|  |  |
| --- | --- |
| **BÁO CÁO**  ***PENETRATION TESTING***  Website Lê Văn Quân  **http://172.16.88.68:8099/huhu** |  |

**Phần 1: Kết quả đánh giá**

|  |  |
| --- | --- |
| **Thời gian đánh giá** | Tháng 08/2024 |
| **Kết quả chung** | Website đang có lỗ hổng liên quan tới bussiness logic, XSS, CSRF |
| **Đề xuất** | Khắc phục các lỗi được đưa ra trong báo cáo. Về dài hạn, cần triển khai đồng bộ các giải pháp ATTT nhằm phát hiện, cảnh báo các hành vi bất thường khi truy cập vào hệ thống, đồng thời triển khai việc theo dõi các lỗ hổng, nguy cơ mới để từ đó có phương án nâng cấp, giảm thiểu các rủi ro này. |
| **Số lượng lỗi nghiêm trọng** | 3 |
| **Số lượng lỗi cao** | 4 |
| **Số lượng lỗi trung bình** | 0 |
| **Số lượng lỗi thấp** | 0 |
| **Ghi chú (nếu có)** |  |

**Phần 2: Thông tin về các lỗ hổng**

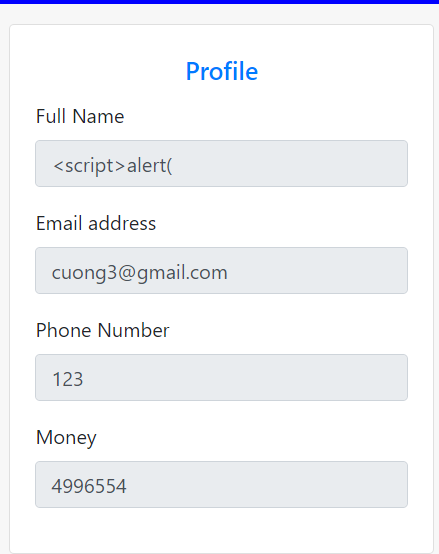
1. **Danh sách lỗ hổng**

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **Tên Lỗi** | **Mô tả** | **Mức độ ảnh hưởng** |
| 1 | Lỗ hổng có thể mua hàng cho người dùng khác | Request POST /huhu/orders hiện rõ thông tin như username, email, … Kẻ tấn công có thể lợi dụng bắt request này để thực hiện việc mua hàng cho người dùng bất kỳ, ảnh hưởng đến uy tín của cửa hàng | Cao |
| 2 | Lỗ hổng thêm sách đã hết hàng vào giỏ hàng | Khi thêm 1 quyển sách bất kỳ vào giỏ hàng, request GET /huhu/cart hiện ra thông tin bid của quyển sách đó, kẻ tấn công có thể thực hiện sửa bid đó thành bid của quyển sách đã hết hàng và mua nó. | Cao |
| 3 | Lỗ hổng XSS ở full-name | Trang web hiện thông tin tài khoản dưới dạng fullname mà chưa kiểm tra đầu vào, kẻ tấn công khi đăng ký với 1 tài khoản chứa đoạn script độc hại thì trang web sẽ bị lỗi XSS, kẻ tấn công có thể lợi dụng để đánh cắp phiên người dùng khác | Nghiêm trọng |
| 4 | Lỗi comment khi load lại trang | Mỗi khi load lại trang web, một bình luận mới sẽ được thêm trùng với bình luận người dùng vừa bình luận | Trung bình |
| 5 | Lỗ hổng CSRF ở các nút thêm vào giỏ hàng, xoá sản phẩm trong giỏ hàng | Request GET /huhu/cart, remove thoả mãn điều kiện 1 cuộc tấn công CSRF, kẻ tấn công có thể lợi dụng để thực hiện 1 cuộc tấn công CSRF, gây khó chịu cho người dùng nếu người dùng không may ấn vào link độc hại | Cao |

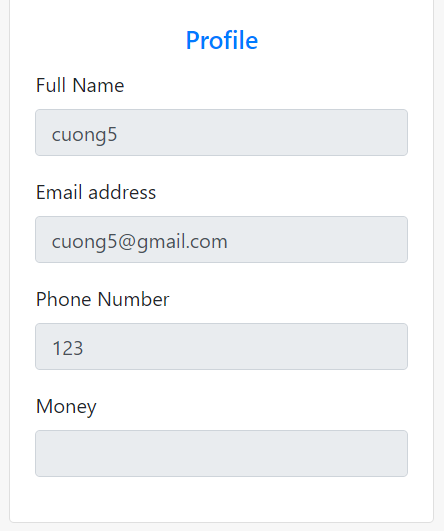
1. **Chi tiết lỗ hổng**
2. **Lỗ hổng có thể mua hàng cho người dùng khác**

* **Mức độ: Cao**
* **Mô tả:** Request POST /huhu/orders hiện rõ thông tin như username, email, … Kẻ tấn công có thể lợi dụng bắt request này để thực hiện việc mua hàng cho người dùng bất kỳ, ảnh hưởng đến uy tín của cửa hàng

Tài khoản của khách hàng đã nạp tiền

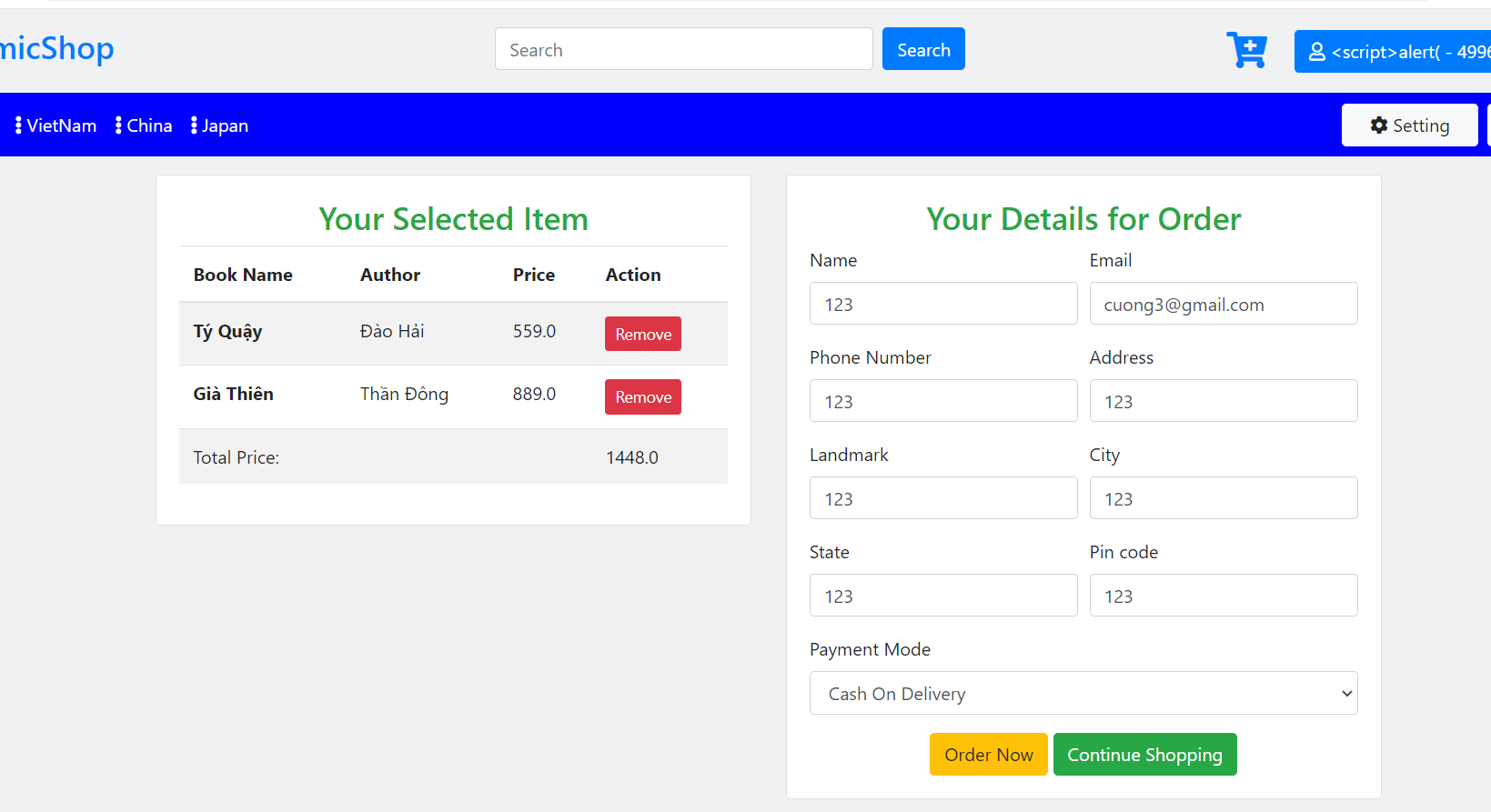


Hình 1. Tài khoản khách hàng có tiền



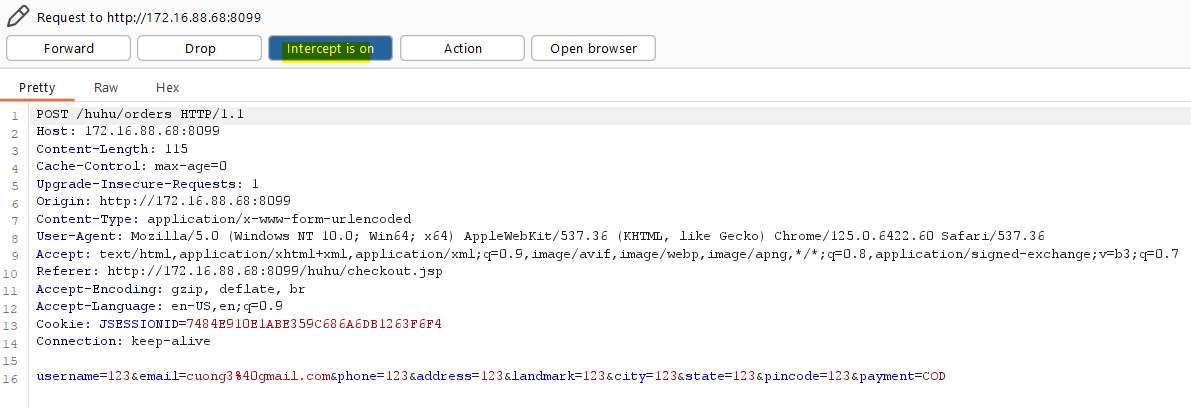
Hình 2.Tài khoản khách hàng không có tiền

Thêm vài cuốn sách vào giỏ hàng và bật “Intercept”



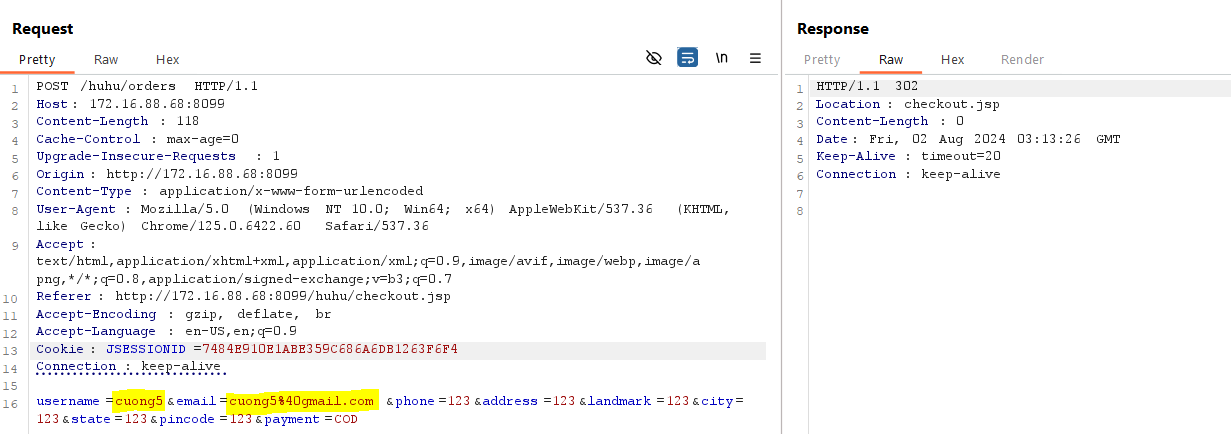
Hình 3. Giỏ hàng khách hàng có tiền

Bắt request nút “Order Now” và gửi sang Repeater để phân tích

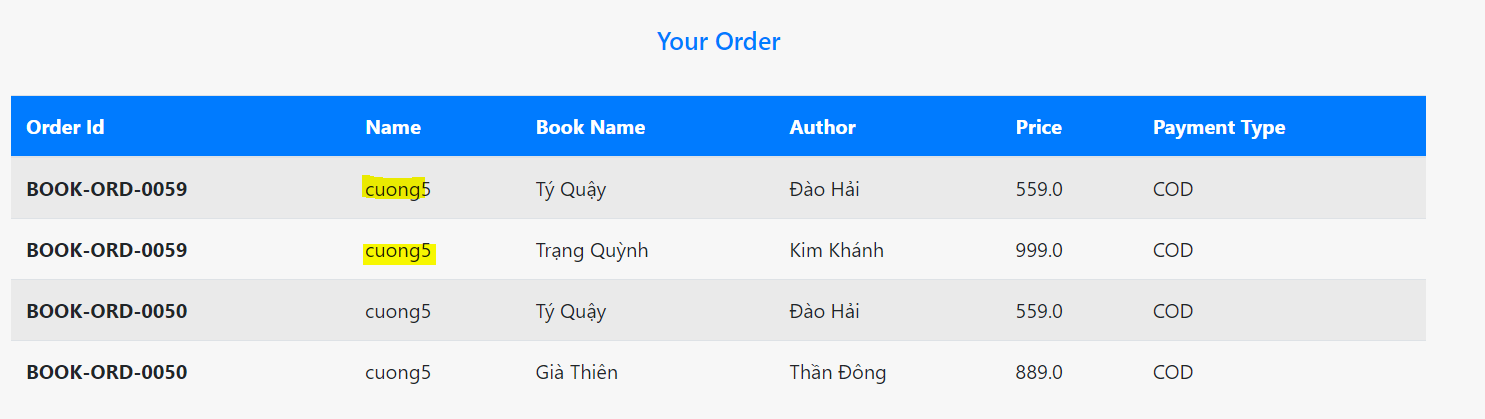


Hình 4. Request nút “Order Now”

Sửa username và email



Hình 5. Sửa username và email thành khách hàng không có tiền

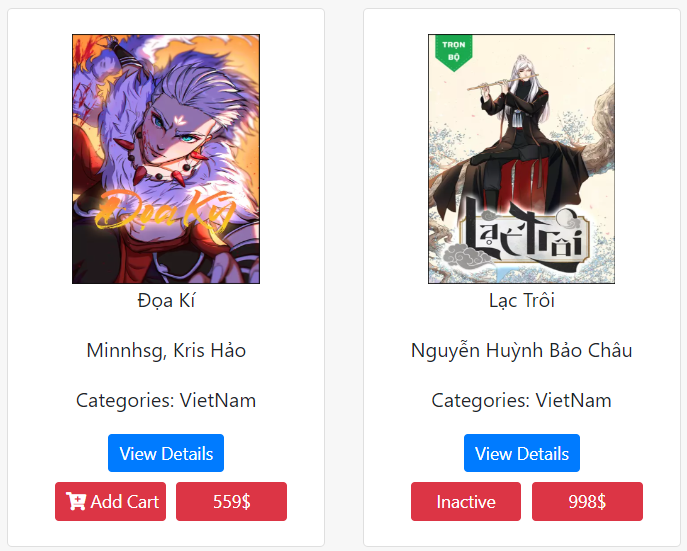


Hình 6. Đơn hàng đã được thêm thành công vào khách hàng cuong5

* **Khuyến nghị:**
* Ẩn hoặc mã hoá các thông tin nhạy cảm
* Sử dụng mã xác thực phiên, đảm bảo chỉ những người có phiên đăng nhập hợp lệ mới có thể thực hiện hành động đặt hàng
* Sử dụng mã CSRF token
* Sử dụng 2FA,captcha để đặt hàng

1. **Lỗ hổng thêm sách đã hết hàng vào trong giỏ hàng**

* **Mức độ: Cao**
* **Mô tả:** Khi thêm 1 quyển sách bất kỳ vào giỏ hàng, request GET /huhu/cart hiện ra thông tin bid của quyển sách đó, kẻ tấn công có thể thực hiện sửa bid đó thành bid của quyển sách đã hết hàng và mua nó.

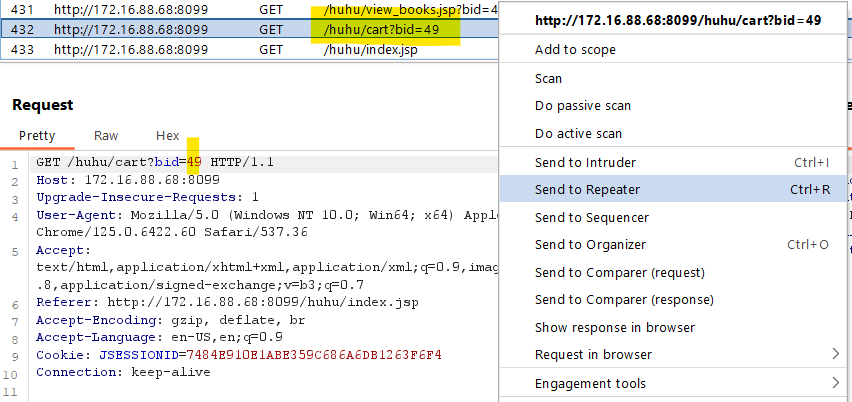


Hình 7. Sách Đoạ kí vẫn còn hàng và sách Lạc Trôi đã hết hàng

|  |  |
| --- | --- |
| Hình 8. Sách Đoạ kí có bid là 49 | Hình 9. Sách Lạc Trôi có bid là 48 |

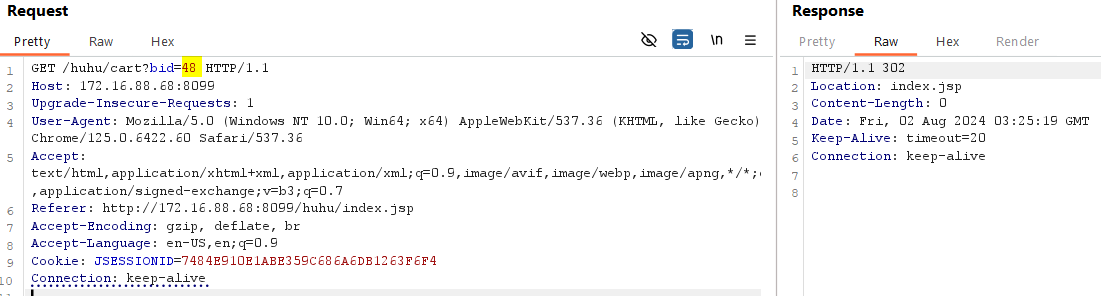
Thêm sách “Đoạ Kí” vào trong giỏ hàng

Bắt request và gửi sang Repeater



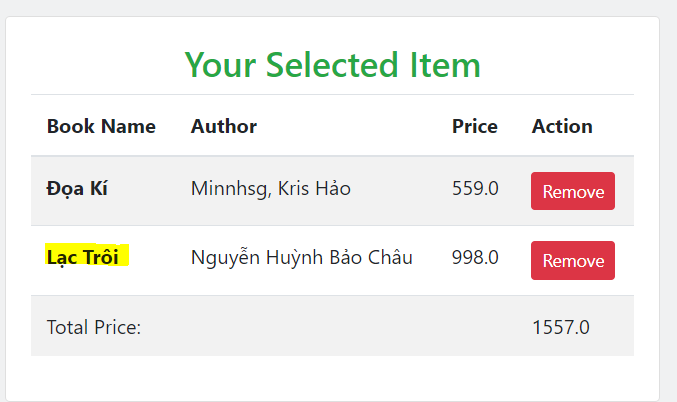
*Hình 10. Request thêm sách “Đoạ Kí” có bid là 49 vào giỏ hàng*

Sửa thành id của sách “Lạc Trôi”



*Hình 11. Sửa bid của request trên thành 48*

“Lạc Trôi” đã được thêm hàng giỏ hàng



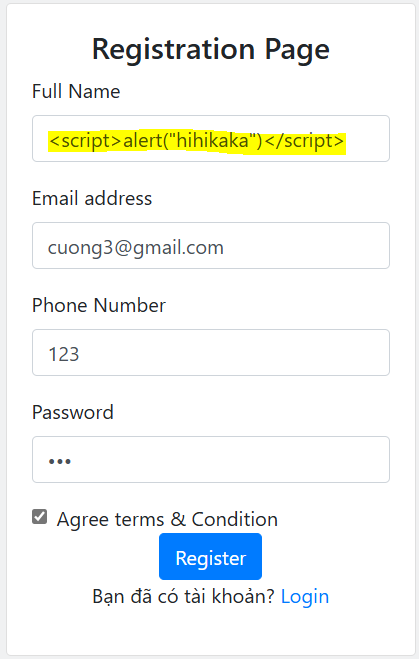
*Hình 12. Sách “Lạc Trôi” mặc dù đã hết hàng nhưng vẫn có thể thêm vào giỏ hàng*

* **Khuyến nghị:**
  + Sử dụng id không đoán được chẳng hạn như UUID hoặc mã hash.
  + Ẩn thông tin nhạy cảm bid

1. **Lỗ hổng XSS ở full-name**

* **Mức độ: Nghiêm trọng**
* **Mô tả:** Trang web hiện thông tin tài khoản dưới dạng fullname mà chưa kiểm tra đầu vào, kẻ tấn công khi đăng ký với 1 tài khoản chứa đoạn script độc hại thì trang web sẽ bị lỗi XSS, kẻ tấn công có thể lợi dụng để đánh cắp phiên người dùng khác.

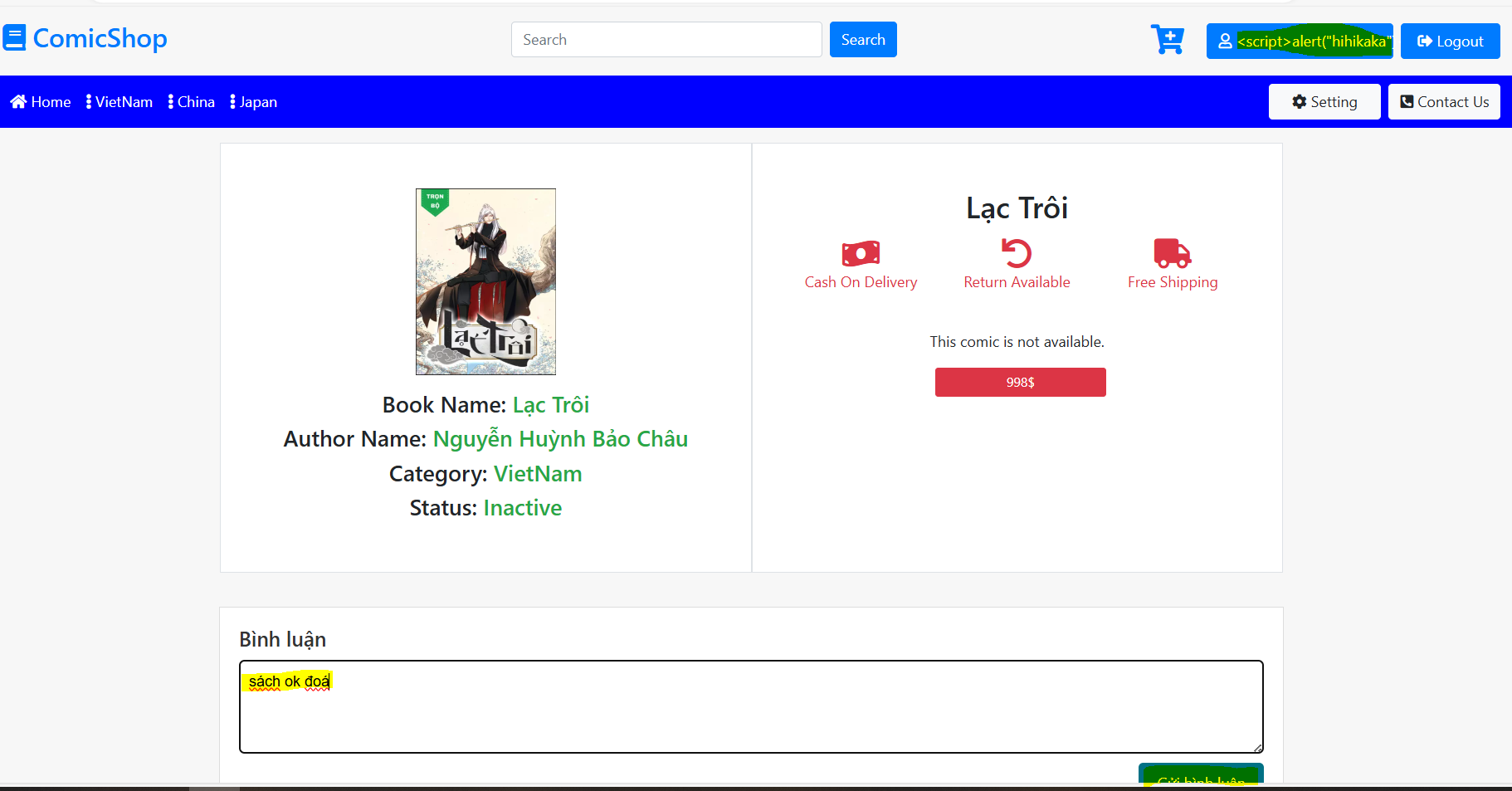
Đăng ký 1 tài khoản có full name dạng <script> …</script>



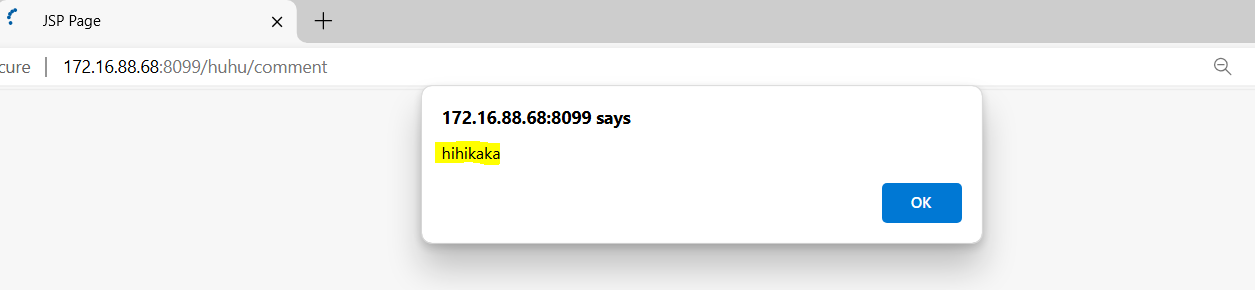
Hình 13. Đăng ký tài khoản



Hình 14. Login với tài khoản vừa đăng ký



Hình 15. Bình luận với đoạn text bất kỳ



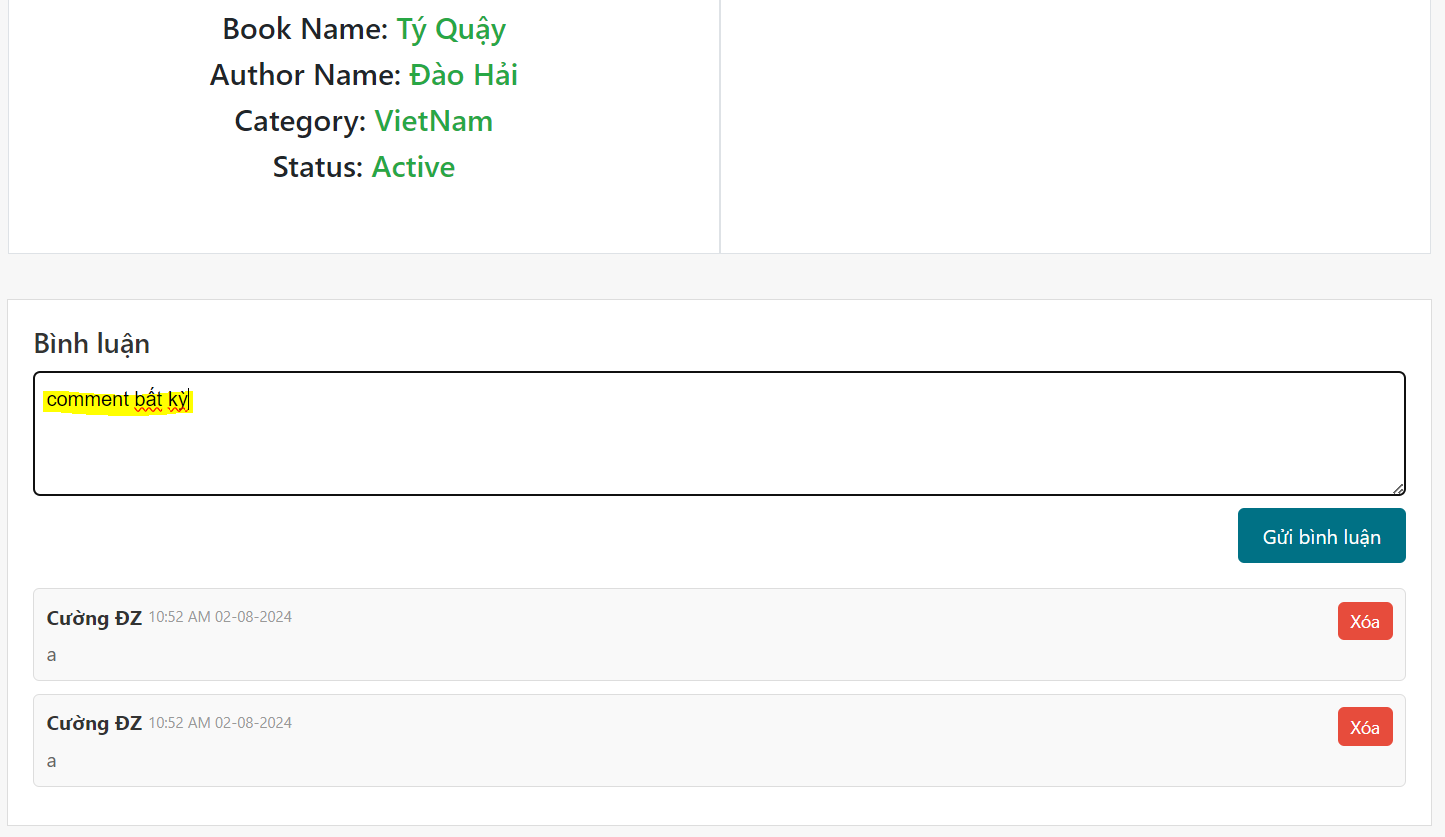
Hình 16. Trang web bị XSS

* **Khuyến nghị:**
  + Sửa phần bình luận không để fullname nữa mà để thành gmail
  + Xử lý đầu vào và đầu ra.
  + Thiết lập CSP.
  + Sử dụng HTTPOnly và Secure Cookies.

1. **Lỗi comment khi load lại trang**

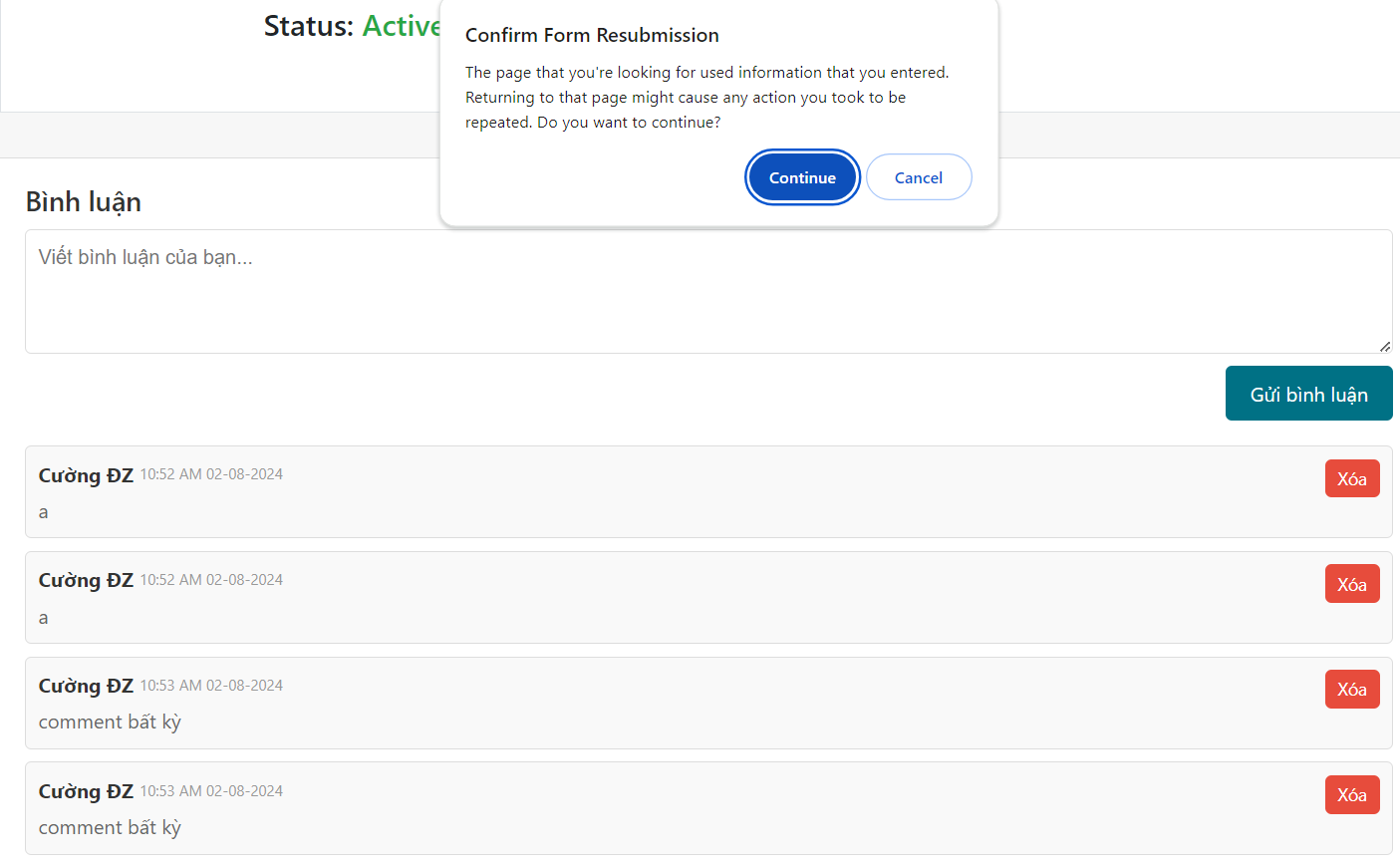
* **Mức độ: Trung bình**
* **Mô tả:** Mỗi khi load lại trang web, một bình luận mới sẽ được thêm trùng với bình luận người dùng vừa bình luận.

Gửi comment bất kỳ



Hình 17. Thực hiện gửi bình luận “comment bất kỳ”

Sau khi load lại trang web thì 1 comment tương tự được tạo ra

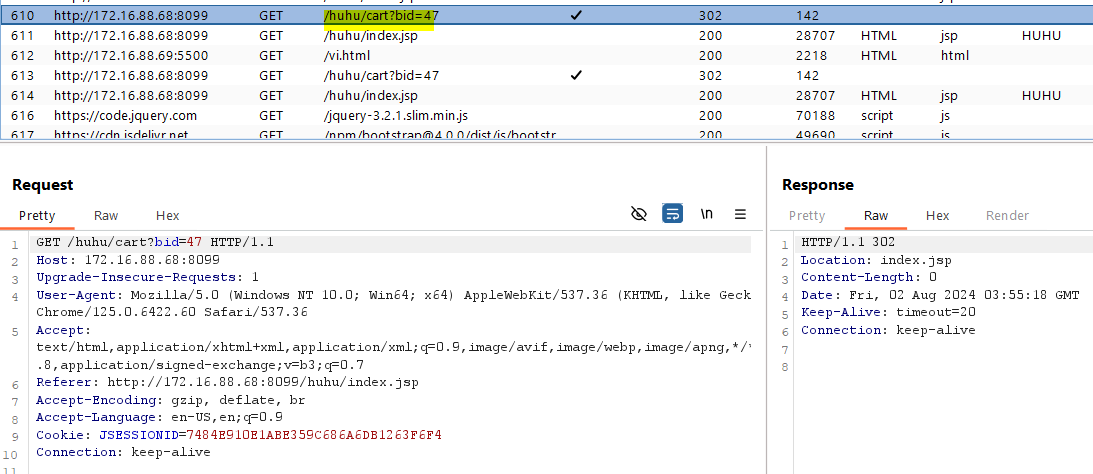


Hình 18. Khi F5 thì bình luận mới cũng là “comment bất kỳ” tự động thêm vào

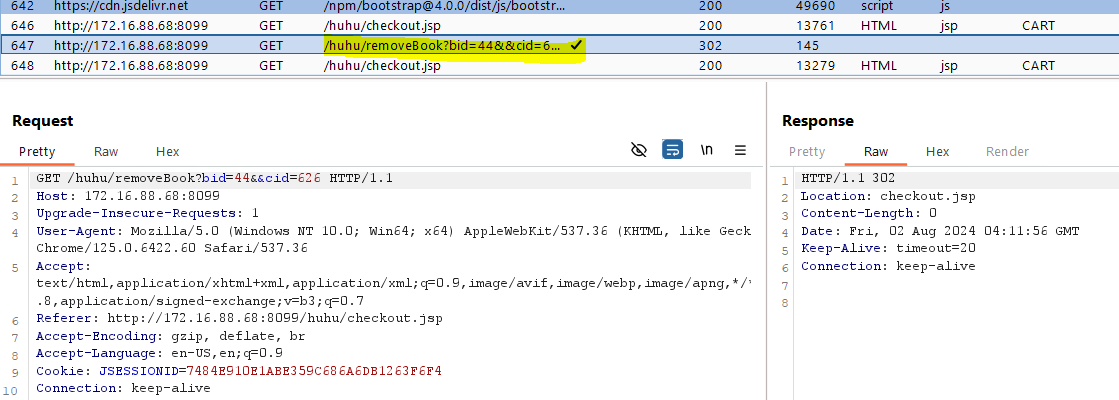
* **Khuyến nghị:**
  + Kiểm tra mã xử lý bình luận.
  + Kiểm tra cơ sở dữ liệu.

1. **Lỗ hổng CSRF ở các nút thêm vào giỏ hàng, xoá sản phẩm trong giỏ hàng**

* **Mức độ: Cao**
* **Mô tả:** Request GET /huhu/cart, remove thoả mãn điều kiện 1 cuộc tấn công CSRF, kẻ tấn công có thể lợi dụng để thực hiện 1 cuộc tấn công CSRF, gây khó chịu cho người dùng nếu người dùng không may ấn vào link độc hại



*Hình 19. Request GET /huhu/cart?bid=47, thêm sản phẩm có mã 47 vào trong giỏ hàng*



*Hình 20. Request nút xoá sản phẩm trong giỏ hàng*

Tạo đoạn mã CSRF để thực hiện tự động thêm giỏ hàng sách có id = 47

<html>

  <!-- CSRF PoC - generated by Burp Suite Professional -->

  <body>

    <form action="http://172.16.88.68:8099/huhu/cart">

      <input type="hidden" name="bid" value="47" />

      <input type="submit" value="Submit request" />

    </form>

    <script>

      history.pushState('', '', '/');

      document.forms[0].submit();

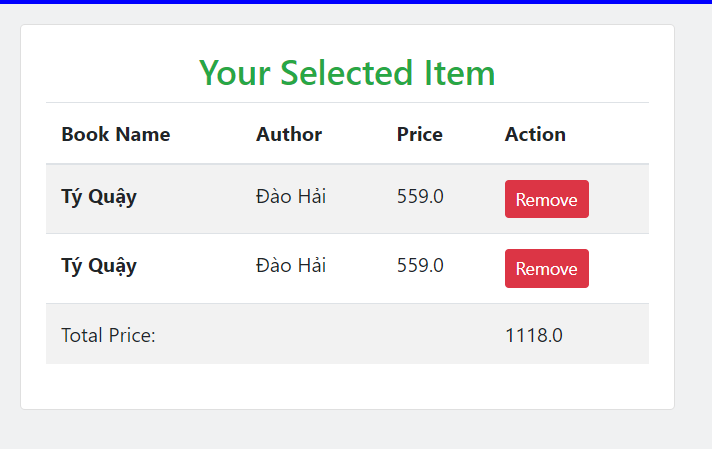
    </script>

  </body>

</html>

*Hình 21. Tạo yêu cầu độc hại*

Sau đó lừa nạn nhân vào link: <http://172.16.88.69:5500/vi.html>, sách tự động thêm vào giỏ hàng



*Hình 21. Khi người dùng không may click vào link độc hại, sản phẩm tự động thêm vào giỏ hàng*

* **Khuyến nghị:**
  + Sử dụng CSRF Token.
  + Sử Dụng HTTP Headers.
  + Yêu cầu xác thực lại
  + Sử dụng 2FA, captcha.