**REPORT**

**VULNBLOG**

**Error-based SQL injection**

Error-based SQL injection là lỗ hổng khi người dùng nhập các payload nhạy cảm vào các trường được kiểm soát bởi người dùng như ô nhập tài khoản, mật khẩu của trang đăng nhập,.. và ứng dụng phản hồi lại bằng các thông báo lỗi liên quan tới cơ sở dữ liệu.

**Boolean-based SQL injection**

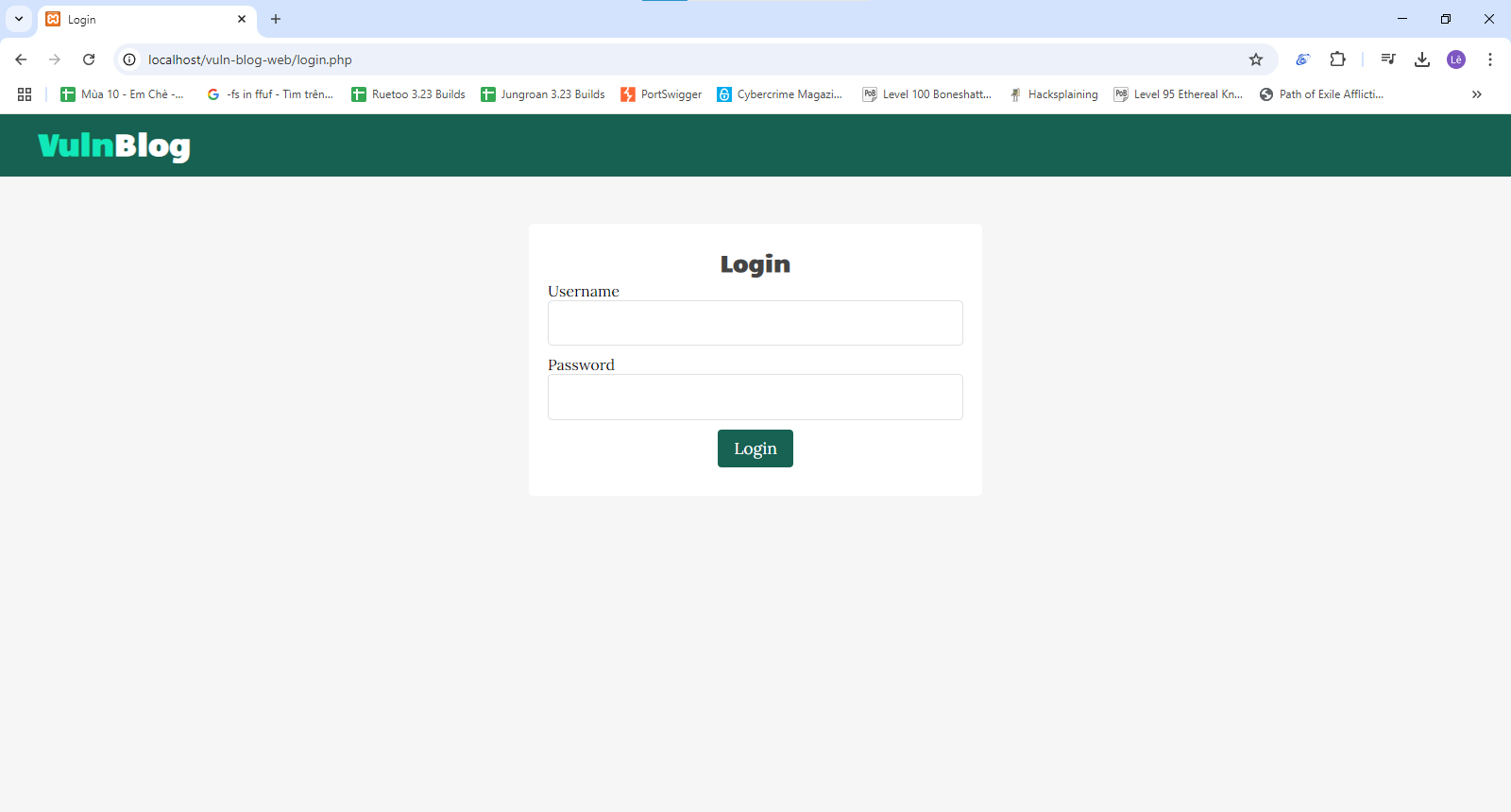
Boolean-based SQL injection là lỗ hổng khi người dùng nhập vào các payload có các toán tử so sánh như OR 1=1 hoặc AND 1=2 để kiểm tra phản hồi của ứng dụng.

**Time-based SQL injection**

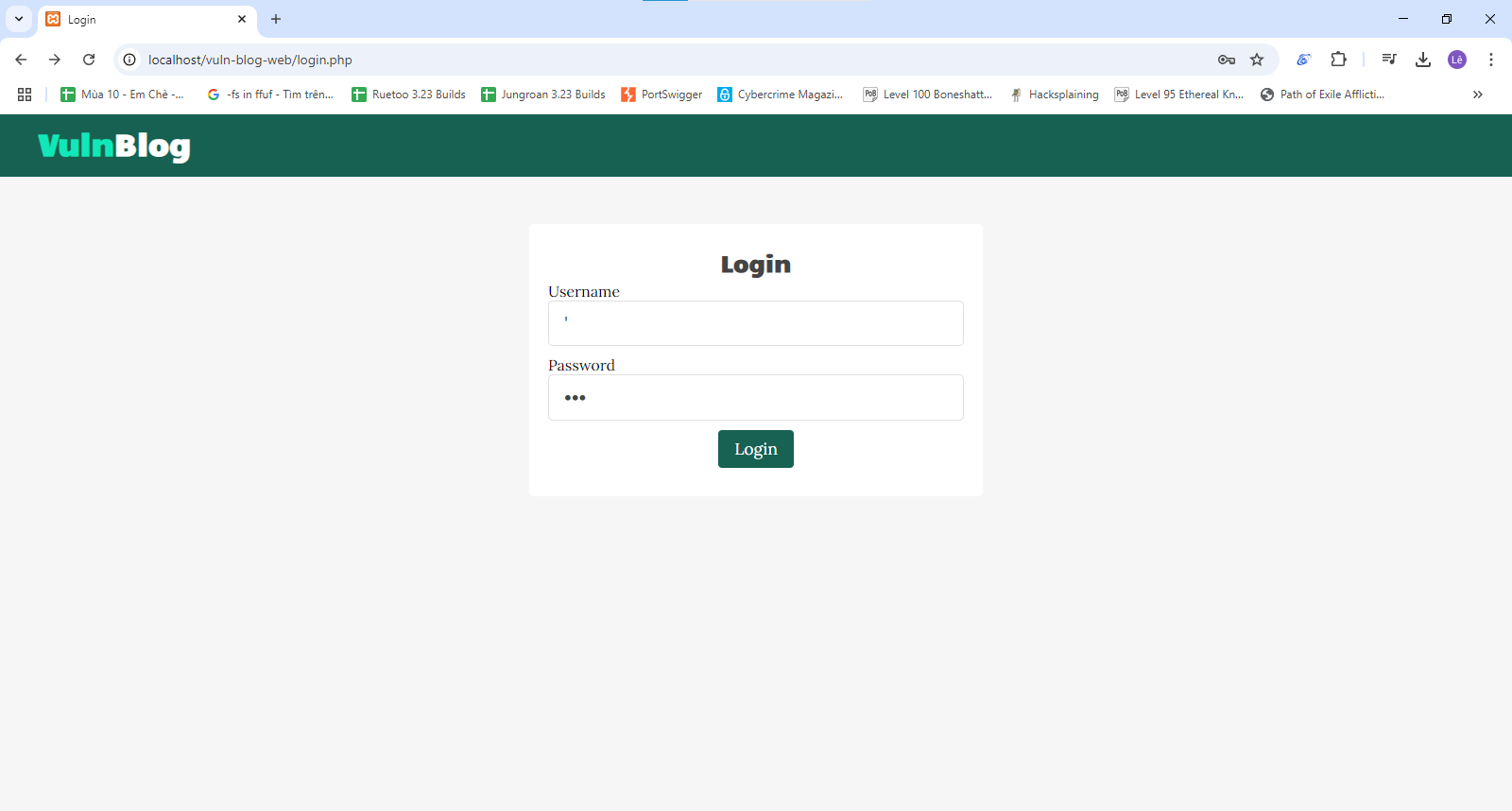
Time-based SQL injection là lỗ hổng khi ứng dụng không trả về một thông báo lỗi nào với input của người dùng, khi này chúng ta có thể kiểm tra bằng các payload như SLEEP(5), để xem ứng dụng có thực hiện câu lệnh này hay không, nếu trang web thực hiện load lại trang trong 5 giây thì trang web có thể đã bị SQL injection.

**1. Chức năng Login**

Truy cập vào trang web, ta nhận được một form để điền các thông tin đăng nhập.



Ta sẽ thử nhập vào payload như “ ‘ ” ở trường username và 123 cho trường password để kiểm tra phản hồi của ứng dụng:

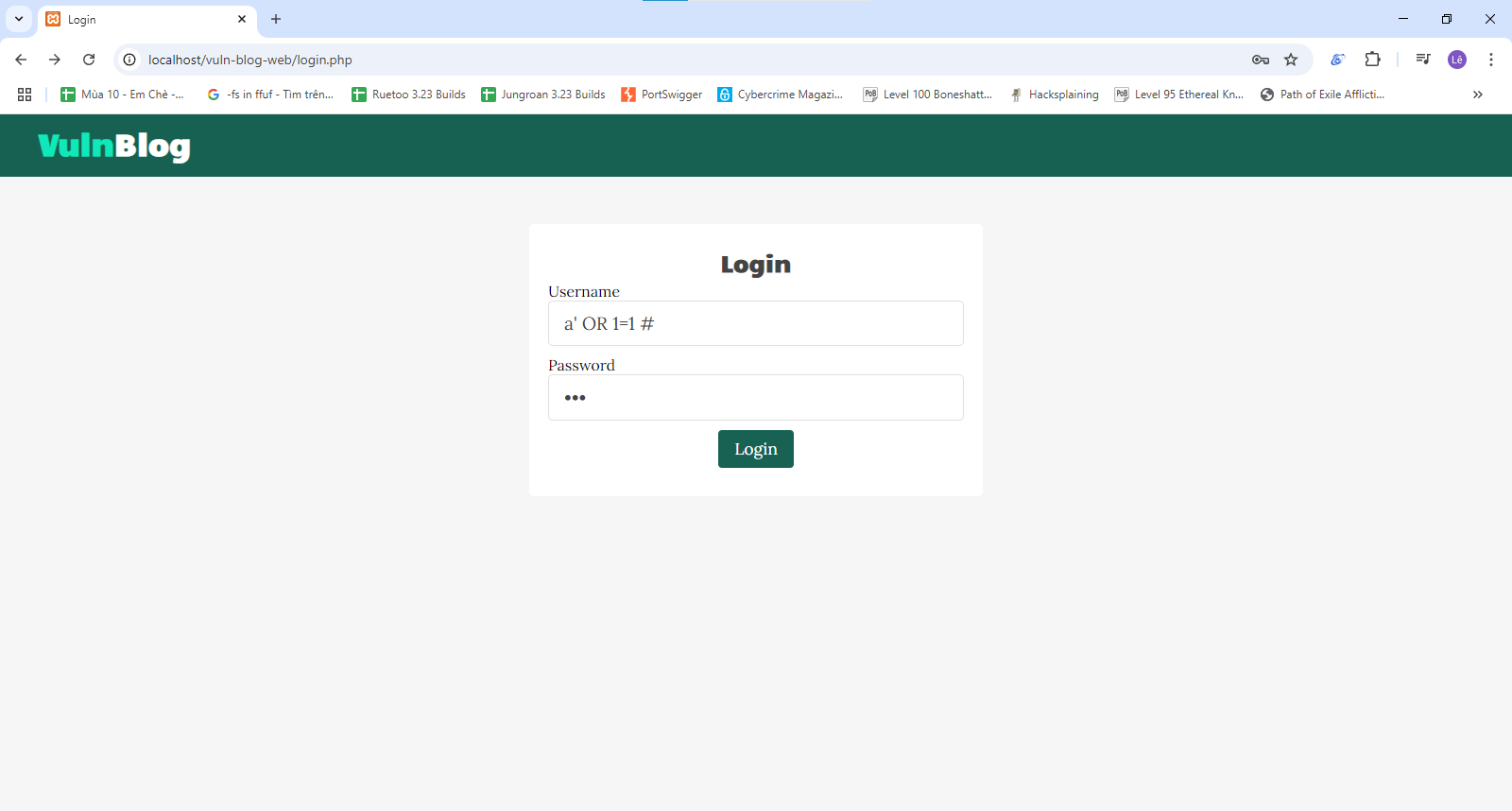


Kết quả là ứng dụng trả về một thông báo lỗi:

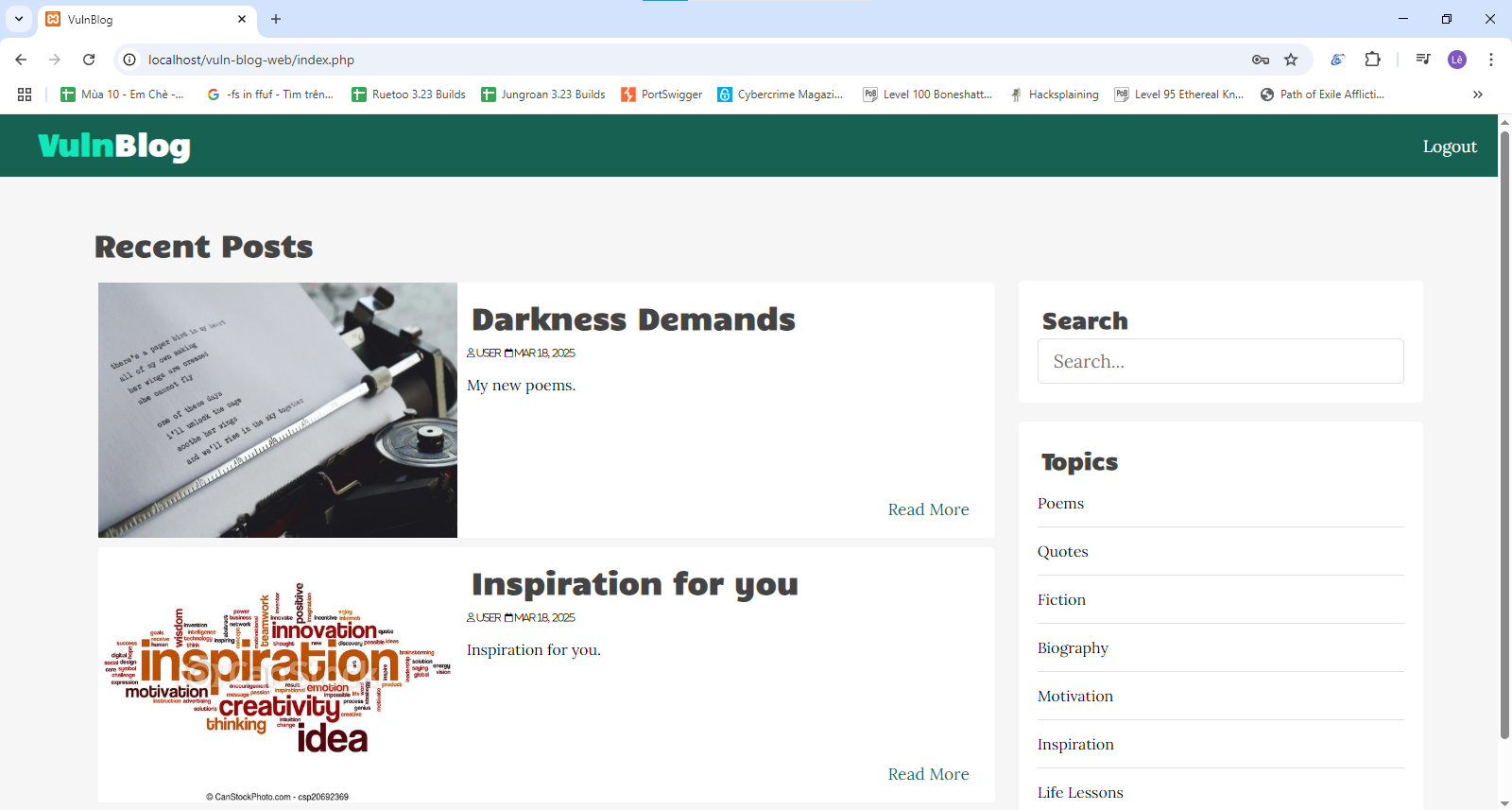


Có thể ứng dụng đã bị lỗ hổng SQL injection. Đây là một dạng của Error-based SQL injection. Thông báo lỗi cho chúng ta biết ứng dụng sử dụng cơ sở dữ liệu MYSQL.

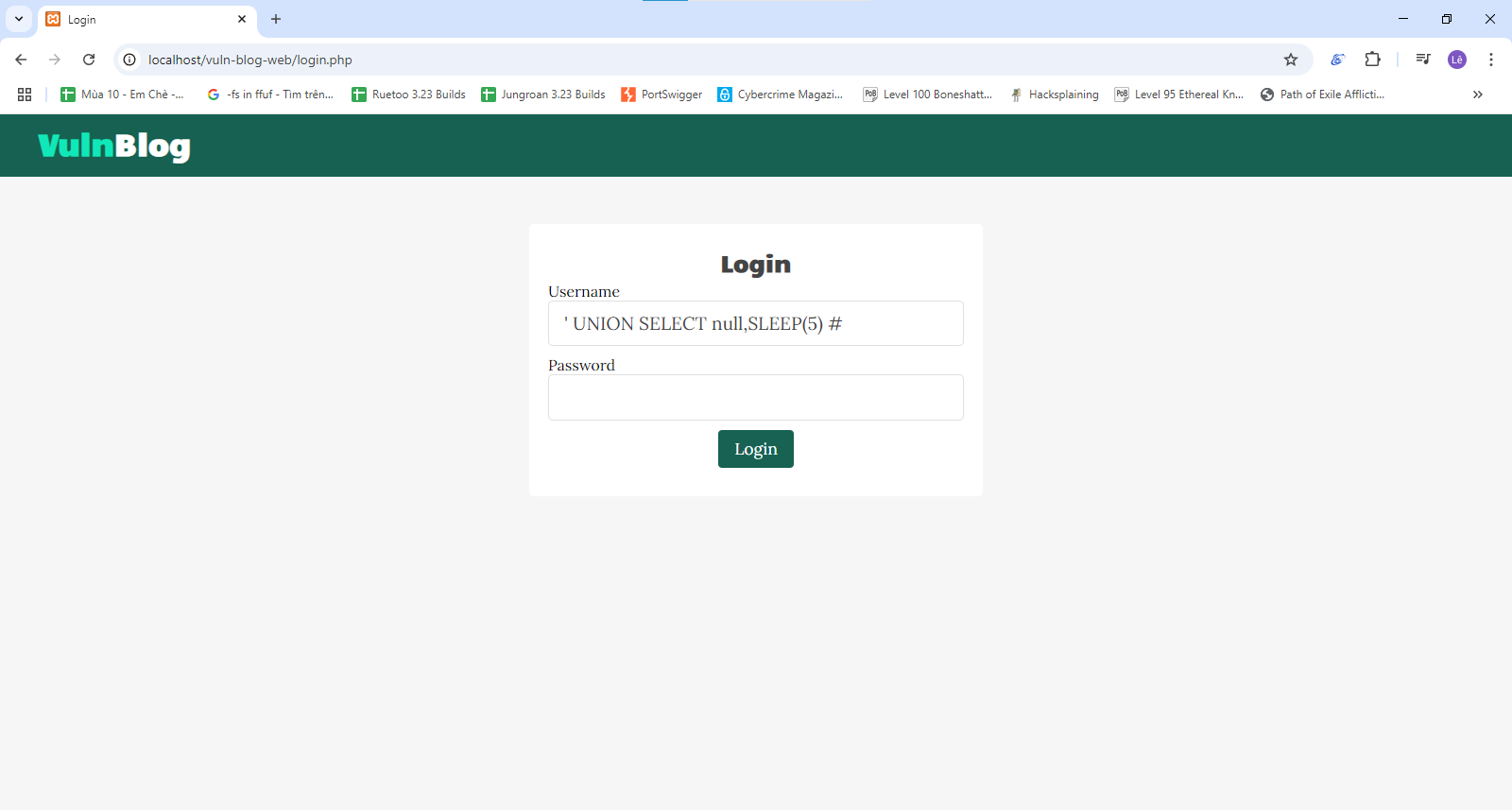
Chúng ta có thể sử dụng payload dạng Boolean SQL injection như “ a’ OR 1=1 # ” để xem có thể bypass được chức năng login hay không:



Kết quả là ta đã truy cập thành công trang chính của ứng dụng:

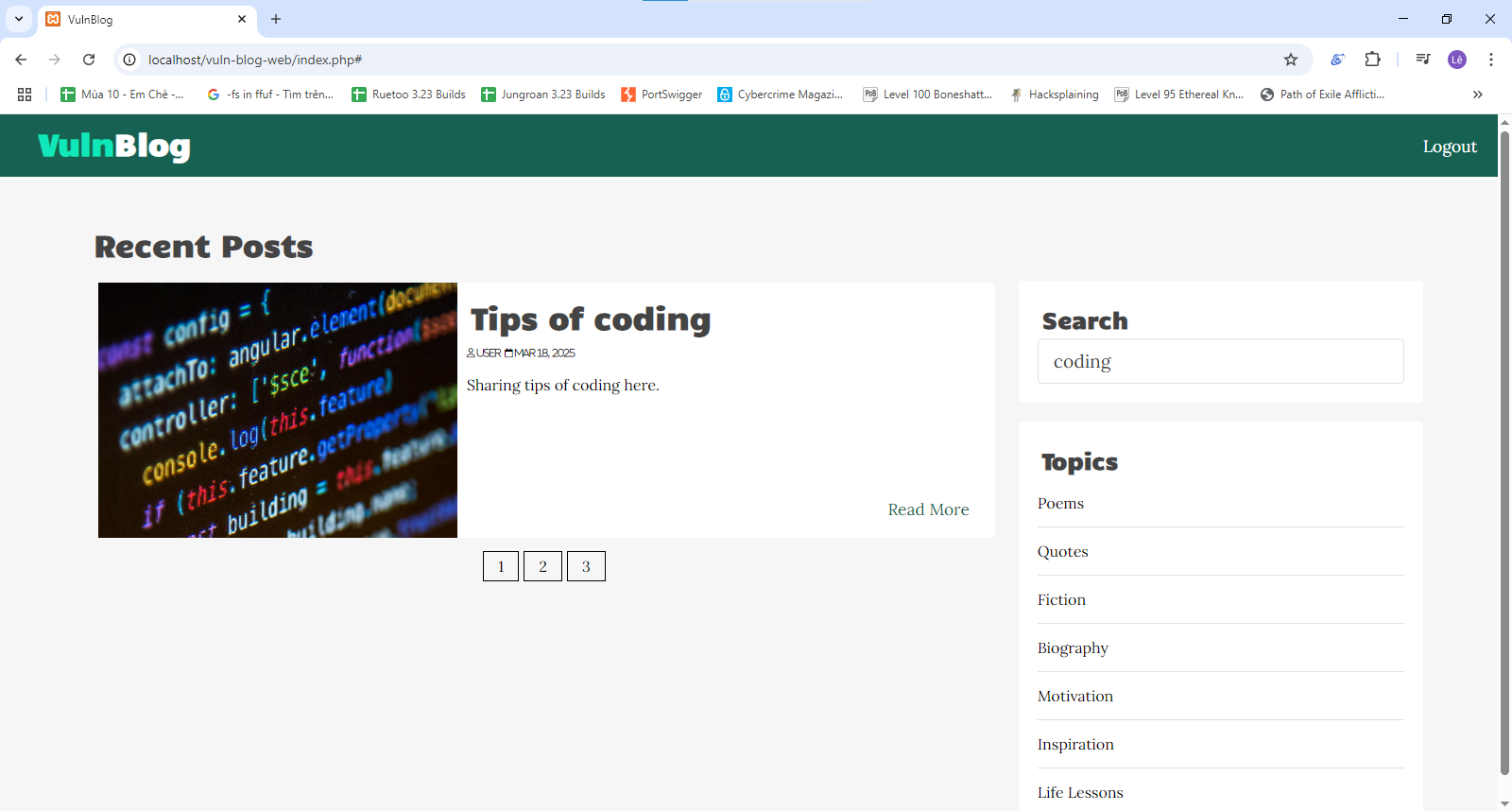


Ngoài ra, ta có thể kiểm tra lỗ hổng SQL injection theo kiểu Time-based với payload “ ' UNION SELECT null,null,null,SLEEP(5) # “ bằng cách sử dụng UNION, vì không biết table hiện tại có bao nhiêu cột nên ta sẽ thử bằng cách từ từ thêm các giá trị null thay cho các cột đến khi đúng số cột có trong table thì câu lệnh SLEEP sẽ được thực hiện:

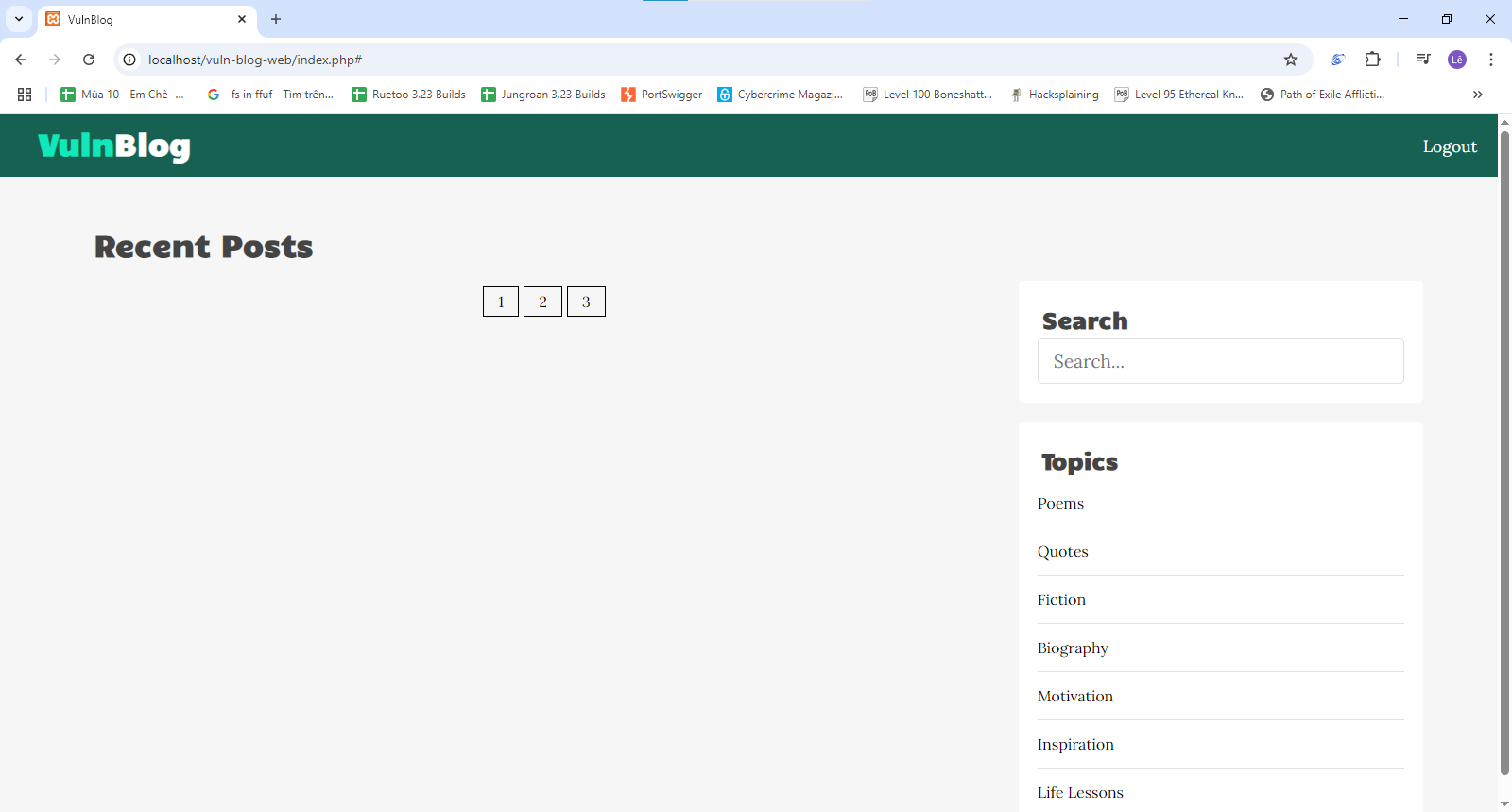


**2. Chức năng Search**

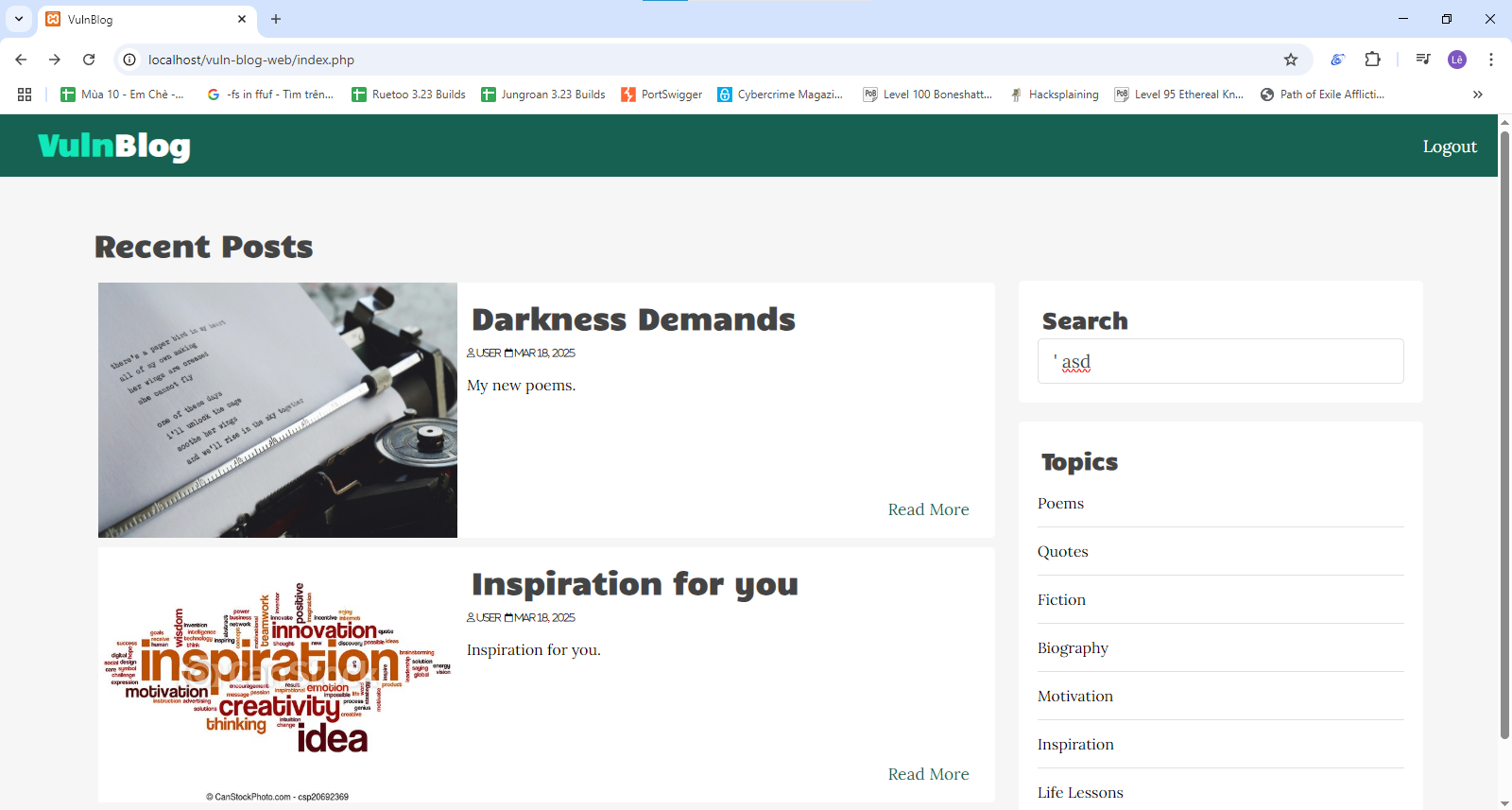
Tiếp cận trang web, ta có một chức năng khác là chức năng search, cho phép trang web hiển thị các blog phù hợp với từ khóa tìm kiếm của chúng ta:



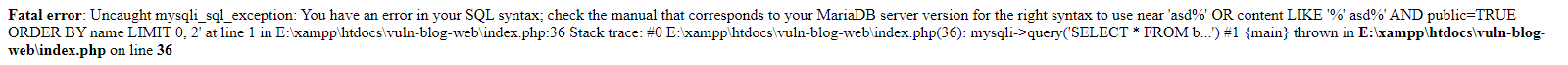
Chúng ta sẽ kiểm tra xem chức năng này có lỗ hổng SQL injection không với payload như trên “ ‘ “:



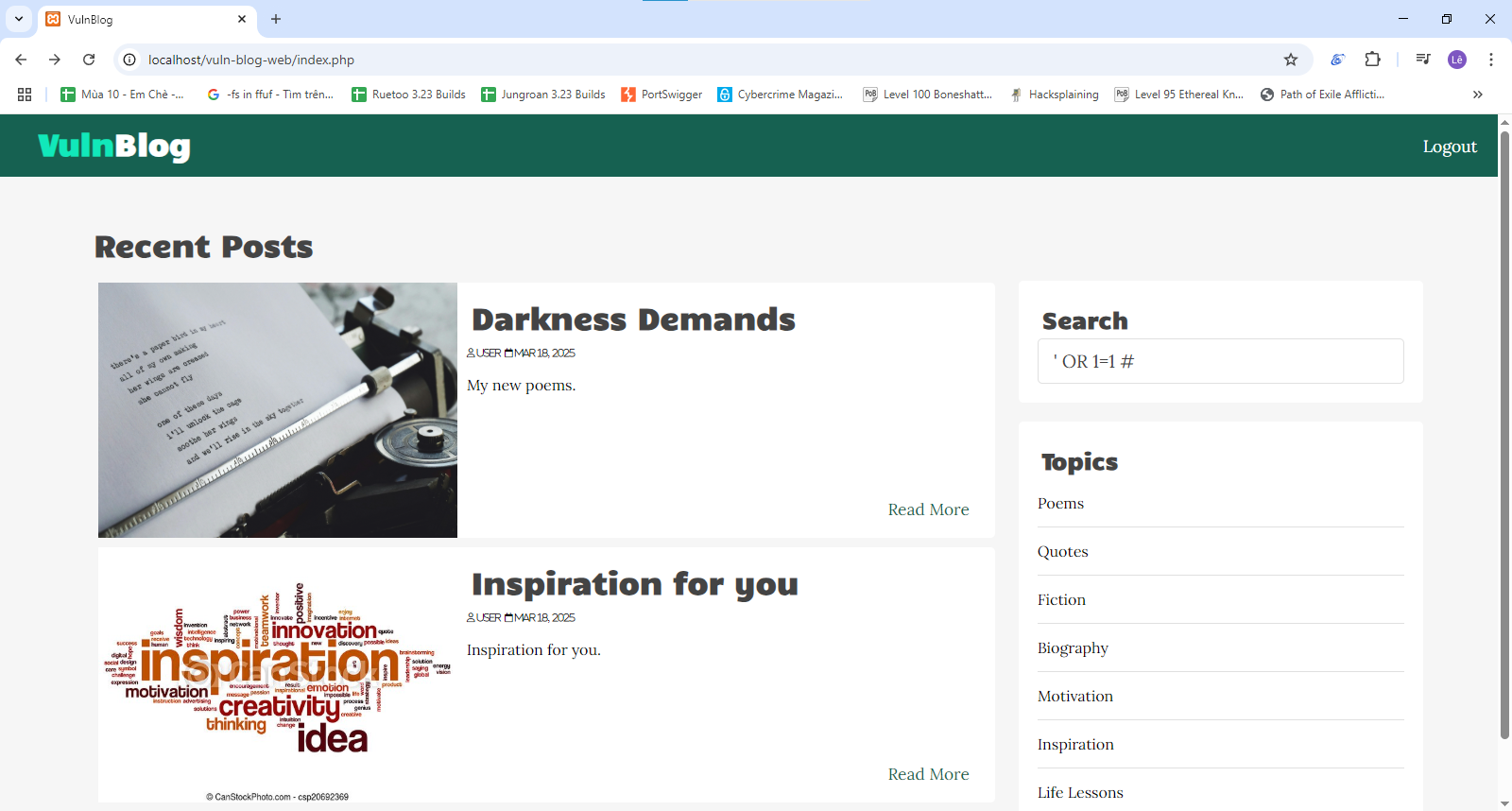
Trang web không trả lại bất kì blog nào cũng như thông báo lỗi nào. Chúng ta sẽ thử thêm một vài payload “ ‘asd “:



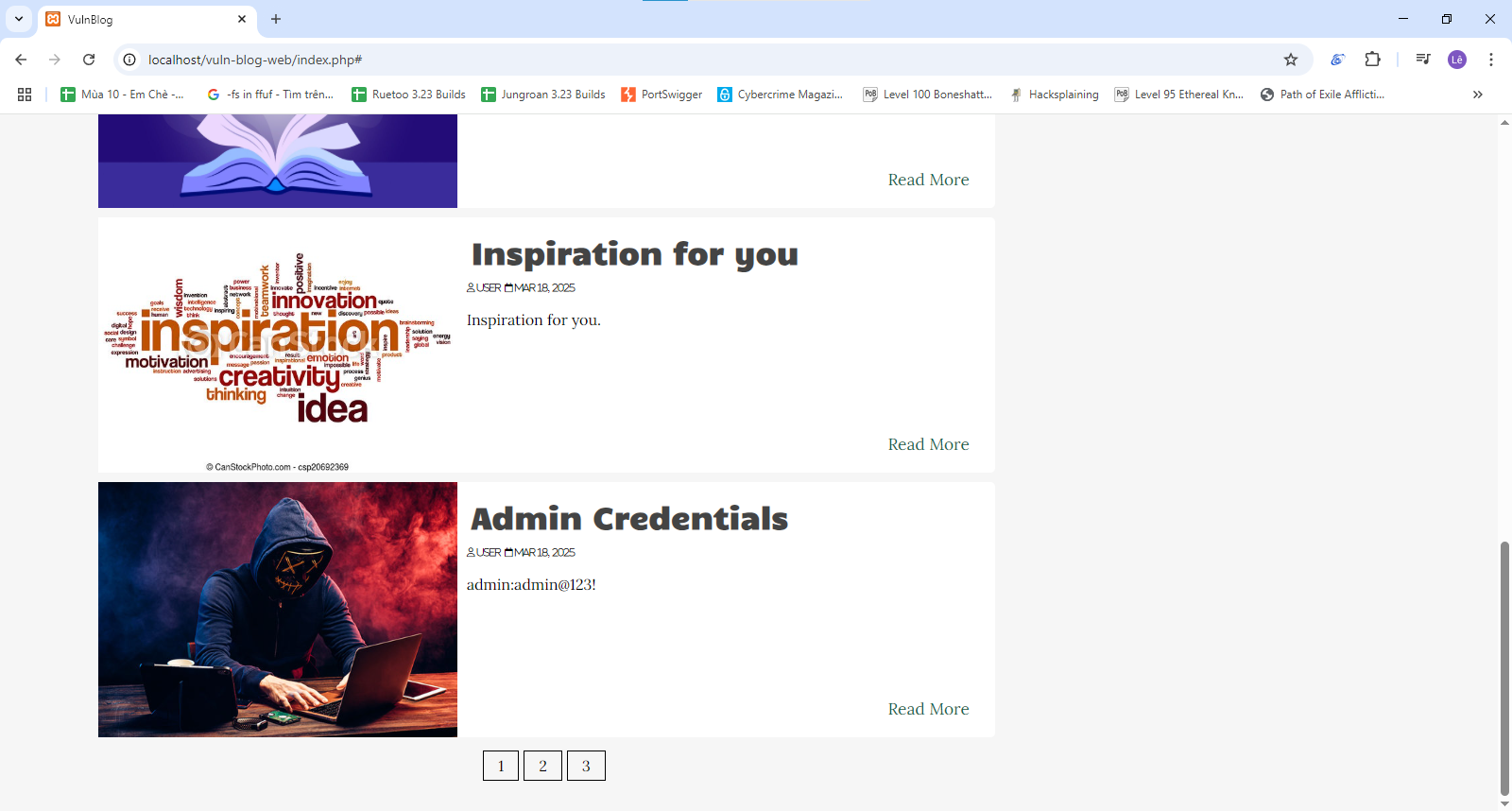
Kết quả là ứng dụng trả về một thông báo lỗi:



Thông báo lỗi cho ta thấy được câu truy vấn của chức năng search, và ở đây có một trường tên là public đang được đặt giá trị TRUE. Ta sẽ thử sử dụng payload Boolean SQL injection để kiểm tra xem có các blog nào đang bị ẩn hay không:



Kết quả là trang web trả về một blog bị ẩn có chứa thông tin của admin:



Chúng ta cũng có thể thực hiện lỗ hổng Time-based SQL injection với payload: “ ' UNION SELECT null,null,null,SLEEP(5) # “:

