BÁO CÁO THỰC HÀNH

Môn học: Bảo mật web và ứng dụng

Lab 2: Tổng quan các lỗ hổng bảo mật web thường gặp (Phần 2)

GVHD: Ngô Khánh Khoa

1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: NT213.O21.ATTN

STT	Họ và tên	MSSV	Email
1	Hà Thị Thu Hiền	21522056	21522056@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:[1]

STT	Công việc	Kết quả tự đánh giá
1	Bài tập 1: A06	100%
2	Bài tập 2: A07	100%
3	Bài tập 3: A08	100%
4	Bài tập 4: A09	100%
5	Bài tập 5: A10	100%

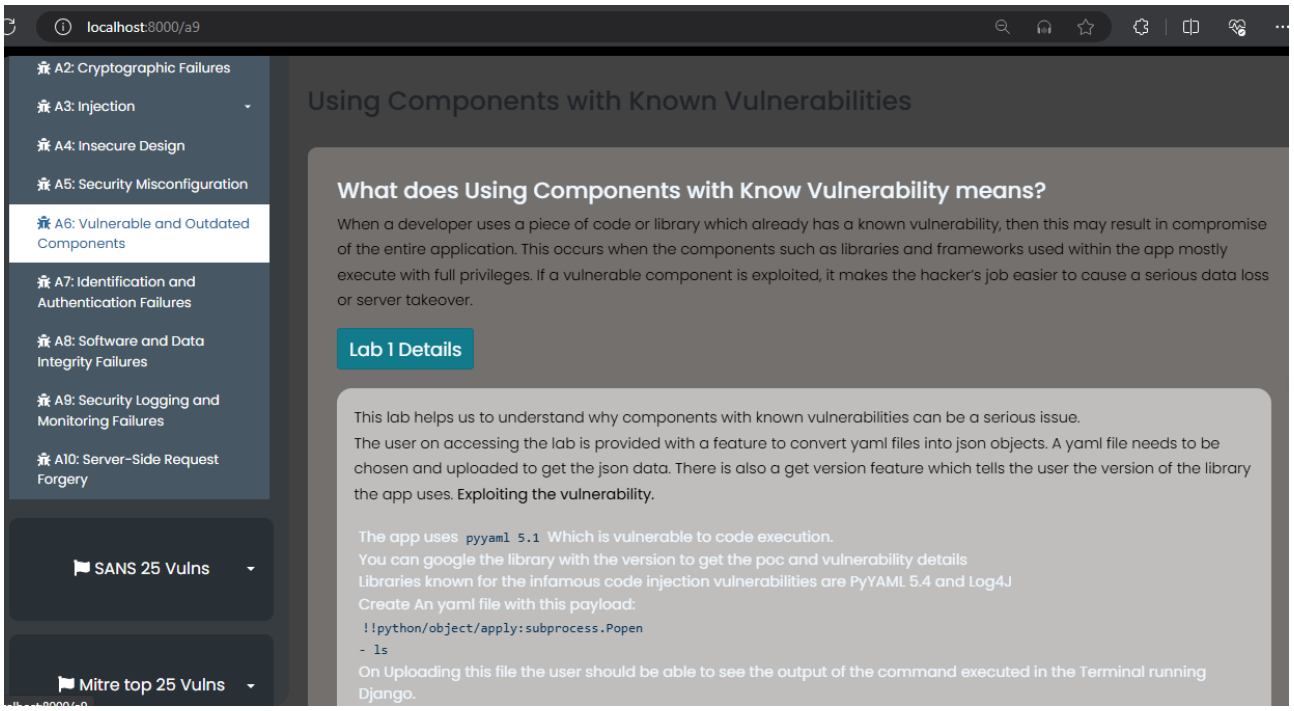
Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

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

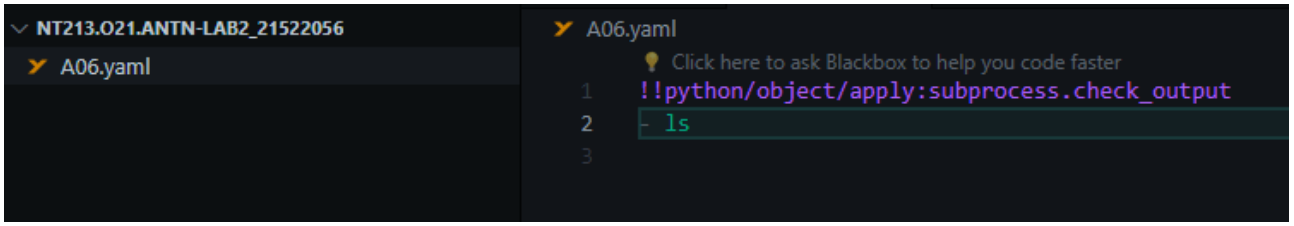
BÁO CÁO CHI TIẾT

1. Bài tập 1: A06:2021 – Vulnerable and Outdated Components

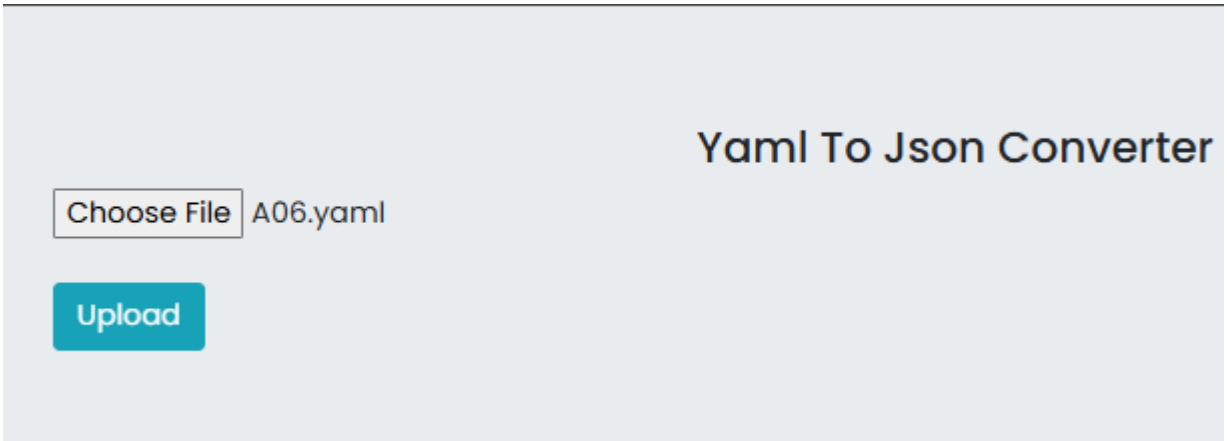
- **Tiêu đề:** Vulnerable and Outdated Components
- **Mô tả lỗ hổng:** Lỗ hổng về các thành phần dễ tấn công và lỗi thời là một vấn đề phổ biến trong bảo mật phần mềm. Đây là lỗ hổng xuất hiện khi một ứng dụng sử dụng các thành phần (như thư viện, framework, module) mà đã có lỗ hổng bảo mật hoặc đã cũ, không được bảo trì. Kẻ tấn công có thể tận dụng những lỗ hổng này để thực hiện các cuộc tấn công như xâm nhập hệ thống, thực thi mã độc, đánh cắp dữ liệu hoặc kiểm soát hệ thống.
    - **Tóm tắt:**



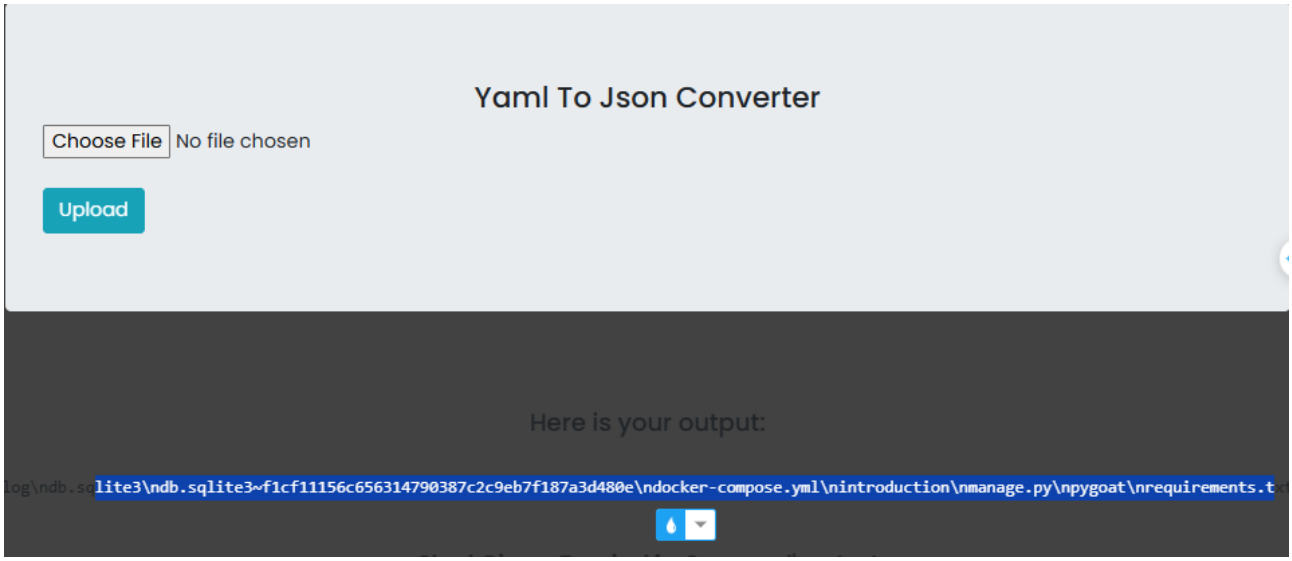
- **Các bước để thực hiện lại và bằng chứng:**
    - 1. bước 1: Tạo file A06.yaml như sau:



    - 2. bước 2: Thực hiện upload file A06.yaml



- 3. bước 3: Xem kết quả



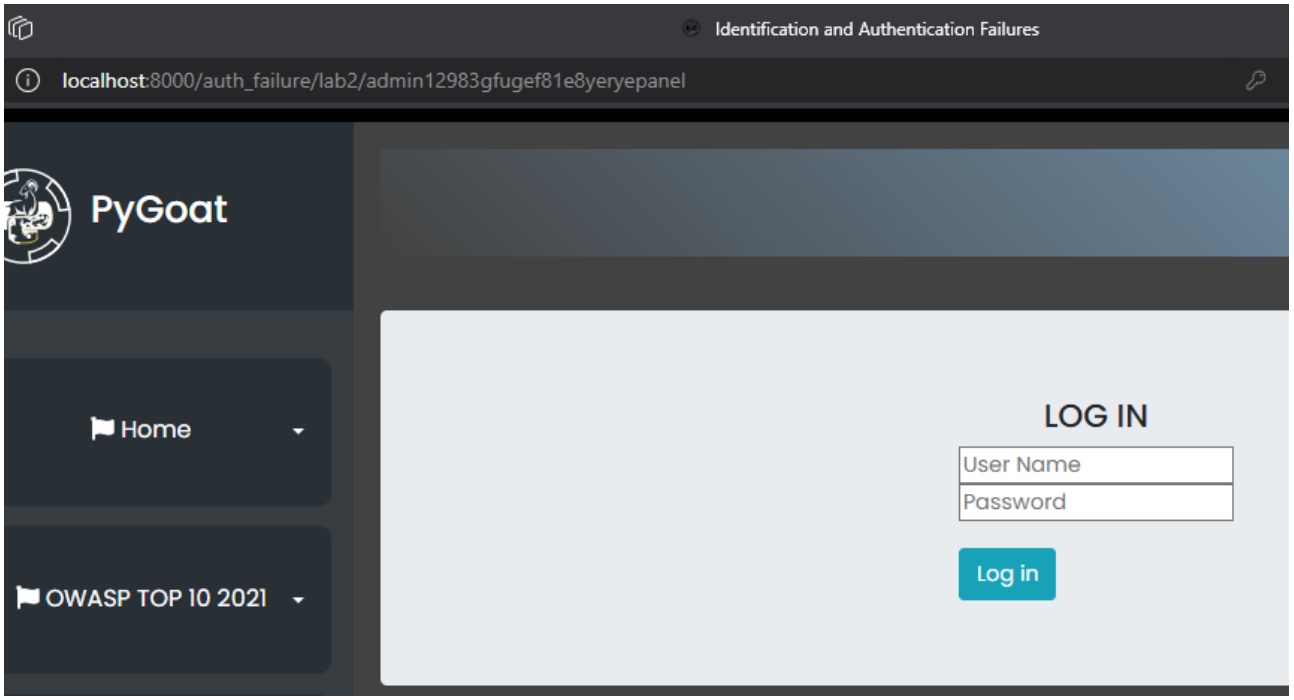
- **Mức độ ảnh hưởng của lỗ hổng:** Cao
- **Khuyến cáo khắc phục:**
  - Lọc dữ liệu người dùng tải lên để loại bỏ các đoạn mã độc.
  - Tắt quyền thực thi cho các tệp được người dùng tải lên để ngăn chặn việc thực thi mã độc.
  - Cập nhật phiên bản của các thành phần sử dụng để loại bỏ các lỗ hổng bảo mật đã biết và tăng cường tính ổn định của hệ thống.
  - Thực hiện kiểm tra và xác thực dữ liệu đầu vào từ người dùng để ngăn chặn các cuộc tấn công injection và các kỹ thuật tấn công khác.

2. Bài tập 2: A07:2021 – Identification and Authentication Failures

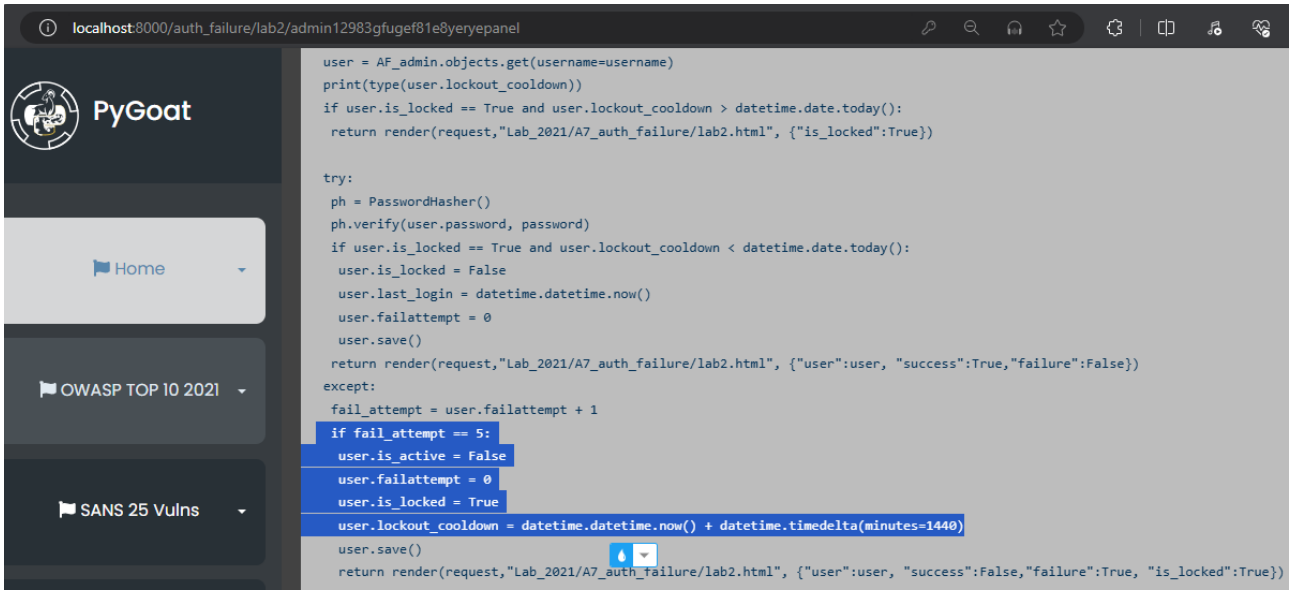
- **Tiêu đề:** Identification and Authentication Failures
  - **Mô tả lỗ hổng:** Đây là lỗ hổng cho phép người dùng đăng nhập bằng tài khoản admin, ta không cần đăng nhập thành công, chỉ cần spam thực hiện phá hoại để chặn tài khoản admin truy cập trong 1 ngày
    - Lỗi nhận dạng và xác thực có thể xảy ra khi các chức năng liên quan đến danh tính, xác thực hoặc quản lý phiên của người dùng không được triển khai dúng cách hoặc không được bảo vệ đầy đủ.

- Kẻ tấn công có thể khai thác các lỗi nhận dạng và xác thực bằng cách xâm phạm mật khẩu, khoá, mã thông báo phiên hoặc khai thác các lỗi triển khai khác để giả định danh tính của người dùng khác, tạm thời hoặc vĩnh viễn.

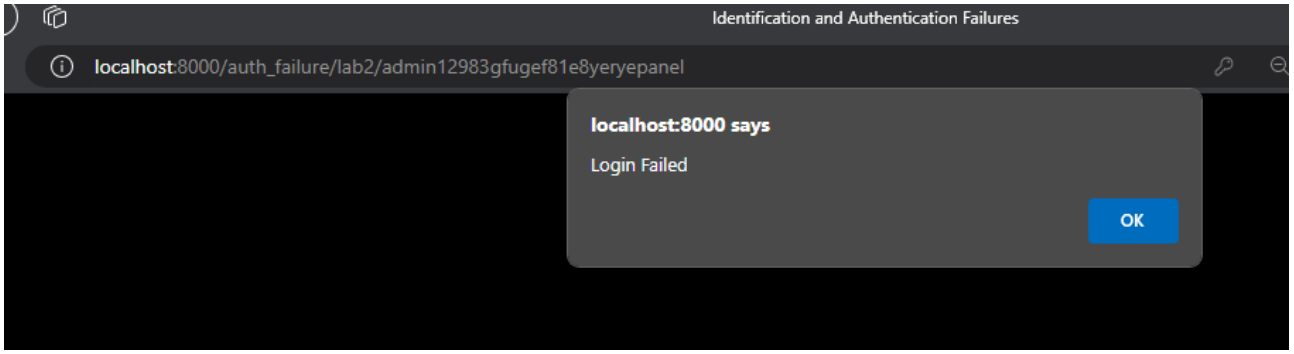
- **Tóm tắt:**



- **Các bước để thực hiện lại và bằng chứng:**
  - 1. bước 1: Nhìn vào code ta thấy, khi thử nhập password của 1 tài khoản sai 5 lần thì tài khoản đó sẽ bị khoá và đặt thời gian cooldown cho việc đăng nhập lại là 24 giờ (1440 phút).



- 2. bước 2: Nhập user là admin, password sai tuỳ ý 5 lần, lần 1 2 3 4 5 sẽ hiển thị



- 3. bước 3: Lần nhập thứ 6 sẽ hiển thị **Account Locked**

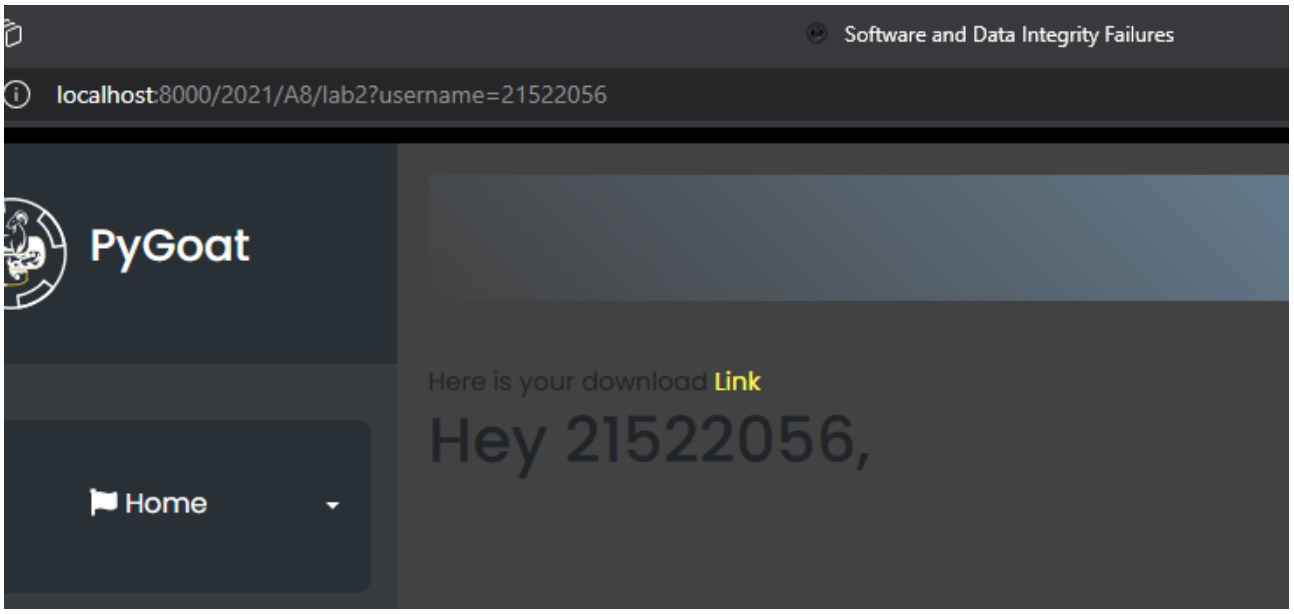
- **Mức độ ảnh hưởng của lỗ hổng:** Cao
- **Khuyến cáo khắc phục:**
  - Cài đặt cơ chế phòng thủ:
    - Thực hiện kiểm tra và xác thực chặt chẽ trong quá trình đăng nhập để đảm bảo rằng chỉ những người dùng được ủy quyền mới có thể truy cập vào tài khoản admin.
  - Sử dụng các biện pháp bảo mật bổ sung:
    - Áp dụng các biện pháp bảo mật như xác minh hai yếu tố hoặc xác thực đa yếu tố để tăng cường bảo mật trong quá trình đăng nhập.
  - Giới hạn số lần đăng nhập thất bại:
    - Thực hiện việc giới hạn số lần đăng nhập thất bại trước khi tài khoản bị khóa. Nếu có quá nhiều lần đăng nhập thất bại, hãy đảm bảo rằng tài khoản sẽ không bị khóa quá lâu và cần có các biện pháp phục hồi.
  - Xử lý cookie an toàn:
    - Kiểm tra và xác thực cookie để đảm bảo rằng không có thông tin quyền hạn người dùng nào được sửa đổi bởi người dùng.
  - Ghi nhật ký và giám sát:
    - Ghi nhật ký hoạt động đăng nhập và giám sát các hành vi đáng ngờ như việc đăng nhập thất bại nhiều lần từ cùng một địa chỉ IP hoặc tài khoản.
  - Thông báo và cảnh báo:
    - Thông báo cho người quản trị hệ thống về các hoạt động đăng nhập đáng ngờ hoặc các cố gắng tấn công để họ có thể thực hiện biện pháp phòng thủ phù hợp.

3. Bài tập 3: A08:2021 – Software and Data Integrity Failures

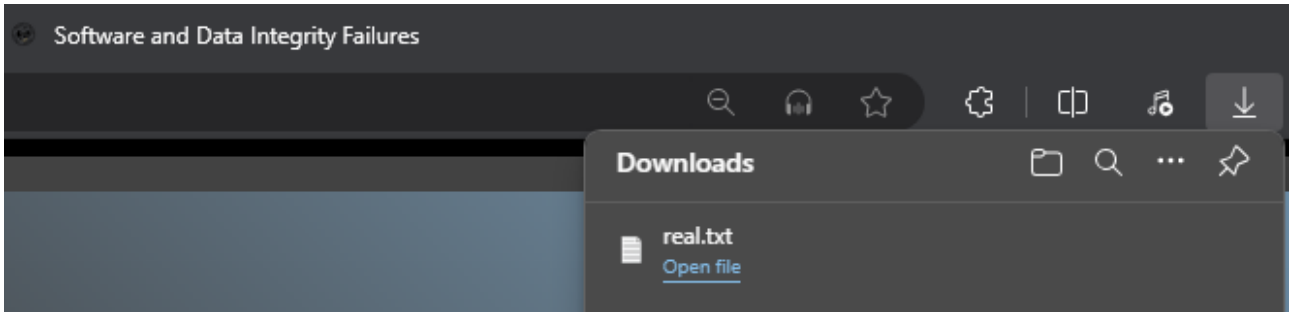
- **Tiêu đề:** Software and Data Integrity Failures
  - **Mô tả lỗ hổng:** Ta có thể sử dụng Inject script JS, XSS attack**,** để trang web thực thi script từ đó lấy được thông tin ta muốn.
    - Các lỗi về tính toàn vẹn của phần mềm và dữ liệu liên quan đến mã và cơ sở hạ tầng không bảo vệ chống lại các vi phạm về tính toàn vẹn.

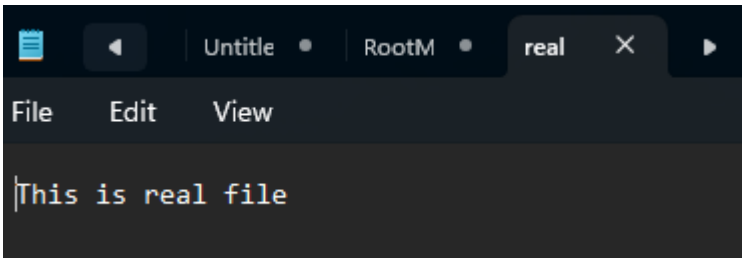
- Điều này có thể xảy ra khi sử dụng phần mềm từ các nguồn và kho lưu trữ không đáng tin cậy hoặc thậm chí là phần mềm bị can thiệp tại nguồn, trong quá trình chuyển tiếp hoặc thậm chí là trong bộ đệm của endpoint.

- **Tóm tắt:**

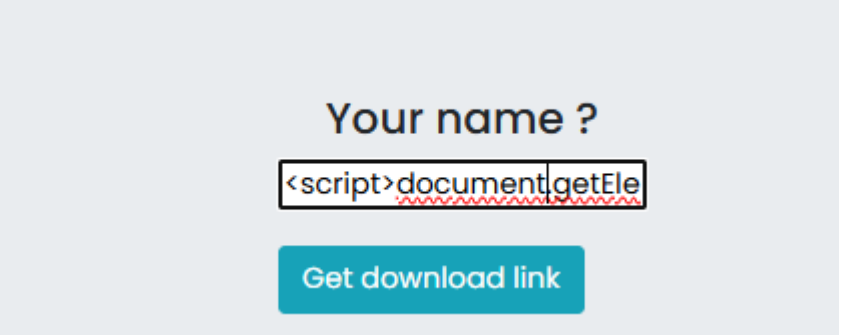


- **Các bước để thực hiện lại và bằng chứng:**
  - 1. bước 1: Click vào link để tải file về

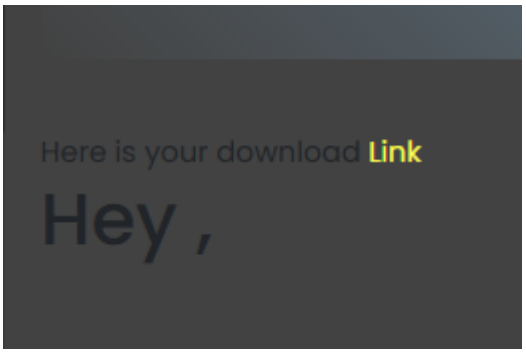


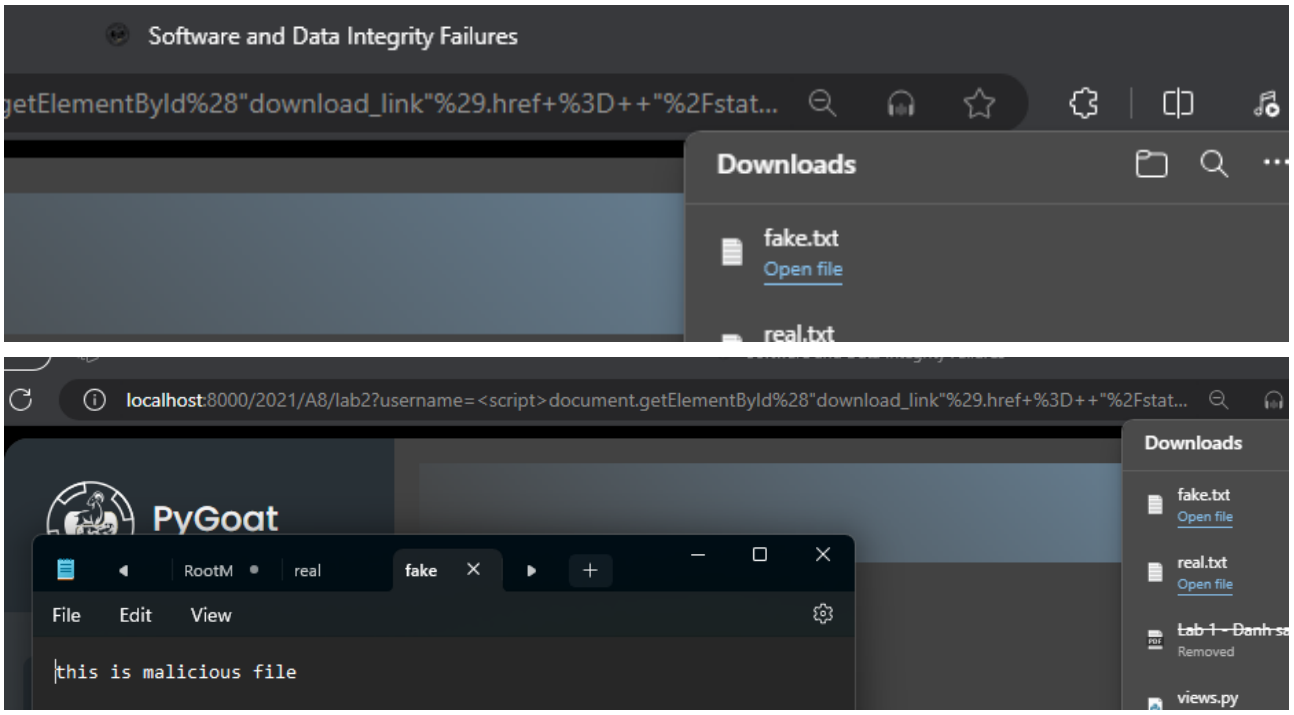


  - 2. bước 2: Inject script JS, XSS attack, để thay vì download file real như bình thường thì bây giờ thực thi script sẽ download file fake.txt.
    <script>document.getElementById("download_link").href = "/static/fake.txt";</script>



- 3. bước 3: Xem kết quả





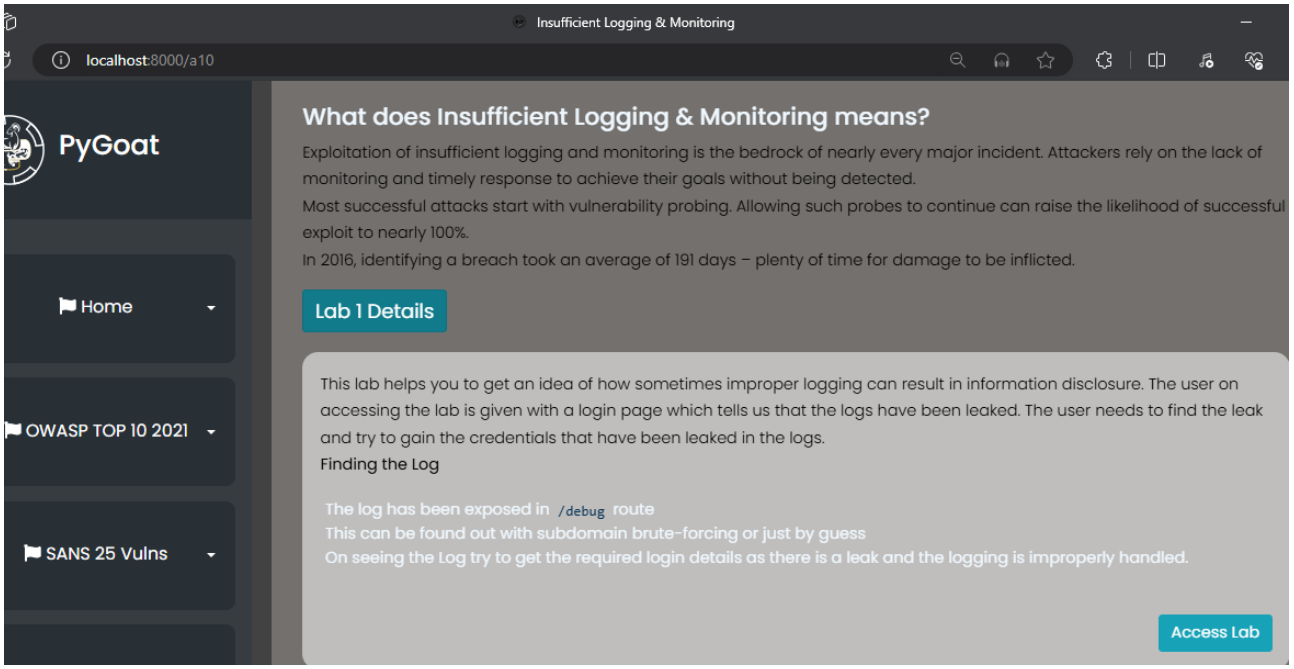
- **Mức độ ảnh hưởng của lỗ hổng:** Nghiêm trọng
- **Khuyến cáo khắc phục:**
  - Xử lý và xác thực đầu vào:
    - Thực hiện kiểm tra và xác thực đầu vào từ người dùng, đặc biệt là trong các kịch bản tải xuống hoặc tải tệp.
    - Loại bỏ hoặc làm sạch các dữ liệu không tin cậy và nguy hiểm như mã JavaScript không được tin cậy.
  - Sử dụng phần mềm đáng tin cậy:

- Sử dụng phần mềm từ các nguồn và kho lưu trữ đáng tin cậy, tránh sử dụng phần mềm từ nguồn không đáng tin cậy hoặc bị can thiệp.

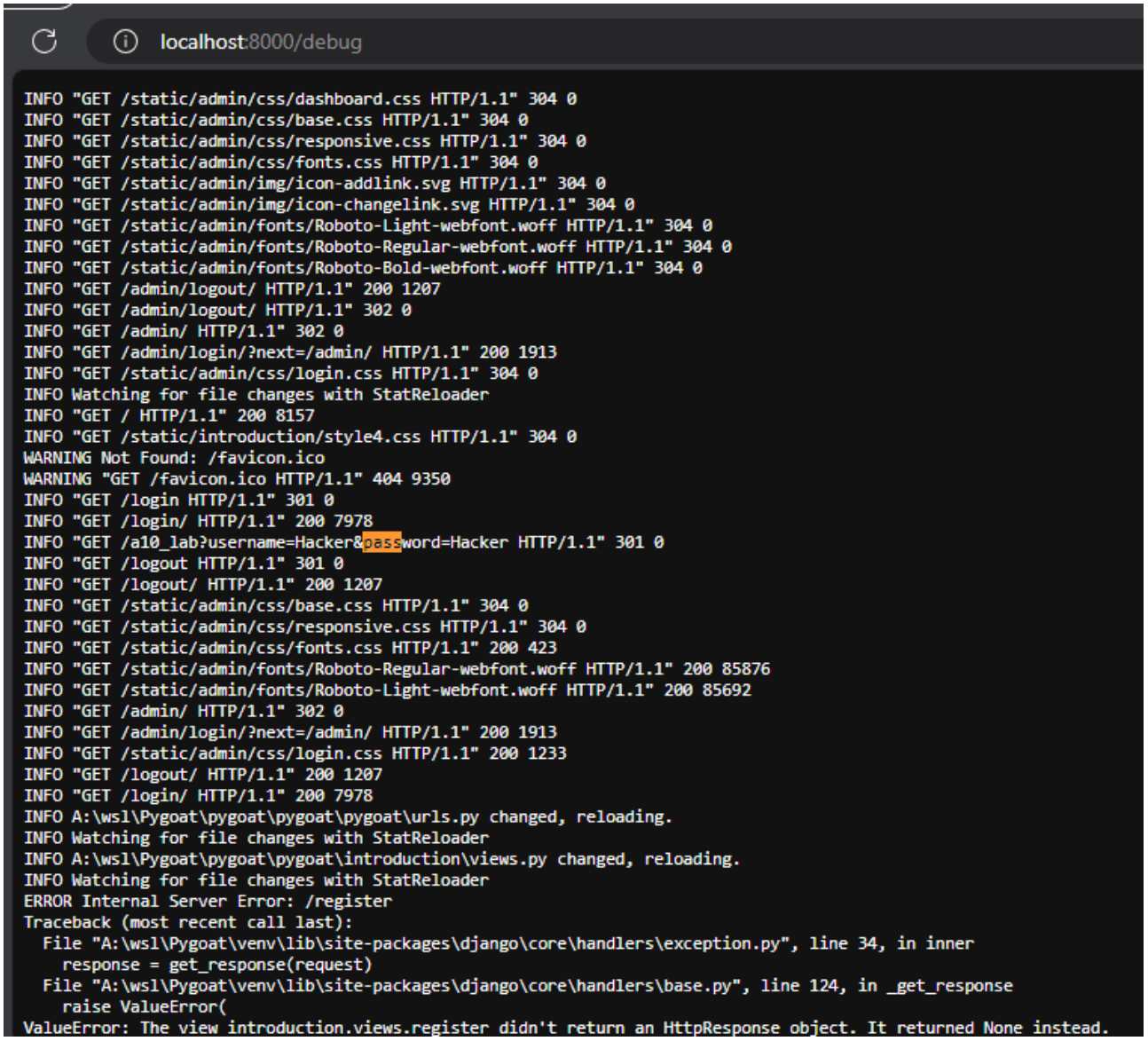
- Kỹ thuật mã hóa:
  - Sử dụng mã hóa để bảo vệ dữ liệu quan trọng trong quá trình chuyển tiếp và lưu trữ.
  - Đảm bảo rằng tất cả các dữ liệu được truyền từ client đến server và ngược lại được mã hóa đúng cách.
- Bảo vệ chống lại XSS:
  - Áp dụng các biện pháp bảo vệ chống lại Cross-Site Scripting (XSS) như sử dụng Content Security Policy (CSP), kiểm tra và làm sạch đầu ra, và sử dụng các thư viện và framework an toàn.
- Kiểm tra và giám sát liên tục:
  - Thực hiện kiểm tra bảo mật định kỳ để phát hiện và khắc phục các lỗ hổng bảo mật.
  - Giám sát các hoạt động của hệ thống để phát hiện sớm các tấn công và vi phạm tính toàn vẹn dữ liệu.

4. Bài tập 4: A09:2021 – Security Logging and Monitoring Failures

- **Tiêu đề:** Security Logging and Monitoring Failures
- **Mô tả lỗ hổng:** Việc không ghi nhật ký, giám sát hoặc báo cáo đầy đủ các sự kiện bảo mật, chẳng hạn như các lần thử đăng nhập, khiến hành vi đáng ngờ khó bị phát hiện và làm tăng đáng kể khả năng kẻ tấn công có thể khai thác thành công ứng dụng. ➔ Ở bài này ta có thể truy cập vào file ghi log khi không có quyền, khiến các thông tin bị lộ ra ngoài
  - **Tóm tắt:**



  - **Các bước để thực hiện lại và bằng chứng:**
    - 1. bước 1: Truy cập vào http://localhost:8000 /debug

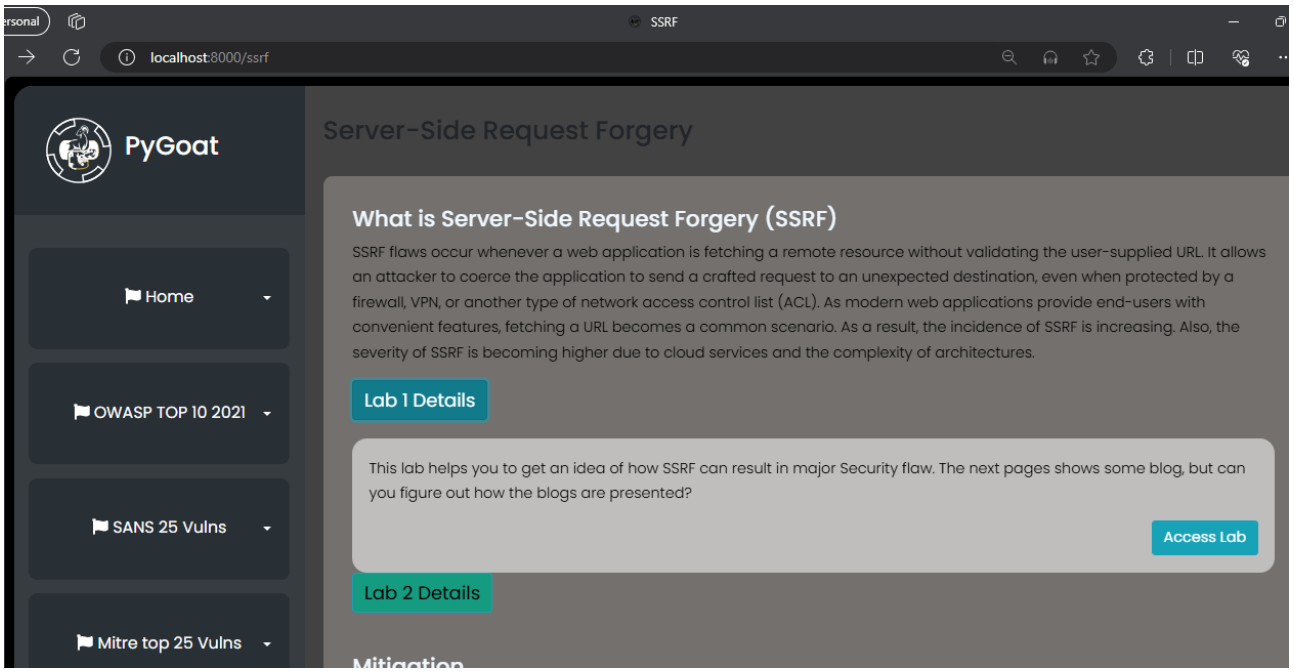


- **Khuyến cáo khắc phục:**
  - Thực hiện bảo vệ truy cập vào file log:
    - Xác định và thiết lập các quy tắc truy cập chặt chẽ cho các file log để đảm bảo rằng chỉ những người dùng cần thiết mới có thể truy cập và đọc được thông tin trong file log.
    - Sử dụng các biện pháp bảo vệ như mã hóa hoặc ký tự hóa để bảo vệ dữ liệu trong file log tránh bị lộ ra ngoài khi bị truy cập trái phép.
  - Tăng cường bảo mật hệ thống:
    - Áp dụng các biện pháp bảo mật để ngăn chặn truy cập không ủy quyền vào các tính năng và tài nguyên nhạy cảm của hệ thống.
    - Xác thực và ủy quyền mọi hoạt động truy cập vào file log để đảm bảo rằng chỉ những người dùng có quyền được phép mới có thể truy cập và thực hiện các thao tác trên file log.
  - Kiểm tra và đánh giá định kỳ:
    - Thực hiện kiểm tra bảo mật định kỳ để đảm bảo rằng các cấu hình ghi nhật ký và giám sát đang hoạt động đúng cách và hiệu quả.

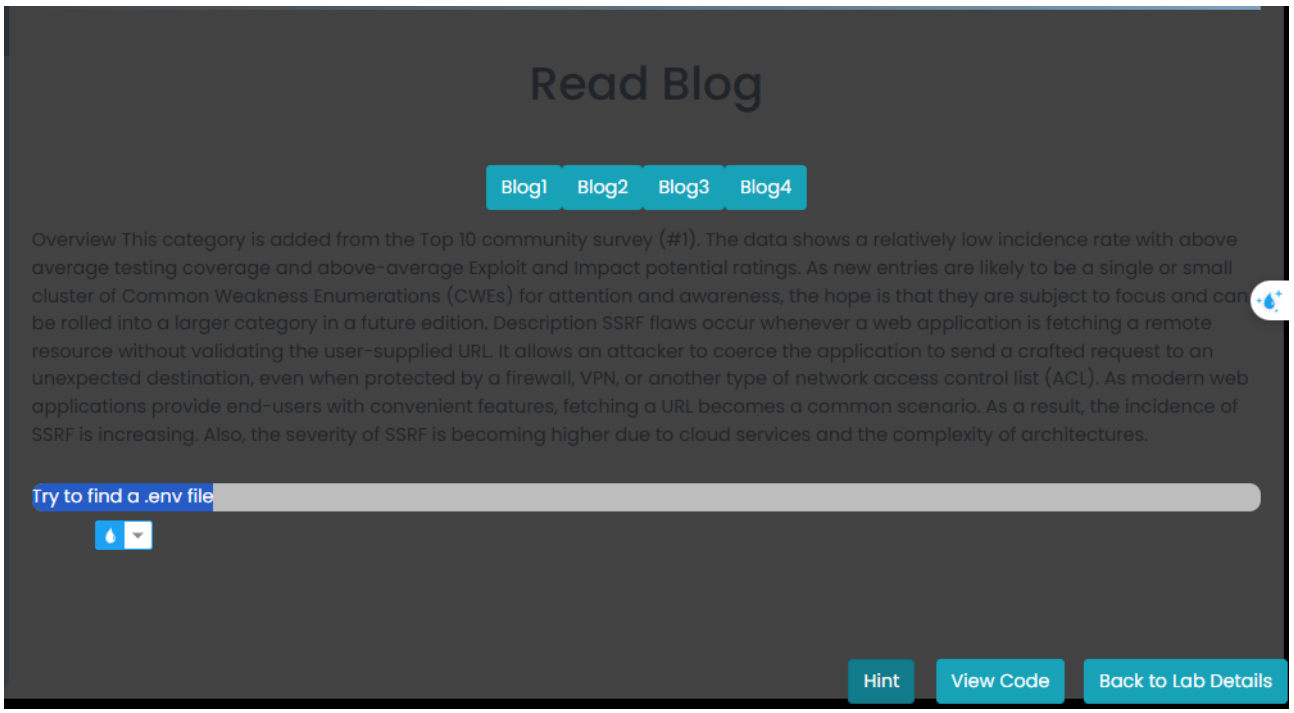
- Đánh giá và cập nhật các biện pháp bảo mật và quy trình giám sát dựa trên kết quả của các kiểm tra bảo mật.

5. Bài tập 5: A10:2021 – Server-Side Request Forgery (SSRF)

- **Tiêu đề:** A10:2021 – Server-Side Request Forgery (SSRF)
- **Mô tả lỗ hổng:** Đây là lỗ hổng cho phép user có thể chỉnh sửa code HTML từ browser và có thể truy cập vào các url vốn không có quyền truy cập.
  - **Tóm tắt:**



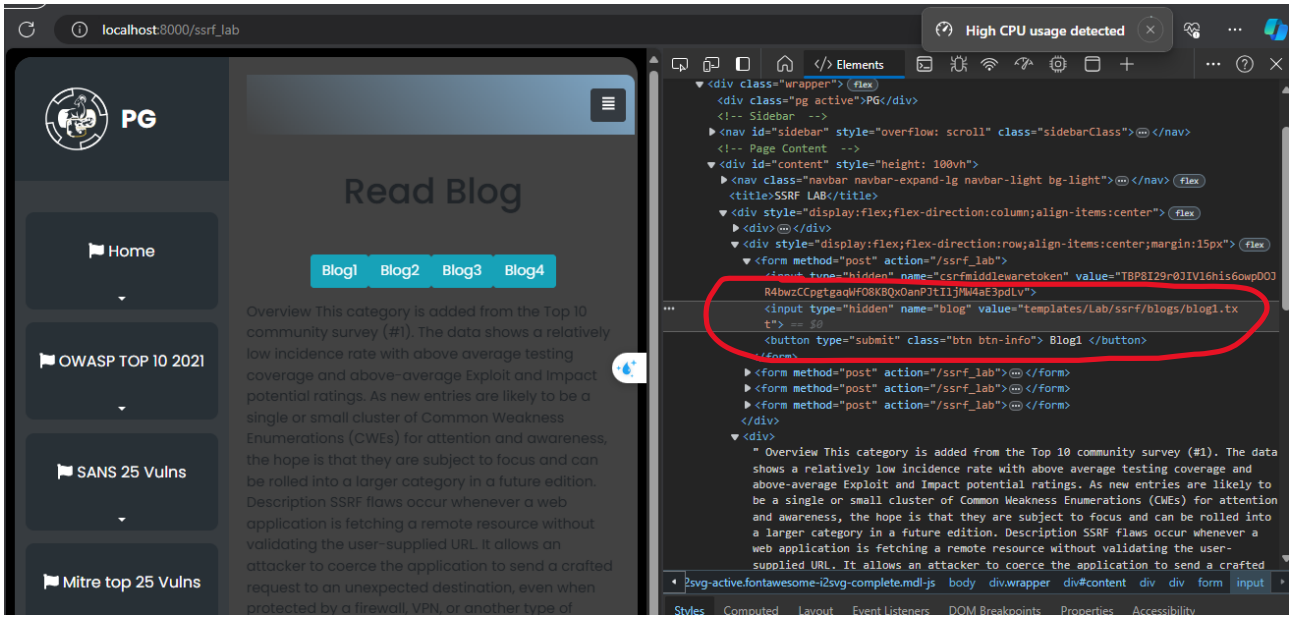
  - **Các bước để thực hiện lại và bằng chứng:**
    - 1. bước 1: Xem hint

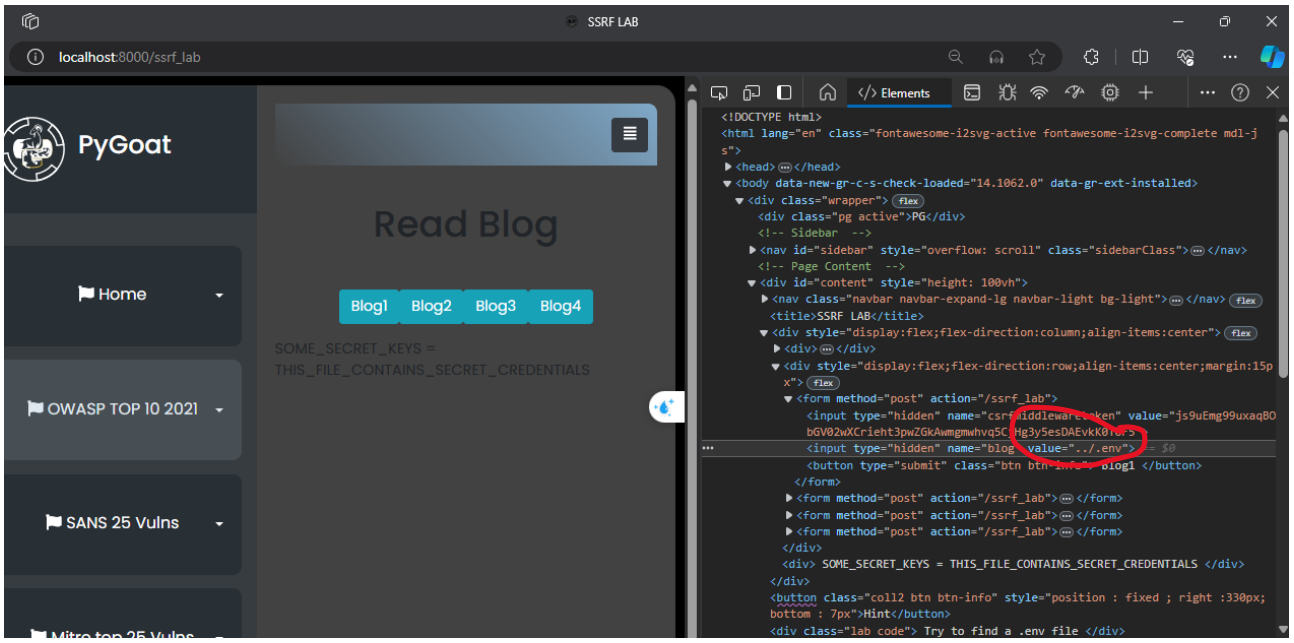


- 2. bước 2: Xem thử code, trong đoạn bôi xanh đường dẫn của tệp được trích xuất từ yêu cầu POST của người dùng và sau đó sử dụng để mở tệp trên server. Điều này có thể dẫn đến tình trạng SSRF nếu người dùng có thể kiểm soát nội dung của yêu cầu POST.
- Một kẻ tấn công có thể gửi yêu cầu POST với một URL bên trong và có thể là một URL nội bộ đến một hệ thống nội bộ, ví dụ như **http://localhost/private-file**. Nếu máy chủ chạy mã này có quyền truy cập vào máy chủ nội bộ và không có biện pháp bảo vệ, nó có thể mở tệp này và trả lại nội dung cho người dùng, cho phép kẻ tấn công truy cập vào tệp không được công bố trước đó.

```
def ssrf_lab(request):
if request.user.is_authenticated:
 if request.method=="GET":
  return render(request,"Lab/ssrf/ssrf_lab.html",{"blog":"Read Blog About SSRF"})
 else:
  file=request.POST["blog"]
  try :
   dirname = os.path.dirname(__file__)
   filename = os.path.join(dirname, file)
   file = open(filename,"r")
   data = file.re
   return render(request,"Lab/ssrf/ssrf_lab.html",{"blog":data})
  except:
   return render(request, "Lab/ssrf/ssrf_lab.html", {"blog": "No blog found"})
else:
 return redirect('login')
```

- 3. bước 3: Mở html của trang web ra xem, và theo thông tin của hint, và lỗ hổng ở trên thì ta thử thay đổi mã html để tìm ra file .env





- **Mức độ ảnh hưởng của lỗ hổng:** Nghiêm trọng
- **Khuyến cáo khắc phục:**
  - Xác thực và kiểm tra đầu vào:
    - Xác thực và kiểm tra đầu vào từ người dùng để đảm bảo rằng chỉ những giá trị hợp lệ được chấp nhận.
    - Đặc biệt cần kiểm tra và hạn chế người dùng chỉ được phép nhập URL nằm trong phạm vi an toàn và không được phép truy cập vào các tài nguyên nội bộ.
  - Sử dụng danh sách trắng (whitelist):
    - Sử dụng danh sách trắng để chỉ cho phép truy cập vào các URL được phép và từ chối mọi URL không được liệt kê trong danh sách này.
  - Hạn chế quyền truy cập và phạm vi tài nguyên:

- Hạn chế quyền truy cập của ứng dụng chỉ vào các tài nguyên cụ thể và không cho phép truy cập vào các tài nguyên nội bộ không an toàn.

- Mã hóa và ủy quyền:
  - Sử dụng các biện pháp bảo mật như mã hóa và ủy quyền để bảo vệ thông tin nhạy cảm và ngăn chặn các tấn công từ bên ngoài.
- Giám sát và ghi nhật ký:
  - Giám sát hoạt động của ứng dụng và ghi nhật ký để phát hiện sớm các hành vi bất thường và các cuộc tấn công.
- Cập nhật và kiểm tra định kỳ:
  - Cập nhật và kiểm tra định kỳ các cơ sở mã nguồn mở và thư viện phụ thuộc để đảm bảo rằng không có lỗ hổng bảo mật nào được tìm thấy và sửa chữa kịp thời.

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