**BÀI THỰC HÀNH SỐ 9**

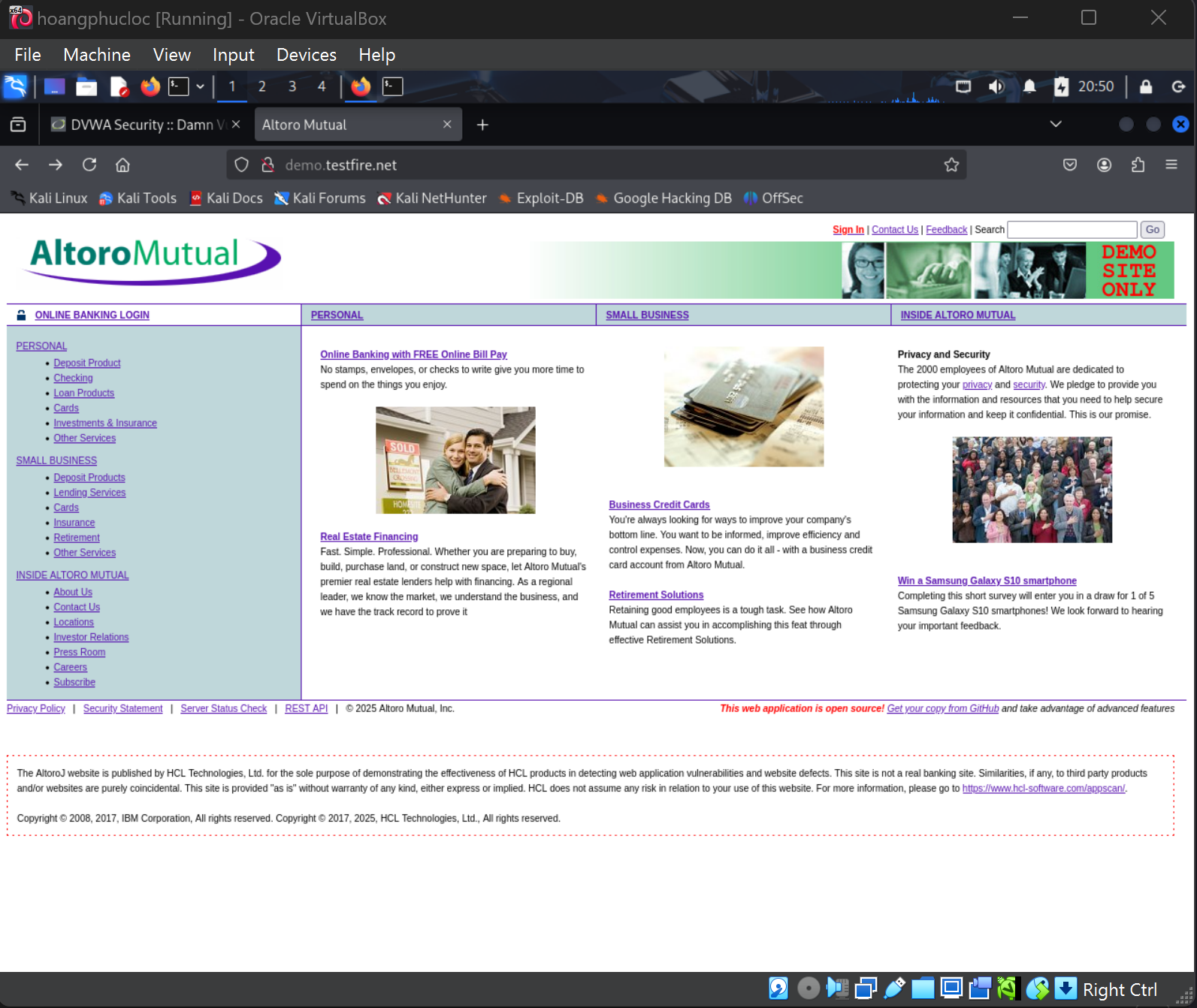
**KIỂM THỬ LỖ HỔNG XSS và CSRF**

Họ và tên: Hoàng Phúc Lộc – 1050080058

**Phần 1: Kiểm thử XSS**

Thực hiện kiểm thử lỗ hổng XSS trên website: chọn 1 trong các trang sau để thử nghiệm

1. <http://demo.testfire.net>



**Câu 1**

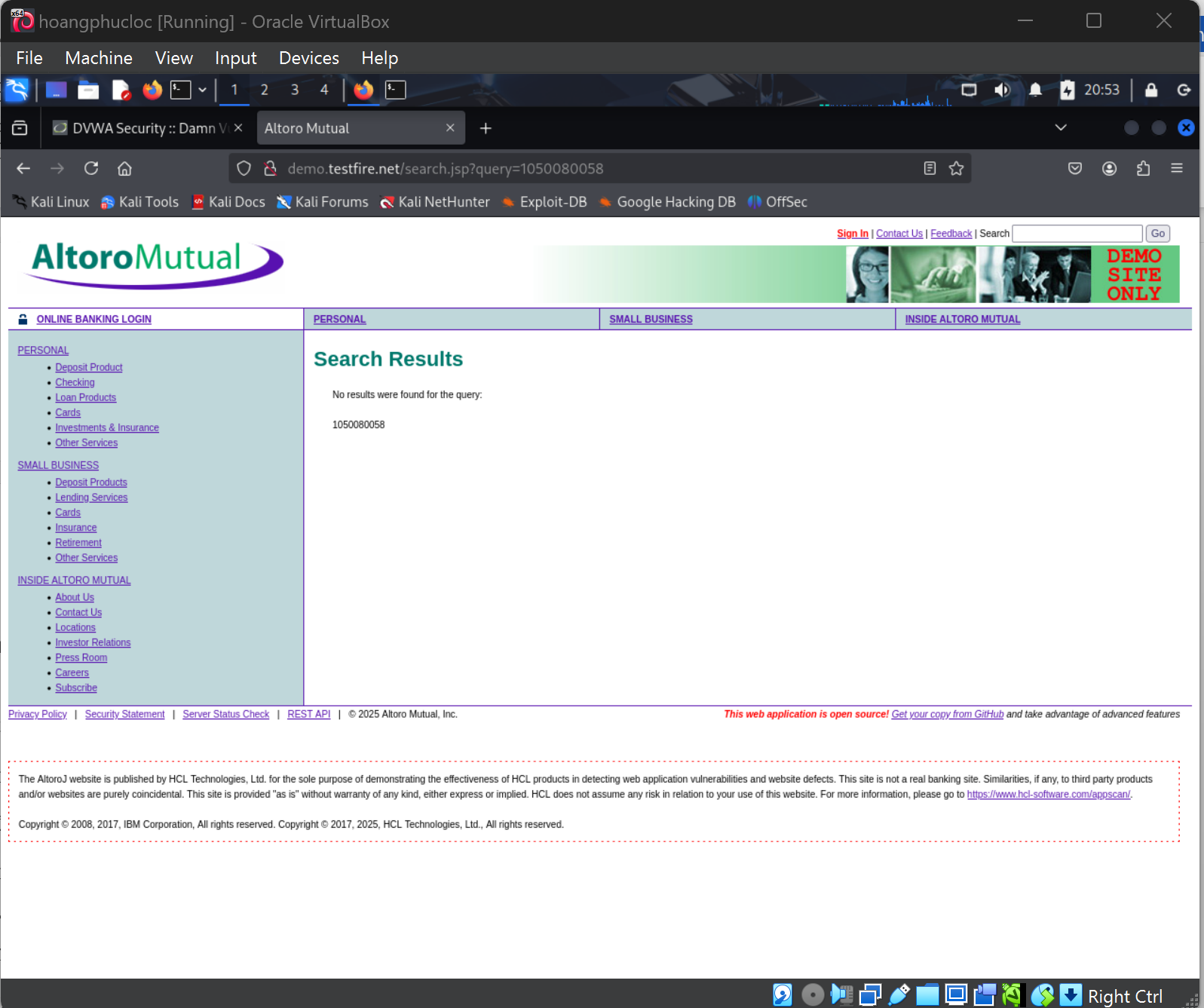
**Thực hiện kiểm thử lỗ hổng XSS để thực thi đoạn mã hiển thị thông báo có chứa 4 số cuối MSSV.**

Hãy giải thích:

- Xác định các tham số đầu vào: 0.25 điểm

Tham số đầu vào: query

http://demo.testfire.net/search.jsp?query=1050080058



- Xác định giá trị đầu vào xuất hiện trong trang kết quả: 0.25 điểm

Giá trị query=1050080058 được phản hồi trực tiếp vào HTML

- Các bước kiểm thử và phán đoán: 0-1 điểm theo các bước kiểm thử và logic trong phán đoán

B1: Truy cập <http://demo.testfire.net/search.jsp>

B2: Quan sát URL và phản hồi HTML: URL truyền query, HTML hiển thị lại query

B3: Thử inject một đoạn JavaScript cơ bản vào ô tìm kiếm: <script>alert(1050080058)</script>

B4: Quan sát kết quả

Phán đoán:

B1: Tìm kiếm 1 từ khoá đơn giản để kiểm tra.

B2: Nếu giá trị hiện nguyên trong HTML, khả năng XSS cao.

B3: Nếu website không chặn, sẽ thực thi alert.

B4: Nếu popup xuất hiện, thì tồn tại lỗ hổng XSS.

- Giá trị kiểm thử cho thấy lỗ hổng và giải thích: 0.5 điểm

Giá trị kiểm thử : <script>alert(1050080058)</script>

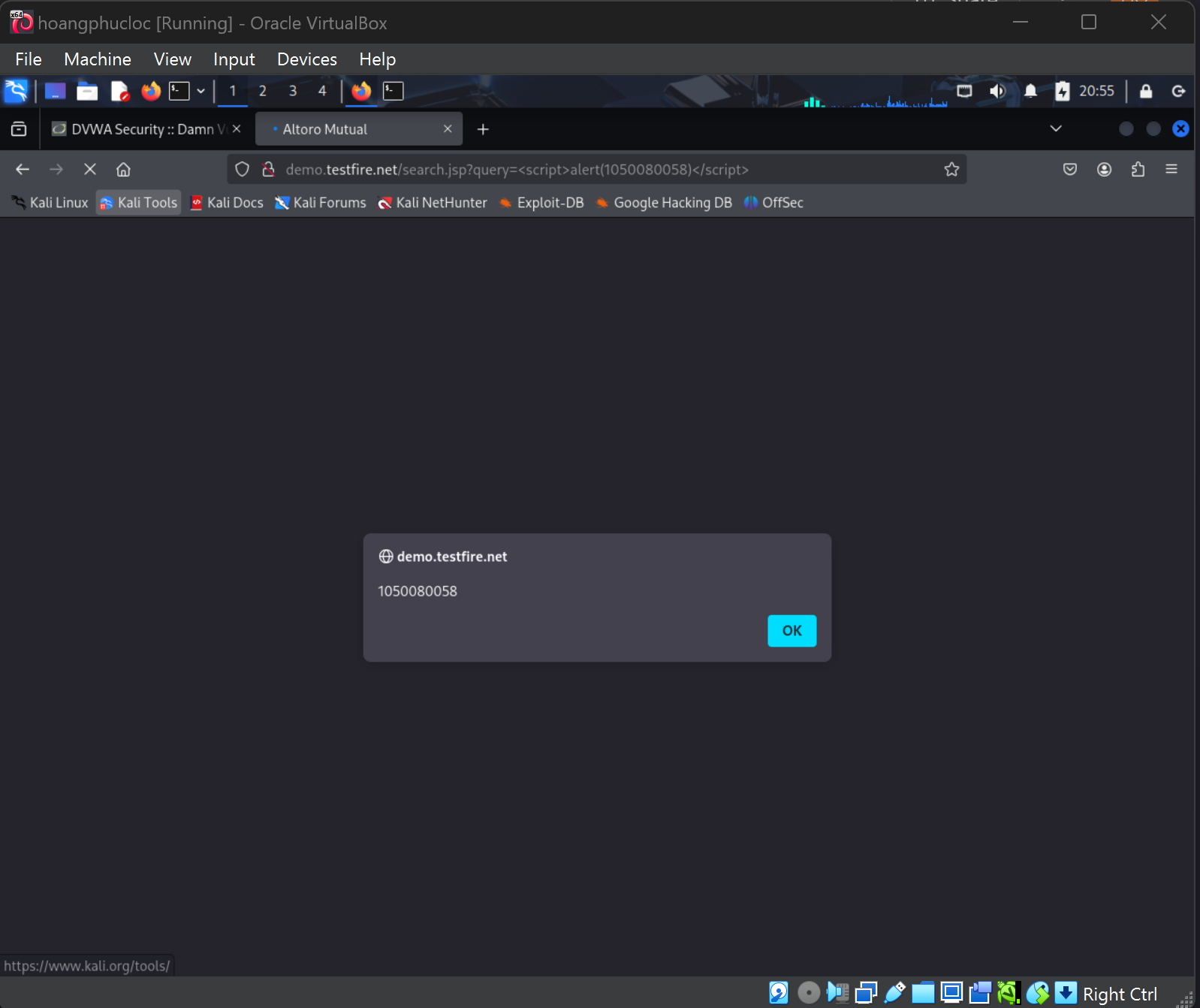
Giải thích:

Khi nhập đoạn này vào ô tìm kiếm và submit:

Nếu trang web không mã hóa hoặc không validate input, thì trình duyệt sẽ thực thi đoạn JavaScript.

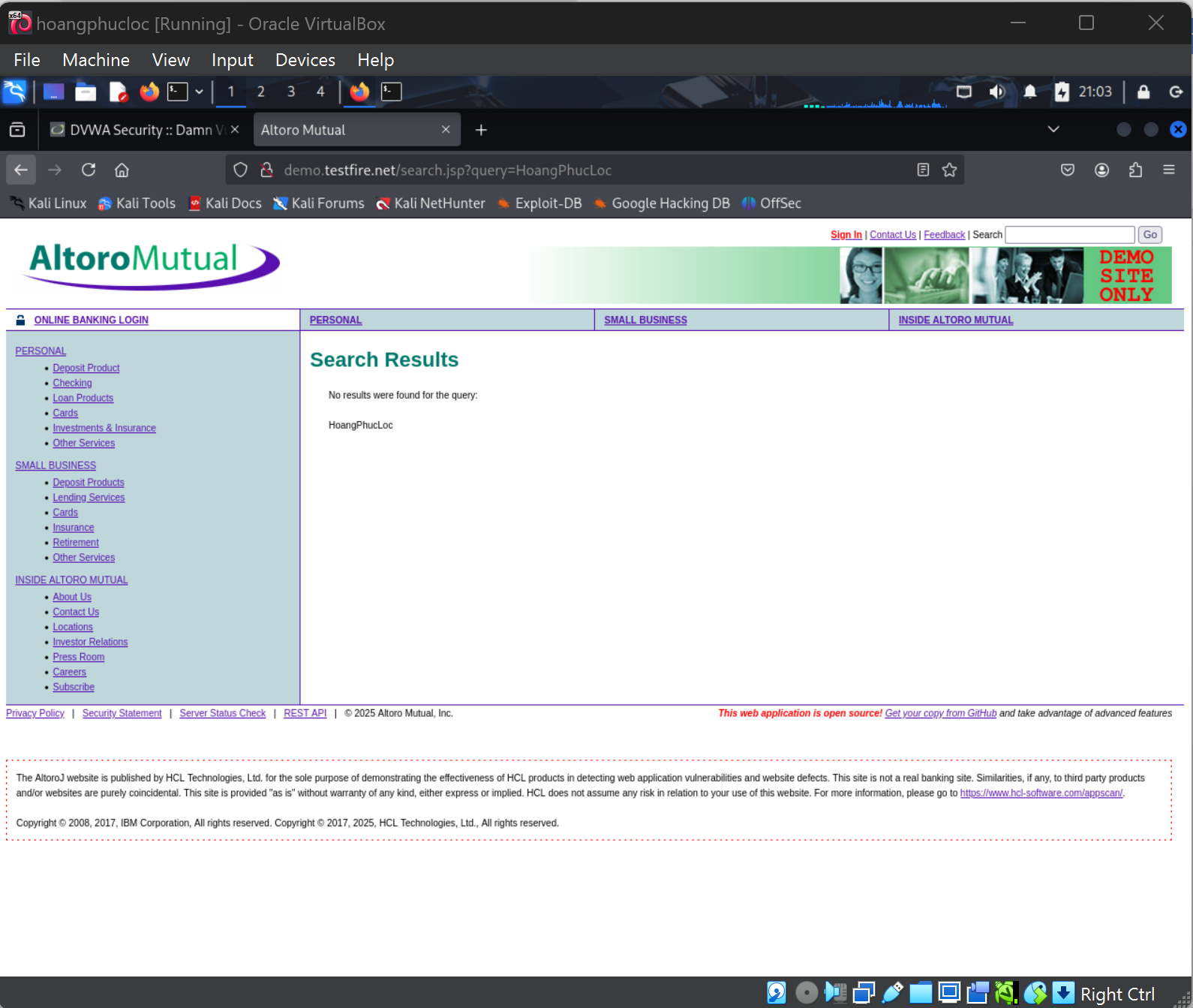
Popup alert(1050080058) xuất hiện ngay lập tức.

Điều này chứng minh rằng website dễ bị tấn công XSS thông qua tham số query.



**Câu 2**

**Thực hiện kiểm thử lỗ hổng XSS để thực thi đoạn mã hiển thị thông báo có chứa tên của sinh viên.**



Điểm cho mỗi tham số đầu vào đã kiểm thử được:

- Xác định các tham số đầu vào: 0.25 điểm

Tham số đầu vào: query

http://demo.testfire.net/search.jsp?query=HoangPhucLoc

- Xác định giá trị đầu vào xuất hiện trong trang kết quả: 0.25 điểm

Giá trị nhập vào (HoangPhucLoc) sẽ xuất hiện lại trong dòng "Search results '...'" ở trang kết quả.

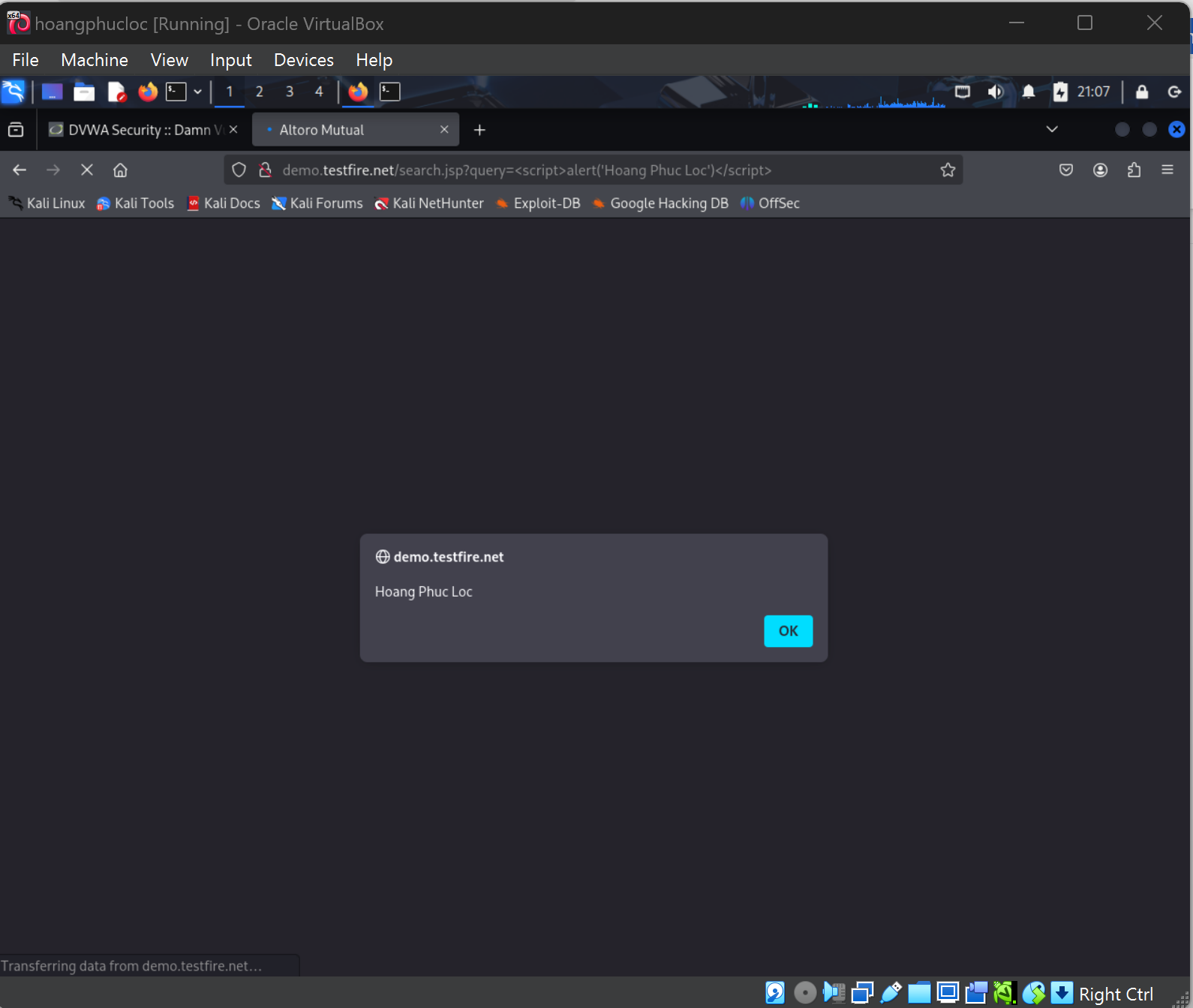
- Các bước kiểm thử và phán đoán: 0-1 điểm theo các bước kiểm thử và logic trong phán đoán

B1: Truy cập trang <http://demo.testfire.net/search.jsp>

B2: Nhập thử 1 chuỗi JavaScript chứa tên sinh viên vào ô tìm kiếm: <script>alert(' Hoang Phuc Loc')</script>

B3: Submit và Quan sát kết quả

Phán đoán: Nếu trình duyệt hiện alert " Hoang Phuc Loc'" thì xác nhận tồn tại lỗ hổng XSS.



- Giá trị kiểm thử cho thấy lỗ hổng và giải thích: 0.5 điểm

Giá trị kiểm thử: <script>alert('Hoang Phuc Loc')</script>

Giải thích:

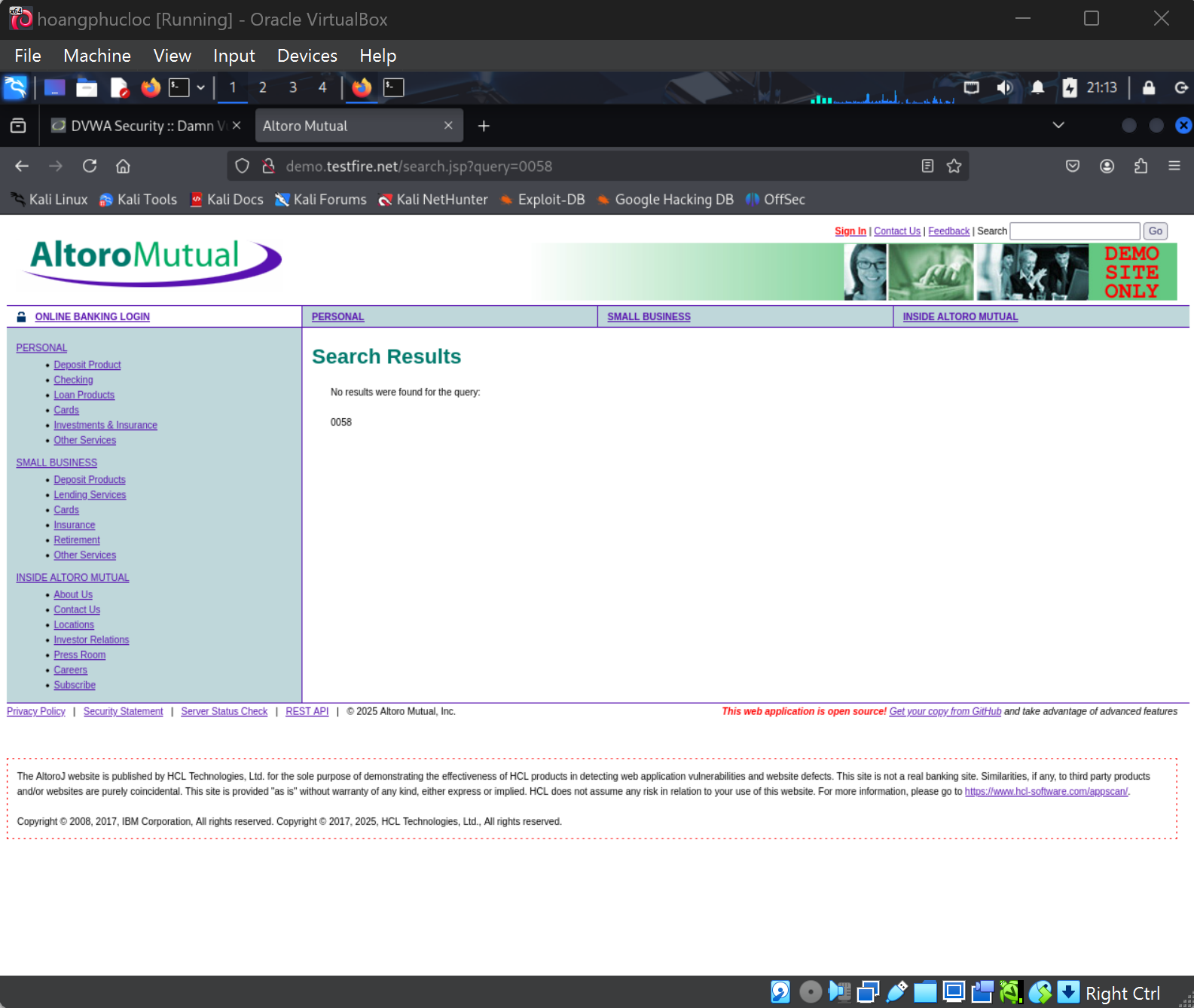
Sau khi nhập và submit, nếu trang web không sanitize input, thì trình duyệt sẽ thực thi script.

Kết quả: Một cửa sổ alert hiện dòng "Hoang Phuc Loc".

Điều này xác nhận rằng website tồn tại lỗ hổng XSS tại tham số query.

**Câu 3**

**Thực hiện kiểm thử lỗ hổng XSS để thực thi đoạn mã hiển thị thông báo có chứa 4 số cuối MSSV.**



Hãy giải thích:

- Xác định các tham số đầu vào: 0.25 điểm

Tham số đầu vào: query

http://demo.testfire.net/search.jsp?query=0058

- Xác định giá trị đầu vào xuất hiện trong trang kết quả: 0.25 điểm

Giá trị query=0058 được phản hồi trực tiếp vào HTML

- Các bước kiểm thử và phán đoán: 0-1 điểm theo các bước kiểm thử và logic trong phán đoán

B1: Truy cập <http://demo.testfire.net/search.jsp>

B2: Quan sát URL và phản hồi HTML: URL truyền query, HTML hiển thị lại query

B3: Thử inject một đoạn JavaScript cơ bản vào ô tìm kiếm: <script>alert(0058)</script>

B4: Quan sát kết quả

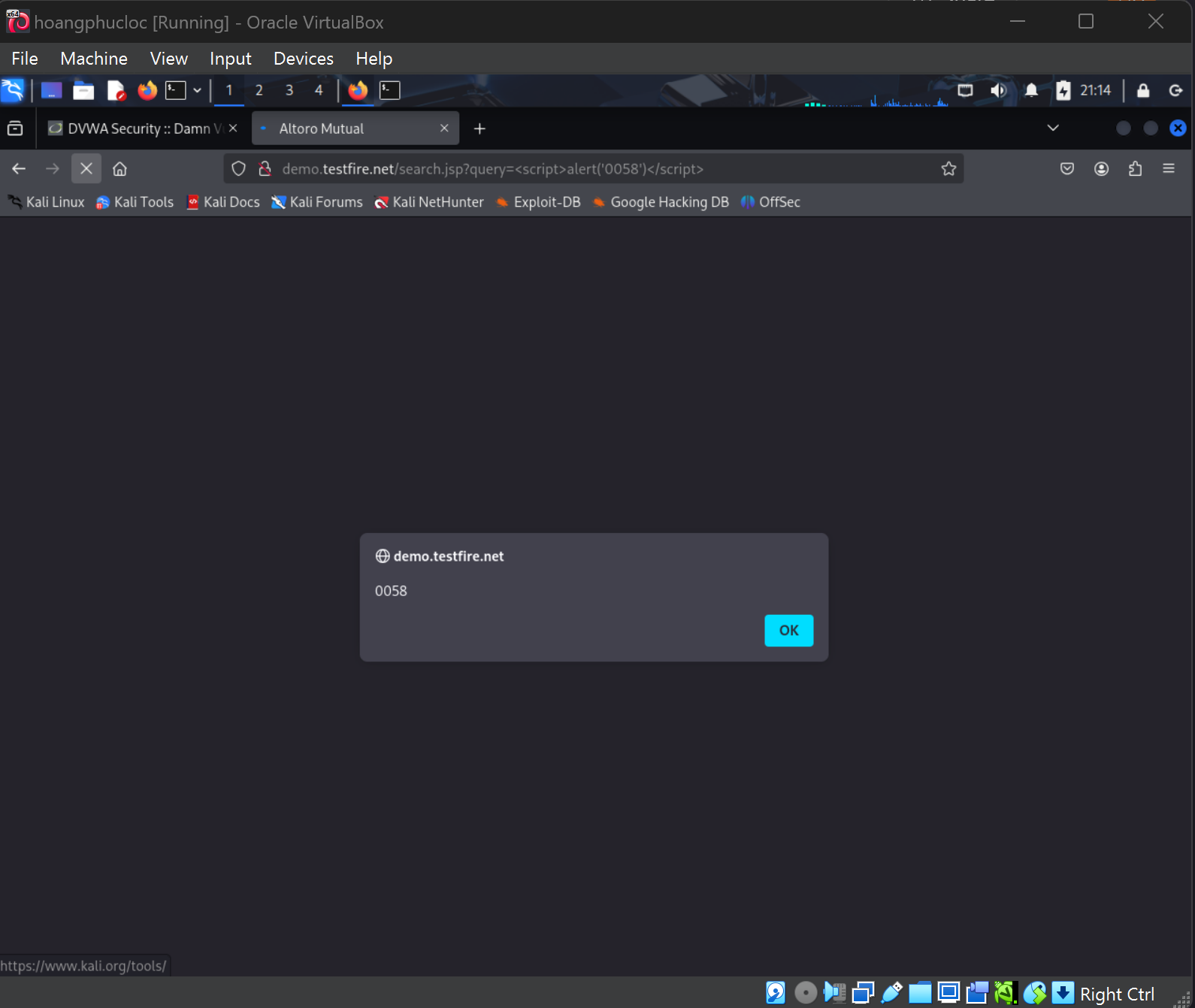
Phán đoán:

B1: Tìm kiếm 1 từ khoá đơn giản để kiểm tra.

B2: Nếu giá trị hiện nguyên trong HTML, khả năng XSS cao.

B3: Nếu website không chặn, sẽ thực thi alert.

B4: Nếu popup xuất hiện, thì tồn tại lỗ hổng XSS.



- Giá trị kiểm thử cho thấy lỗ hổng và giải thích: 0.5 điểm

Giá trị kiểm thử : <script>alert(0058)</script>

Giải thích:

Khi nhập đoạn này vào ô tìm kiếm và submit:

Nếu trang web không mã hóa hoặc không validate input, thì trình duyệt sẽ thực thi đoạn JavaScript.

Popup alert(0058) xuất hiện ngay lập tức.

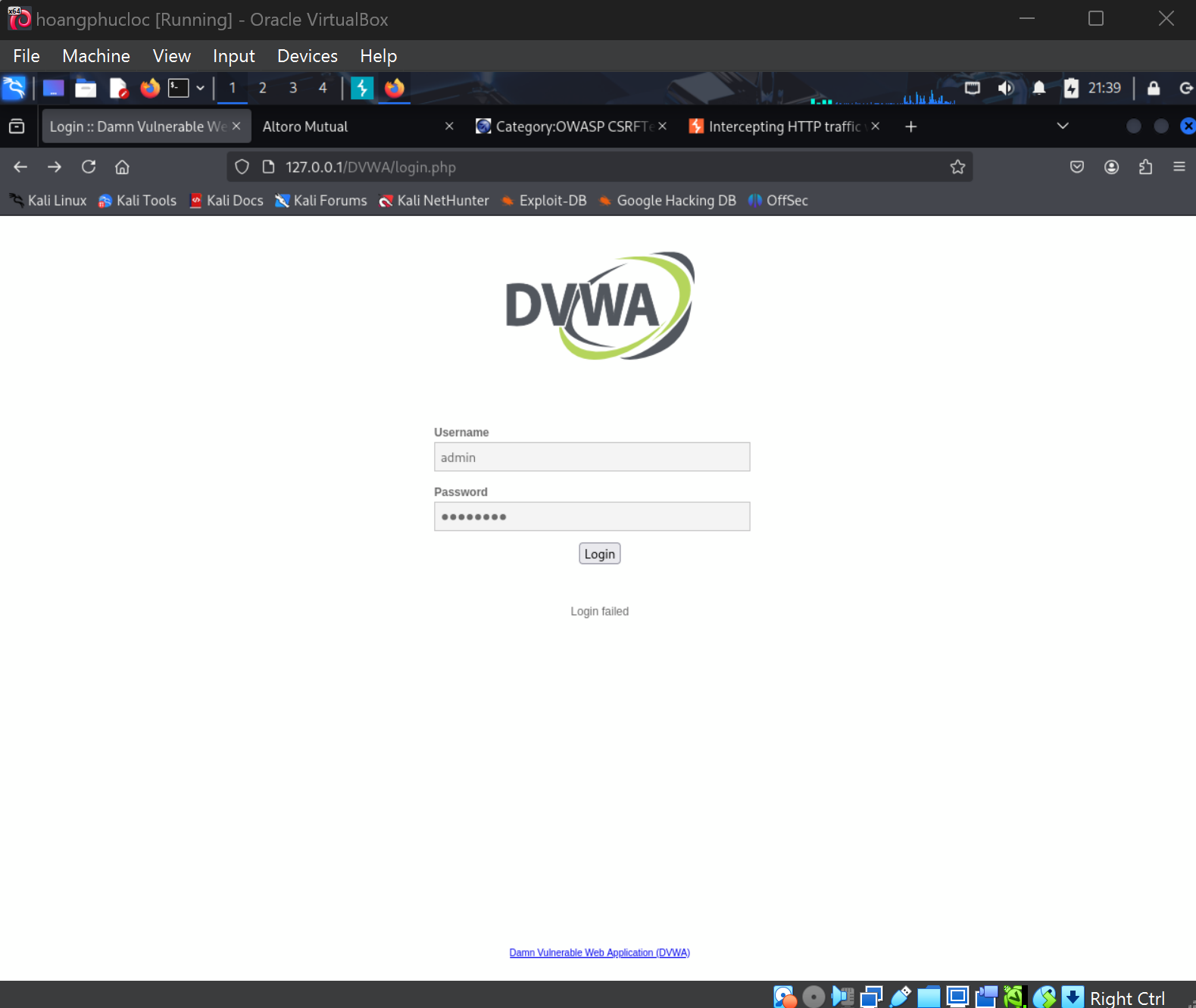
Điều này chứng minh rằng website dễ bị tấn công XSS thông qua tham số query.

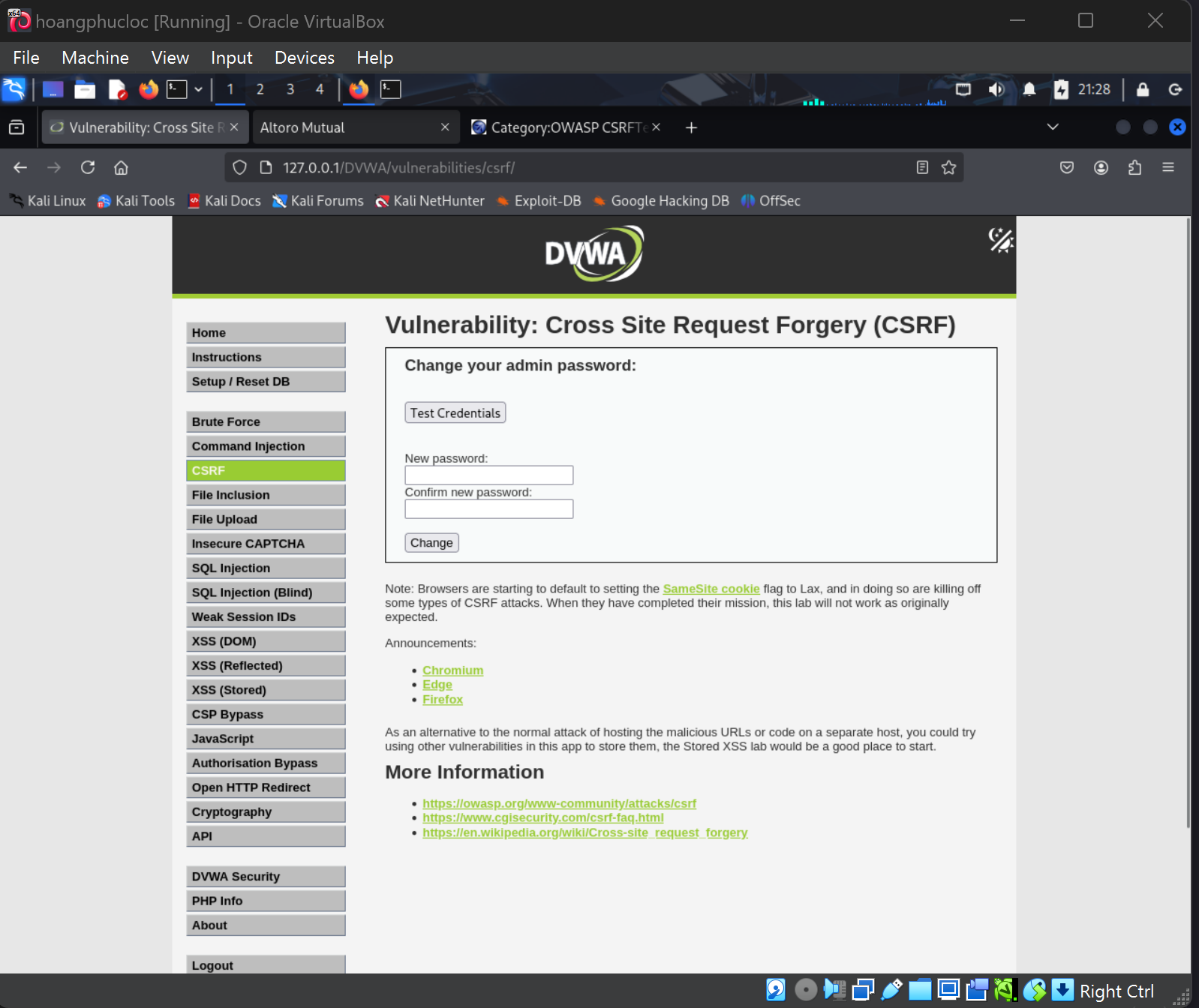
**Phần 2: Kiểm thử CSRF (2 điểm)**

Trình bày các bước thực hiện và kết quả kiểm thử lỗ hổng CSRF trên website chọn 1 trong các trang sau để thử nghiệm

**Bước 1: Chuẩn bị môi trường**

* Truy cập vào trang **DVWA** tại địa chỉ: http://127.0.0.1/DVWA/
* Đăng nhập tài khoản (mặc định):
  + Username: admin
  + Password: password
* Vào phần **"CSRF"** trong menu vulnerabilities.





**Bước 2: Phân tích request gửi lên server**

* Trên giao diện "Change your admin password", thấy 2 ô nhập:
  + New Password
  + Confirm New Password
* Khi submit, trình duyệt gửi **HTTP POST request**:

**Các tham số đầu vào:**

* password\_new: Mật khẩu mới.
* password\_conf: Xác nhận mật khẩu mới.

Bước 3: Tấn công CSRF

http://127.0.0.1/DVWA/vulnerabilities/csrf/ ?password\_new=123456&password\_conf=123456&Change=Change#

Bước 4: Thực hiện tấn công

Quan sát:

* Password bị đổi thành "123456" mà người dùng không hề biết.
* Không có CSRF token bảo vệ.

