BÁO CÁO THỰC HÀNH

**Môn học: Bảo mật web và ứng dụng**

**Lab 1: Tổng quan các lỗ hổng bảo mật thường gặp**

*GVHD: Ngô Khánh Khoa*

1. **THÔNG TIN CHUNG:**

*(Liệt kê tất cả các thành viên trong nhóm)*

Lớp: NT213.O21.ATTN.1

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Họ và tên** | **MSSV** | **Email** |
| 1 | Phạm Ngọc Thơ | 21522641 | 21522641@gm.uit.edu.vn |

1. **NỘI DUNG THỰC HIỆN:[[1]](#footnote-1)**

|  |  |  |
| --- | --- | --- |
| **STT** | **Công việc** | **Kết quả tự đánh giá** |
| 1 | Bài tập A01 | 100% |
| 2 | Bài tập A02 | 100% |
| 3 | Bài tập A03 | 100% |
| 4 | Bài tập A04 | 100% |
| 5 | Bài tập A05 | 100% |

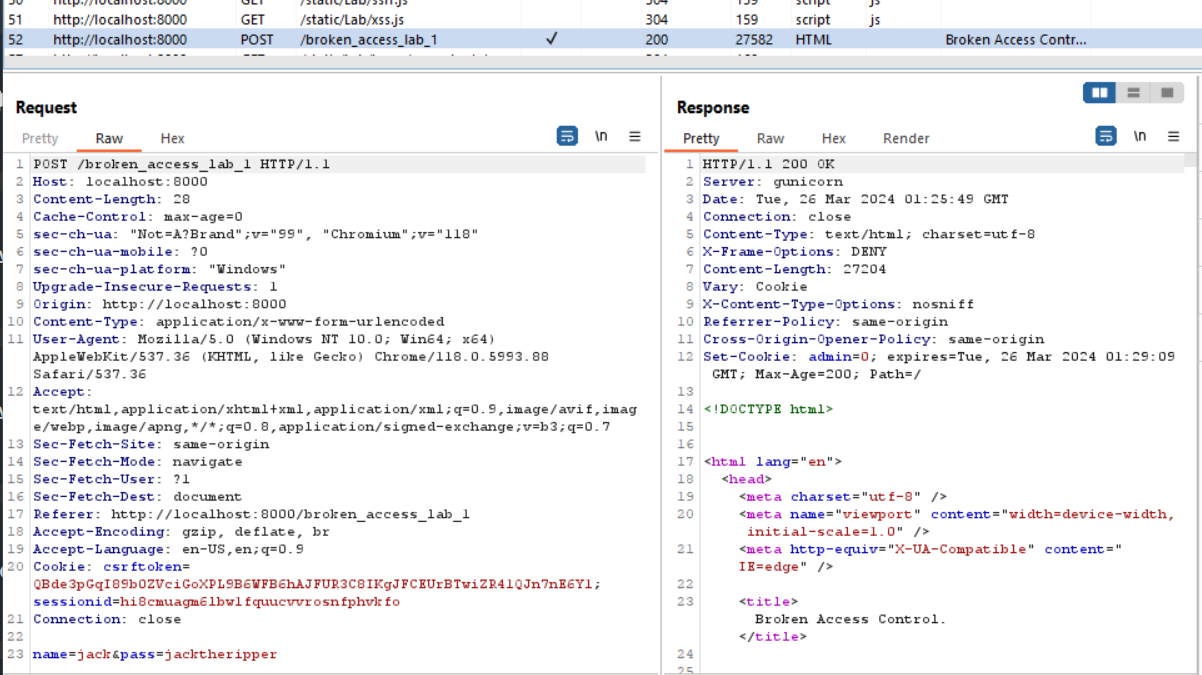
**Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.**

BÁO CÁO CHI TIẾT

1. **A01:2021-Broken Access Control:**

**Chậm lại và suy nghĩ 1:** Câu truy vấn tại đây đang làm gì?

Xét câu truy vấn cuối cùng:



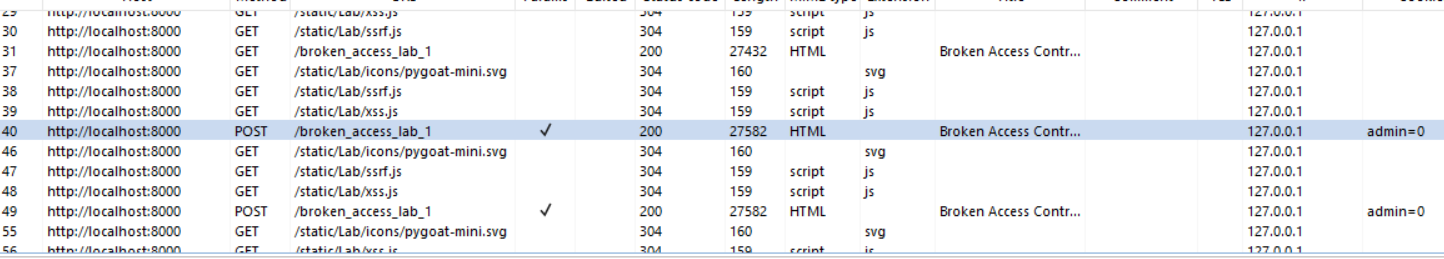
* Phía user gửi request bằng gói tin POST, truy cập vào web http://localhost:8000/broken\_access\_lab\_1. Nội dung gửi gồm name là “jack” và password là “jacktheripper”.
* Tại phía server, set cookie với “admin=0” kèm thời gian hết hạn của cookie. Gửi phản hồi cho user với nội dung “No Secret key for this User”:



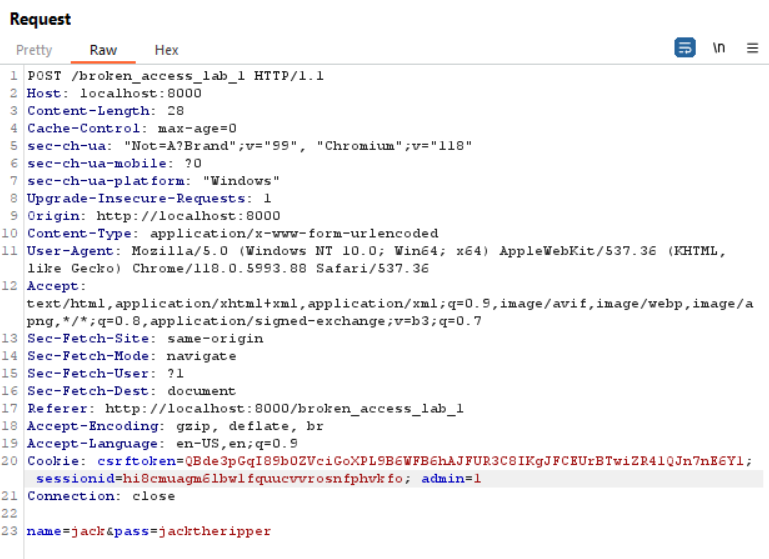
* Sau khi server phản hồi với “admin=0” thì nó đã được thêm vào cookie trong các gói request tiếp theo của user:

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

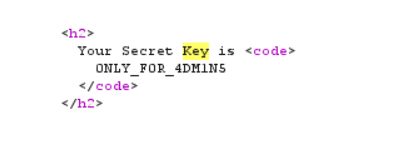
* Chọn 1 gói tin để sửa giá trị “admin” từ 0 thành 1:



* Chuyển sang tab Repeater, tiến hành chỉnh sửa giá trị và chọn “Send” để gửi gói tin đến server:



* Tại gói tin phản hồi, tìm được secret key là “0NLY\_F0R\_4DM1N5”:



**Bài tập 2:** Báo cáo lỗ hổng đang được thực hành. Có thể sử dụng format theo mẫu sau:

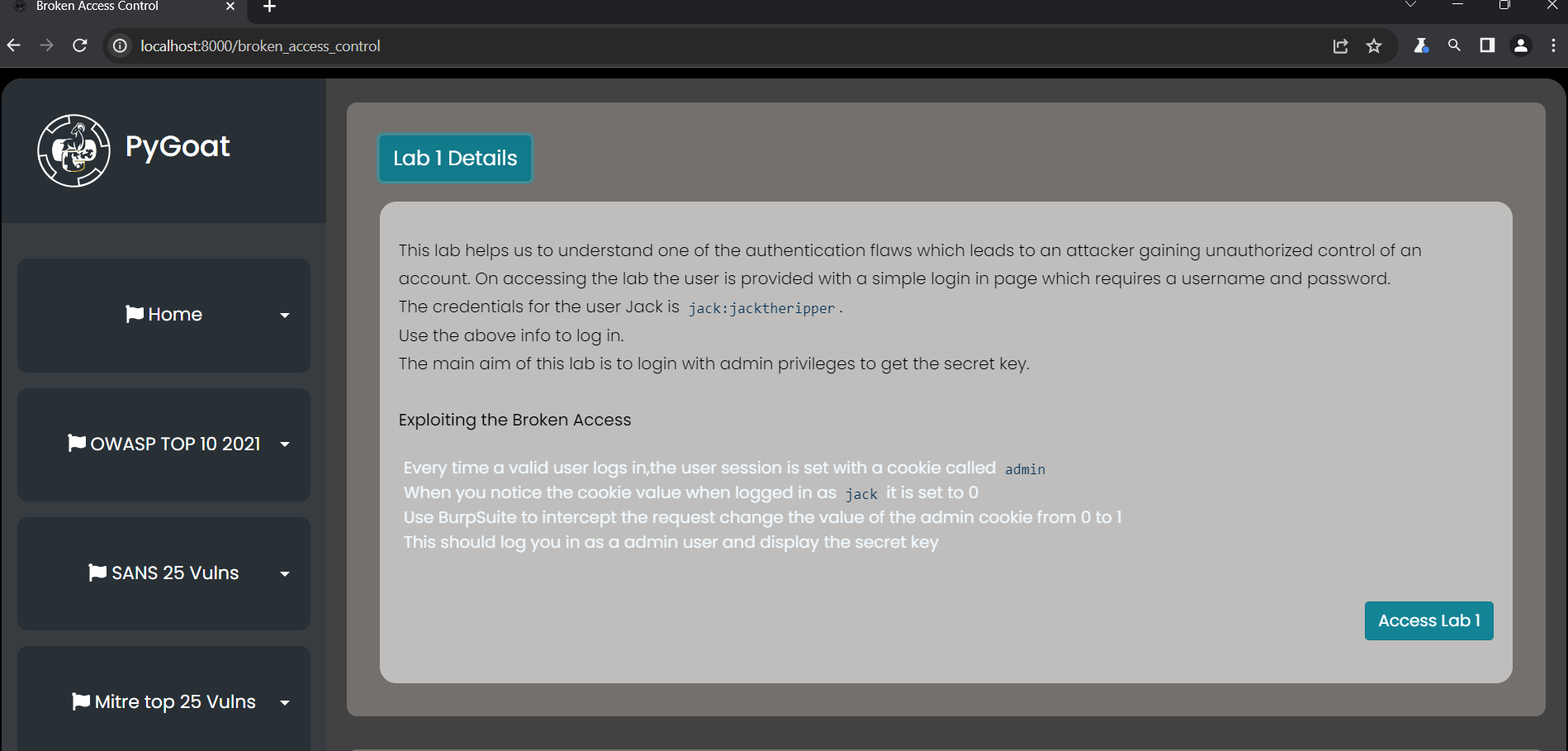
**#Tiêu đề:** Khai thác lỗ hổng Broken Access Control để tìm thông tin nằm ngoài quyền truy cập của người dùng.

**#Mô tả lỗ hổng:**

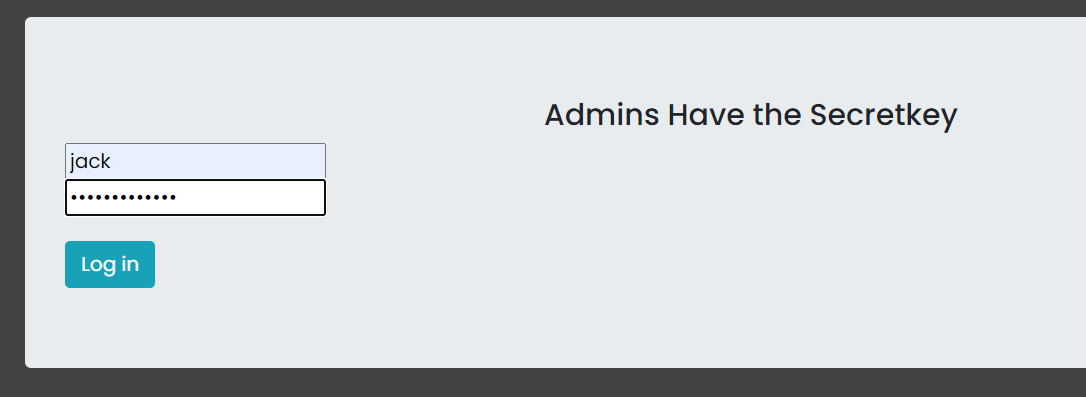
### Tóm tắt: Broken Access Control: là lỗi trong việc phân quyền trong hệ thống. Các lỗi phân quyền thường do thiếu đi các bộ phát hiện lỗi tự động hoặc cách thức kiểm thử hoặc các hàm kiểm thử chưa hiệu quả khiến cho hệ thống bị rò rỉ.

### Các bước để thực hiện lại và bằng chứng:

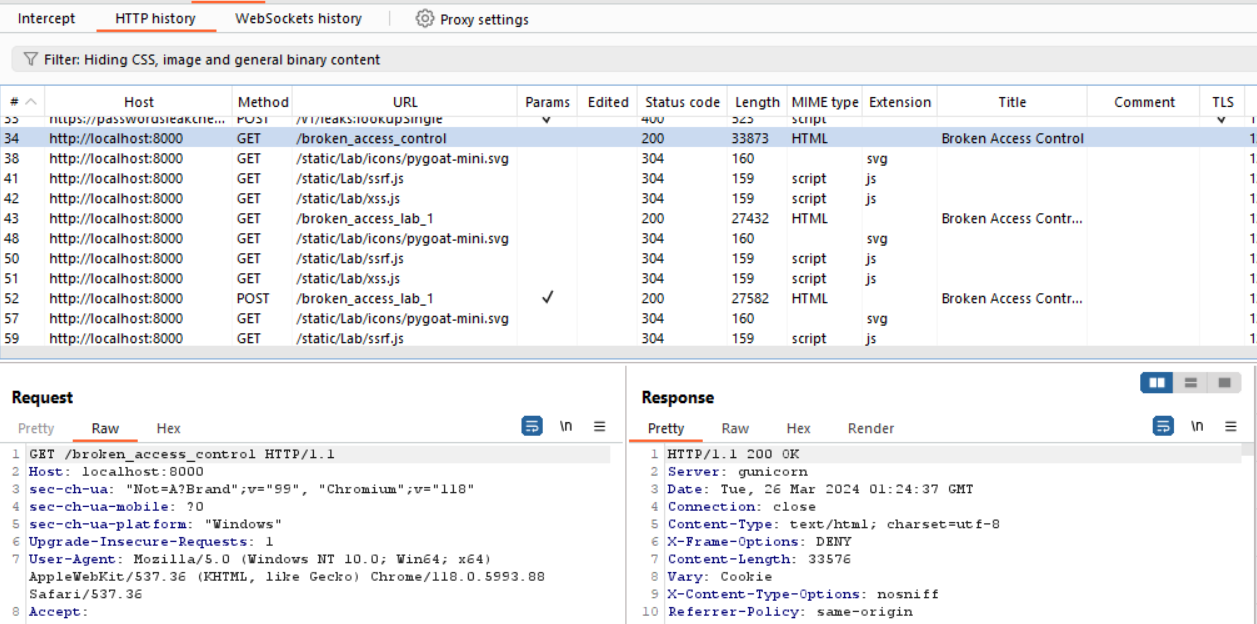
1. bước 1: Truy cập bài thực hành tại http://localhost:8000 => OSWAP TOP 10 2021 => A1: Broken Access Control => Lab 1 Details:



2. bước 2: Đăng nhập với tên và mật khẩu được cung cấp:

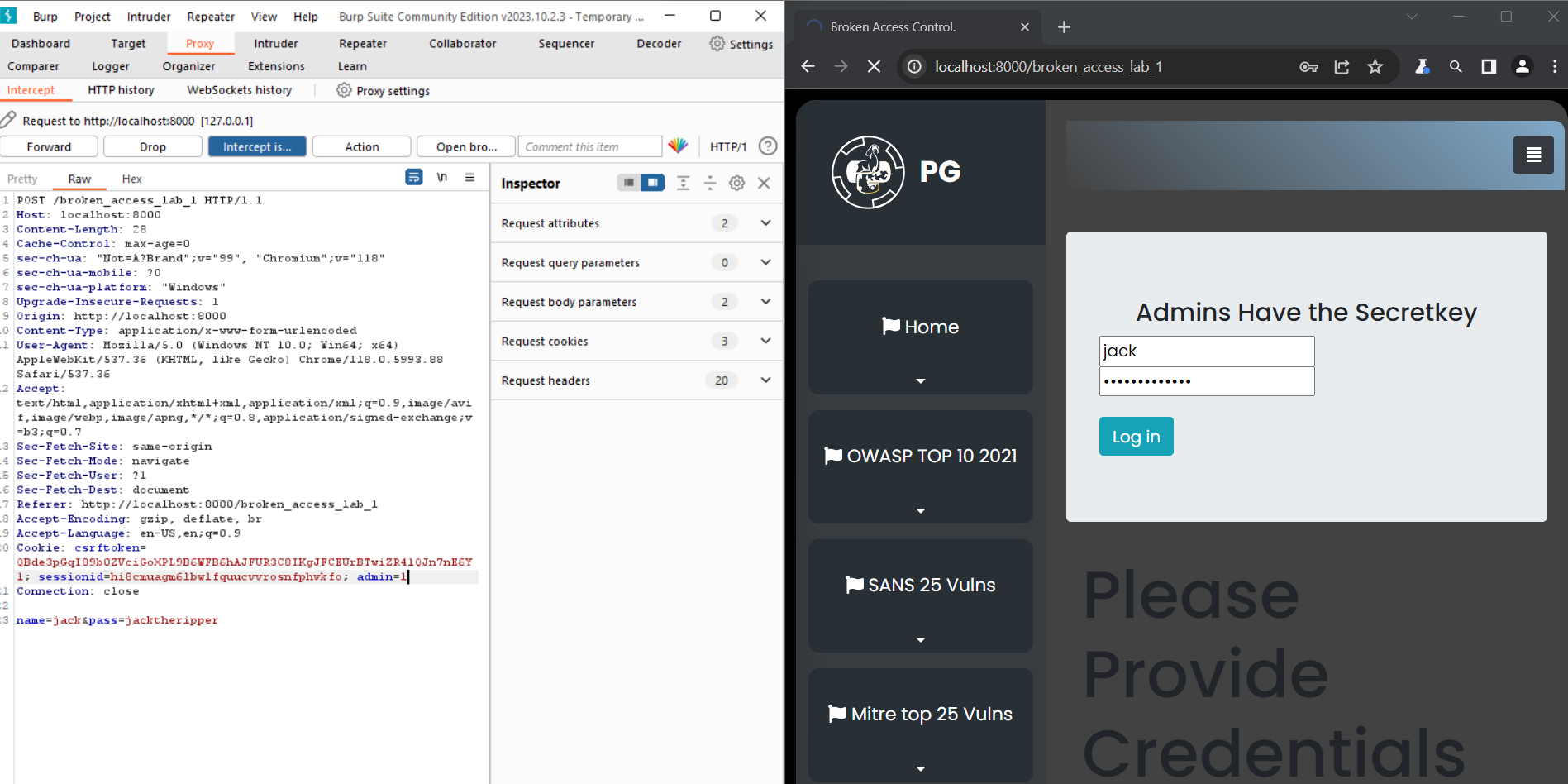


3. bước 3: Trở lại giao diện HTTP history của Burpsuite để kiểm tra lịch sử câu truy vấn để hiểu rõ logic của ứng dụng.

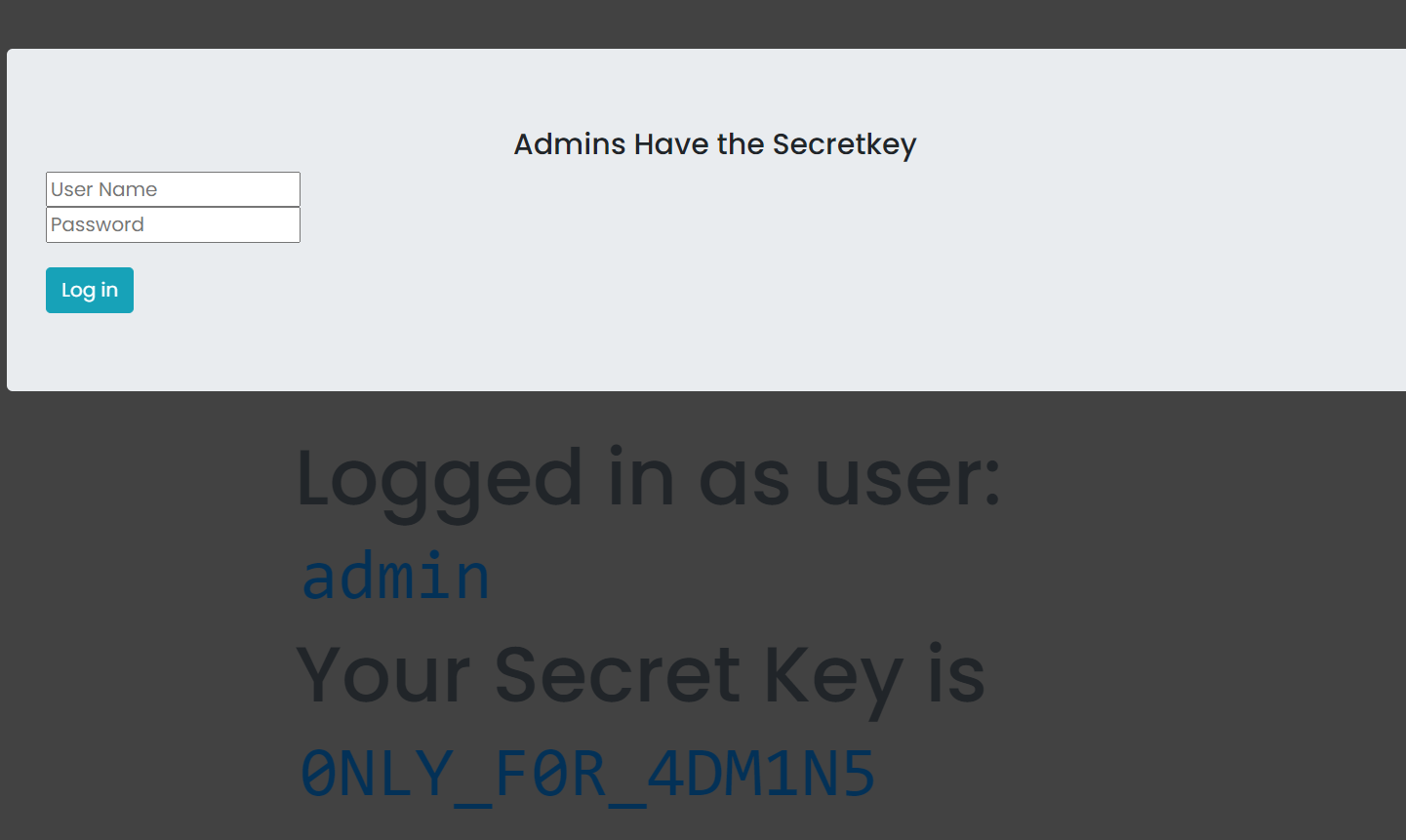


4. bước 4: tiến hành chặn gói tin tại tab Intercept và sửa đổi trực tiếp tại đó:

- Ngay bước đăng nhập, chỉnh lại cookie với “admin=1”, Forward gói tin lên server:



* Kết quả đã đăng nhập được với tư cách admin, trang hiện secret key tương tự như cách 2 (Repeater ở bài tập 1):



### Tài liệu hỗ trợ và tham khảo: lý thuyết tham khảo tại [link](https://viblo.asia/p/top-cac-lo-hong-bao-mat-web-cua-owasp-phan-1-maGK70ebZj2#_khai-niem-14)

\*Liệt kê bất kỳ tài liệu bổ sung nào (ví dụ: ảnh chụp màn hình, nhật ký,...)

**# Mức độ ảnh hưởng của lỗ hổng:** Tác động bảo mật nào mà kẻ tấn công có thể đạt được?

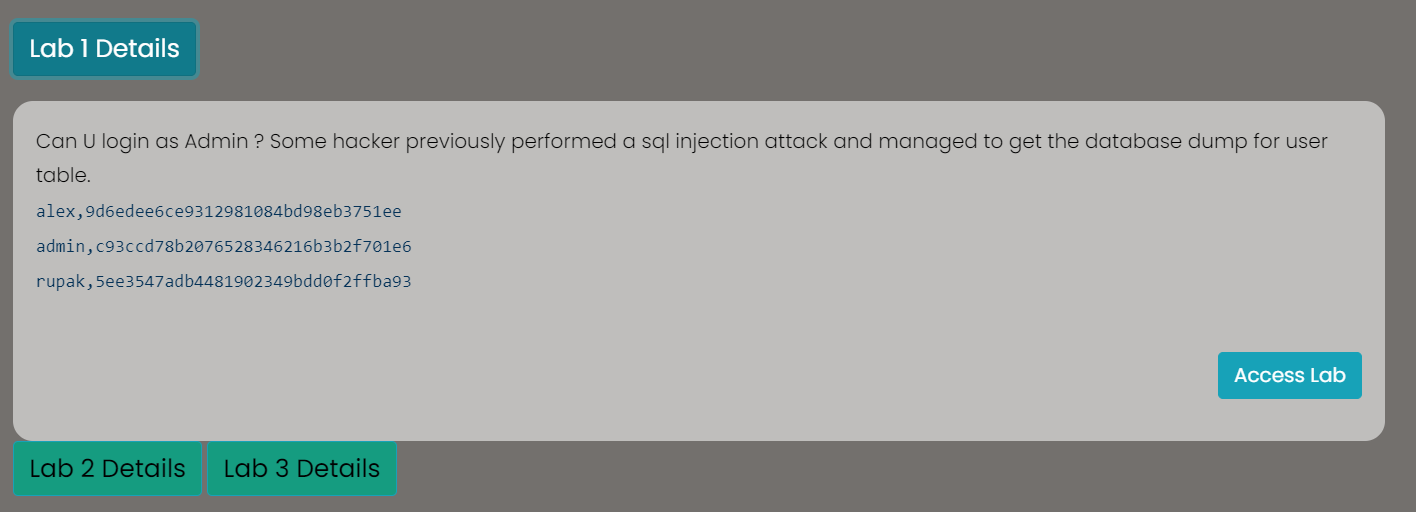
Các hacker có thể truy cập và có các quyền như người dùng hoặc admin, hoặc người dùng có hiểu biết về các hàm của hệ thống cũng có thể gọi và thực thi các hàm này( Ví dụ: Thêm, Sửa, Xóa các bản ghi).

**# Khuyến cáo khắc phục:** Làm thế nào để vá lỗ hổng này?

* Giới hạn quyền truy cập API và controller để giảm thiểu thiệt hại.
* Thực hiện các cơ chế kiểm soát quyền truy cập và thực hiện nó trên toàn ứng dụng.
* JWT tokens nên vô hiệu hóa trên server khi đăng xuất.
* Nên cài đặt các rule ở Model để quản lý các thao tác với database.

1. **A02:2021 – Cryptographic Failures**

**Chậm lại và suy nghĩ 2**: Đoạn chuỗi ký tự trên là gì?



Các đoạn ký tự đọc không hiểu ở trên có thể là password vì tương ứng với mỗi user có 1 đoạn ký tự riêng biệt tạo thành 1 cặp (username, password). Nhìn sơ qua có thể thấy khá giống với kỹ thuật hash md5 nên dùng tool online để decrypt.

**Bài tập 3:** Báo cáo lỗ hổng đang được thực hành. Có thể sử dụng format theo mẫu sau:

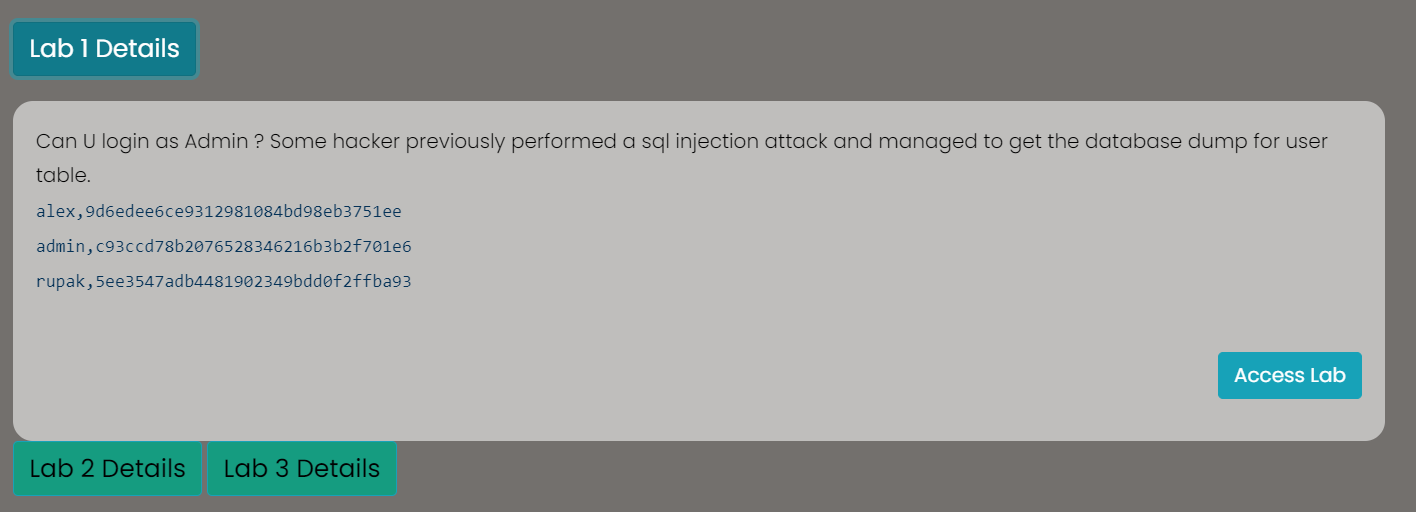
**#Tiêu** **đề:** Lỗ hổng Cryptographic Failures đã trực tiếp gây nên vấn đề lộ/lọt thông tin trong quá trình lưu trữ và truyền tải dữ liệu.

**#Mô tả lỗ hổng:**

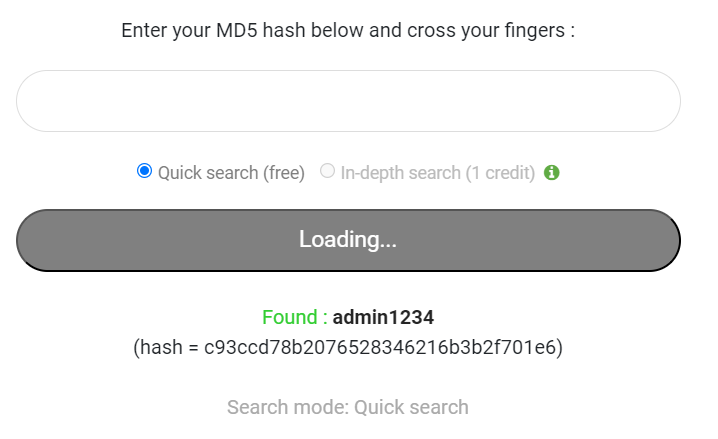
### Tóm tắt: Cryptographic Failures là lỗi bảo mật thông tin không chặt chẽ, dễ bị giải mã.

### Các bước để thực hiện lại và bằng chứng:

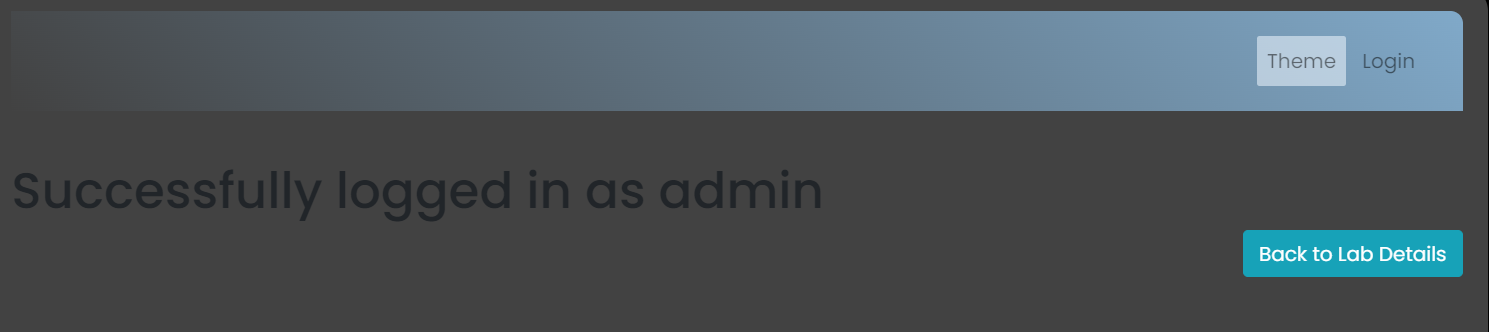
1. bước 1: Đến trang thực hiện lab:



2. bước 2: Để đăng nhập với vai trò admin, em dùng tool thầy đề xuất để decrypt, thu được password là “admin1234”:



3. bước 3: Đăng nhập với thông tin vừa tìm được, thành công:



### Tài liệu hỗ trợ và tham khảo: [tong-quan-ve-top-ten-owasp-2021](https://viblo.asia/p/tong-quan-ve-top-ten-owasp-2021-6J3ZgR7BKmB)

\*Liệt kê bất kỳ tài liệu bổ sung nào (ví dụ: ảnh chụp màn hình, nhật ký,...)

**# Mức độ ảnh hưởng của lỗ hổng:** Phá vỡ tính CIA trong an toàn thông tin, tiết lộ thông tin nhạy cảm, tạo cơ hội để attacker đánh cắp thông tin và xâm phạm hệ thống.

**# Khuyến cáo khắc phục:** Làm thế nào để vá lỗ hổng này?

* Không sử dụng những giao thức cũ như FTP, SMTP,... để vận chuyển dữ liệu nhạy cảm.
* Sử dụng các thuật toán mã hóa mạnh.
* Mã hóa dữ liệu trên đường truyền bằng TLS, HTTPS.
* Lưu trữ password bằng các hàm băm mạnh như Argon2, scrypt, bcrypt,...
* Luôn sử dụng mã hóa được xác thực thay vì chỉ mã hóa.

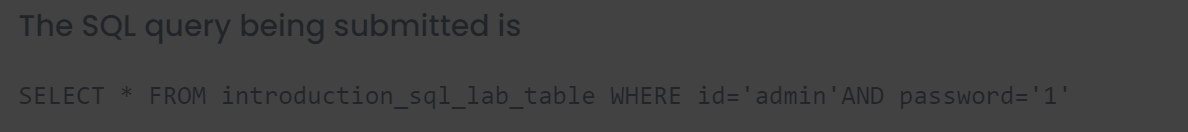
1. **A03:2021 – Injection**

**Chậm lại và suy nghĩ 3:** Nếu trang web thực hành bị lỗi tiêm SQL thì khai thác như thế nào?

* Đối với lab này, câu truy vấn mà chương trình dùng để trích xuất thông tin từ database được trình bày rõ trên màn hình khi ta nhập password sai:

SELECT \* FROM introduction\_sql\_lab\_table WHERE id=’admin’ AND password=’1’

Trong đó, admin và 1 là hai giá trị do em nhập vào. Có thể thấy hai giá trị này, nhập sao thì nó xử lý như vậy mà không thông qua lớp kiểm tra dữ liệu nào, nên ta có thể dựa vào đây để khai thác.



* Để khai thác lỗ hổng SQL, em sẽ nhập password với giá trị như sau:

***ANYTHING’ OR ‘1’=’1***

Ý nghĩa là để duyệt toàn bộ database với điều kiện tìm đến id=’admin’ và password=’anything’ or ‘1’=’1’, nhưng do password có chứa ‘1’=’1’ là điều kiện luôn đúng, do đó kết quả sẽ luôn trả về dù password mình nhập có là giá trị sai. Cuối cùng là đăng nhập thành công!

**Bài tập 4:** Báo cáo lỗ hổng đang được thực hành. Có thể sử dụng format theo mẫu sau:

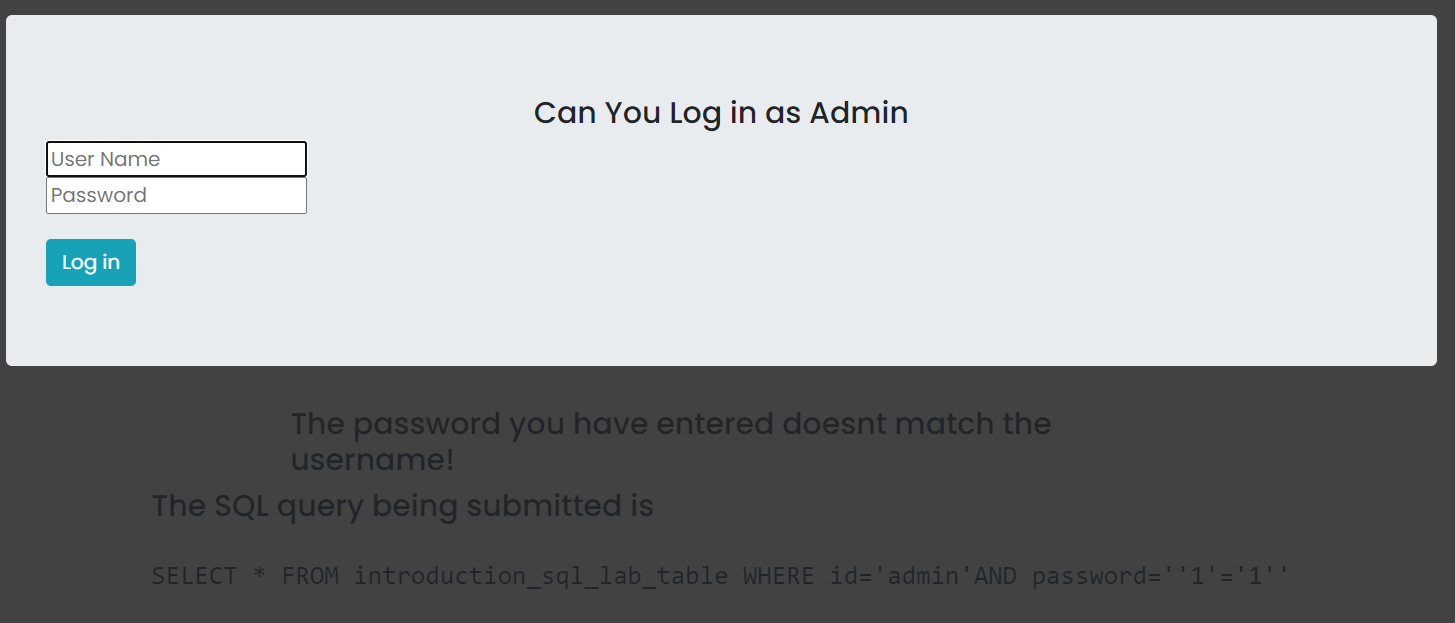
**#Tiêu** **đề:** Injection đang khiến hệ thống trở nên tệ hơn và dữ liệu dễ bị tiết lộ.

**#Mô tả lỗ hổng:** Lỗ hổng này là gì? Trong các bước rõ ràng, làm thế nào để bạn tái tạo nó?.

### Tóm tắt: Injection là lỗ hổng xảy ra do sự thiếu sót trong việc lọc các dữ liệu đầu vào không đáng tin cậy. Khi bạn truyền các dữ liệu chưa được lọc tới Database (Ví dụ như lỗ hổng SQL injection), tới trình duyệt (lỗ hổng XSS), tới máy chủ LDAP (lỗ hổng LDAP Injection) hoặc tới bất cứ vị trí nào khác thì kẻ tấn công có thể chèn các đoạn mã độc để gây ra lộ lọt dữ liệu và chiếm quyền kiểm soát trình duyệt của user.

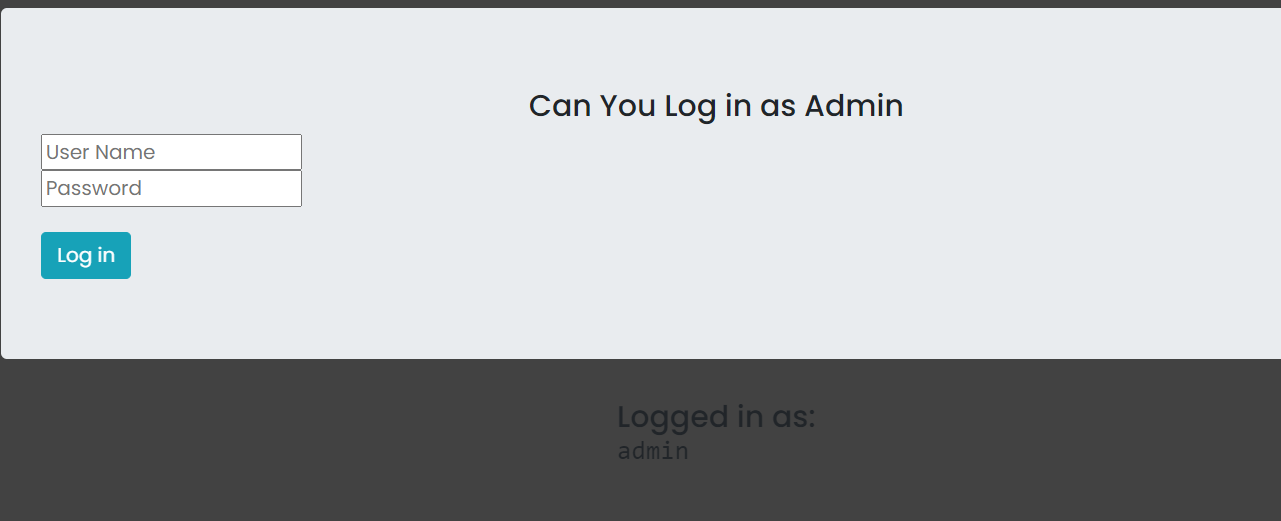
### Các bước để thực hiện lại và bằng chứng:

1. bước 1: Truy cập vào lab, nhập username là admin, password là ‘1’=’1’, nhận được thông báo như sau:



2. bước 2: Lỗi của trang web là xuất ra cả câu truy vấn luôn, nên dựa vào đây ta có thể khai thác lỗ hổng bằng cách nhập password như sau:

***ANYTHING’ OR ‘1’=’1***



3. bước 3: Kết quả là đăng nhập thành công với vai trò admin dù ta không biết mật khẩu chính xác của tài khoản này.

### Tài liệu hỗ trợ và tham khảo: Các trang web trên Google.

\*Liệt kê bất kỳ tài liệu bổ sung nào (ví dụ: ảnh chụp màn hình, nhật ký,...)

**# Mức độ ảnh hưởng của lỗ hổng:** Khai thác SQL thành công có thể đọc dữ liệu nhạy cảm từ cơ sở dữ liệu, sửa đổi dữ liệu cơ sở dữ liệu (Chèn/Cập nhật/Xóa), thực thi các thao tác quản trị trên cơ sở dữ liệu (chẳng hạn như tắt DBMS), khôi phục nội dung của một tệp nhất định có trên tệp DBMS hệ thống và trong một số trường hợp đưa ra lệnh cho hệ điều hành. Đôi khi, còn chiếm được quyền kiểm soát.

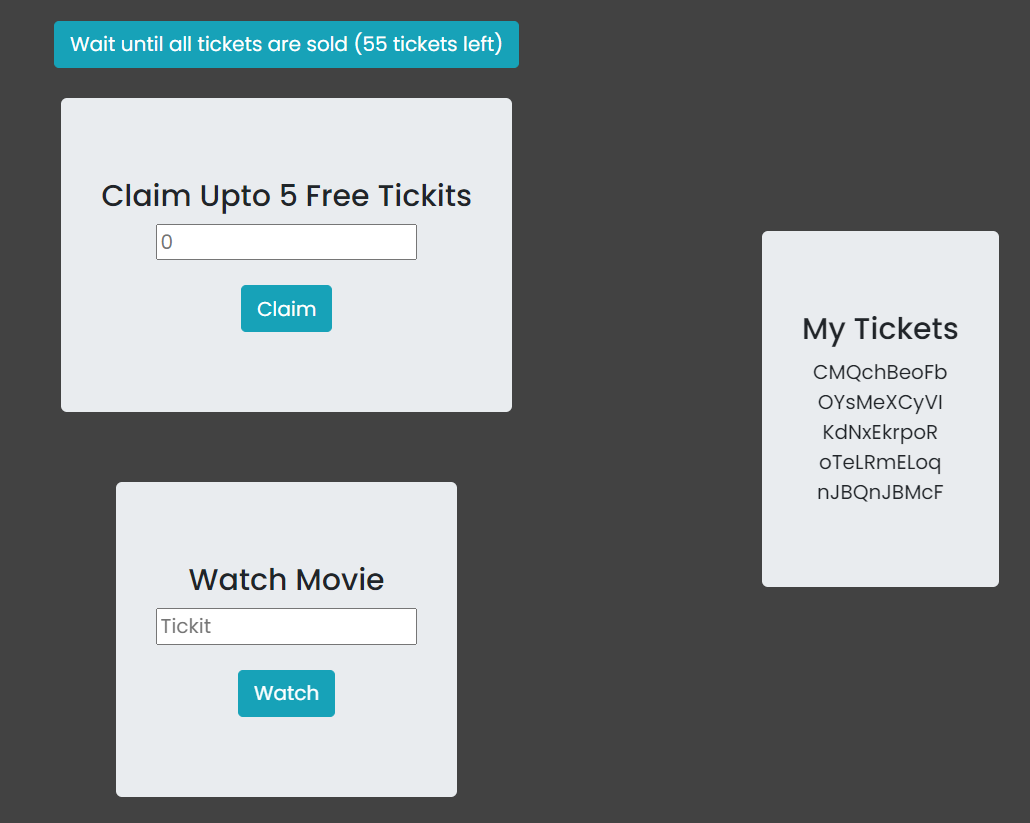
**# Khuyến cáo khắc phục**: phải tách dữ liệu được gửi lên khỏi câu lệnh thực thi trực tiếp và các query.

* Kiểm tra và lọc dữ liệu đầu vào như giới hạn kích thước, loại bỏ các kí tự đặc biệt,...
* Sử dụng thủ tục lưu trữ (Store procedures) trong CSDL. Dữ liệu được truyền vào thủ tục thông qua các tham số thay vì truyền trực tiếp -> tách dữ liệu khỏi mã lệnh.
* Không hiển thị Exception hay thông báo lỗi để tránh kẻ tấn công có thể suy đoán kết quả.
* Thiết lập quyền hạn phù hợp cho user.
* Sử dụng các công cụ dò quét lỗ hổng bảo mật
* Backup dữ liệu thường xuyên

1. **A04:2021 – Insecure Design**

**Chậm lại và suy nghĩ 4**: 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 không giới hạn số lượng tài khoản mà một người có thể tạo ra. Ví dụ có thể giới hạn bằng cách set 1 số điện thoại chỉ tạo được 1 tài khoản. Lỗ hổng này có thể bị khai thác bằng cách tạo ra nhiều tài khoản, ở lab này thì cần tạo ra 12 tài khoản là được. 11 tài khoản đầu, mỗi tài khoản lấy 5 vé là hết 55 vé, do đó tài khoản thứ 12 sẽ được xem phim miễn phí:



Một cách khác là có thể cài đặt mặc định thanh toán online ngay khi mua vé, để tránh tình trạng “đặt cho vui”. Nếu đây là “đặt cho vui” thì 55 vé đó trên hệ thống hiển thị “sold out” trong khi thực tế đến ngày chiếu phim, lại không có ai đến nhận vé, gây thất thoát cho rạp phim.

**Bài tập 5:** Báo cáo lỗ hổng đang được thực hành. Có thể sử dụng format theo mẫu sau:

**#Tiêu** **đề:** Thiết kế không an toàn gây ảnh hưởng đến tính ổn định và logic của ứng dụng, nếu không kịp thời phát hiện, hệ thống sẽ bị đình công, gây thất thoát lớn về tài sản.

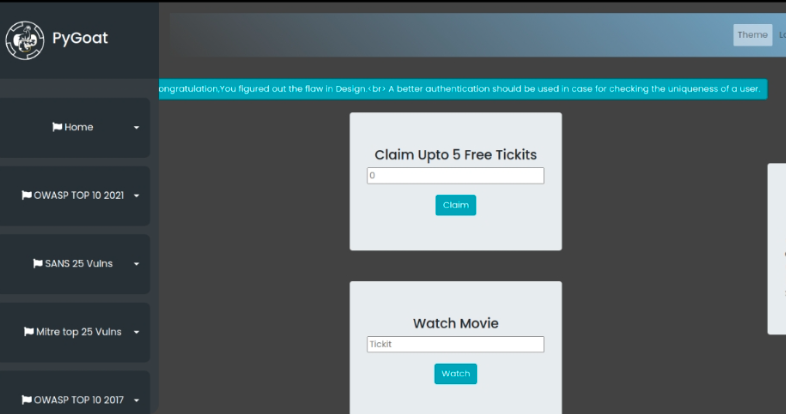
**#Mô tả lỗ hổng:**

### Tóm tắt: Thiết kế không an toàn là “thiết kế kiểm soát bị thiếu hoặc không hiệu quả”. Một trong những yếu tố góp phần vào thiết kế không an toàn là thiếu hồ sơ rủi ro vốn có trong phần mềm hoặc hệ thống đang được phát triển và do đó không xác định được mức độ thiết kế bảo mật là cần thiết.

### Các bước để thực hiện lại và bằng chứng:

1. bước 1: tạo 12 tài khoản khác nhau, mỗi tài khoản lấy 5 vé.

2. bước 2: Tài khoản cuối cùng sold out nên xem được phim miễn phí:



### Tài liệu hỗ trợ và tham khảo: Google và các link đã đề cập ở trên.

\*Liệt kê bất kỳ tài liệu bổ sung nào (ví dụ: ảnh chụp màn hình, nhật ký,...)

**# Mức độ ảnh hưởng của lỗ hổng:** Gây thiệt hại về tài chính. Làm sai logic chương trình.

**# Khuyến cáo khắc phục:**

* Sử dụng những thư viện mẫu thiết kế an toàn.
* Kiểm tra tính hợp lí ở mỗi cấp ứng dụng.
* Tách các lớp phần trên hệ thống và các lớp mạng.
* Nghiên cứu và tiến hành lập hồ sơ rủi ro để nâng cao tính chặt chẽ của ứng dụng và dự đoán trước những nguy cơ có thể xảy ra.
* Đảm bảo tuân thủ các nguyên tắc trong quá trình phát triển phần mềm như SDLC, SSDLC.

1. **A05:2021 – Security Misconfiguration**

**Chậm lại và suy nghĩ 5**: X-Host is None là gì? Có kiểm soát được X-Host không.

* X-Host is None là một phần trong thông báo lỗi của trang web.
* Có thể kiểm soát và thay đổi được X-Host.

**Bài tập 6:** Báo cáo lỗ hổng đang được thực hành. Có thể sử dụng format theo mẫu sau:

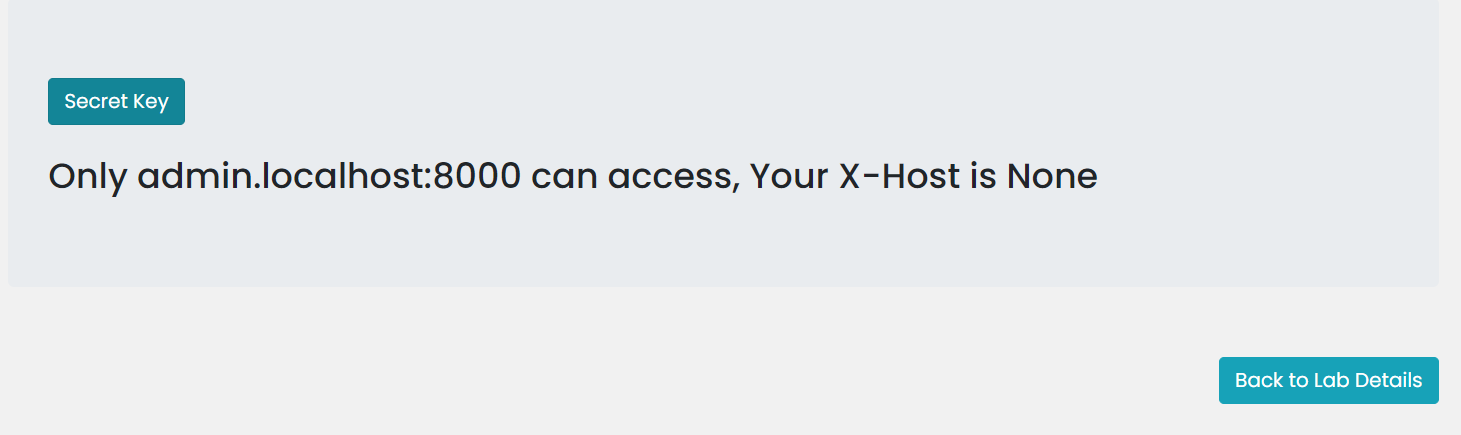
**#Tiêu** **đề:** Cấu hình bảo mật sai là tạo cơ hội để attacker “chạm” vào hệ thống của bạn!

**#Mô tả lỗ hổng:**

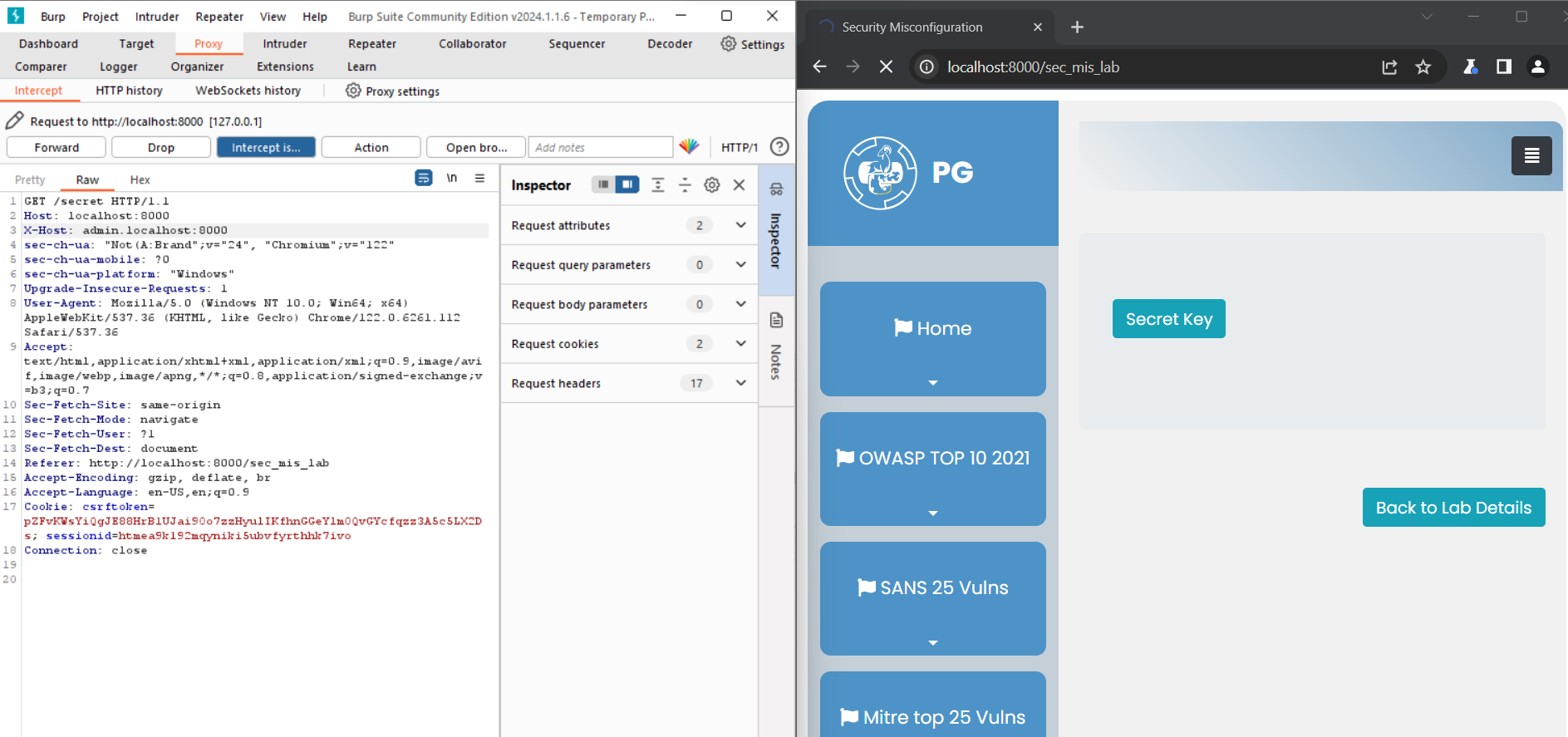
### Tóm tắt: Security misconfiguration hay lỗi cấu hình sai bảo mật là lỗ hổng phổ biến nhất trong danh sách và thường là kết quả của việc sử dụng cấu hình mặc định hoặc thông báo hiển thị lỗi có chứa quá nhiều thông tin. Ví dụ: một ứng dụng có thể hiển thị lỗi mô tả quá nhiều thông tin có thể tiết lộ các lỗ hổng trong ứng dụng. Điều này có thể được giảm thiểu bằng cách loại bỏ bất kỳ tính năng không sử dụng nào trong code và đảm bảo rằng các thông báo lỗi sẽ mang tính tổng quát chung chung hơn.

### Các bước để thực hiện lại và bằng chứng:

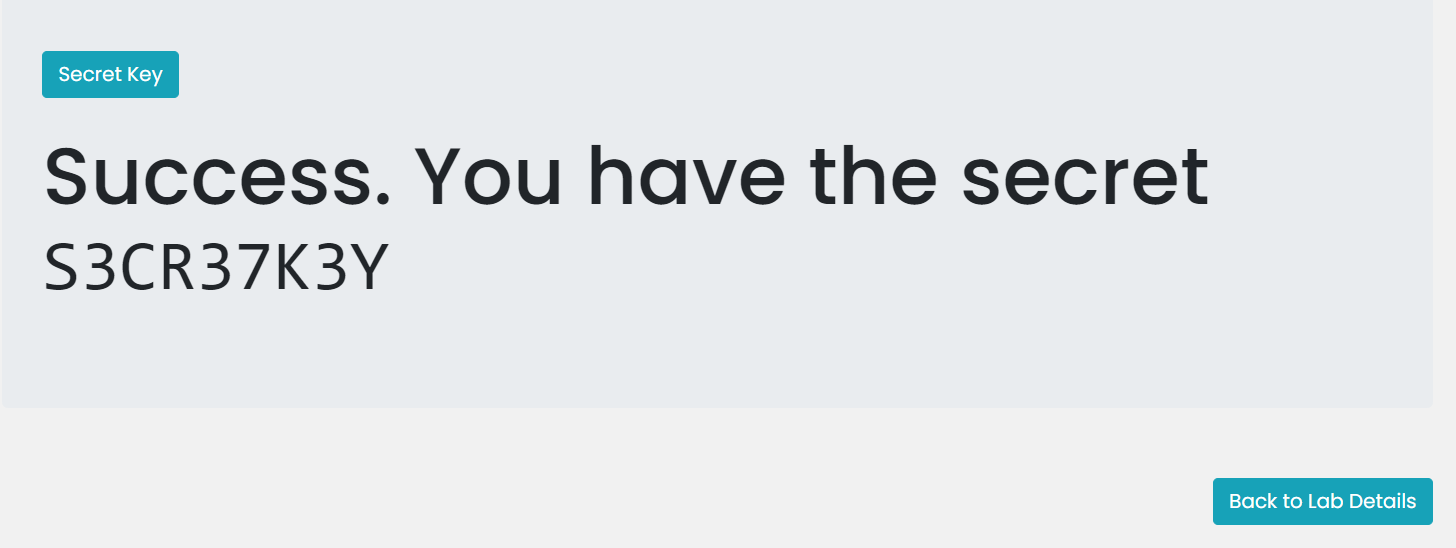
1. bước 1: Em không làm gì cả, chỉ đơn giản là vào challenge và Access thử xem hiển thị gì, kết quả là:



2. bước 2: Để có thể kết nối được, thì phải là admin. Vậy làm sao để trở thành admin? Quan sát chỗ thông báo, thấy chỉ rõ “admin.localhost:8000” mới có thể truy cập được. Lợi dụng điều này, em sẽ tiến hành chặn gói tin, sau đó chèn thêm trường X-Host như trên vào header của gói tin để lấy được secret key:



3. bước 3: Chọn “Forward” để gửi gói tin đi, và nhận được secret key:



### Tài liệu hỗ trợ và tham khảo: [kien-thuc/security/owasp](https://vinsep.com/kien-thuc/security/owasp-la-gi-top-10-owasp-la-gi/)

\*Liệt kê bất kỳ tài liệu bổ sung nào (ví dụ: ảnh chụp màn hình, nhật ký,...)

**# Mức độ ảnh hưởng của lỗ hổng:** cho phép 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.

**# Khuyến cáo khắc phục:**

* Loại bỏ những tài nguyên, tính năng không cần thiết.
* Cung cấp sự hiệu quả và an toàn giữa các thành phần.
* Liên tục cập nhật những phiên bản mới nhất.
* Thiết kế quy trình an toàn.

---

***Sinh viên đọc kỹ yêu cầu trình bày bên dưới trang này***

# **YÊU CẦU CHUNG**

* Sinh viên tìm hiểu và thực hiện bài tập theo yêu cầu, hướng dẫn.
* Nộp báo cáo kết quả chi tiết những việc (**Report**) bạn đã thực hiện, quan sát thấy và kèm ảnh chụp màn hình kết quả (nếu có); giải thích cho quan sát (nếu có).
* Sinh viên báo cáo kết quả thực hiện và nộp bài.

**Báo cáo:**

* File .DOCX và .PDF. Tập trung vào nội dung, không mô tả lý thuyết.
* Nội dung trình bày bằng Font chữ Times New Romans/ hoặc font chữ của mẫu báo cáo này (UTM Neo Sans Intel/UTM Viet Sach)– cỡ chữ 13. Canh đều (Justify) cho văn bản. Canh giữa (Center) cho ảnh chụp.
* Đặt tên theo định dạng: [Mã lớp]-ExeX\_GroupY. (trong đó X là Thứ tự Bài tập, Y là mã số thứ tự nhóm trong danh sách mà GV phụ trách công bố).

*Ví dụ: [*NT101.K11.ANTT*]-Exe01\_Group03.*

* Nếu báo cáo có nhiều file, nén tất cả file vào file .ZIP với cùng tên file báo cáo.
* Không đặt tên đúng định dạng – yêu cầu, sẽ **KHÔNG** chấm điểm bài nộp.
* Nộp file báo cáo trên theo thời gian đã thống nhất tại courses.uit.edu.vn.

**Đánh giá**:

* Hoàn thành tốt yêu cầu được giao.
* Có nội dung mở rộng, ứng dụng.

*Bài sao chép, trễ, … sẽ được xử lý tùy mức độ vi phạm.*

**HẾT**

1. Ghi nội dung công việc, các kịch bản trong bài Thực hành [↑](#footnote-ref-1)