

BÁO CÁO THỰC HÀNH

Môn học: Bảo mật web và ứng dụng

Tên chủ đề:

Tổng quan các lỗ hổng bảo mật web

thường gặp

GVHD: Ngô Đức Hoàng Sơn

1. THÔNG TIN CHUNG:

Lớp: NT213.P12.ANTT

STT	Họ và tên	MSSV	Email
1	Hồ Vĩnh Khánh	22520633	22520633@gmail.com

2. NỘI DUNG THỰC HIỆN:¹

STT	Nội dung	Tình trạng
1	Bài tập 1	100%
2	Bài Tập 2	100%
3	Bài Tập 3	100%
4	Bài Tập 4	100%
5	Bài Tập 5	100%
6	Bài Tập 6	100%
Điểm tự đánh giá		10/10

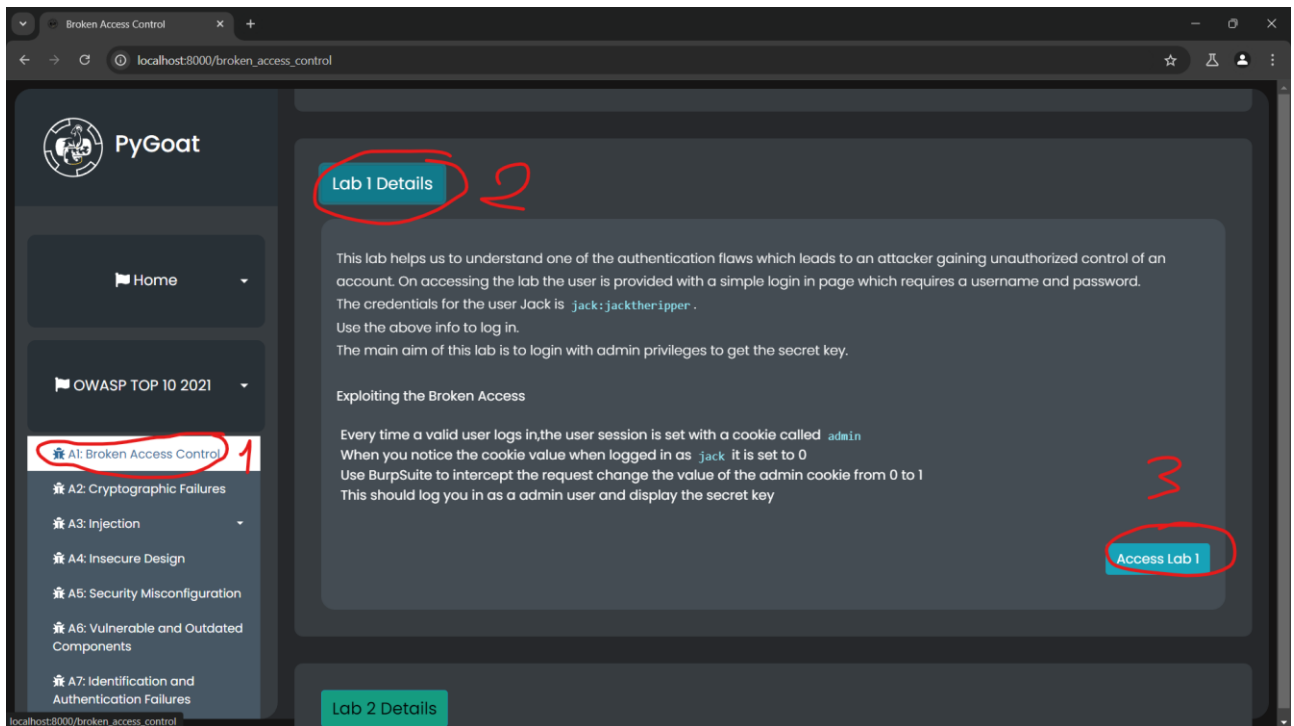
Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

¹ Ghi nội dung công việc, các kịch bản trong bài Thực hành

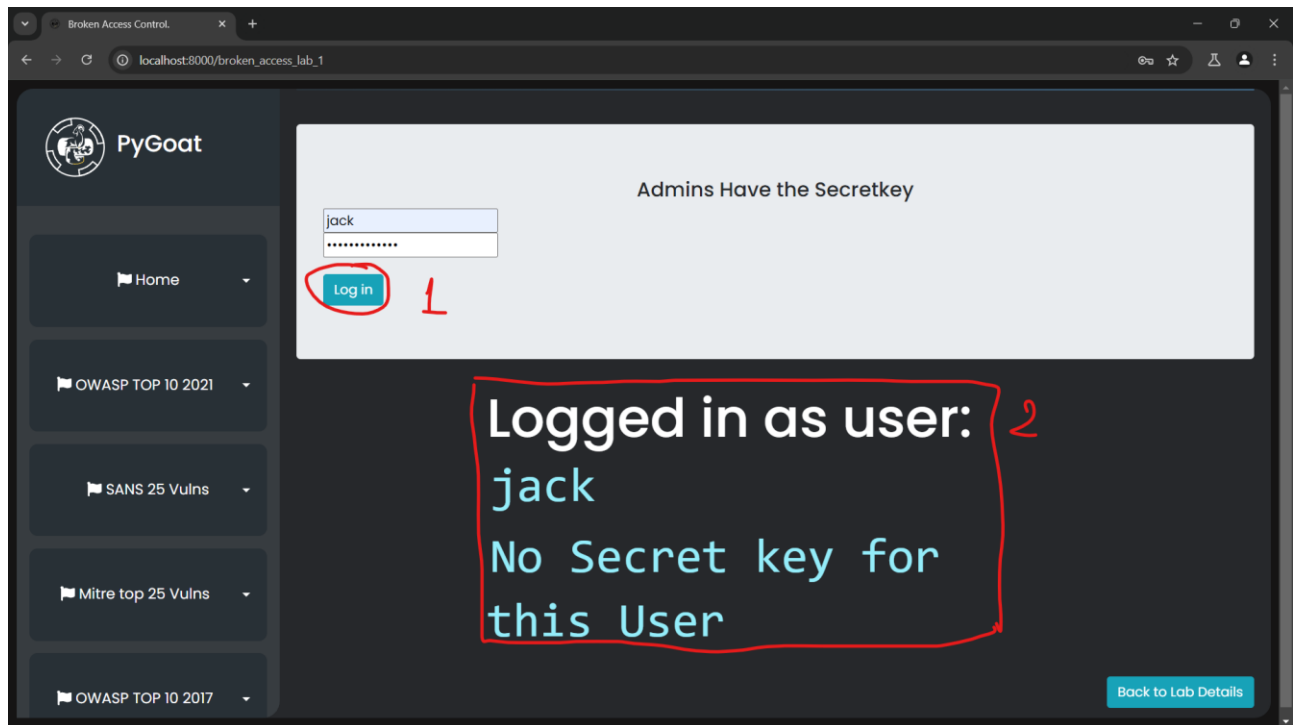
BÁO CÁO CHI TIẾT

1. Bài Tập 1: Sử dụng repeater để thực hành bài tập trên

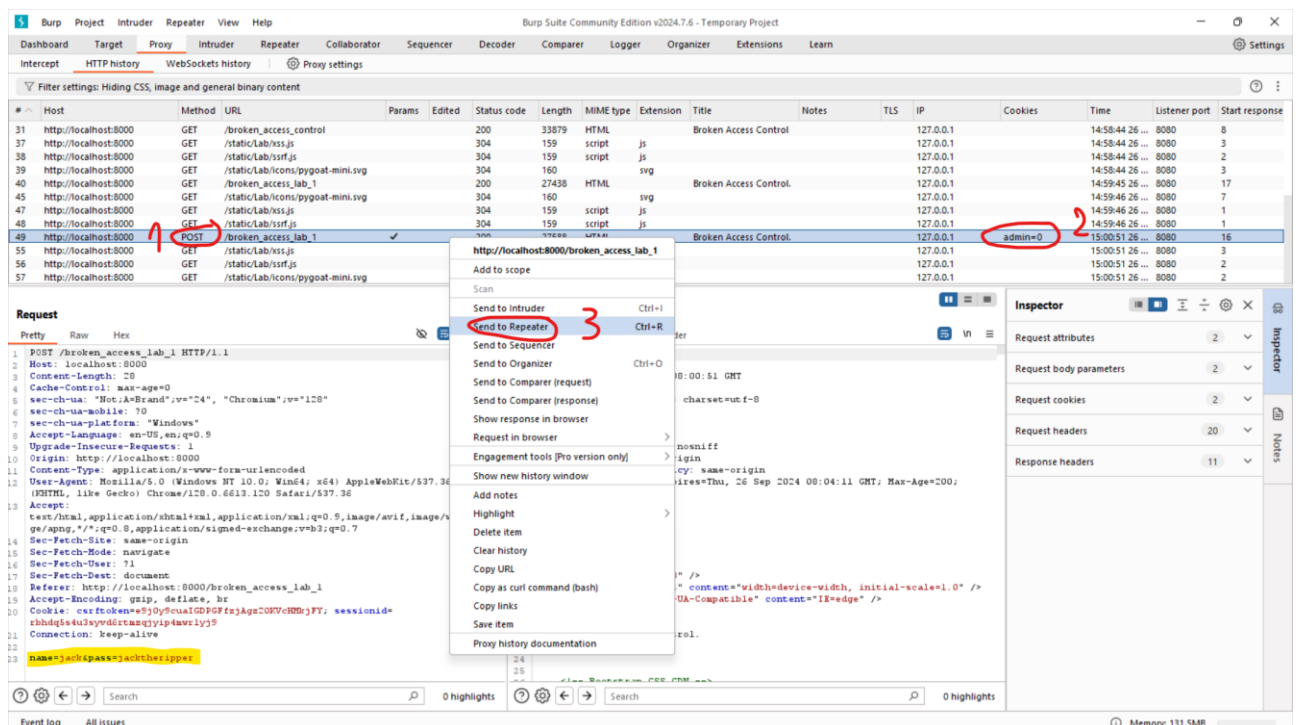
- **Bước 1:** : Truy cập bài thực hành tại <http://localhost:8000> => OWASP TOP 10 2021 => A1: Broken Access Control => Lab 1 Details



- **Bước 2:** Đăng nhập vào trang web với tài khoản và mật khẩu được cung cấp của user Jack.

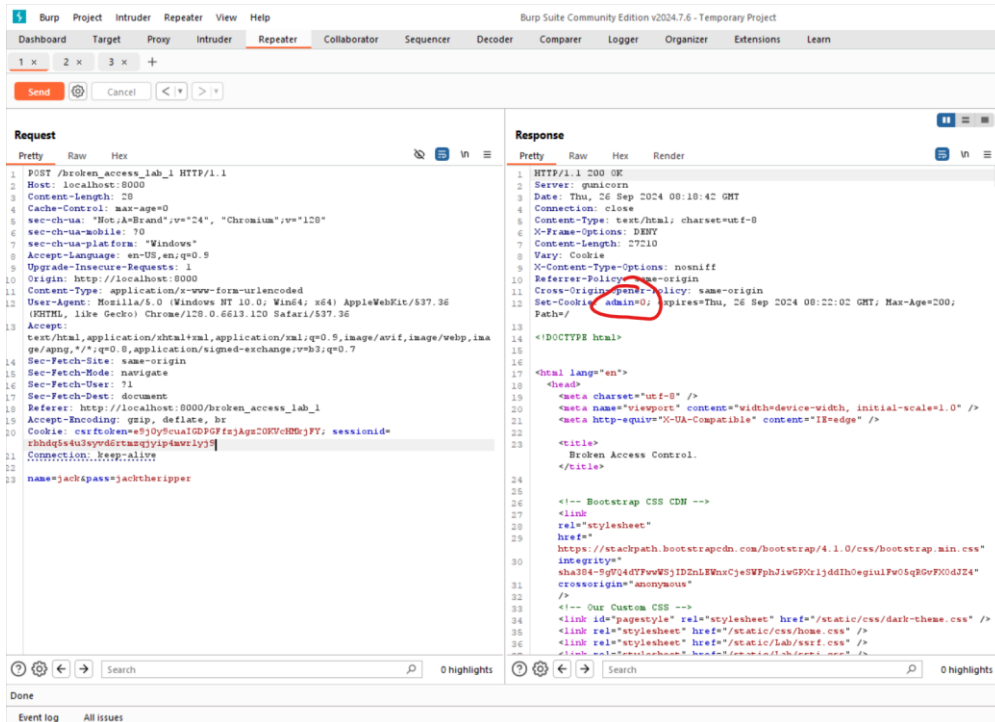


- **Bước 3:** Trở lại giao diện HTTP history của Burpsuite để kiểm tra lịch sử câu truy vấn.

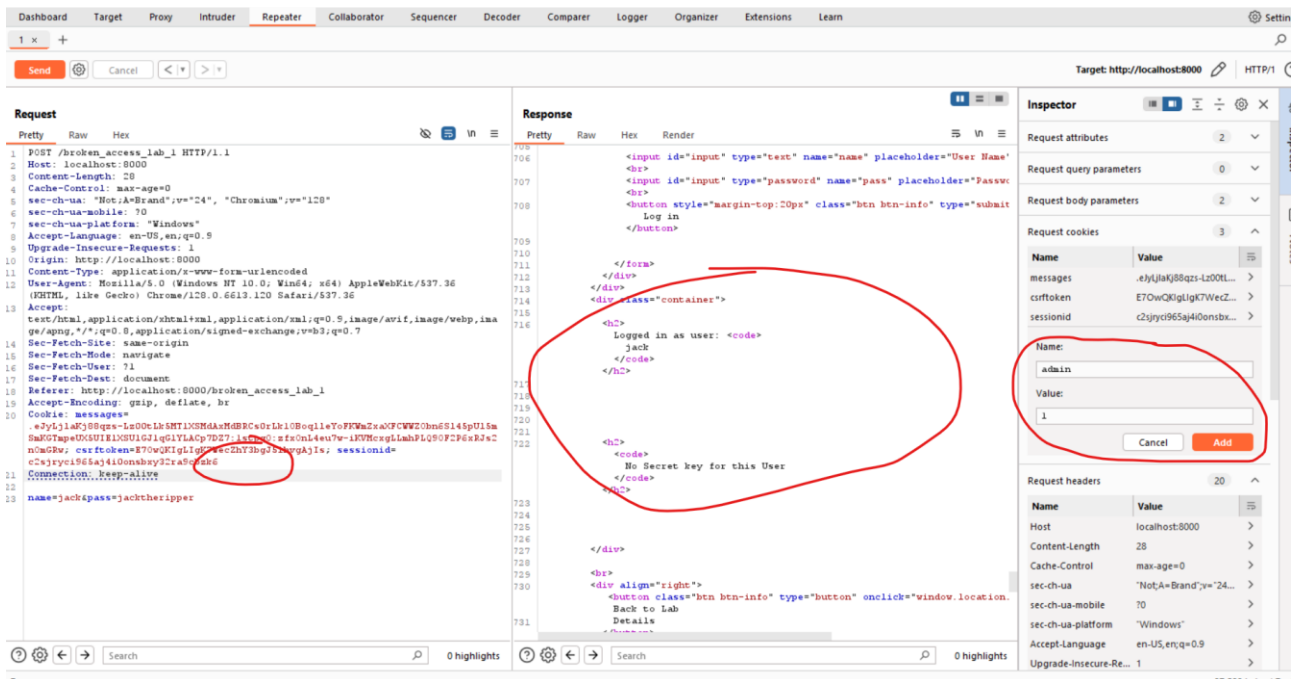


- Tại đây ta thấy dòng truy vấn mới phương thức POST có chứa name và password. Và cookie admin=0. => Do đó dòng 49 là dòng đáng lưu ý và nghi ngờ.

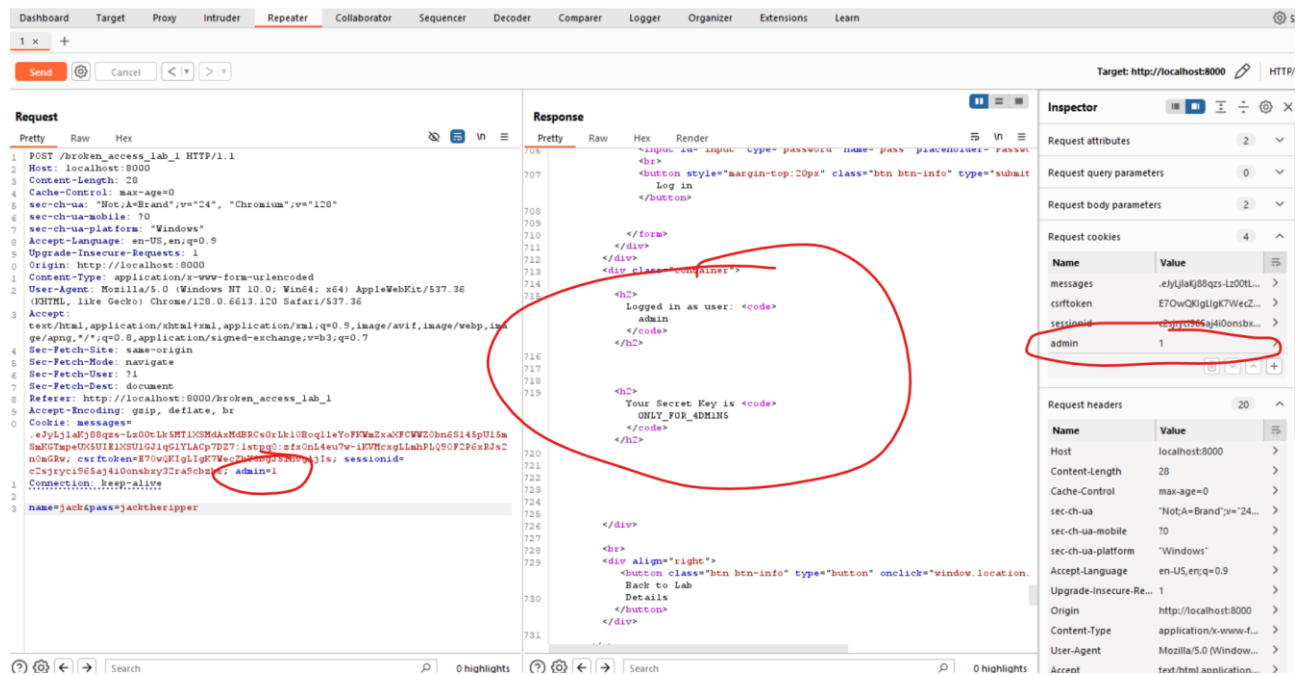
- Nhấp chuột phải và chọn Repeater hoặc ctrl+R, chuyển qua tab repeater, ở giao diện này sẽ có 2 phần chính là request và response.



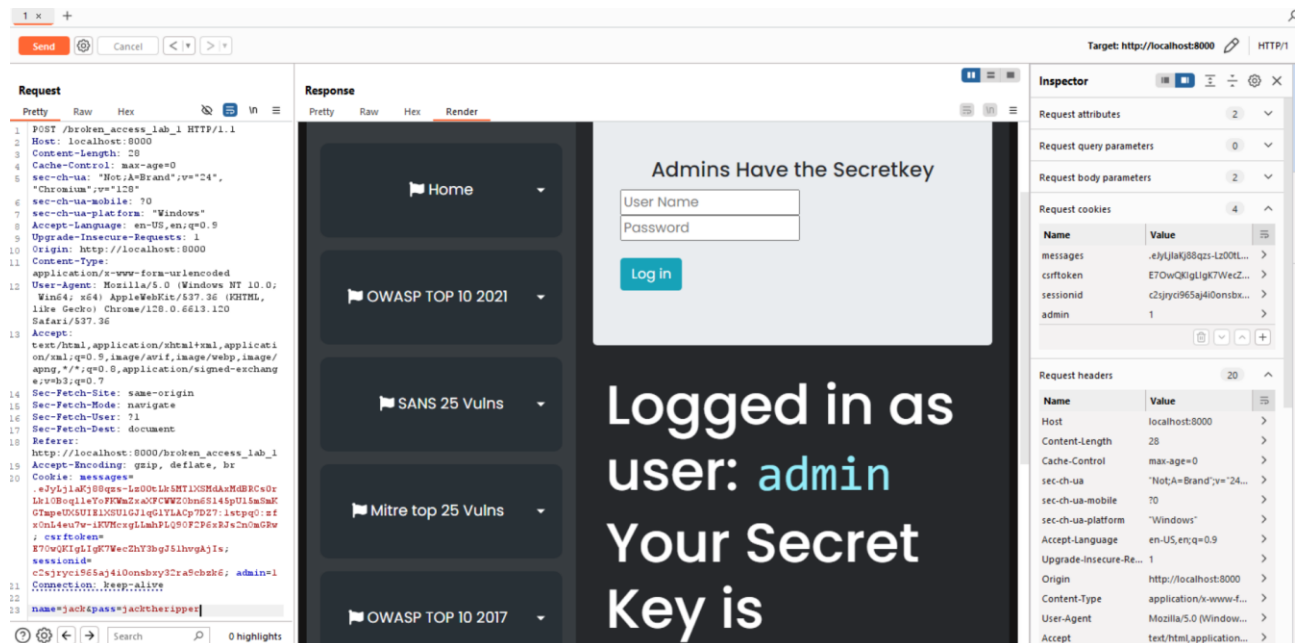
Ta tiến hành thêm admin=1 vào request bằng cách nhập trực tiếp hoặc add thêm giá trị cookie bằng cách điền name và value trong thẻ request cookies. Sau đó bấm send để gửi đi.



Kết quả nhận được ở thẻ response như sau



Tương đương



2. Bài tập 2: Báo cáo lỗ hổng

- Tiêu đề: A01:2021-Broken Access Control
- Mô tả lỗ hổng:

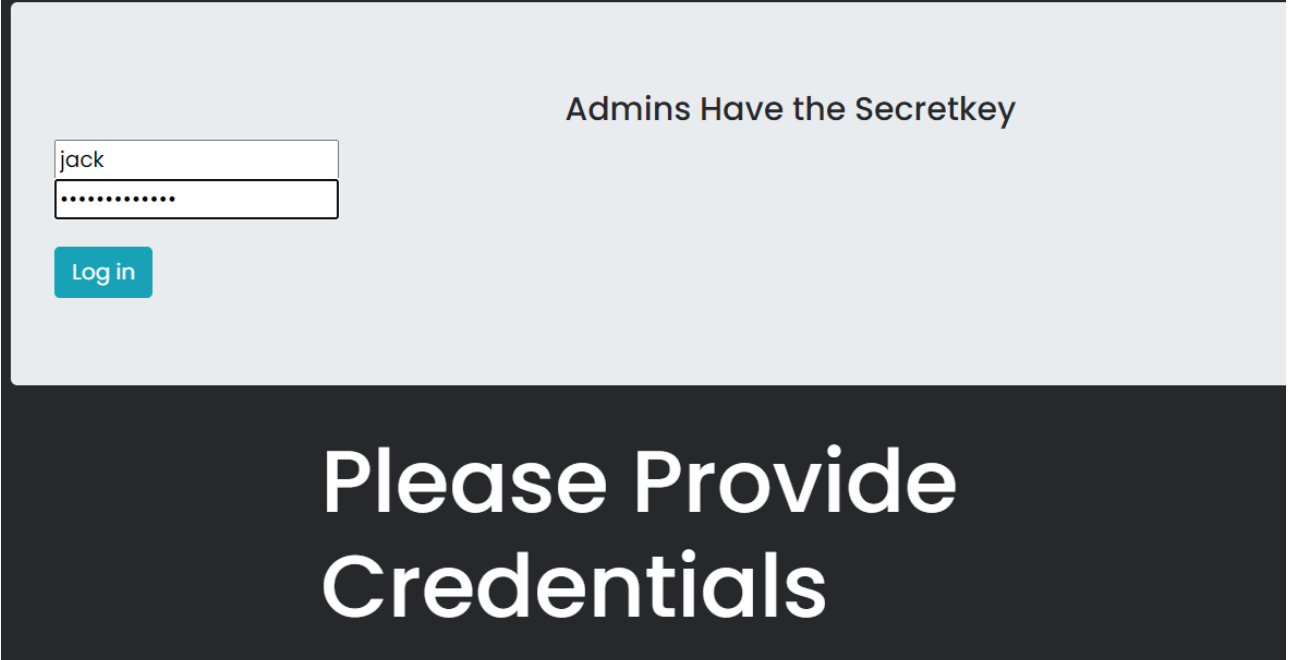
Lỗi kiểm soát truy cập hỏng (Broken Access Control) đã di chuyển lên từ vị trí thứ 5 lên vị trí thứ nhất. Các lỗi này xảy ra vì thiếu các tính năng phát hiện lỗi tự động hoặc các

hàm kiểm tra và đánh giá chưa hiệu quả do lỗi ở việc phân quyền trong hệ thống. Hacker có thể lợi dụng lỗi này để truy cập vào quyền người dùng để thêm, sửa, xóa các bản ghi.

- Tóm tắt:

Khi truy cập vào phòng thí nghiệm, người dùng được cung cấp một trang đăng nhập đơn giản yêu cầu tên người dùng và mật khẩu. Thông tin đăng nhập của người dùng Jack là jack:jacktheripper. Sử dụng thông tin trên để đăng nhập. Mục đích chính của phòng thí nghiệm này là đăng nhập với đặc quyền quản trị viên để lấy khóa bí mật.

- Các bước để thực hiện lại và bằng chứng:
 - Bước 1: Đăng nhập trang web bằng tài khoản đã được cung cấp (jack:jacktheripper), dùng BurpSuite để chặn gói tin đăng nhập.



Admins Have the Secretkey

jack

.....

Log in

Please Provide Credentials

Giao diện chặn gói tin BurpSuite

Request

```

1 POST /broken_access_lab_1 HTTP/1.1
2 Host: localhost:8000
3 Content-Length: 20
4 Cache-Control: max-age=0
5 sec-ch-ua: "Not;A=Brand";v="24", "Chromium";v="120"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Windows"
8 Accept-Language: en-US,en;q=0.9
9 Upgrade-Insecure-Requests: 1
10 Origin: http://localhost:8000
11 Content-Type: application/x-www-form-urlencoded
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6613.120 Safari/537.36
13 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
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: http://localhost:8000/broken_access_lab_1
19 Accept-Encoding: gzip, deflate, br
20 Cookie: messages=.eyJ1aXJ88qzs-Lz00Lk...; csrfToken=E70wQKlgK7WeZ...; sessionId=c2jyrc965aj4i0onsby...; admin=0
21 Connection: keep-alive
22
23 name=jack&pass=jacktheripper
    
```

Inspector

Name	Value
messages	.eyJ1aXJ88qzs-Lz00Lk...
csrfToken	E70wQKlgK7WeZ...
sessionId	c2jyrc965aj4i0onsby...
admin	0

- Bước 2: Thay đổi cookie admin=1 trong tab request và bấm forward. Hoặc xóa và thêm request cookies ở tab inspector.

Request

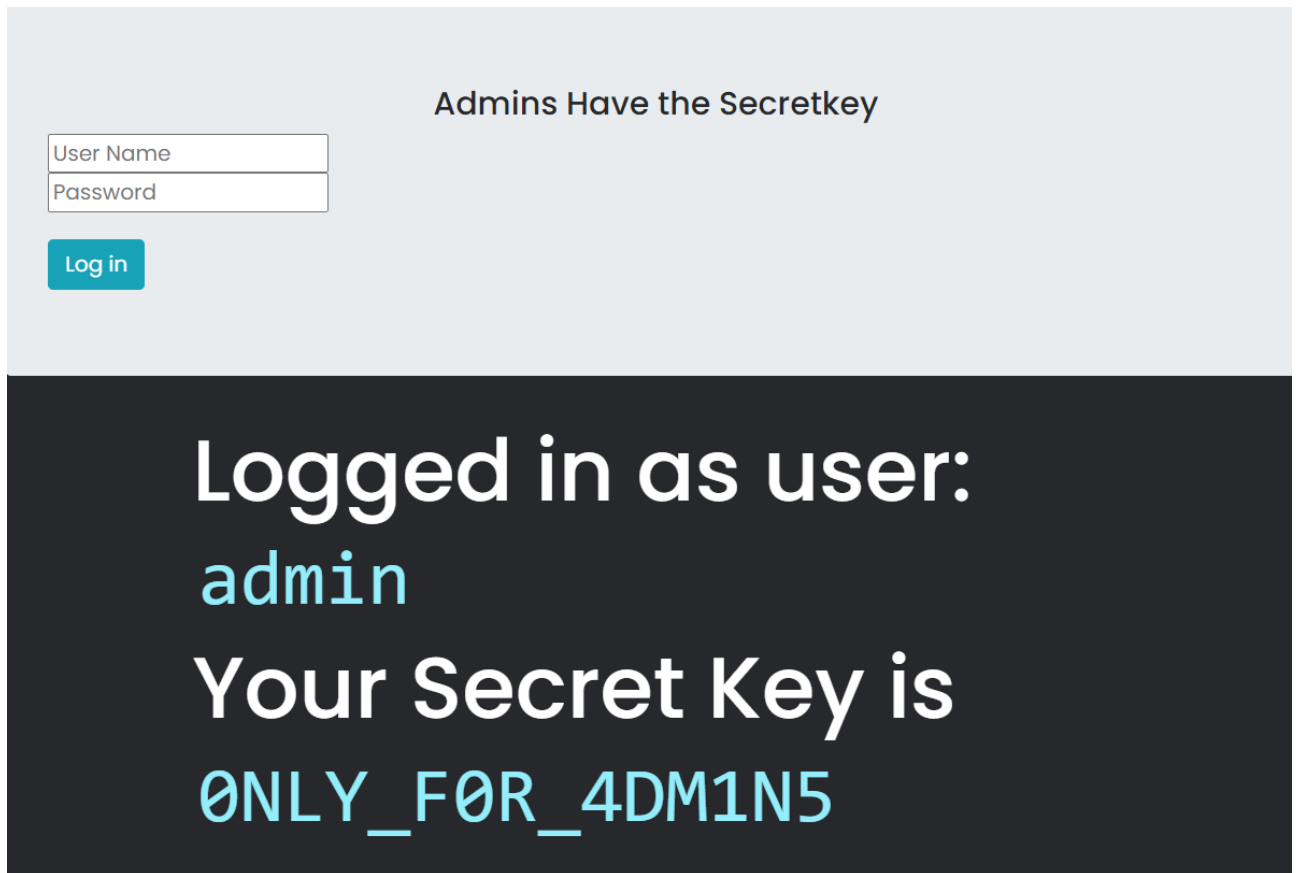
```

1 POST /broken_access_lab_1 HTTP/1.1
2 Host: localhost:8000
3 Content-Length: 20
4 Cache-Control: max-age=0
5 sec-ch-ua: "Not;A=Brand";v="24", "Chromium";v="120"
6 sec-ch-ua-mobile: ?0
7 sec-ch-ua-platform: "Windows"
8 Accept-Language: en-US,en;q=0.9
9 Upgrade-Insecure-Requests: 1
10 Origin: http://localhost:8000
11 Content-Type: application/x-www-form-urlencoded
12 User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/120.0.6613.120 Safari/537.36
13 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
14 Sec-Fetch-Site: same-origin
15 Sec-Fetch-Mode: navigate
16 Sec-Fetch-User: ?1
17 Sec-Fetch-Dest: document
18 Referer: http://localhost:8000/broken_access_lab_1
19 Accept-Encoding: gzip, deflate, br
20 Cookie: messages=.eyJ1aXJ88qzs-Lz00Lk...; csrfToken=E70wQKlgK7WeZ...; sessionId=c2jyrc965aj4i0onsby...; admin=1
21 Connection: keep-alive
22
23 name=jack&pass=jacktheripper
    
```

Inspector

Name	Value
messages	.eyJ1aXJ88qzs-Lz00Lk...
csrfToken	E70wQKlgK7WeZ...
sessionId	c2jyrc965aj4i0onsby...
admin	1

Kết quả này là đăng nhập với tư cách Quản trị viên



- Tài liệu hỗ trợ và tham khảo:
 - Hướng dẫn thực hành lab 1 Tổng quan các lỗ hổng bảo mật web thường gặp Thực hành môn Bảo mật web và ứng dụng
 - Github: <https://github.com/adeyosemanputra/pygoat/tree/master>
- **Mức độ ảnh hưởng của lỗ hổng:** Hacker có thể lợi dụng lỗi này để truy cập vào quyền người dùng để thêm, sửa, xóa các bản ghi. Có thể chiếm quyền kiểm soát, làm lộ dữ liệu của trang web
- **Khuyến cáo khắc phục:**
 - Sử dụng các kỹ thuật quản lý phiên thích hợp
 - Sử dụng các Token như JWT để ủy quyền cho người dùng
 - Kiểm tra kỹ lưỡng và kiểm tra các biện pháp kiểm soát quyền truy cập để đảm bảo chúng hoạt động như thiết kế

3. Bài tập 3: Báo cáo lỗ hổng Cryptographic Failures

- **Tiêu đề:** A02:2021 – Cryptographic Failures
- **Mô tả lỗ hổng:**

Lỗ hổng này là lỗi mã hóa, là nguyên nhân chính dẫn đến việc lộ lọt dữ liệu nhạy cảm. Các lỗ hổng phổ biến bao gồm sử dụng mật khẩu cố định, thuật toán mã hóa yếu, và

thiếu tính ngẫu nhiên. Cần xác định mức độ bảo vệ cần thiết cho dữ liệu khi truyền tải và lưu trữ. Thông tin nhạy cảm như mật khẩu, số thẻ tín dụng, hồ sơ y tế cần được bảo vệ đặc biệt, nhất là khi thuộc phạm vi của các quy định như GDPR hay PCI DSS.

- Tóm tắt:

Một số hacker trước đây đã thực hiện một cuộc tấn công tiêm nhiễm sql và tìm cách lấy được kết xuất cơ sở dữ liệu cho bảng người dùng.

alex,9d6edee6ce9312981084bd98eb3751ee

admin,c93ccd78b2076528346216b3b2f701e6

rupak,5ee3547adb4481902349bdd0f2ffba93

Từ đó ta thấy account admin có đoạn mã hóa có thể là password được mã hóa.

Các bước để thực hiện lại và bằng chứng:

- Bước 1:

Nhận dạng dữ liệu mã hóa, ta thấy được độ dài đoạn mã có 32 ký tự. Dựa vào tài liệu thực hành cung cấp ta biết đó là mã băm MD5. Hoặc có thể nhận dạng mã hóa bằng trang:

https://hashes.com/en/tools/hash_identifier

- Bước 2:

Mã băm này khá yếu và đã bị bruteforce và dễ dàng tìm kiếm thông qua các trang web online trên mạng. Ở đây chúng ta sẽ thử trang: <https://www.md5online.org/md5-decrypt.html>

Enter your MD5 hash below and cross your fingers :

c93ccd78b2076528346216b3b2f701e6

☒ Quick search (free) ☐ In-depth search (1 credit) 

Decrypt

Found : admin1234

(hash = c93ccd78b2076528346216b3b2f701e6)

Search mode: Quick search

- Bước 3:

Ta có thể tìm được password của tài khoản admin là **admin1234**

- Tài liệu hỗ trợ và tham khảo:
 - Hướng dẫn thực hành lab 1 Tổng quan các lỗ hổng bảo mật web thường gặp Thực hành môn Bảo mật web và ứng dụng
 - Github: <https://github.com/adeyosemanputra/pygoat/tree/master>
- **Mức độ ảnh hưởng của lỗ hổng:** Lộ các dữ liệu nhạy cảm quan trọng hoặc xâm phạm hệ thống.
- **Khuyến cáo khắc phục:**
 - Không lưu trữ dữ liệu nhạy cảm khi không cần thiết
 - Đảm bảo giao thức truyền an toàn
 - Tránh sử dụng các hàm băm lỗi thời MD5, SHA1, PKCS số 1 v1.5
 - Mã hóa cần xác thực thay vì chỉ mã hóa đơn thuần

4. Bài tập 4: Báo cáo lỗ hổng

- **Tiêu đề:** A03:2021 – Injection

- Mô tả lỗ hổng:

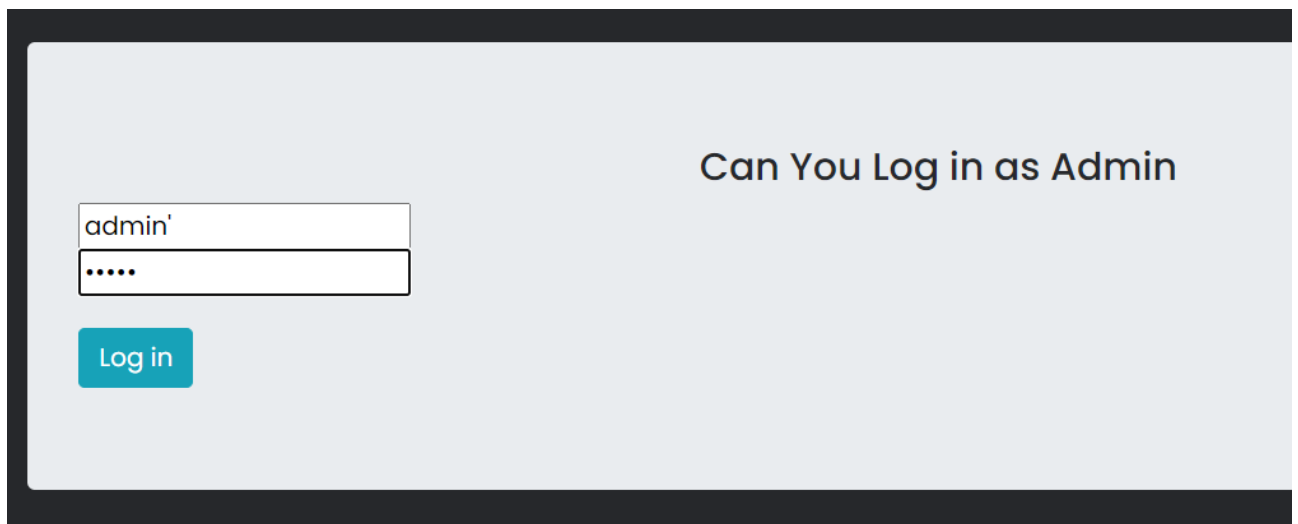
Lỗi chèn mã độc (Injection) sẽ lợi dụng lỗ hổng từ các câu lệnh truy vấn của ứng dụng. Hacker sẽ lợi dụng chèn một đoạn mã SQL để nắm quyền thực thi website và khai thác dữ liệu cơ sở dữ liệu (database) khi khách hàng nhập vào những biểu mẫu (form) trên trang web.

○ Tóm tắt:

Người dùng sẽ truy cập bài thực hành với trang đăng nhập được cung cấp. Người dùng có thể thử đăng nhập với tài khoản admin. Lỗi tiêm SQL có thể được nhận ra thông qua một vài thủ thuật như tiêm một kí tự ' vào bất kỳ trường nào. Nếu kết quả là một lỗi SQL thì lỗi tiêm SQL có thể đã xảy ra.

○ Các bước để thực hiện lại và bằng chứng:

- Bước 1: Tiêm vào trang đăng nhập một kí tự ' để kiểm tra có lỗi SQL hay không.



The screenshot shows a web application interface for a login page. The title is "Can You Log in as Admin". There are two input fields: the first is for the username, containing the text "admin'", and the second is for the password, containing masked characters ".....". Below the password field is a blue button labeled "Log in".

Ta thấy kết quả trên màn hình như sau:

Can You Log in as Admin

User Name
Password

Log in

The password you have entered doesnt match the username!

The SQL query being submitted is

```
SELECT * FROM introduction_sql_lab_table WHERE id='admin''AND password='admin'
```

⇒ Có lỗi SQL

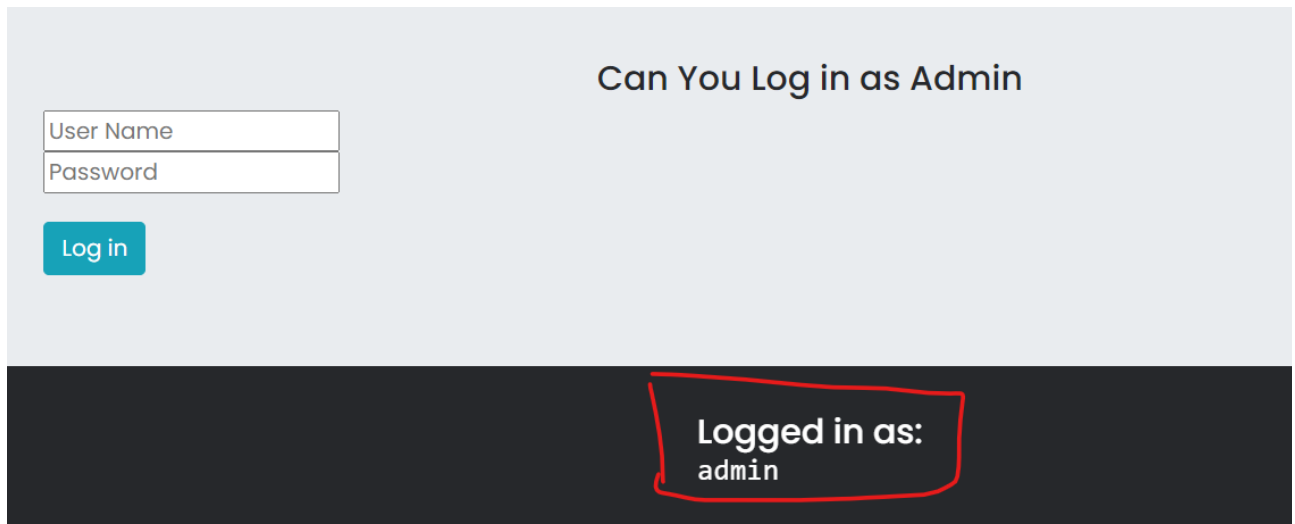
- Bước 2: Tận dụng lỗi hỏng trên bài tập yêu cầu chúng ta đăng nhập với quyền quản trị viên. Ta dùng ' OR '1' ='1 chèn vào để pass login của trang web.

Can You Log in as Admin

admin' OR '1' ='1
.....

Log in

Đăng nhập thành công



- Tài liệu hỗ trợ và tham khảo:
 - Hướng dẫn thực hành lab 1 Tổng quan các lỗ hổng bảo mật web thường gặp Thực hành môn Bảo mật web và ứng dụng
 - Github: <https://github.com/adeyosemanputra/pygoat/tree/master>
- **Mức độ ảnh hưởng của lỗ hổng:** Hacker sẽ lợi dụng chèn một đoạn mã SQL để nắm quyền thực thi website và khai thác dữ liệu cơ sở dữ liệu (database) khi khách hàng nhập vào những biểu mẫu (form) trên trang web.
- **Khuyến cáo khắc phục:**
 - Nên xác thực tất cả thông tin do người dùng cung cấp (zero just)
 - Cài đặt ít đặc quyền nhất có thể để giảm thiểu thiệt hại khi bị tấn công
 - Dùng các lệnh có sẵn, không dùng đầu vào trực tiếp

5. Bài tập 5: Báo cáo lỗ hổng

- **Tiêu đề:** A04:2021 – Insecure Design
- **Mô tả lỗ hổng:**

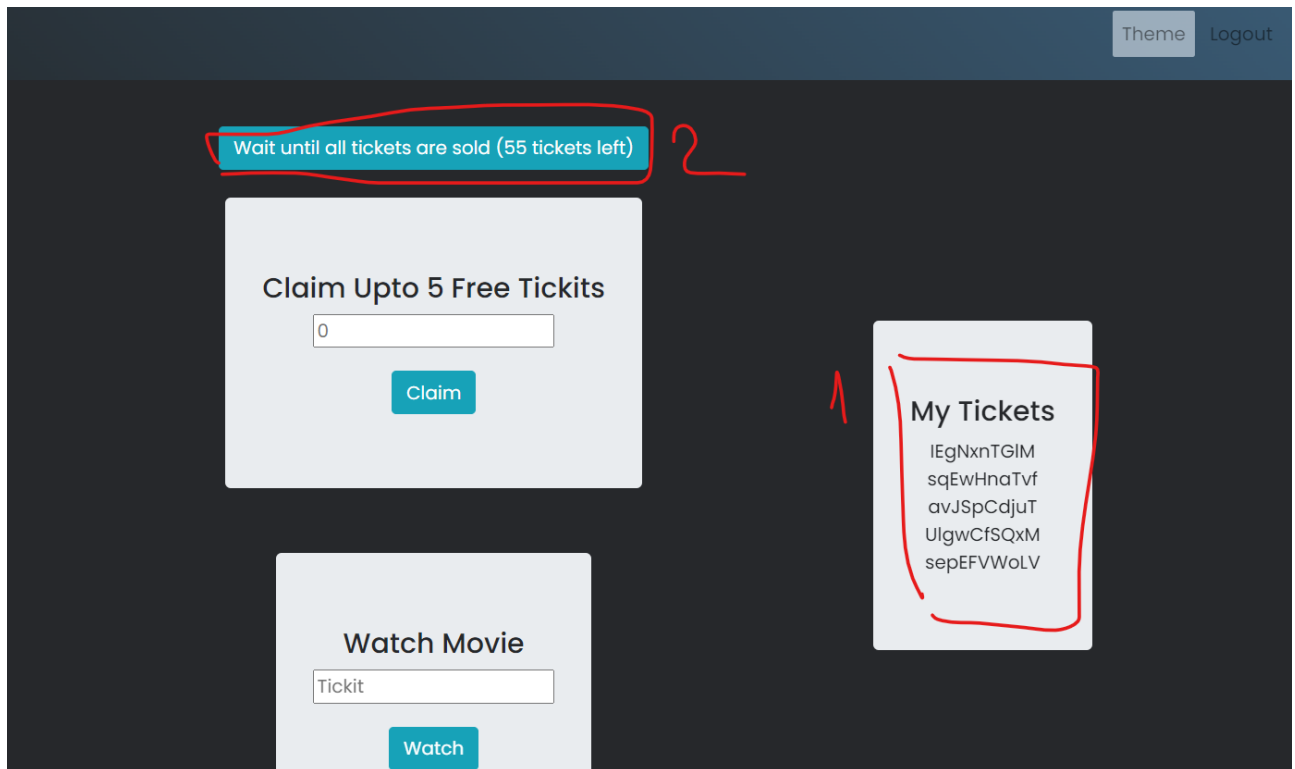
Lỗi thiết kế không bảo mật (Insecure Design) là một trong những danh mục mới trong top 10 lỗ hổng OWASP 2021. Tin tặc lợi dụng khai thác các sai sót trong thiết kế website để tấn công người dùng.

- Tóm tắt:

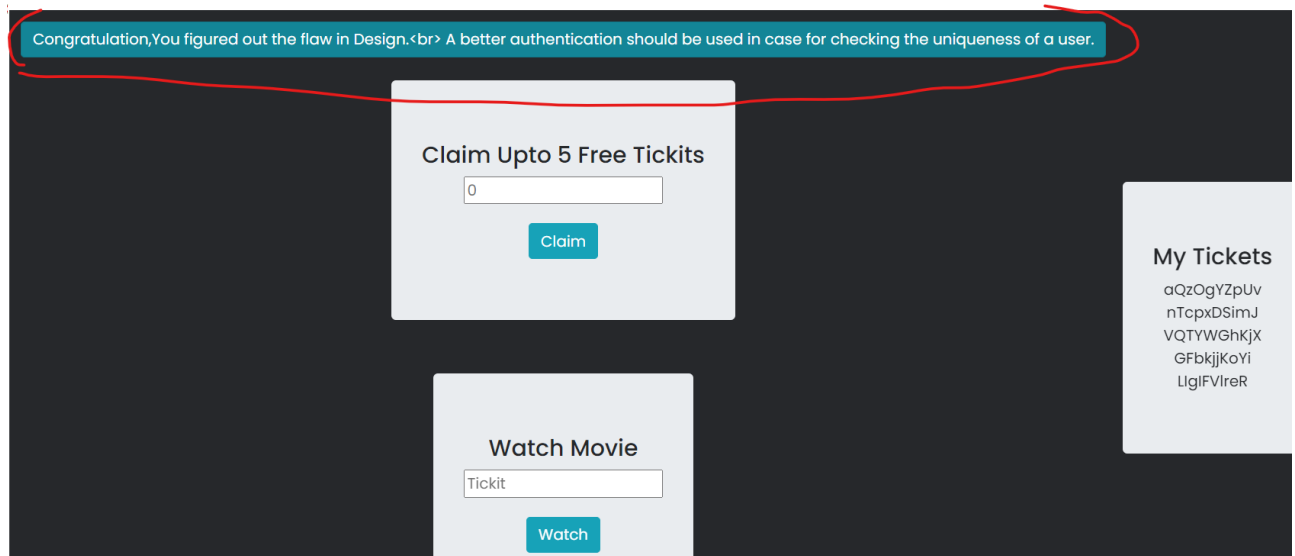
Trang web này cung cấp vé miễn phí cho mọi người (tối đa 5 vé mỗi người). Và bộ phim sẽ được công chiếu khi toàn bộ vé được bán hết. Người ta không thể nhận được nhiều hơn 5 vé miễn phí. Nhưng có một lỗi thiết kế lớn. Người ta có thể nhận được tất cả các vé bằng cách tạo nhiều tài khoản. Trong trường hợp này, 5 vé mỗi trang và tổng số yêu cầu là 60, vì vậy ta chỉ cần tạo 12 tài khoản và yêu cầu 5 vé từ mỗi tài khoản

- Các bước để thực hiện lại và bằng chứng:

- Bước 1: Vào trang web và lấy 5 vé xem phim. Chúng ta có thể nhấp vào nút claim thêm lần nữa để xem được số vé còn lại trong hệ thống



- Bước 2: Tạo thủ công hoặc tự động thêm 11 tài khoản nữa và nhận thêm 5 vé cho mỗi tài khoản cho đến khi hết vé. Sau khi toàn bộ vé đã bán hết ai có vé có thể bấm vào watch movie để xem phim



- Tài liệu hỗ trợ và tham khảo:
 - Hướng dẫn thực hành lab 1 Tổng quan các lỗi hỏng bảo mật web thường gặp Thực hành môn Bảo mật web và ứng dụng
 - Github: <https://github.com/adeyosemanputra/pygoat/tree/master>

- **Mức độ ảnh hưởng của lỗ hổng:** Tin tặc lợi dụng khai thác các sai sót trong thiết kế website để tấn công người dùng. Và có thể làm chậm trễ hoặc dừng các dịch vụ của hệ thống.
- **Khuyến cáo khắc phục:**
 - o Cần tăng độ xác thực, xác minh tài khoản một cách chặt chẽ hơn (có thể định danh cá nhân/kyc)
 - o Quản lí người dùng và tài nguyên, hạn chế lãng phí tài nguyên
 - o Cần phân tích thiết kế hệ thống logic hơn, cần các chuyên gia thiết kế có kiến thức và kinh nghiệm để đảm bảo việc xây dựng, vận hành và bảo vệ hệ thống

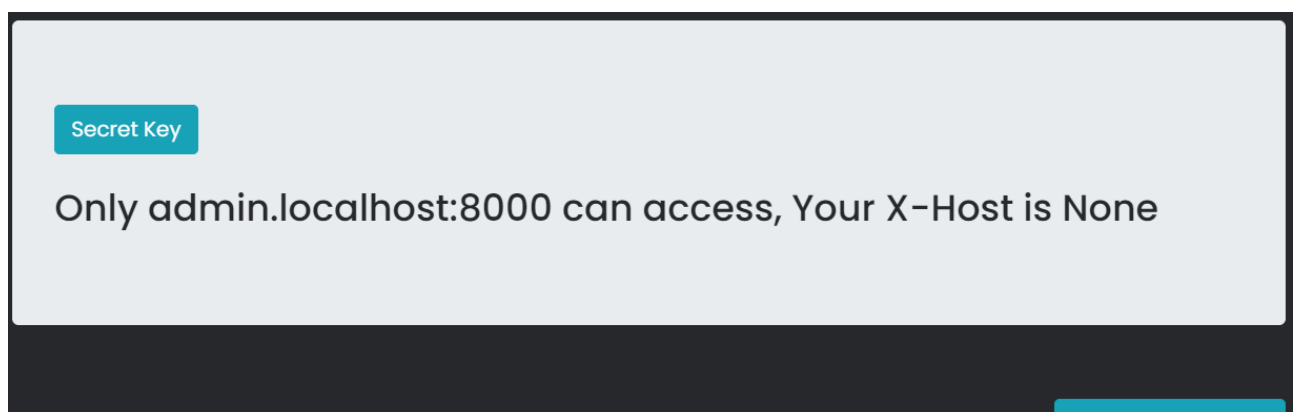
6. Bài tập 6: Báo cáo lỗ hổng

- **Tiêu đề:** A05:2021 – Security Misconfiguration
- **Mô tả lỗ hổng:**

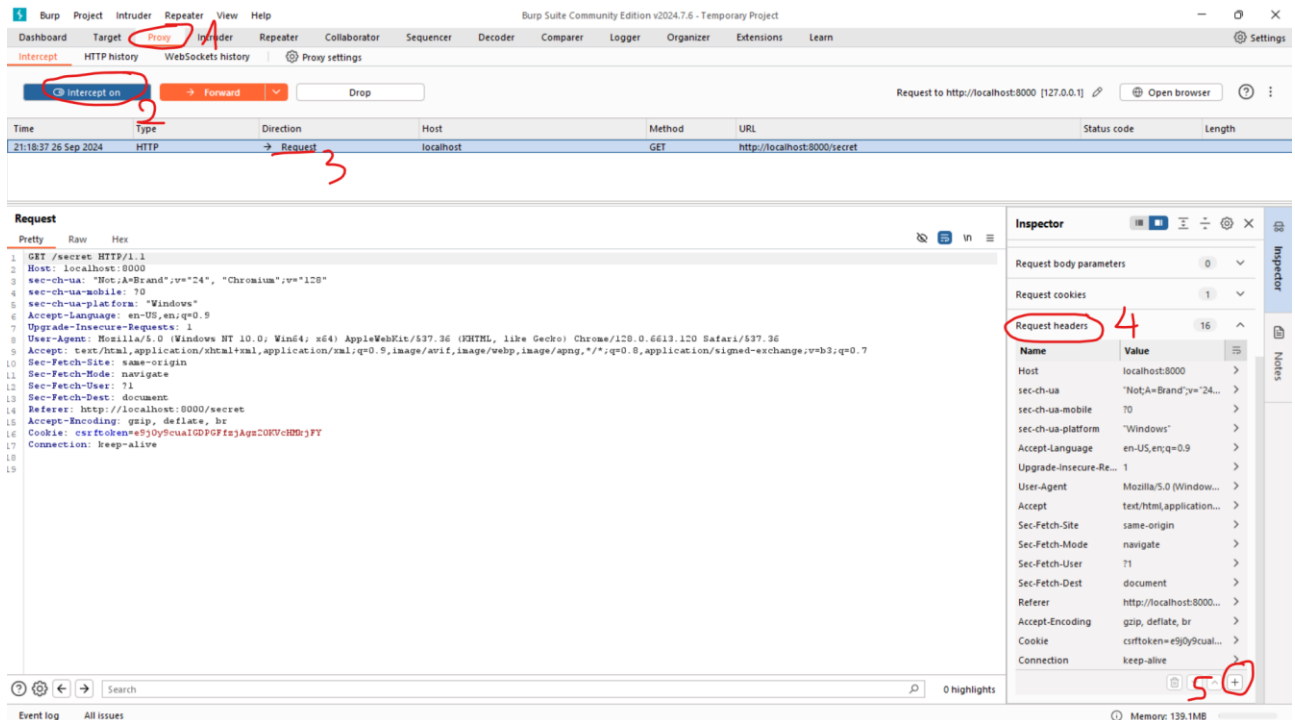
Lỗi cấu hình bảo mật sai (Security Misconfiguration) di chuyển tăng một bậc lên vị trí thứ 6 do nhà quản trị không cập nhật các cấu hình bảo mật mới được cập nhật mỗi ngày. Dẫn đến tình trạng cấu hình bảo mật được xây dựng không chắc chắn tại các tài cơ sở hạ tầng của website, nền tảng khung phần mềm (framework), máy chủ,...

- o Tóm tắt:

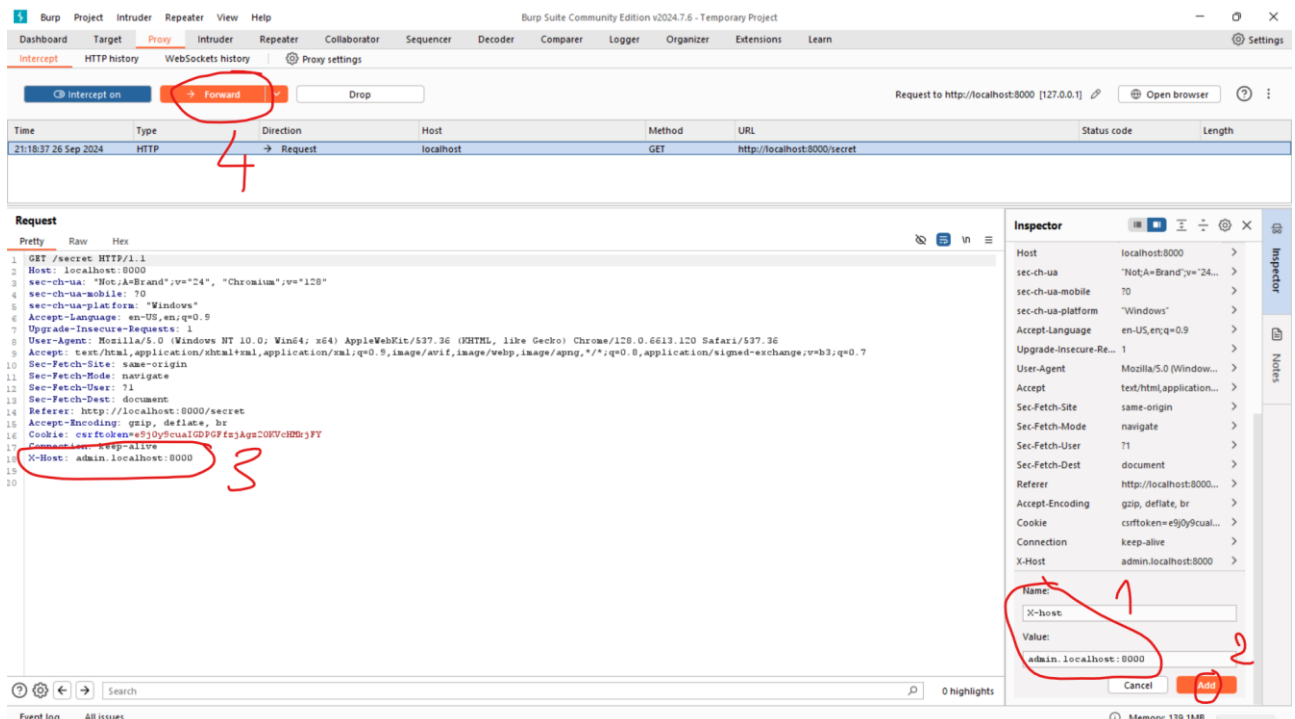
Lab này có cấu hình bảo mật sai. Nó có một nút tiết lộ khóa bí mật nhưng nó chỉ có thể truy cập được nếu quản trị viên truy cập vào nó. Và thông báo X-Host lúc này là None. Vậy nên để truy cập vào, ta cần gán giá trị cho header X-Host và giá trị của nó phải là admin.localhost:8000



- o Các bước để thực hiện lại và bằng chứng:
 - Bước 1: Ta dùng BurpSuite để chặn request, chú ý tab inspector bên góc phải.

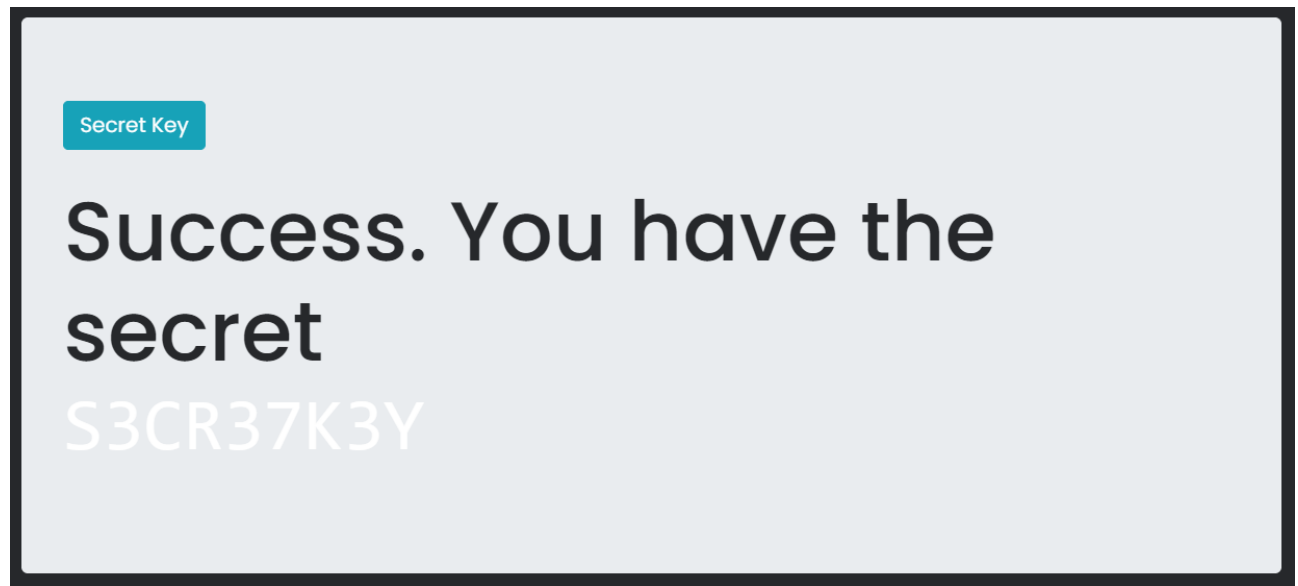


- Bước 2: Chọn vào button + để thêm header X-host: admin.localhost:8000. Sau khi nhấn add header được thêm vào tab request. Sau đó ta nhấn forward.



Trang web sẽ trả về cho ta

Success. You have the **Success. You have the secret S3CR37K3Y** như hình bên dưới là đã thành công



- Tài liệu hỗ trợ và tham khảo:
 - Hướng dẫn thực hành lab 1 Tổng quan các lỗi hỏng bảo mật web thường gặp Thực hành môn Bảo mật web và ứng dụng
 - Github: <https://github.com/adeyosemanputra/pygoat/tree/master>
- **Mức độ ảnh hưởng của lỗi hỏng:** Những lỗi hỏng như vậy thường khiến kẻ tấn công truy cập trái phép vào một số dữ liệu hoặc chức năng hệ thống. Đôi khi, những sai sót như vậy dẫn đến sự xâm phạm toàn bộ hệ thống. Tác động kinh doanh phụ thuộc vào nhu cầu bảo vệ của ứng dụng và dữ liệu.
- **Khuyến cáo khắc phục:**
 - Không cài đặt trả về các giá trị bảo mật như khóa, mật khẩu, ...
 - Phân quyền đúng các role, folder, file, ...
 - Cập nhật hệ thống và các tính năng bảo mật thường xuyên
 - Thiết lập thông báo lỗi để không show ra các thông tin nhạy cảm trong hệ thống
 - Không dùng các framework web sinh ra các account hoặc mật khẩu mặc định

HẾT