

# BÁO CÁO THỰC HÀNH

Môn học: Bảo mật web và ứng dụng

Tên chủ đề: Tổng quan về các lỗ hổng bảo mật web thường gặp

GVHD: Ngô Đức Hoàng Sơn

**Nhóm: 12**

## 1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT213.O21.ANTT.1

STT	Họ và tên	MSSV	Email
1	Nguyễn Triệu Thiên Bảo	21520155	21520155@gm.uit.edu.vn
2	Trần Lê Minh Ngọc	21521195	21521195@gm.uit.edu.vn
3	Huỳnh Minh Khuê	21522240	21522240@gm.uit.edu.vn

## 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Nội dung	Tình trạng	Trang
1	Yêu cầu 1	100%	2 – 5
2	Yêu cầu 2	100%	5 – 7
3	Yêu cầu 3	100%	7 – 8
4	Yêu cầu 4	100%	8 – 11
5	Yêu cầu 5	100%	11 – 13
Điểm tự đánh giá			10/10

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

<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

# BÁO CÁO CHI TIẾT

## Bài tập 1

### A01:2021-Broken Access Control

**Tiêu đề:** Broken Access Control - Lỗ hổng kiểm soát truy cập bị hỏng, lỗ hổng này giúp attacker có quyền truy cập vào các tài nguyên không được cho phép.

**Mô tả lỗ hổng:**

Broken Access Control - Lỗ hổng kiểm soát truy cập bị hỏng chỉ các vấn đề liên quan đến việc vi phạm nguyên tắc đặc quyền tối thiểu hoặc từ chối theo mặc định, lỗi phân quyền,... khiến cho người dùng thực sự không thể truy cập vào tài nguyên mà họ được phép, hoặc những người dùng thông thường nhưng có được đặc quyền của quản trị viên.

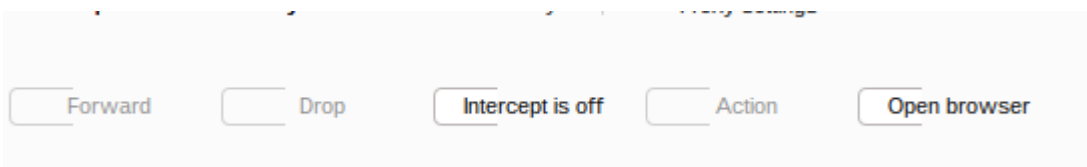
**Trả lời câu hỏi:** Câu truy vấn tại đây đang làm gì?

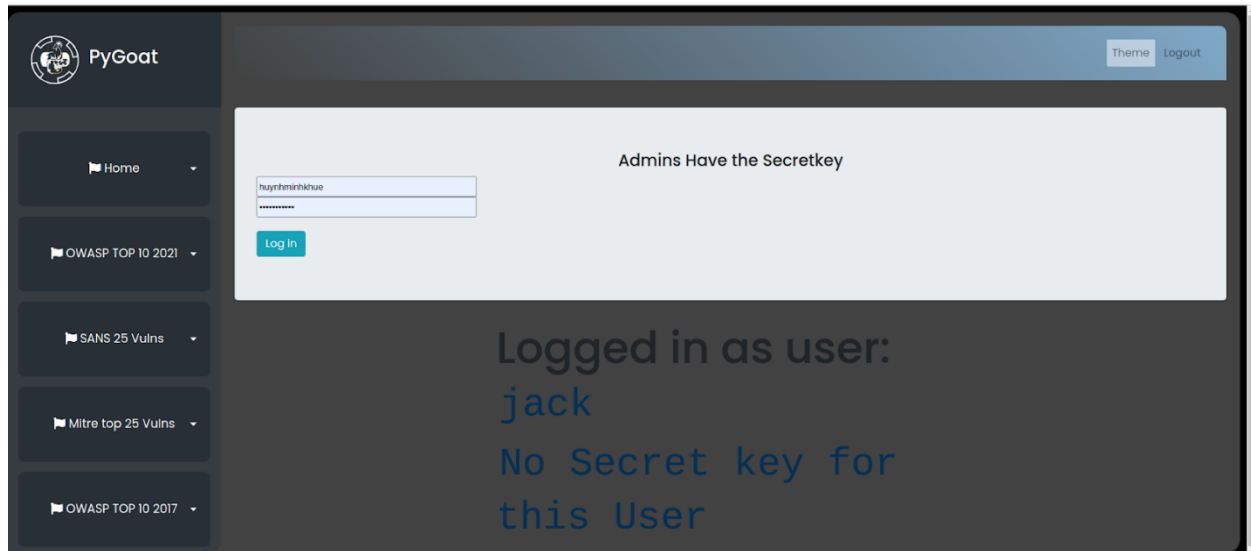


⇒ Câu truy vấn tại đây được sử dụng để gửi thông tin biểu mẫu (tên tài khoản, mật khẩu) lên server để yêu cầu đăng nhập.

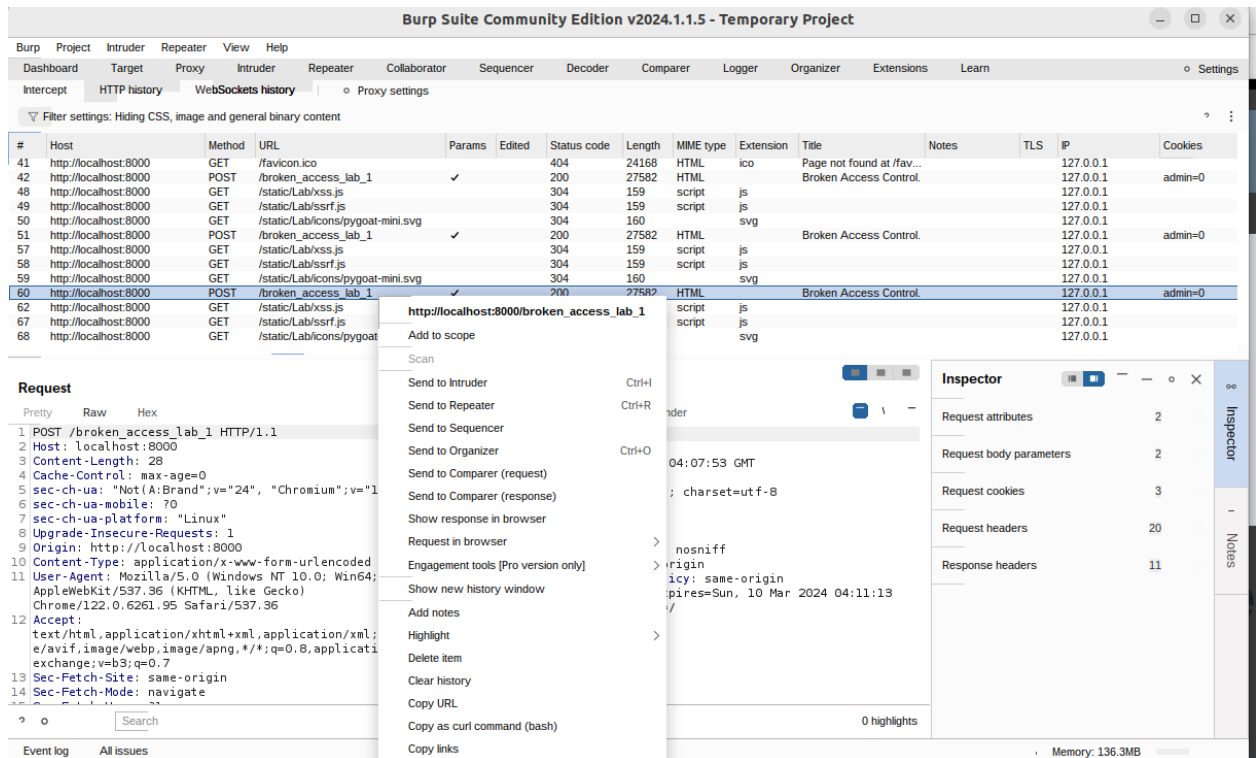
**Các bước thực hiện:** sử dụng repeater

**Bước 1:** Mở Intercept để chặn gói tin. Đăng nhập với tên tài khoản “jack” và mật khẩu “jacktheripper”.

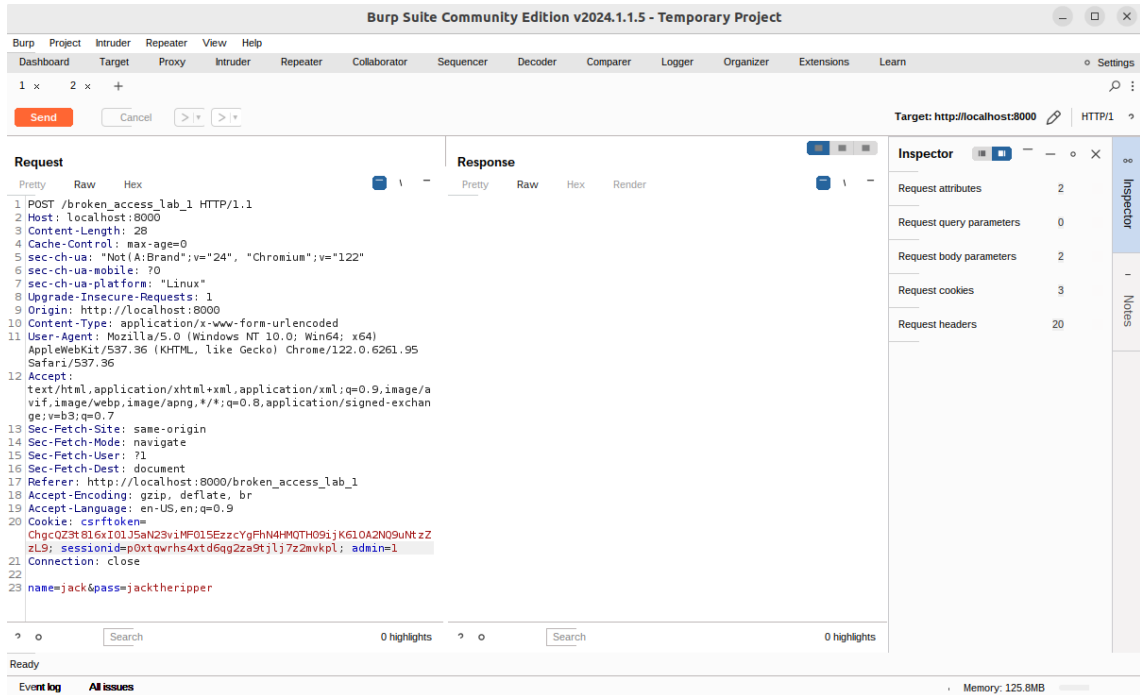




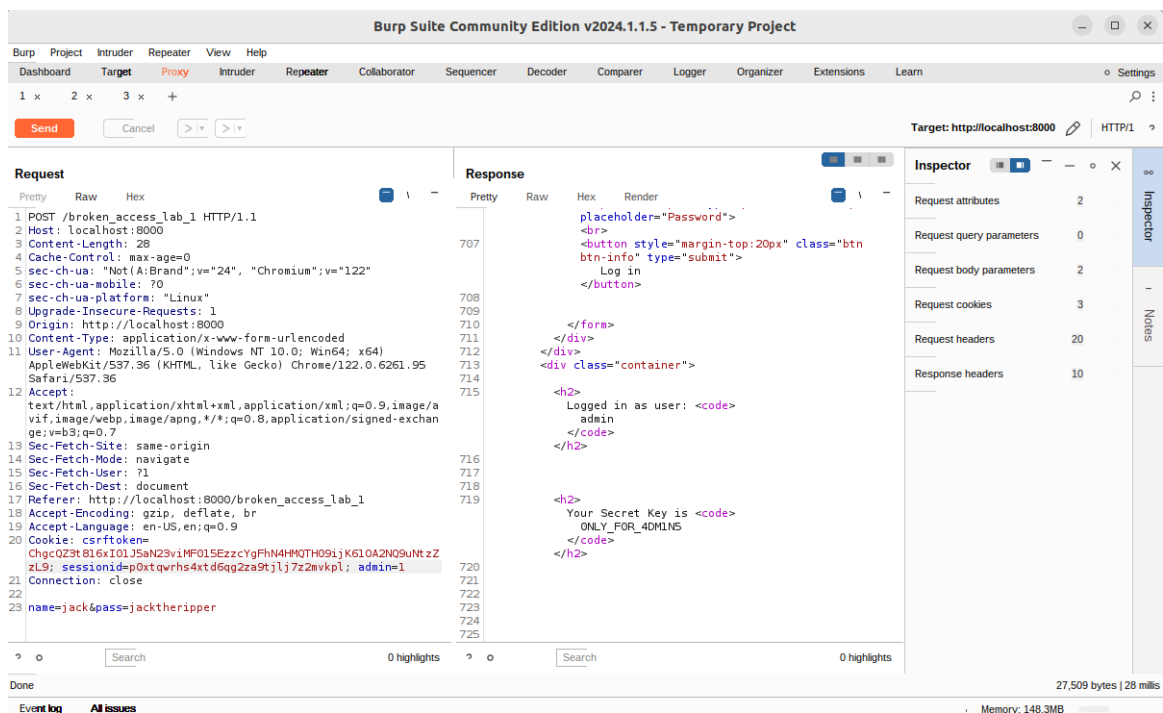
**Bước 2:** Mở tab HTTP history để xem lịch sử các câu truy vấn đã gửi. Chọn câu truy vấn phương thức POST với URL là broken\_access\_lab\_1 rồi gửi đến repeater.



**Bước 3:** Mở tab Repeater lên, ta thấy nội dung của request, chỉnh sửa ở phần cookie: sửa admin=0 thành admin=1. Bấm “Send” để gửi request đi.



**Bước 4:** Sau khi gửi request đi, ta nhận được response. Trong nội dung của response có chứa secret key của admin.



### Mức độ ảnh hưởng của lỗi hỏng:

Lỗi hỏng này có thể gây cản trở người dùng thực sự muốn truy cập vào tài nguyên hệ thống, làm mất thời gian, hao tổn tài nguyên, vi phạm tính sẵn sàng, cũng như cho phép tất cả mọi người đều có quyền như một quản trị viên, có thể chỉnh sửa, thêm bớt dữ liệu tùy ý, làm nhiễu loạn thông tin, khiến cho thông tin bị sai lệch,... vi phạm tính vẹn toàn trong an toàn thông tin (như trong bài này, attacker có thể chỉnh sửa thông tin request để có được secretkey của một quản trị viên).

**Khuyến cáo khắc phục:**

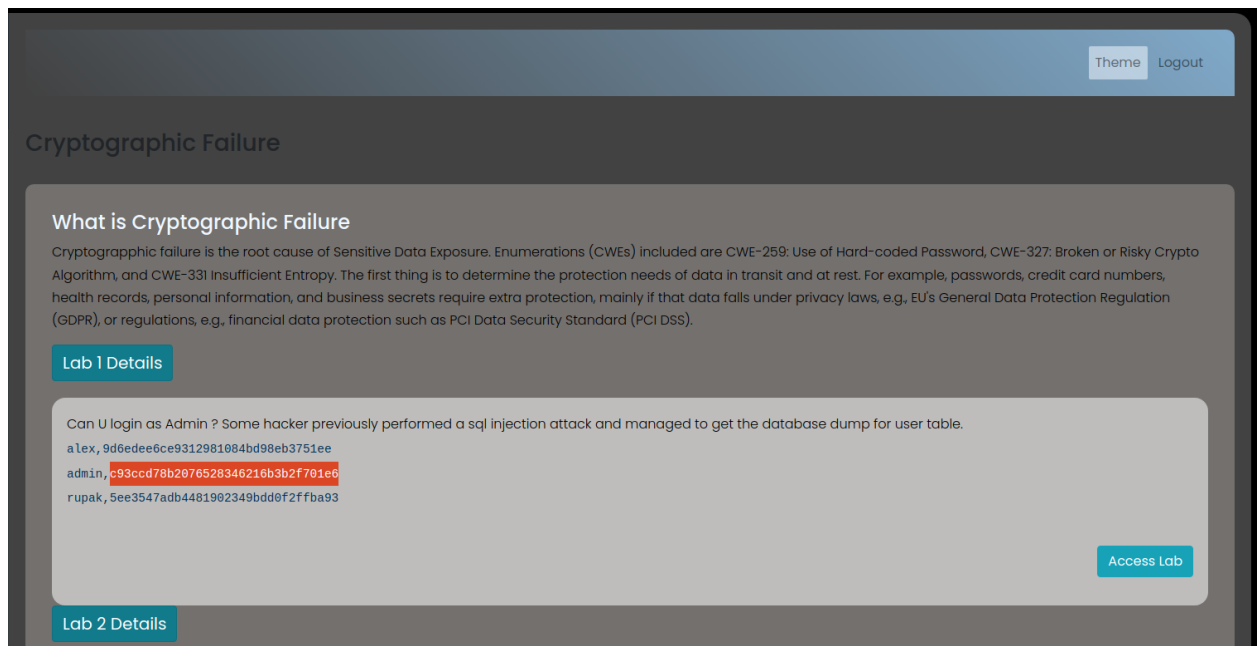
Cần phải kiểm tra và đánh giá quyền truy cập, chỉ phân quyền truy cập một số lượng tài nguyên cần thiết cho các đối tượng người dùng nhất định, không nên phân quyền ngang nhau, cũng như sử dụng các phương pháp xác thực mạnh mẽ để đảm bảo người dùng được xác định chính xác và có quyền truy cập tương ứng.

**Bài tập 2****A02:2021 – Cryptographic Failures**

**Tiêu đề:** Cryptographic Failures - Lỗ hổng Lỗi mật mã ảnh hưởng đến sự bảo mật thông tin riêng tư của người dùng như mật khẩu, thông tin số thẻ tín dụng,...

**Mô tả lỗ hổng:**

Cryptographic Failure - Lỗ hổng Lỗi mật mã chỉ các vấn đề liên quan đến việc sử dụng mật mã trong ứng dụng không đúng cách hoặc thiếu an toàn, gây ra các lỗi bảo mật. Kẻ tấn công có thể dễ dàng lợi dụng lỗ hổng này (trong trường hợp bài này là sử dụng thuật toán mã hóa đơn giản để mã hóa mật khẩu lưu trong cơ sở dữ liệu) để tìm ra thông tin đã được mã hóa.



**Trả lời câu hỏi:** Đoạn chuỗi ký tự trên là gì?

⇒ Đoạn chuỗi ký tự trong phần “Lab 1 Details” là mã hash MD5 của mật khẩu của tài khoản “admin” được lưu trong cơ sở dữ liệu.

**Các bước thực hiện:**

Bước 1: Copy mã hash MD5 của tài khoản admin.

Theme Logout

### Cryptographic Failure

#### What is Cryptographic Failure

Cryptographic failure is the root cause of Sensitive Data Exposure. Enumerations (CWEs) included are CWE-259: Use of Hard-coded Password, CWE-327: Broken or Risky Crypto Algorithm, and CWE-331: Insufficient Entropy. The first thing is to determine the protection needs of data in transit and at rest. For example, passwords, credit card numbers, health records, personal information, and business secrets require extra protection, mainly if that data falls under privacy laws, e.g., EU's General Data Protection Regulation (GDPR), or regulations, e.g., financial data protection such as PCI Data Security Standard (PCI DSS).

**Lab 1 Details**

Can U login as Admin ? Some hacker previously performed a sql injection attack and managed to get the database dump for user table.

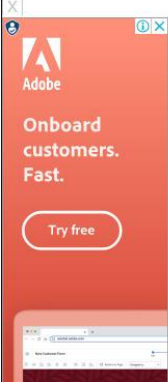
```
alex,9d6edee6ce9312981084bd98eb3751ee
admin,c93ccd78b2076528346216b3b2f701e6
rupak,5ee3547adb4481902349bdd0f2ffba93
```

Access Lab

**Lab 2 Details**

**Bước 2:** Nhập mã hash cần tìm và bấm “Decrypt” để giải mã. Đã giải mã ra được mật khẩu của tài khoản “admin”. Mật khẩu là: admin1234.

← → ↺ <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

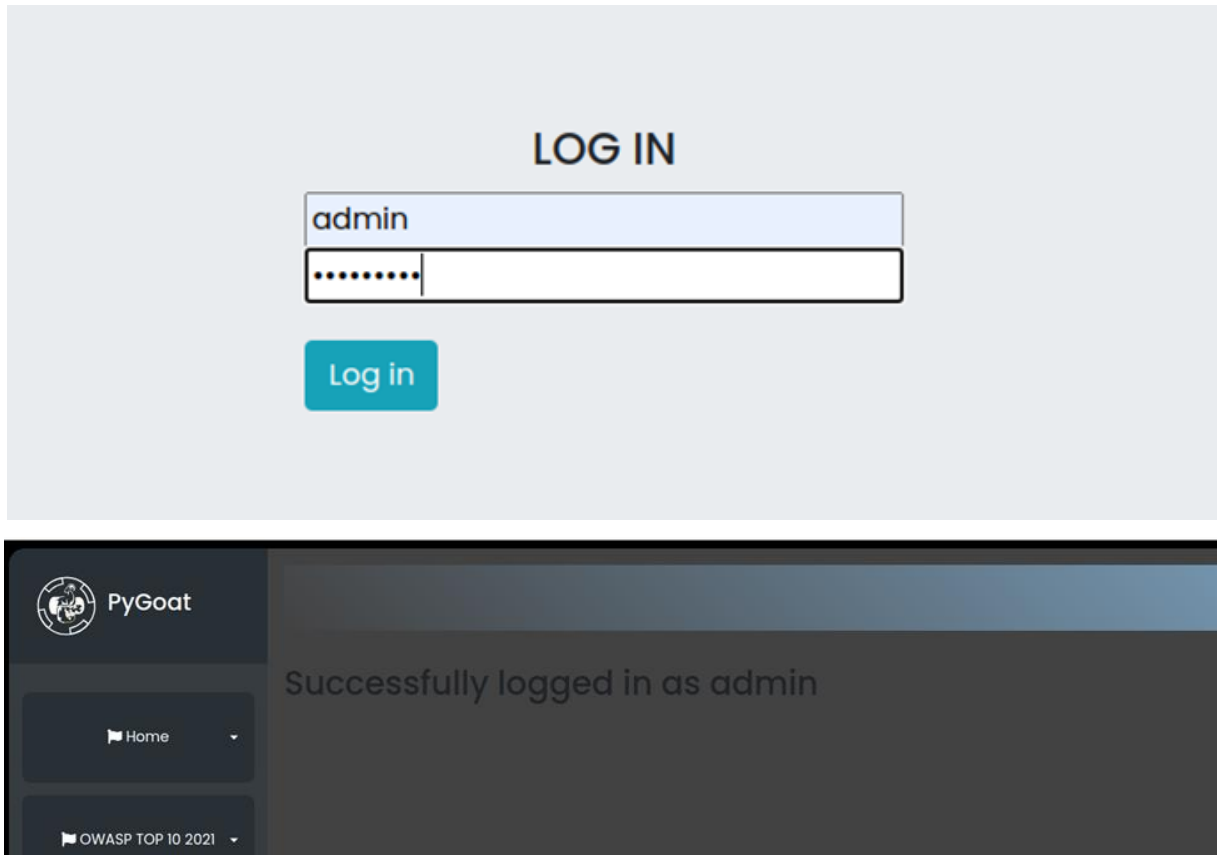
Enter your MD5 hash below and cross your fingers :

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

Loading...

Found : admin1234  
(hash = c93ccd78b2076528346216b3b2f701e6)

Bước 3: Đăng nhập vào tài khoản “admin” bằng mật khẩu vừa tìm được “admin1234”. Kết quả đăng nhập thành công.



### Mức độ ảnh hưởng của lỗ hổng:

Lỗi này giúp cho kẻ tấn công có được thông tin nhạy cảm, riêng tư của nạn nhân một cách dễ dàng và có thể sử dụng chúng để trục lợi hoặc thực hiện các hành vi phi pháp (trong bài này là mật khẩu của tài khoản “admin”).

### Khuyến cáo khắc phục:

Sử dụng các thuật toán mã phức tạp hơn cũng như là các hàm băm an toàn để mã hóa các thông tin quan trọng, bảo vệ dữ liệu trong truyền thông bằng cách sử dụng các giao thức truyền thông an toàn, thiết kế mật mã an toàn và kiểm tra, cập nhật hệ thống bảo mật định kỳ.

## Bài tập 3

### A03:2021 – Injection

**Tiêu đề:** Injection là kiểu tấn công liên quan tới một trình biên dịch và một đoạn payload được chèn vào để khiến trình biên dịch thực hiện các hành động mà nhà phát triển không mong muốn

**Mô tả lỗ hổng:** SQL Injection là một kiểu tấn công phổ biến của Injection. Đây là một kiểu tấn công nhắm vào cơ sở dữ liệu SQL, cho phép người dùng cấp các tham số của riêng họ cho một truy vấn SQL, thường dẫn đến một cơ sở dữ liệu bị xâm phạm như kẻ tấn công có thể đọc dữ liệu trái phép, dữ liệu bị thay đổi, thực thi quyền của người quản trị,..



**Trả lời câu hỏi:** Nếu trang web thực hành bị lỗi tiêm SQL thì khai thác như thế nào?

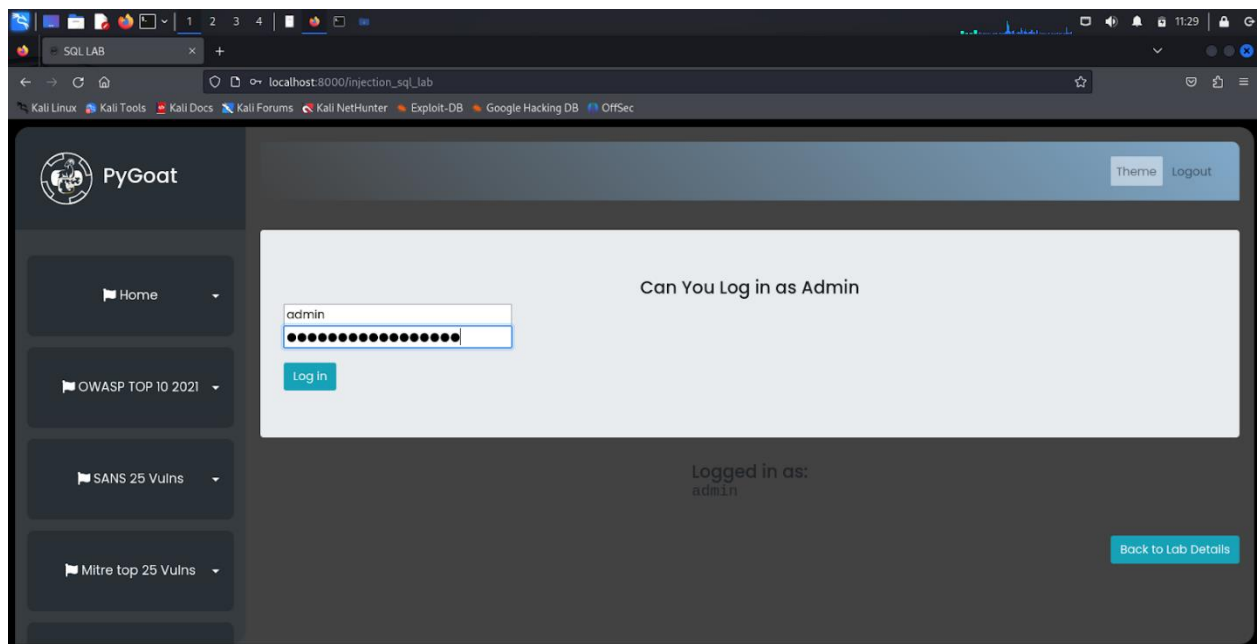
⇒ Nếu trang web thực hành bị lỗi team SQL thì có thể bị khai thác bằng cách thử các username phổ biến như “admin” và chọn password có thêm cụm từ “OR ‘1’ == ‘1’”

**Các bước thực hiện:**

**Bước 1:** Ta biết được trang web sử dụng câu truy vấn SQL sau cho quá trình đăng nhập  
"SELECT \* FROM introduction\_login WHERE user='"+name+"'AND  
password='"+password+'"

Như vậy ta chỉ cần nhập username là admin và nhập mật khẩu bất kì, rồi thêm điều kiện OR ‘1’ == ‘1’ vào cuối là ta có quyền hạn truy cập của admin.

**Bước 2:** Thực hiện nhập username và password như đã phân tích ở trên



**Mức độ ảnh hưởng của lỗ hổng:** Làm lộ dữ liệu trong database, gây hậu quả từ nhẹ cho tới vô cùng nghiêm trọng. Dữ liệu sẽ bị sửa chữa hoặc xóa toàn bộ dữ liệu khiến hệ thống ngừng hoạt động

**Khuyến cáo khắc phục:** Cần lọc dữ liệu từ người dùng, cũng như không được dùng cộng chuỗi khi truy vấn SQL. Ngoài ra không được hiển thị lỗi trả về khi truy vấn vì có thể bị kẻ tấn công lợi dụng để khai thác SQL Injection.

## Bài tập 4

### A04:2021 – Insecure Design

**Tiêu đề:** Insecure Design - Lỗ hổng thiết kế không an toàn ảnh hưởng đến sự bảo mật và toàn vẹn thông tin và tài sản của người dùng, doanh nghiệp (trường hợp bài này là lấy vé xem phim vượt quá số lượt quy định).

**Mô tả lỗ hổng:**

Insecure Design - Lỗ hổng thiết kế không an toàn chỉ những lỗi, thiết kế không đảm bảo đủ mức độ bảo mật, dễ bị tấn công hoặc không thể ngăn chặn được các mối đe dọa an ninh. Các lỗ hổng này có thể cho phép tin tặc xâm nhập vào hệ thống, truy cập dữ liệu nhạy cảm, hoặc thực hiện các hành động không được phép.



**Trả lời câu hỏi:** Lỗi thiết kế không an toàn nằm ở đâu? Chú ý là web được tạo nhiều tài khoản.

⇒ Lỗi thiết kế không an toàn nằm ở việc đăng kí tài khoản, một người được đăng kí số lượng tài khoản không giới hạn và chương trình cũng không có cơ chế kiểm tra tài khoản người dùng.

**Các bước thực hiện:**

**Bước 1:** Mỗi người được phép nhận 5 vé xem phim miễn phí nên chúng ta sẽ điền số 5 - số vé tối đa nhận được và nhấn nút "claim" để lấy vé về. Phía bên tay phải là các ký tự của 5 vé xem phim chúng ta vừa lấy được.

The screenshot displays a web application interface with a dark background. On the left, there are two light gray rectangular boxes. The top box is titled 'Claim Upto 5 Free Tickits' and contains a text input field with the number '0' and a teal 'Claim' button below it. The bottom box is titled 'Watch Movie' and contains a text input field with the word 'Tickit' and a teal 'Watch' button below it. On the right side, there is a light gray rectangular box titled 'My Tickets' which lists five alphanumeric strings: NICgFbkZSb, wGDUMatIAU, WmeZyZzFxX, XLNyaTTTIG, and GzrWbKuwcv.

**Bước 2:** Chúng ta không thể xem phim bây giờ bởi vì phải chờ đến khi 60 vé xem phim được bán hết (còn lại 55 vé). Tuy nhiên, chương trình không giới hạn tài khoản một người có thể đăng ký và không xác thực tài khoản. Chúng ta sẽ lợi dụng lỗ hổng này bằng cách tạo nhiều tài khoản ảo để lấy hết số vé còn lại.

Trên cùng một máy tính, chúng ta sẽ tạo các tài khoản khác và kiểm tra số vé còn lại sau mỗi lần lấy vé.

⇒ Có thể lấy vé được và số vé ngày càng giảm.

The image displays two sequential screenshots of a web application interface for claiming movie tickets. Both screenshots show a 'Login' form on the left, a 'Claim Upto 5 Free Tickets' section in the middle, and a 'My Tickets' list on the right.

**Top Screenshot:**

- Wait until all tickets are sold (50 tickets left)**
- Login Form:** Username: minhngoc, Password: [masked]. Buttons: Login, ClickHere to register | Login with Google.
- Claim Upto 5 Free Tickets:** Input field: 0, Claim button.
- Watch Movie:** Input field: Ticket, Watch button.
- My Tickets:** kJAhvuqmQH, vzdfOMsLkc, LokailfKuN, ycVjqauKRv, pmvzqlYaa.

**Bottom Screenshot:**

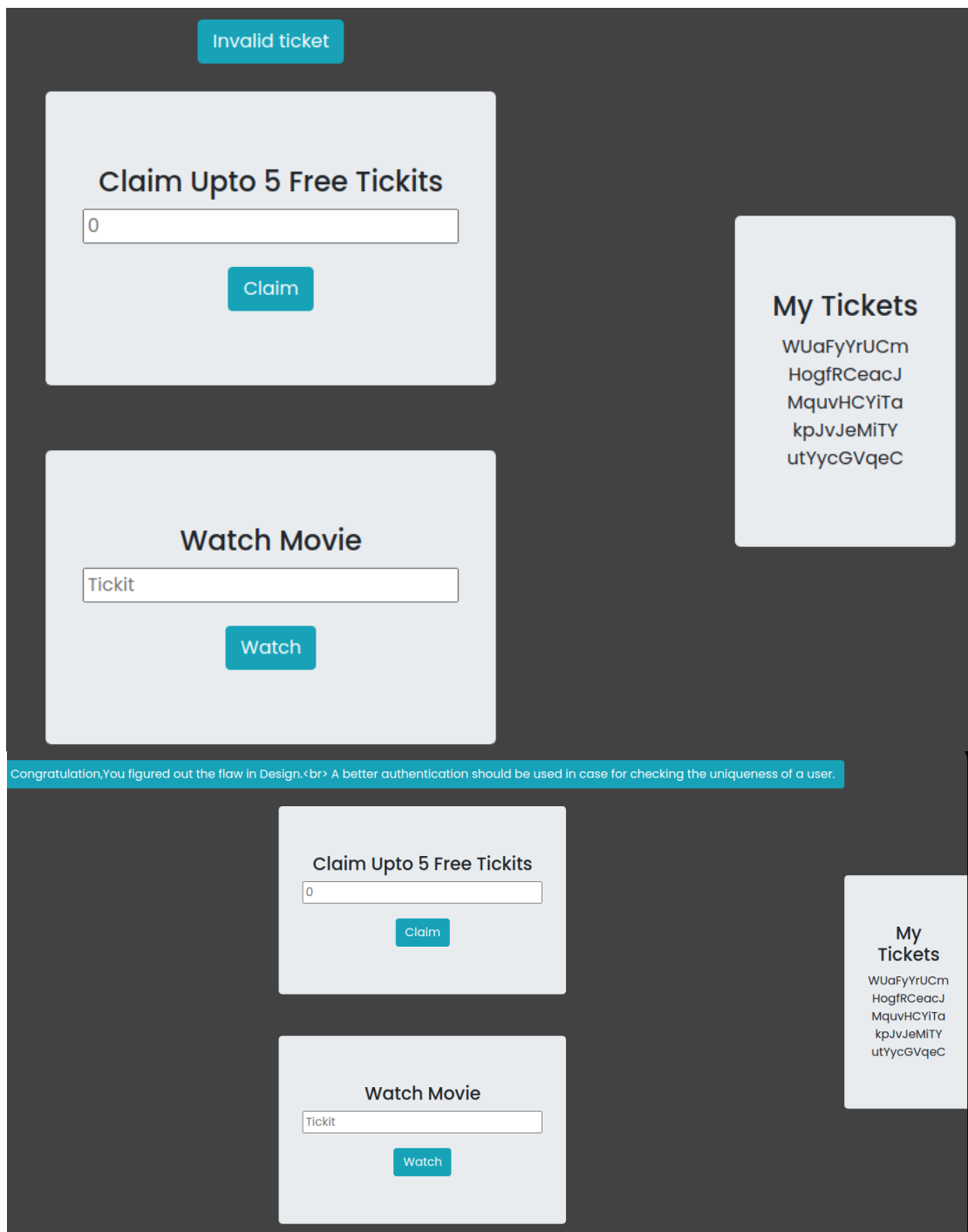
- Wait until all tickets are sold (45 tickets left)**
- Login Form:** Username: MinhKhue, Password: [masked]. Buttons: Login, ClickHere to register | Login with Google.
- Claim Upto 5 Free Tickets:** Input field: 0, Claim button.
- Watch Movie:** Input field: Ticket, Watch button.
- My Tickets:** XTpXpHRuWA, yfoULnBUBg, jKYzTYNZdR, JryiRuASvz, jcuHKMcERE.

⇒ Có thể lấy vé được và số vé ngày càng giảm.

**Bước 3:** Sau khi sử dụng 12 tài khoản để lấy hết 60 vé, bây giờ chúng ta đã có thể xem phim rồi.

**Mức độ ảnh hưởng của lỗ hổng:** Kẻ tấn công có thể lợi dụng các thiết kế không an toàn này để đánh cắp, tráo đổi hoặc phá hỏng thông tin, tài sản của người dùng, hệ thống, doanh nghiệp,... Trong bài này là một người có thể lấy tất cả số vé xem phim của 12 người.

**Khuyến cáo khắc phục:** Sử dụng một thiết kế an toàn hơn, ví dụ, tạo ra có chế kiểm tra, xác thực người dùng (bằng email, số điện thoại,...), giới hạn số tài khoản mà một người có thể đăng ký.



## Bài tập 5

### A05:2021 – Security Misconfiguration

**Tiêu đề:** Cấu hình bảo mật sai có thể xảy ra ở mọi tầng trong 1 ứng dụng như dịch vụ mạng, web server, database, framework,... Cấu hình bảo mật sai sẽ dẫn tới truy cập trái phép tới hệ thống và dữ liệu. Thiệt hại của lỗ hổng này có thể từ nhẹ tới nghiêm trọng

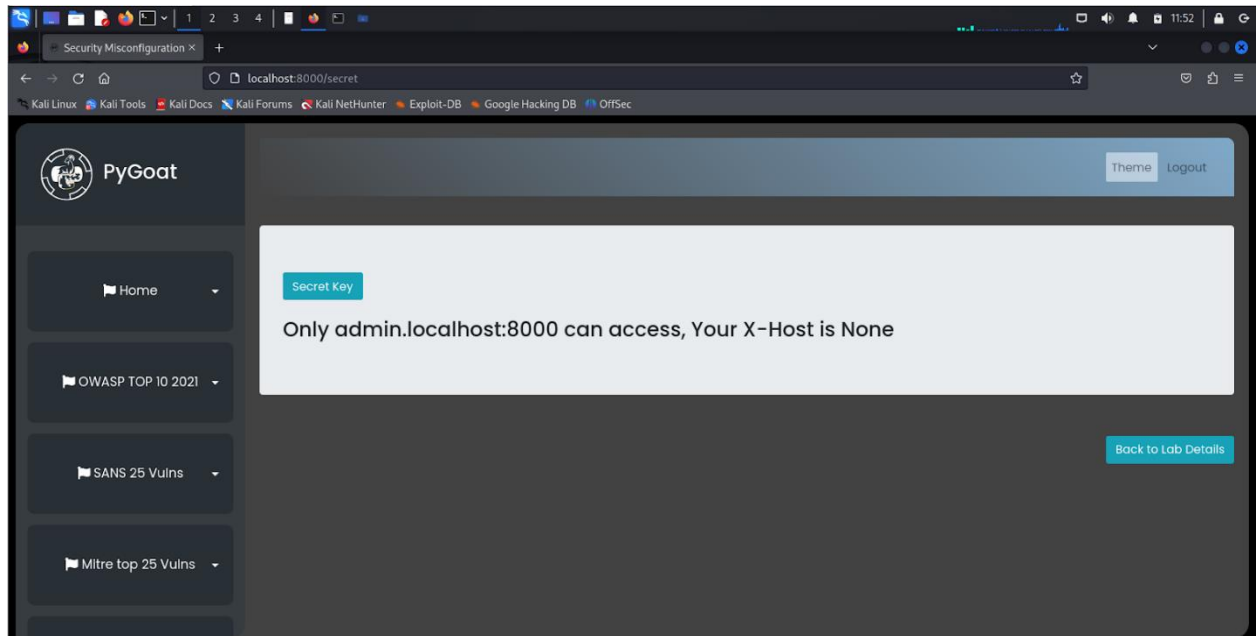
**Mô tả lỗ hổng:** Lỗ hổng trong bài liên quan tới việc thông báo lỗi được thông báo chi tiết tới người dùng. Chính nhờ việc này, kẻ tấn công có thể đoán ra được payload tấn công để tiến hành xâm nhập trái phép

**Trả lời câu hỏi:** X-Host is None là gì? Có kiểm soát được X-Host không.

⇒ X-Host is None nói tới một trường X-Host được tự thiết kế. Trong các gói tin request thông thường không có trường này. Ta có thể kiểm soát được X-Host bằng cách bắt gói tin request, thêm trường này vào gói tin rồi tiếp tục forward tới server.

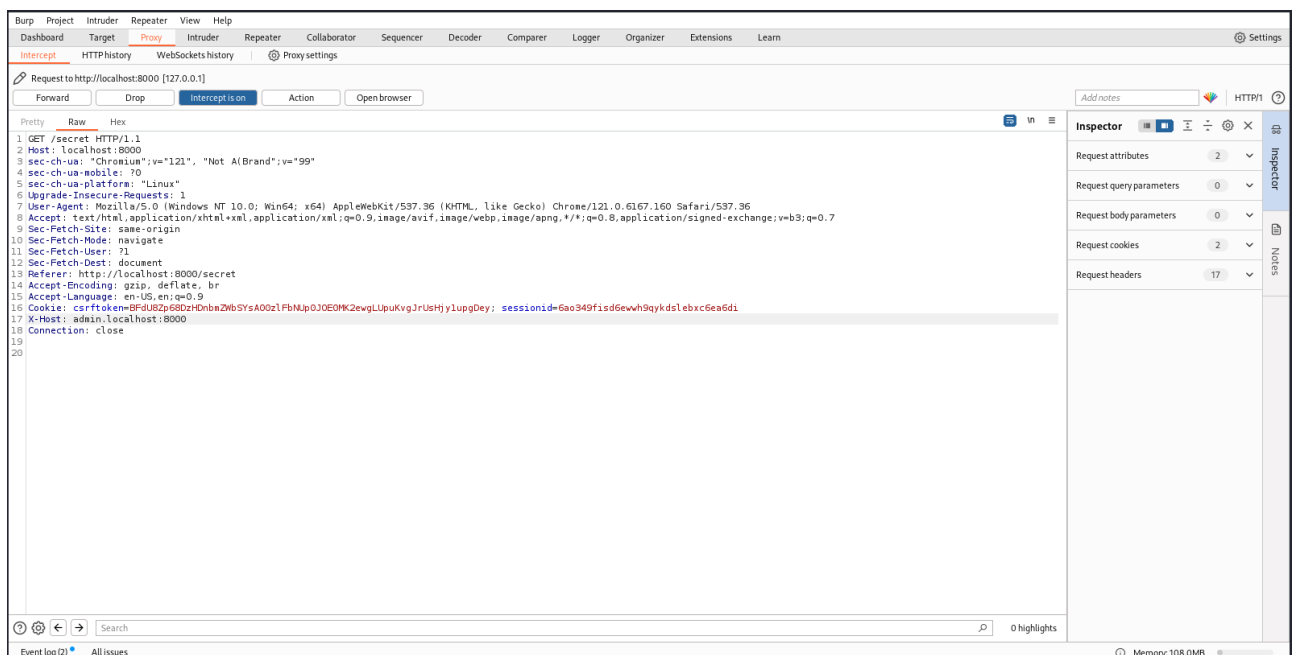
**Các bước thực hiện:**

**Bước 1:** Khi nhấn tìm Secret Key ta sẽ nhận được thông báo lỗi như sau:

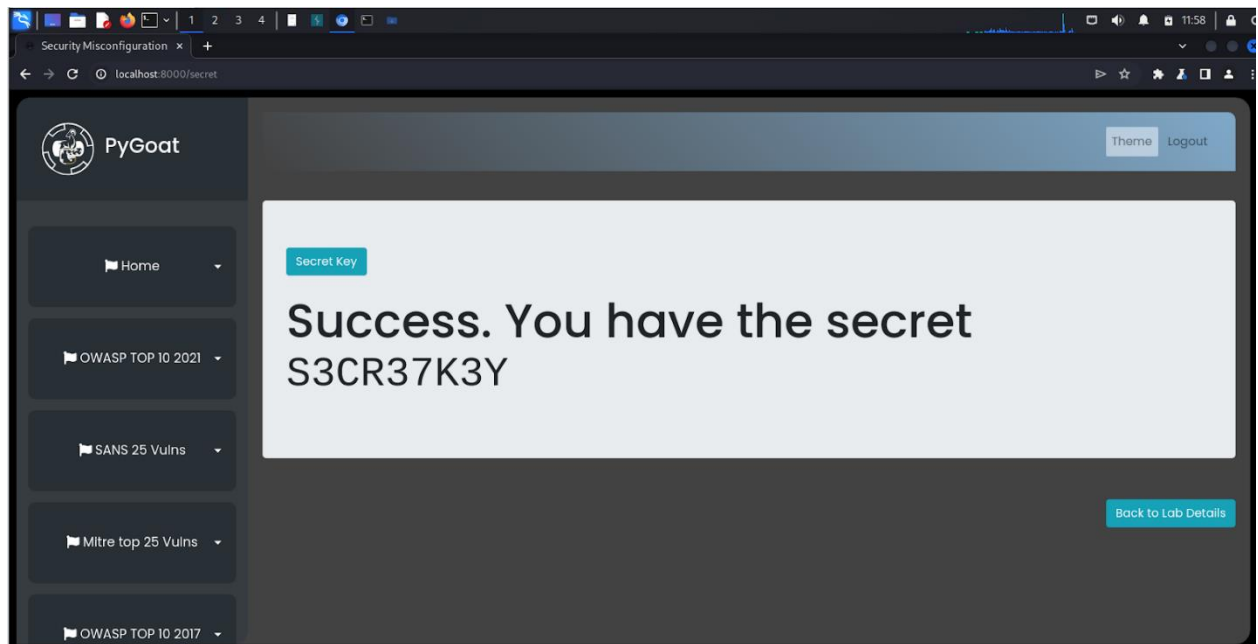


Nghĩa là trong gói tin của ta không có trường X-Host và chỉ có X-Host: admin.localhost:8000 mới có quyền truy cập

**Bước 2:** Mở chế độ Intercept của Burp Suite và bắt gói tin request gửi lên server. Thêm trường X-Host: admin.localhost:8000 vào gói tin



**Bước 3:** Forward gói tin và nhận được secret key cần tìm



**Mức độ ảnh hưởng của lỗ hổng:** Lỗi như vậy thường cho kẻ tấn công quyền truy cập bất hợp pháp đến những chức năng hay dữ liệu hệ thống. Thỉnh thoảng lỗi này cũng có thể giúp kẻ tấn công chiếm toàn bộ hệ thống.

**Khuyến cáo khắc phục:** Cập nhật các bản vá và cập nhật cho các phần mềm của hệ thống. Sử dụng hệ thống có kiến trúc vững chắc, có thể phân tách và bảo vệ các thành phần riêng biệt. Quét và kiểm tra định kì hệ thống để phát hiện những cấu hình sai hoặc bị thiếu.