Các lỗi zap quét

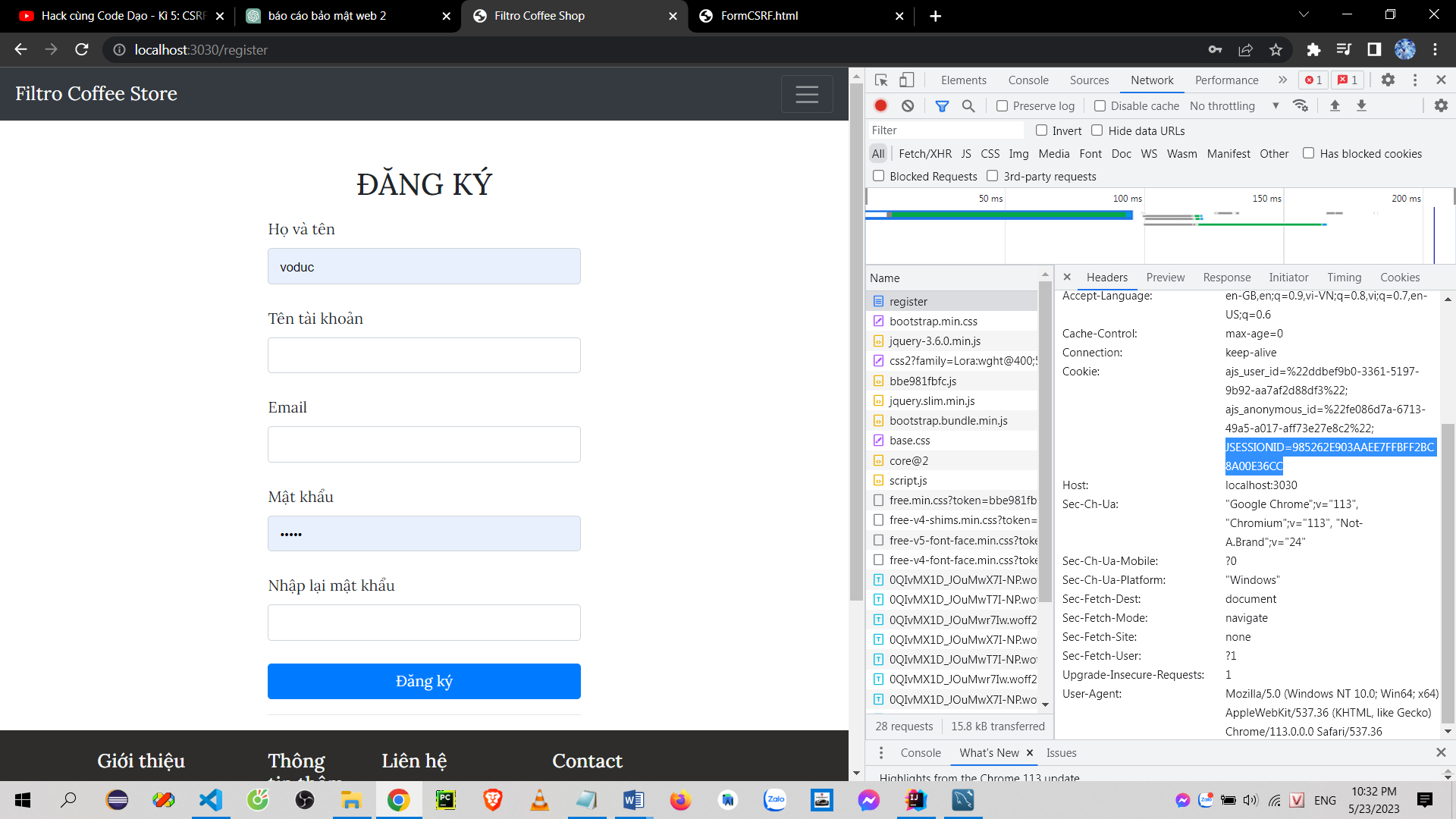
# Absence of Anti-CSRF Tokens

CSRF là gì

CSRF là viết tắt của Cross-Site Request Forgery, có nghĩa là giả dạng request từ site khác mà không phải là site chính chủ.

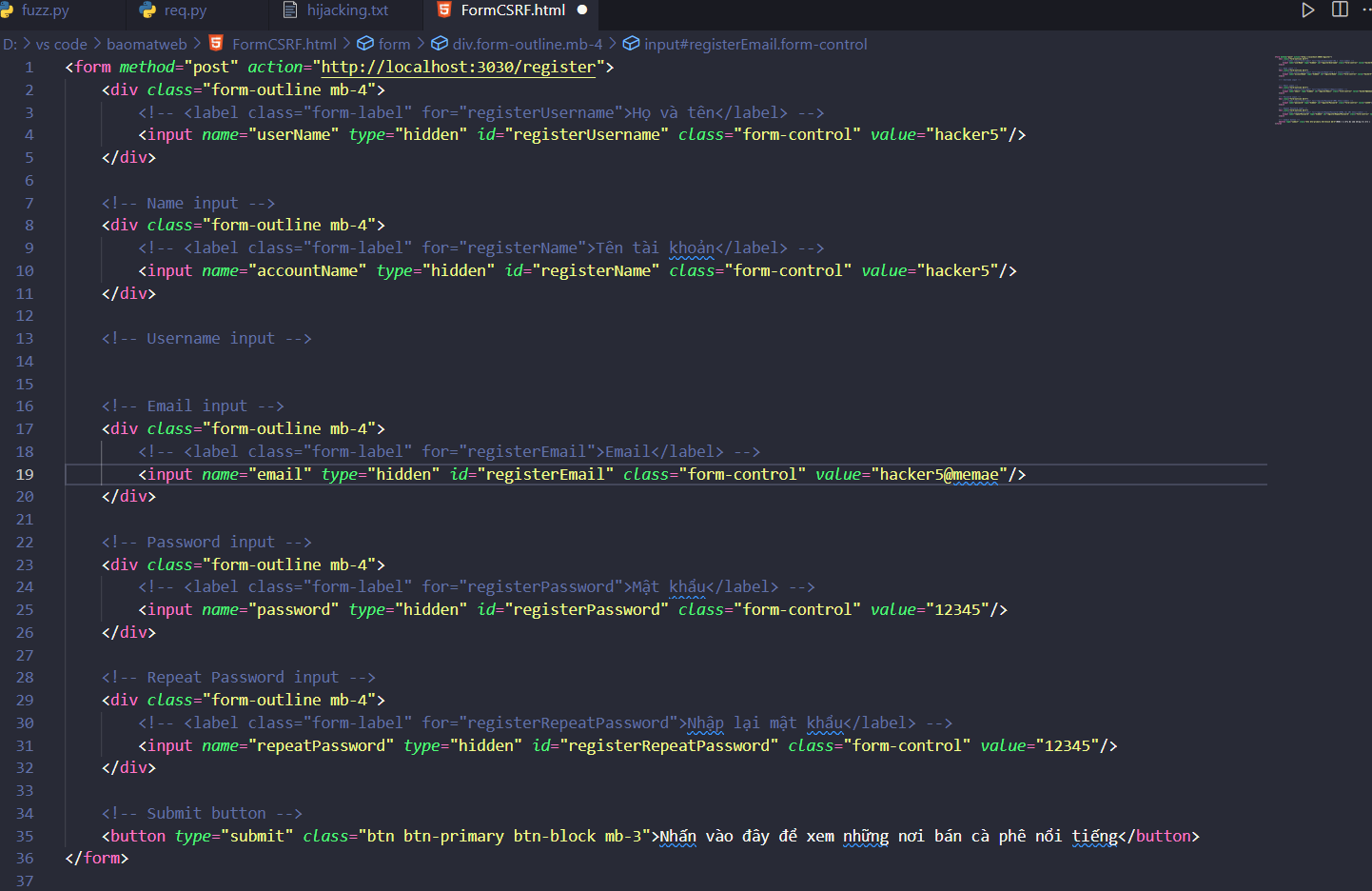
Khi người dùng đến trang web của bạn, người dùng đó sẽ có 1 cookie chứng tỏ người dùng đó là ai, và nếu code không cẩn thận thì hacker có nhiều cách để gửi thông tin request đó tới server của bạn, sử dụng cookie của người dùng để làm những hành động không mong muống.

Ví dụ hacker sẽ sử dụng trang web của hacker để gửi request tới server của mình gắn kèm cookie của người dùng. Lỗi này xuất hiện khi người dùng tương tác với một trang web nào đó không phải của bạn nhưng trang web này giả dạng một request người dùng tới trang web của mình, trang web của mình thấy cookie nên sẽ nghỉ là đây là người dùng bình thường nhưng nó là request do hacker tạo ra. Ví dụ như hacker có danh sách gmail, tên của người dùng, hacker tiến hành gửi gmail có nội dung chứa form hoặc một nút bấm gì đấy, khi người dùng nhấn vào thì ngay lập tức, trang web giả mạo của hacker sẽ gửi 1 request đăng ký tài khoản mới tới server của chúng ta. Nếu số lượng người dùng quá lớn sẽ gây tràn cơ sở dữ liệu, tải dữ liệu chậm,...

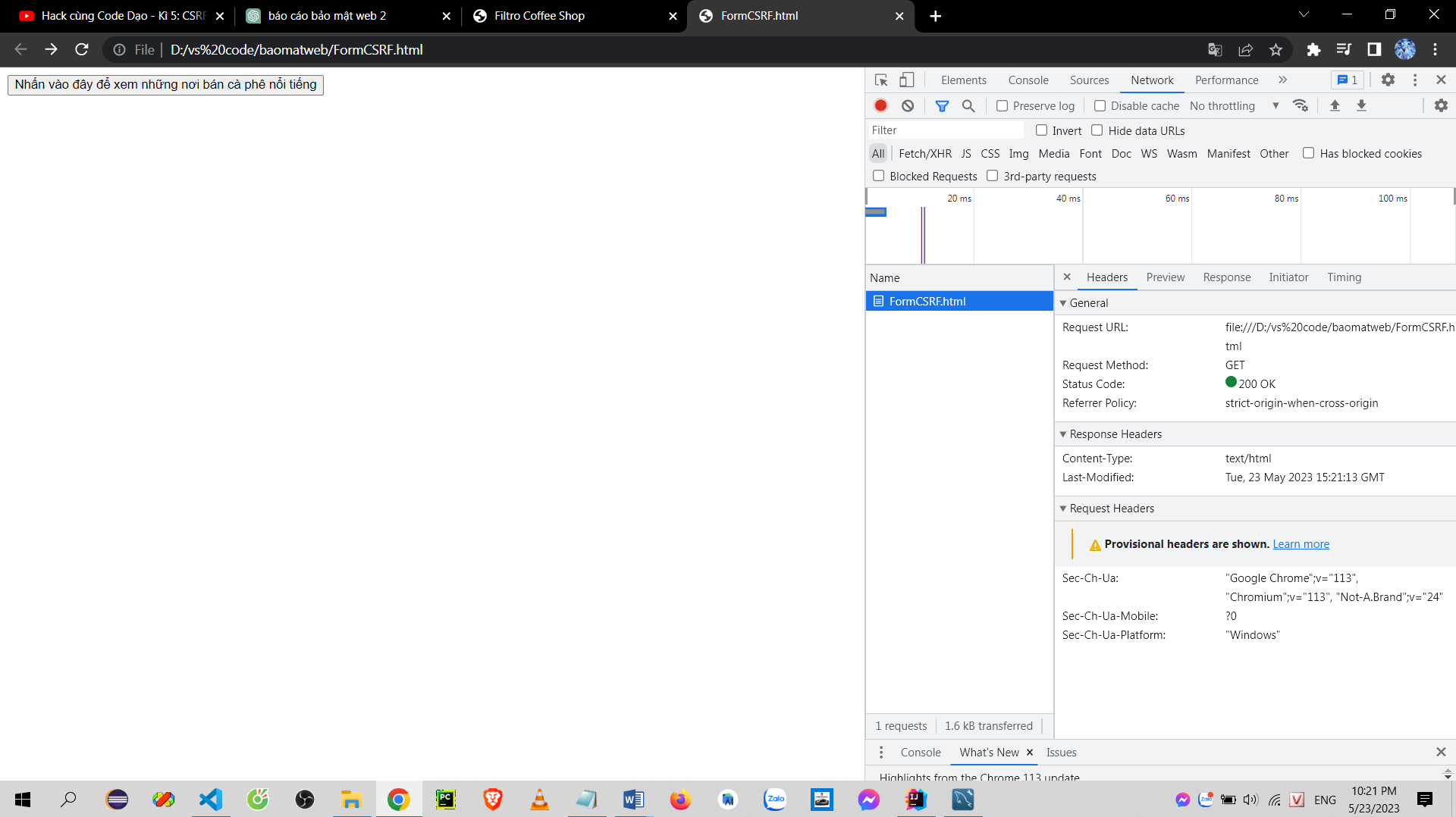


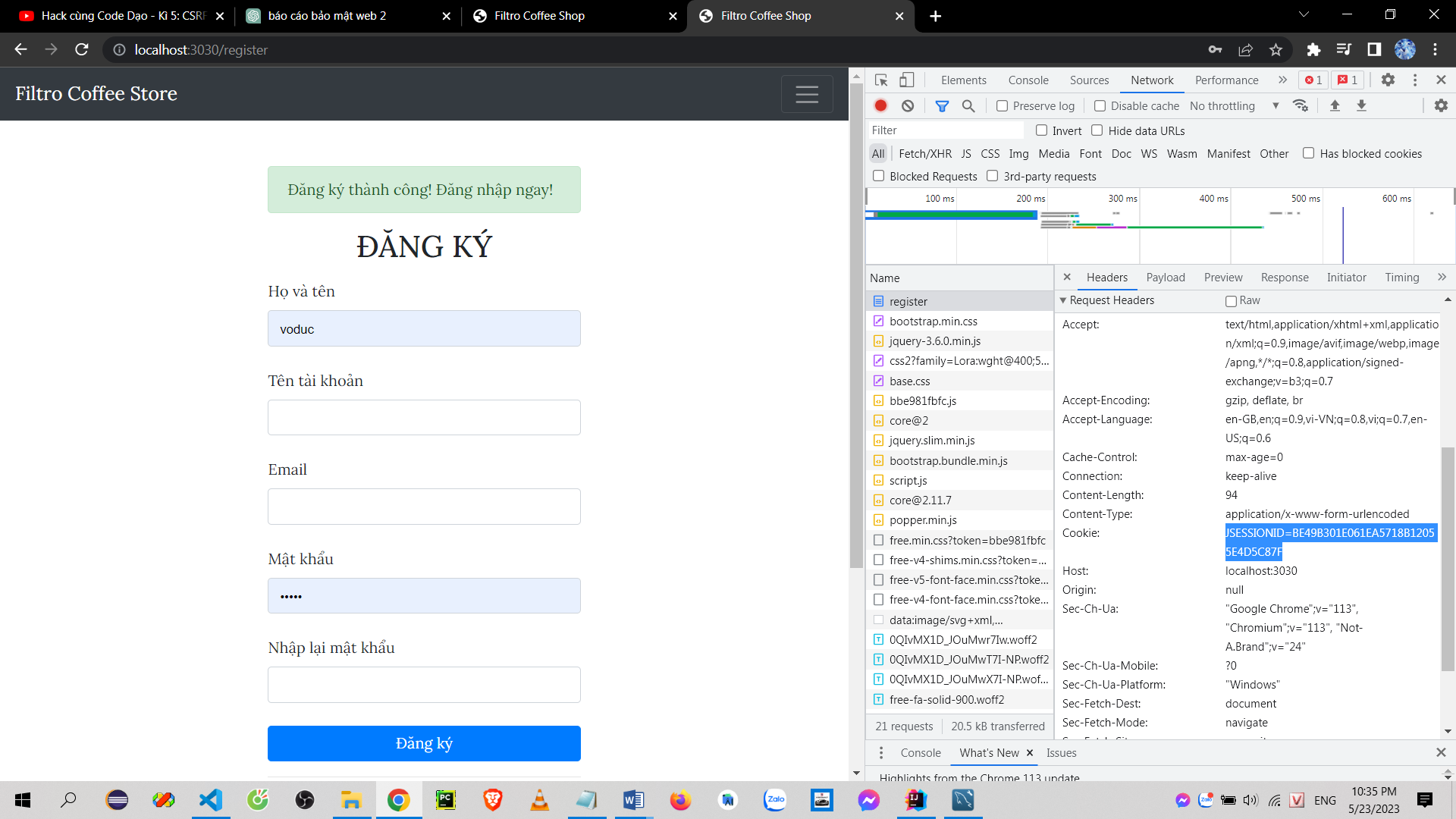
Cookie: JSESSIONID=BE49B301E061EA5718B12055E4D5C87F

Khi người dùng vào trang đăng ký thì server thì cookie tương ứng được tạo ra, hacker có thể sử dụng cookie này để tạo 1 tài khoản mà người dùng không mong muốn khi họ thao tác trên web của hacker.



Web của hacker là một form với các trường ẩn, thiết lập giá trị mặc định, sẽ tạo tài khoản userName: hacker5, accountName: hacker5, email: hacker5@memae, password: 12345 sau khi người dùng nhấn nút “Nhấn vào đây để xem những nơi bán cà phê nổi tiếng”.

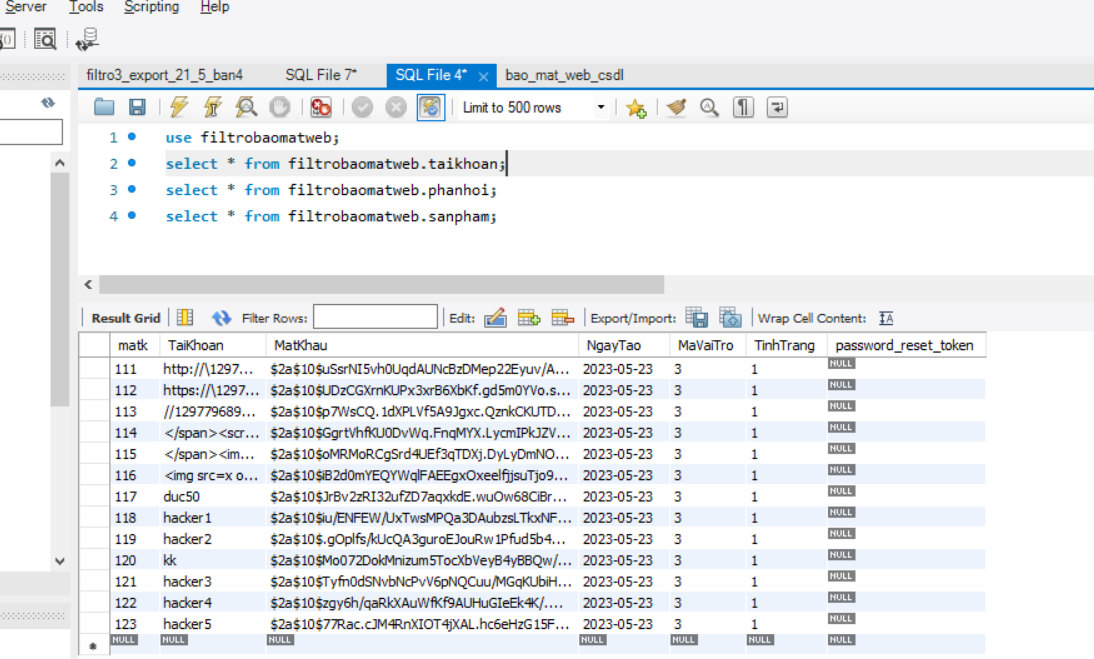




Cookie: JSESSIONID=BE49B301E061EA5718B12055E4D5C87F

(Để cookie 2 bên giống nhau thì sau khi load trang đăng ký bên người dùng thì người dùng nhấn nút để bên trang giả mạo để đồng bộ cookie, do cookie tự thay đổi theo thời gian cố định)

Sau khi người dùng nhấn nút, thì một tài khoản hacker đã tự động đăng ký dựa trên cookie của người dùng, giả mạo danh tính.

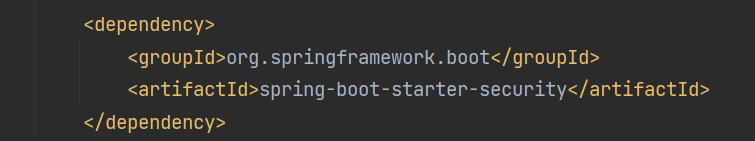


Hậu quả: ngoài đăng ký một tài khoản không mong muốn để bị mạo danh, còn có thể bị thay đổi mật khẩu, thực hiện thanh toán trái phép trên trang thương mại điện tử, đăng bài viết hoặc gửi tin nhắn giả mạo, xoá dữ liệu.

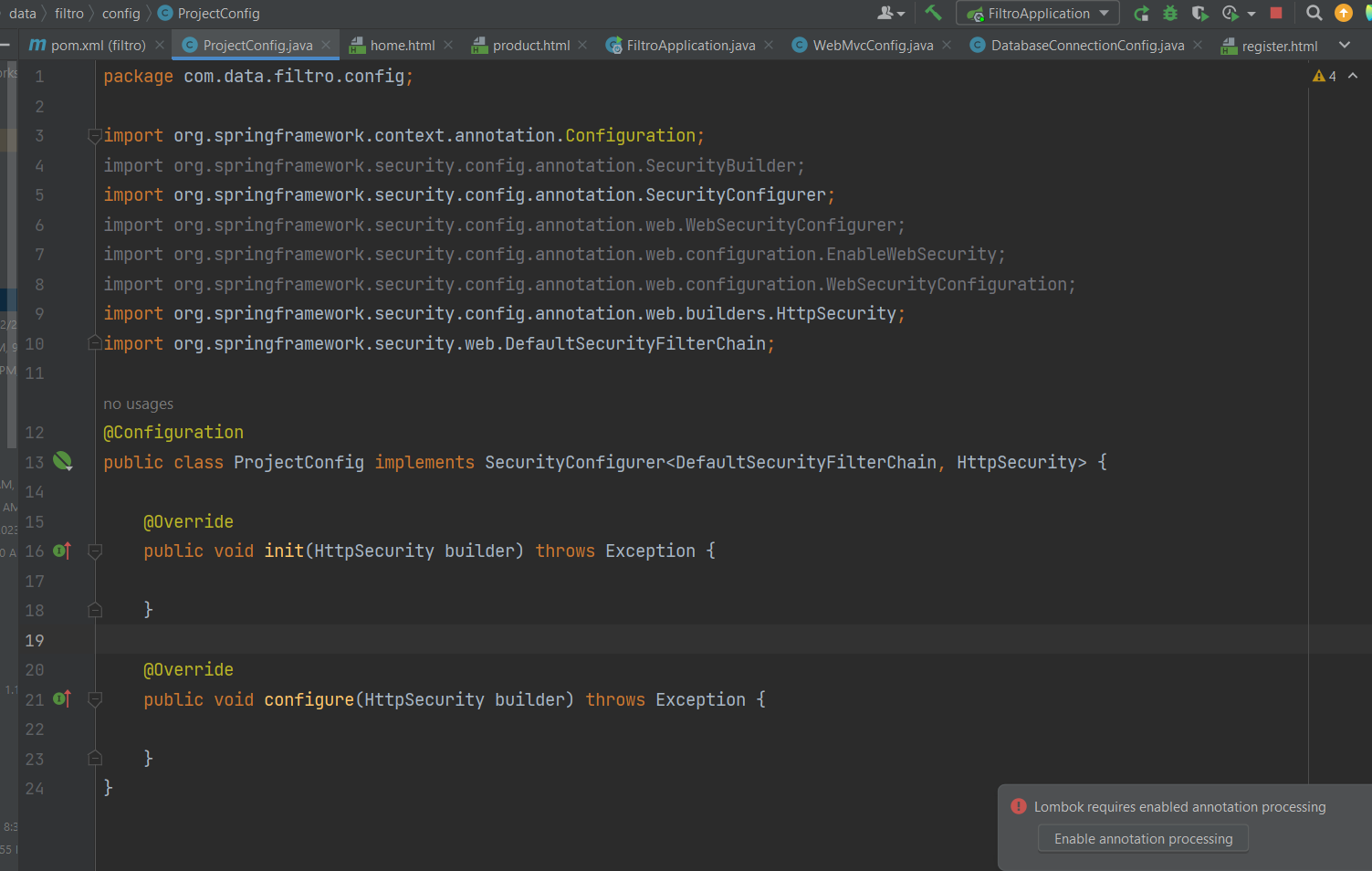
Xử lý

Vì thế CSRF tokens là một dạng mã được tạo ra và gắn kết với mỗi yêu cầu người dùng. Khi người dùng truy cập vào trang web, máy chủ sẽ sinh ra một mã token duy nhất cho phiên làm việc của người dùng. Khi người dùng thực hiện một hành động từ trang chính chủ như gửi biểu mẫu thì mã token sẽ được gửi cùng với yêu cầu. Trang web sẽ kiểm tra xem mã token có khớp với trạng thái người dùng hay không.

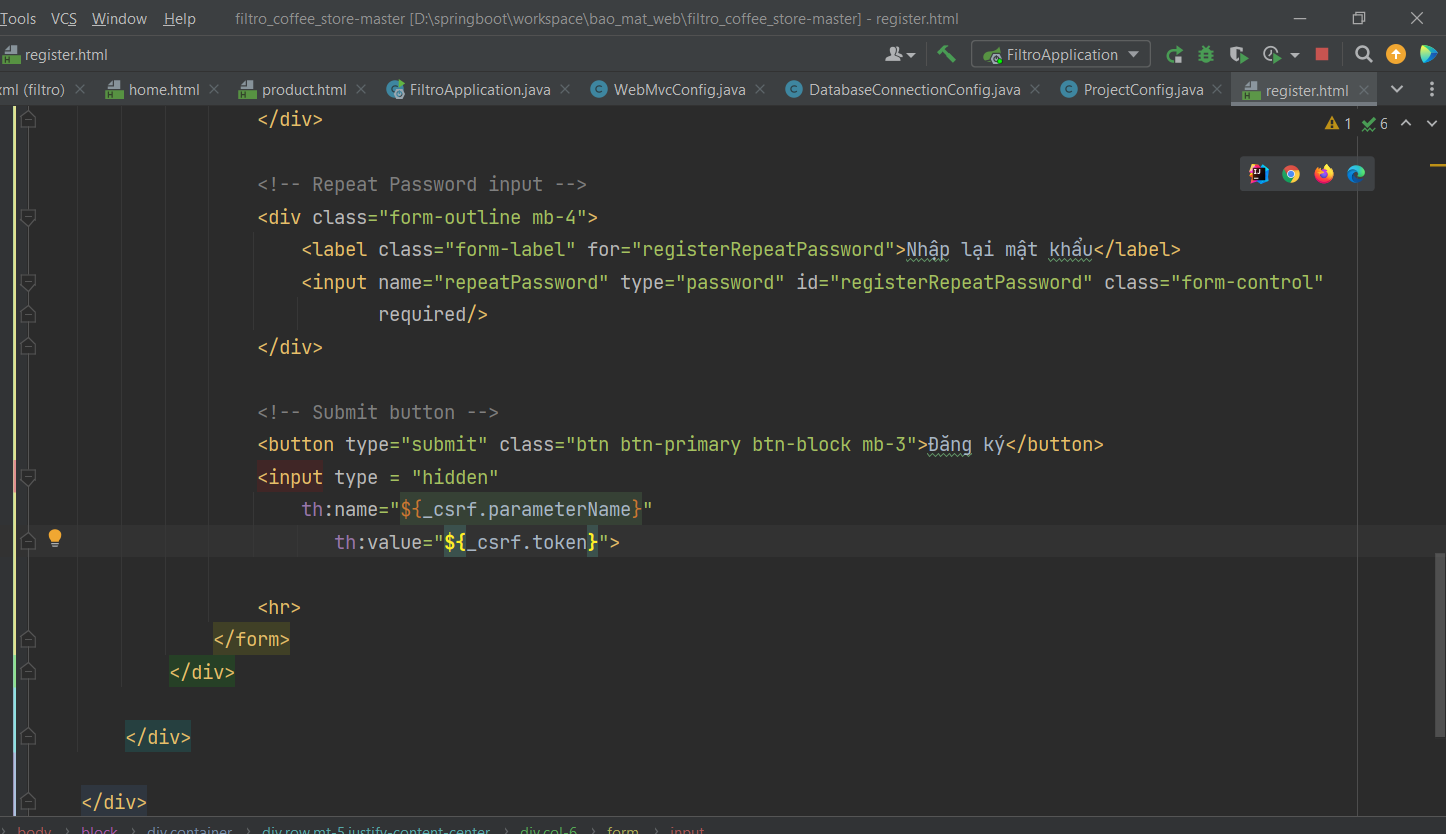
Trong file pom thêm dependency



Tạo 1 file config để cấu hình cho web



Khi phương thức configure được gọi, thì mặc định http.csrf() sẽ được bật

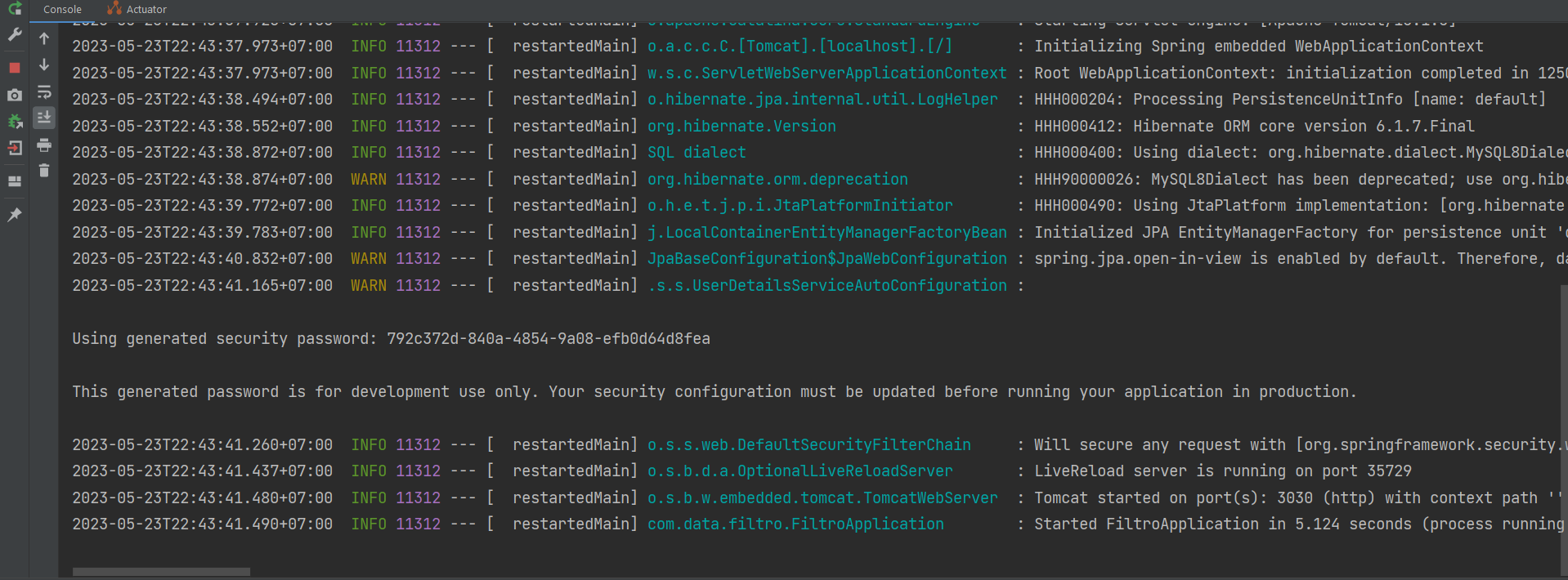


Thêm <input type = "hidden"

th:name="${\_csrf.parameterName}"

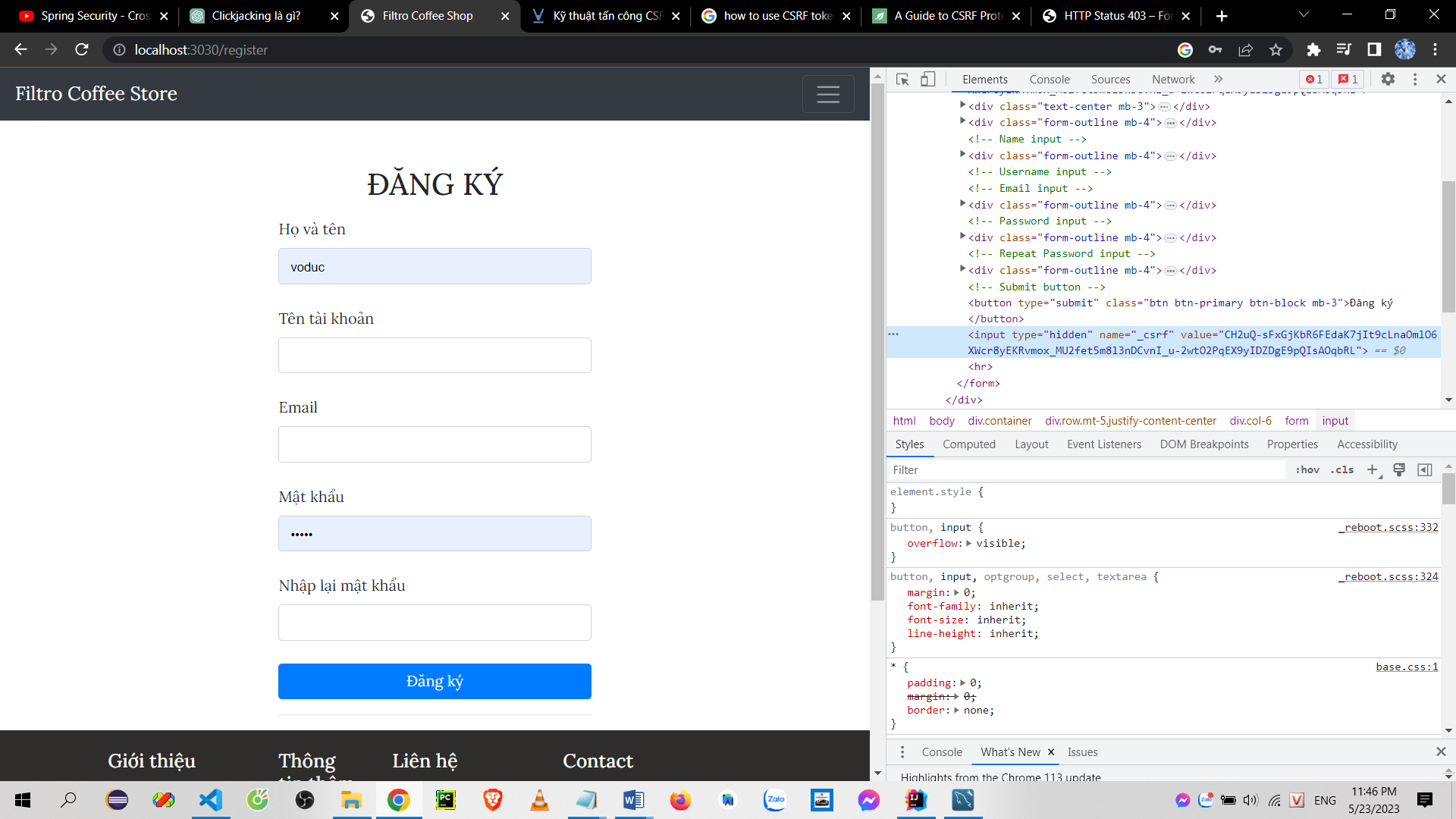
th:value="${\_csrf.token}">

dưới nút Submit của form

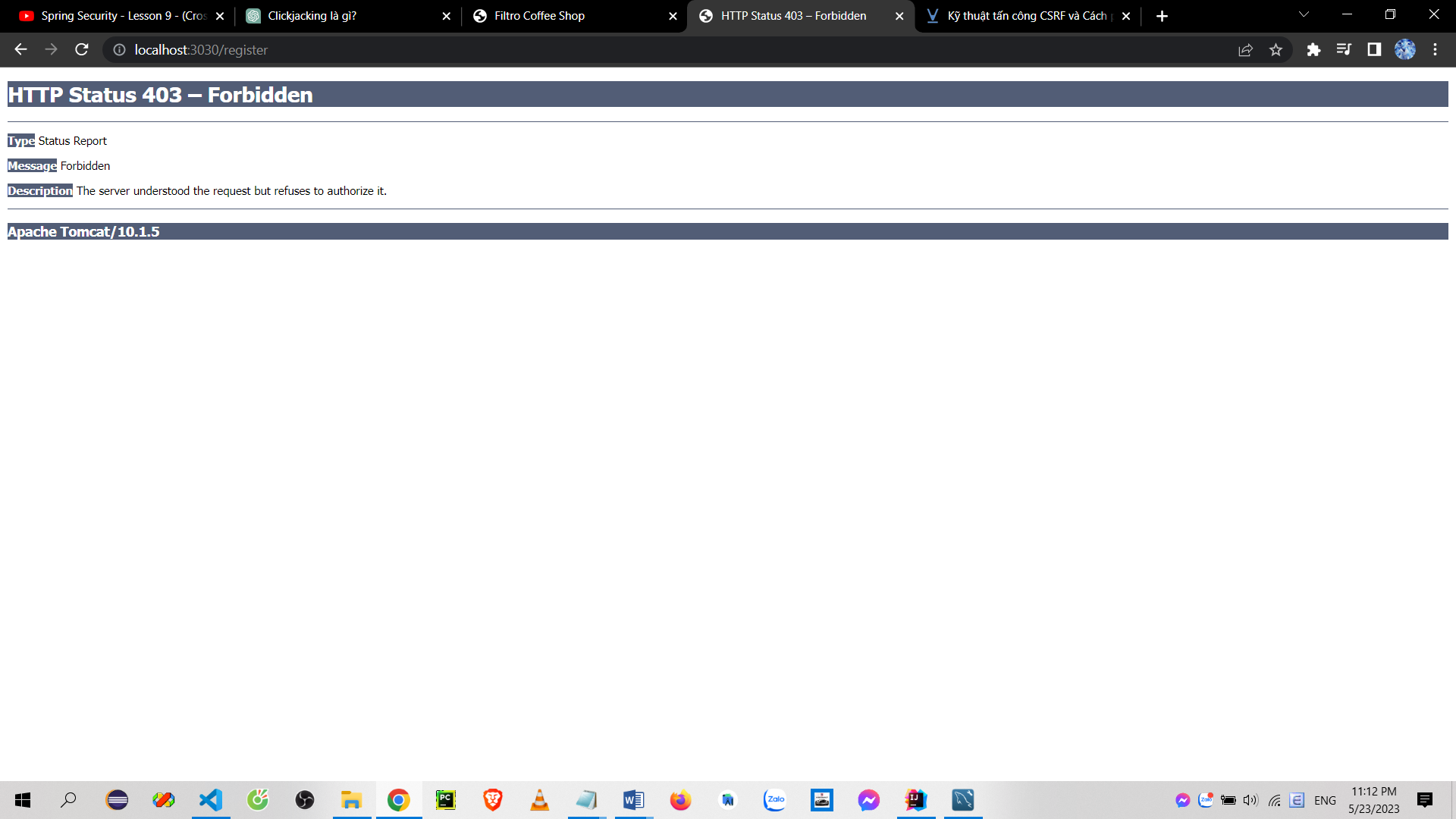


Using generated security password: 792c372d-840a-4854-9a08-efb0d64d8fea

Khi truy cập trực tiếp thì form đăng ký sẽ được cấp token CSRF



Bên web giả mạo thì không truy cập được thì không nhận được CSRF token được gửi từ server, trừ khi truy cập trực tiếp



(Sửa file pom, thêm file ProjectConfig, thêm 2 dòng trong register html)

Content Security Policy (CSP) Header Not Set

Missing Anti-click jacking Header

Cross-Site Scripting