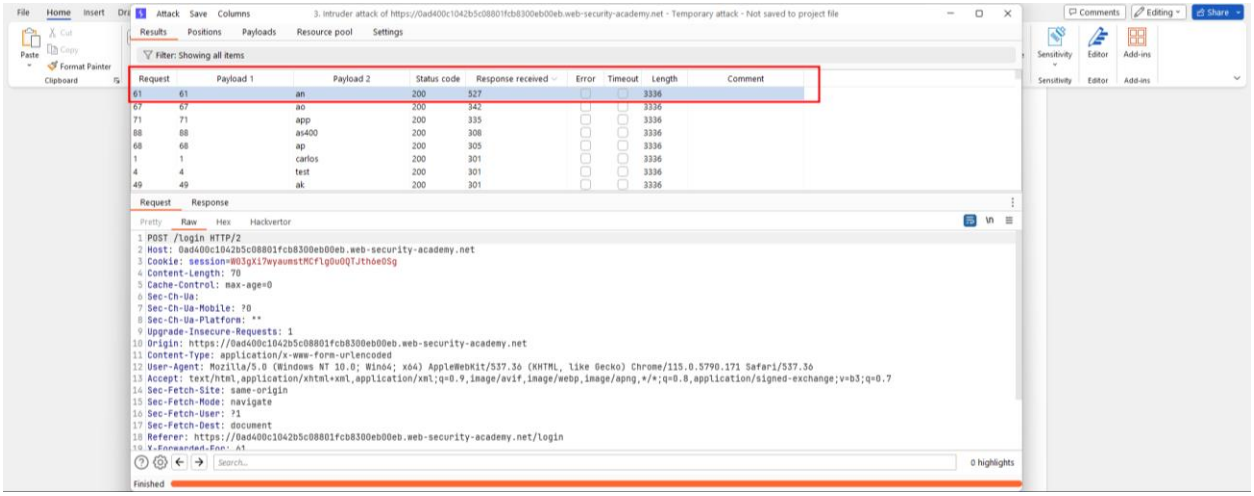


⇨ Giờ ta đã cấu hình xong, tiếp theo ta sẽ **Start attack.**

⇨ Với Payload set 1 sẽ được dùng cho vị trí ở header **X-Forwarded-For**, và Payload set 2 sẽ dùng cho vị trí **username.**

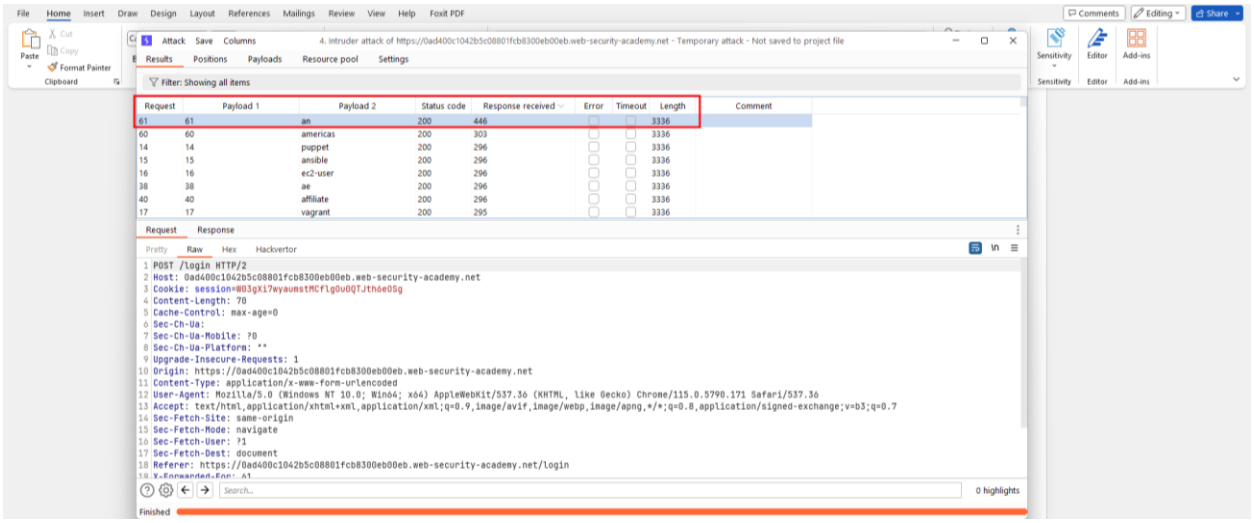
6. Sau khi quá trình brute-force xong thì từ **Columns > Response received**.

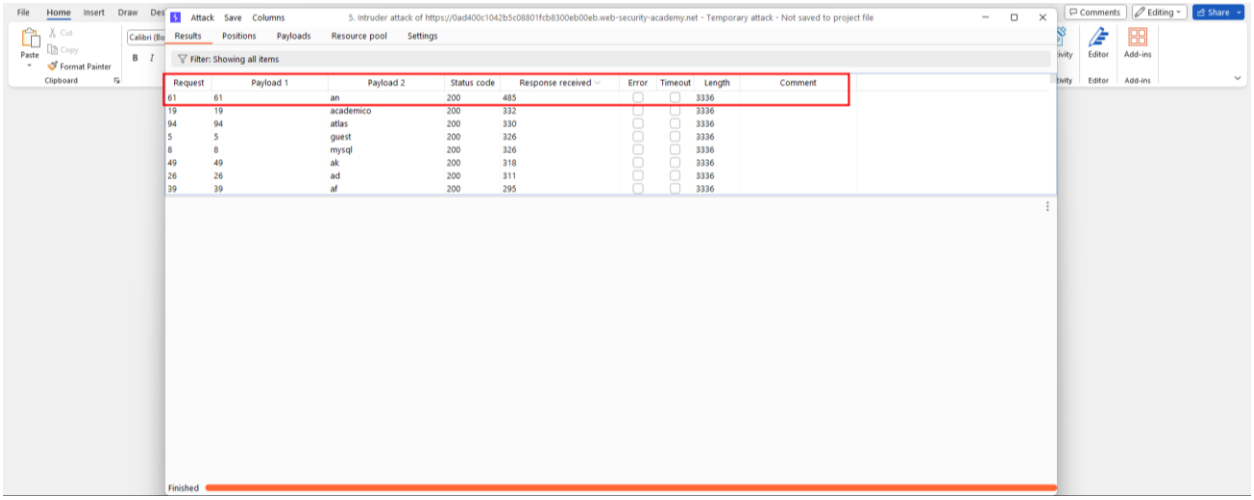




⇨ Có thể thấy request 61 với username "**an**" có khoản cách thời gian response dài hơn với các response còn lại.

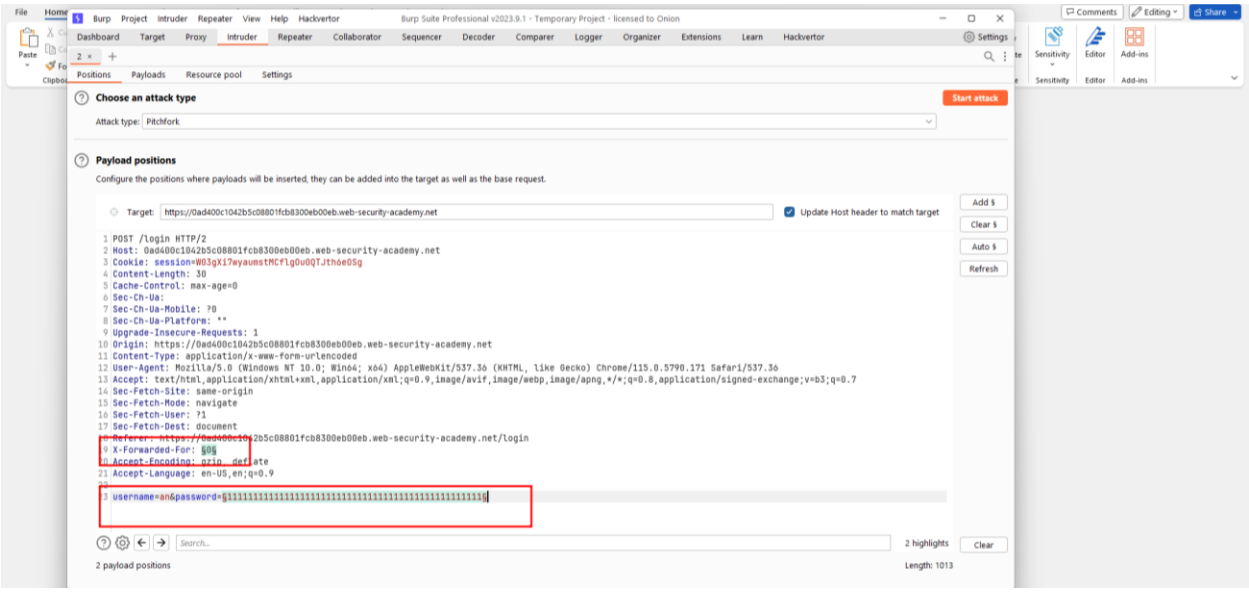
⇨ Để chứng minh điều này là đúng thì ta sẽ chạy lại thêm 2 lần nữa



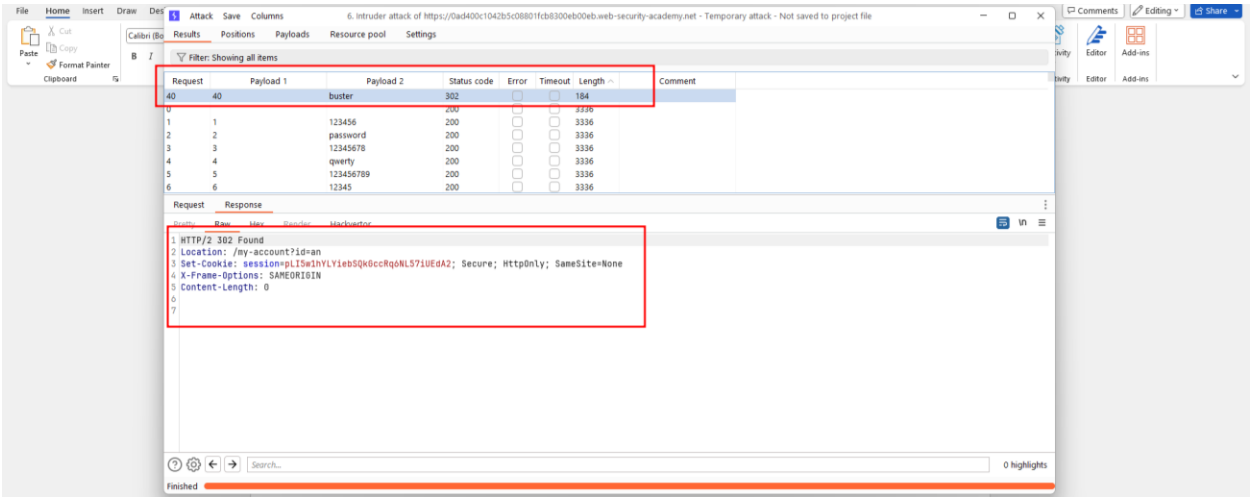


Vậy ta kết luận rằng tài khoản với username là “an” có tồn tại nên cần thêm một khoản thời gian thao tác nên mới lâu hơn

7. Để tìm được password thì ta sẽ làm tương tự như trên là thay thế lại vị trí Payload thành password và thay username thành “**an**”

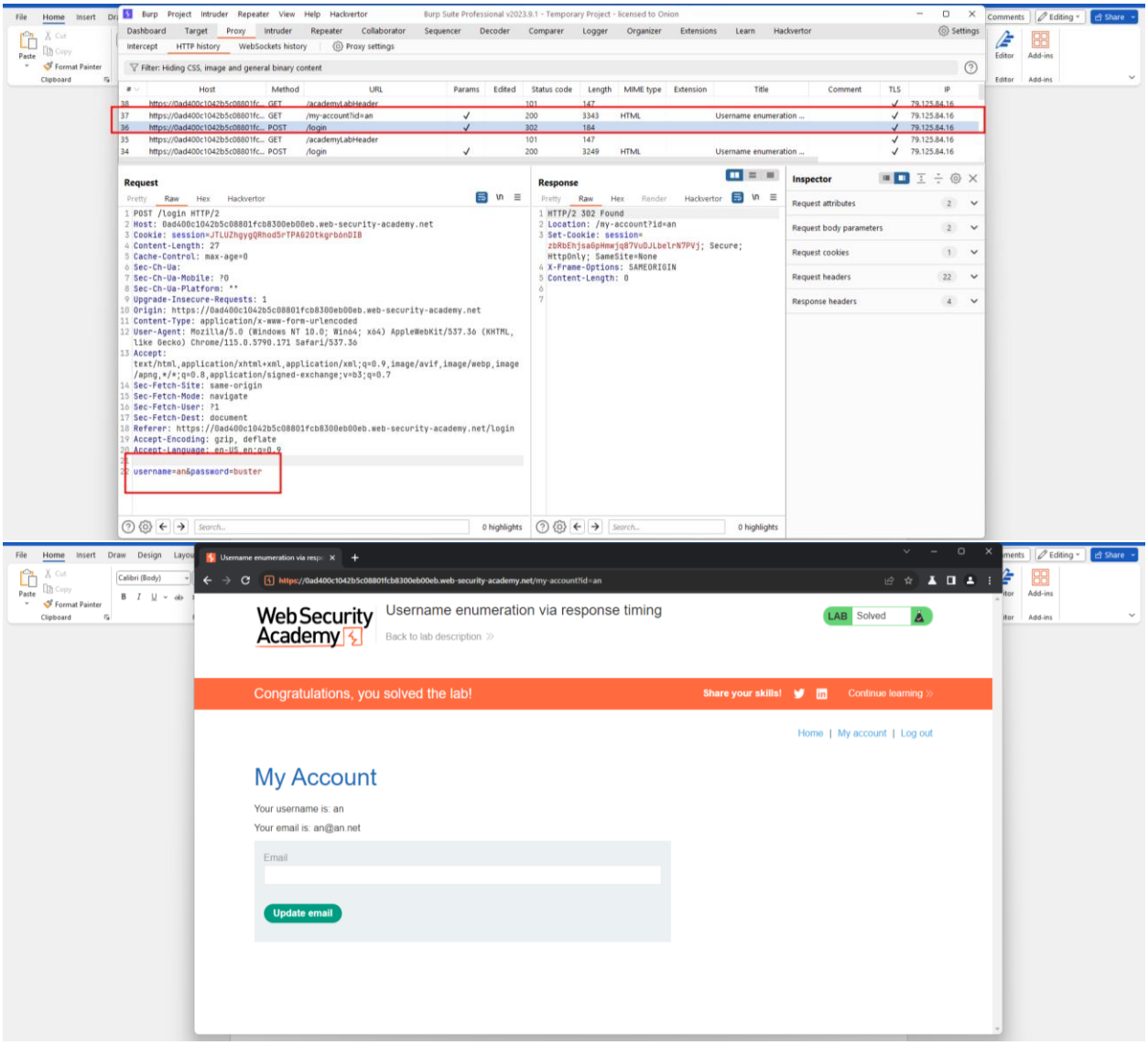


Thay vì kiểm tra Response received thì ta chỉ cần kiểm tra Length của response vì khi password tìm đúng nó sẽ trả về HTTP Status 302 như bài lab phía trên



⇨ Vậy là ta đã tìm được password là “**buster**”

8. Giờ ta sẽ thử dùng account đã tìm được để login vào: **an:buster**



⇨ Có thể thấy được là ta đã login được vào tài khoản.