**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ**

**THÔNG TIN 2**



**ĐỒ ÁN MÔN HỌC**

**AN TOÀN ỨNG DỤNG WEB VÀ CSDL**

**ĐỀ TÀI : BẢO MẬT TRANG WEB BÁN HÀNG ĐIỆN**

**THOẠI DI ĐỘNG**

**Giáo viên hướng dẫn : ThS. PHAN NGHĨA HIỆP**

**Nhóm SV thực hiện : N20DCAT016 – Đỗ Diệu Hiền**

**: N20DCAT014 – Ngô Đức Tuấn**

**: N20DCAT052 – Phan Xuân Tấn**

**: N20DCAT064 – Trần Quang Việt**

**: N20DCAT025 – Trần Trung Kiên**

***Thành phố Hồ Chí Minh, ngày 21 tháng 12 năm 2023***

**VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

**CƠ SỞ THÀNH PHỐ HỒ CHÍ MINH**

**KHOA CÔNG NGHỆ**

**THÔNG TIN 2**



**ĐỒ ÁN MÔN HỌC**

**AN TOÀN ỨNG DỤNG WEB VÀ CSDL**

**ĐỀ TÀI : BẢO MẬT TRANG WEB BÁN HÀNG ĐIỆN**

**THOẠI DI ĐỘNG**

**Giáo viên hướng dẫn : ThS. PHAN NGHĨA HIỆP**

**Nhóm SV thực hiện : N20DCAT016 – Đỗ Diệu Hiền**

**: N20DCAT054 – Ngô Đức Tuấn**

**: N20DCAT052 – Phan Xuân Tấn**

**: N20DCAT064 – Trần Quang Việt**

**: N20DCAT025 – Trần Trung Kiên**

***Thành phố Hồ Chí Minh, ngày 21 tháng 12 năm 2023***

ĐÁNH GIÁ NHẬN XÉT CỦA GIẢNG VIÊN

- Điểm số:

- Điểm chứ:

Thành phố Hồ Chí Minh, ngày….tháng…..năm 2023

Giảng viên

**LỜI CẢM ƠN**

Báo cáo đồ án: “bảo mật trang web bán điện thoại di động” với đề tài này là là kết quả của quá trình cố gắng không ngừng của nhóm chúng em và được sự giúp đỡ, động viên khích lệ, cùng với những tài liệu hữu ích của thầy Phan Nghĩa Hiệp. Qua trang viết này nhóm em xin gửi lời cảm ơn chân thành đến thầy người luôn hỗ trợ chúng em trong suốt thời gian vừa qua.

Người thực hiện

**Mục lục**

[**DANH MỤC HÌNH** 6](#_Toc154044783)

[**LỜI GIỚI THIỆU** 7](#_Toc154044784)

[**1.** **Chương 1: Cơ sở lý thuyết** 8](#_Toc154044785)

[**1.1** **Cơ sở lý thuyết thiết kế hệ thống web** 8](#_Toc154044786)

[**1.1.1** **Giao diện: HTML & CSS** 8](#_Toc154044787)

[**1.1.2** **Backend: Java(MVC)** 8](#_Toc154044788)

[**1.2** **Cơ sở lý thuyết và các lỗi bảo mật và cách phòng chống** 9](#_Toc154044789)

[**1.2.1** **XSS** 9](#_Toc154044790)

[**1.2.2** **SQL Injection** 12](#_Toc154044791)

[**1.2.3** **Mã hóa base64** 13](#_Toc154044792)

[**1.2.4** **Mã hóa Bcrypt** 14](#_Toc154044793)

[**1.2.5** **Brute Force Attack** 15](#_Toc154044794)

[**1.2.6** **CSRF** 16](#_Toc154044795)

[**1.2.7** **SQL Injection in the DBMS (Lỗi chèn mã SQL)** 17](#_Toc154044796)

[**1.2.9** **Sao lưu và backup dữ liệu** 21](#_Toc154044798)

[**1.2.10** **Default and Weak Passwords** 23](#_Toc154044799)

[**1.2.11** **Excessive User & Group Privileges** 23](#_Toc154044800)

[**1.2.12** **Privilege Escalation** 23](#_Toc154044801)

[**1.2.13** **Logic Attacks** 24](#_Toc154044802)

[**2.** **Chương 2: Chương trình DEMO và thực nghiệm** 26](#_Toc154044803)

[**3.** **Chương 3: Tổng kết** 28](#_Toc154044804)

[**3.1** **Các lỗi đã khắc phục** 28](#_Toc154044805)

[**3.2 Các lỗi còn chưa khắc phục được** 29](#_Toc154044806)

[**TÀI LIỆU THAM KHẢO** 29](#_Toc154044809)

[BẢNG PHÂN CÔNG NHIỆM VỤ 30](#_Toc154044810)

# **DANH MỤC HÌNH**

* *Hình 1:Mô hình mvc*
* *Hình 2: Mô hình tấn công xss*
* *Hình 3: Mô hình tấn công sql*
* *Hình 4: Mô hình tấn công csrf*
* *Hình 5: Mô hình mã hóa base 64*
* *Hình 6: Mô hình mã hóa*
* *Hình 7: Mô hình tấn công Brute Force Attack*
* *Hình 8: Mô hình tấn công ddos*
* *Hình 9:Backup sao lưu dữ liệu*
* *Hình 10: Backup sao lưu dữ liệu*
* *Hình 11: Backup sao lưu dữ liệu*
* *Hình 12:Giao diện trang web bán hàng*
* *Hình 13:Mô hình tấn công bằng csrf*

# **LỜI GIỚI THIỆU**

Ngày nay cùng với sự phát triển ngày càng đi lên của đất nước là sự bùng nổ của công nghệ thông tin, công nghệ ngày càng phát triển thì sự phức tạp của các trang web hiện đại ngày càng tăng, đi theo đó là những mối nguy hại tìm ẩn vô cùng to lớn, các cuộc tấn công mạng ngày càng diễn ra nhiều hơn, ảnh hưởng nặng nề về tài sản, rò rỉ thông tin quan trọng, thiệt hại cho nhiều cá nhân và các doanh nghiệp lớn. Với sự phát triển không ngừng của công nghệ, những cuộc tấn công mạng thông qua internet cũng ngày càng trở nên đa dạng hơn một số các loại tấn công như: malware – tấn công bằng phần miềm độc hại, tấn công giả mạo, tấn công trung gian, khai thác lỗ hổng…. Và không thể không nói đến đó là các tấn công lên website như: từ chối dịch vụ phân tán (DDOS), tấn công bằng mã độc (XSS), SQL Injection. Tấn công DDOS có nhiều hình thức tấn công như: tấn công gây nghẽn mạng, tấn công SYN flood, tấn công khếch đại DNS,… Mục tiêu của các hình thức tấn công này gây quá tải hệ thống mạng bằng lượng truy cập lớn, làm cạn tài nguyên của máy chủ, ngăn chặn việc nhận các yêu cầu kết nối mới, làm quá tải hệ thống bằng các bộ giải mã DNS.

Nhận thấy rằng đây là một trong những vấn đề hết sức nghiêm trọng ảnh hưởng xâm phạm đến các thông tin của các cá nhân, doanh nghệp lớn và hơn hết là sự mong muốn học hỏi của các thành viên trong nhóm cũng như mong muốn đóng góp ý kiến được kết hợp từ lý thuyết và thực tiễn đã được học trong trường,

Nhóm em đã chọn đề tài: “Phòng chống tấn công cho website bán điện thoại di động”. Trong phạm vi của báo cáo, chúng em sẽ trình bày những vấn đề sau:

Chương 1: Cơ sở lý thuyết của đề tài.

Chương 2: Demo và thực nghiệm.

Chương 3: Kết quả thực nghiệm.

Trên cơ sở đó, báo cáo của nhóm được hoàn thiện với sự nỗ lực của các thành viên trong nhóm. Mặc dù vậy, chắc chắn sẽ không tránh khỏi những thiếu sót, nhóm em mong nhận được những lời nhận xét, chia sẻ và góp ý đến từ các thầy cô và các bạn để báo cáo của nhóm em được hoàn chỉnh hơn.

1. **Chương 1: Cơ sở lý thuyết**
   1. **Cơ sở lý thuyết thiết kế hệ thống web**
      1. **Giao diện: HTML & CSS**

HTML viết tắt Hyper Text Markup Language (hay ngôn ngữ đánh dấu siêu văn bản) được thiết kế ra để tạo nên các trang web.  HTML mô tả cấu trúc của một trang web về mặt ngữ nghĩa và các dấu hiệu ban đầu được bao gồm cho sự xuất hiện của tài liệu.

Cùng với CSS và JavaScript tạo nên bộ 3 nền tảng cho World Wide Web.

Cú pháp html được cấu tạo bởi các thẻ có đóng mở, như cặp thẻ <div></div>, thẻ <h1></h1>, …và những thẻ không cần thẻ đóng như thẻ <img > thẻ <input

* + 1. **Backend: Java(MVC)**

JAVA là một trong những ngôn ngữ lập trình hướng đối tượng. Nó được sử dụng trong phát triển phần mềm, trang web, game hay ứng dụng trên các thiết bị di động.. Java được khởi đầu bởi James Gosling và bạn đồng nghiệp ở Sun MicroSystem năm 1991. Ban đầu Java được tạo ra nhằm mục đích viết phần mềm cho các sản phẩm gia dụng, và có tên là Oak. Java được phát hành năm 1994, đến năm 2010 được Oracle mua lại từ Sun MicroSystem.

Java được tạo ra với tiêu chí “Viết (code) một lần, thực thi khắp nơi” (Write Once, Run Anywhere – WORA). Chương trình phần mềm viết bằng Java có thể chạy trên mọi nền tảng (platform) khác nhau thông qua một môi trường thực thi với điều kiện có môi trường thực thi thích hợp hỗ trợ nền tảng đó.

Mô hình MVC:

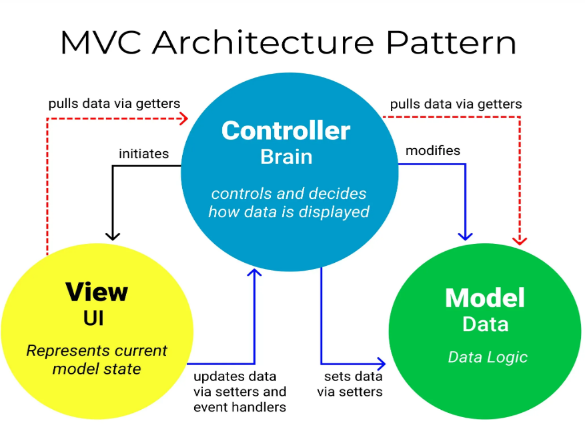
MVC được viết tắt bởi 3 từ: Model – View – Controller, là một mẫu thiết kế nhằm mục tiêu chia tách phần giao diện và code để dễ quản lý, phát triển và bảo trì.

Mô hình MVC chia một ứng dụng thành ba phần tương tác được với nhau, mỗi thành phần có một nhiệm vụ riêng biệt và độc lập với các thành phần khác.

Model: quản lý, xử lý dữ liệu

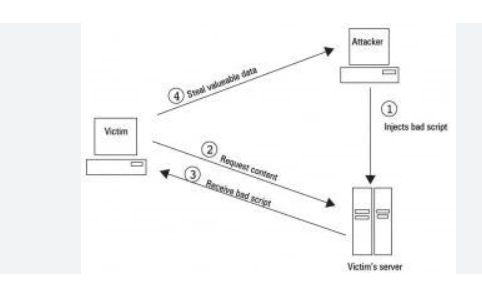
View: giao diện, hiển thị dữ liệu cho người dùng

Controller: điều khiển sự tương tác giữa Model và View

*Mô hình MVC* (1)

* 1. **Cơ sở lý thuyết và các lỗi bảo mật và cách phòng chống**
     1. **XSS**
        1. **Khái niệm**

XSS là tên viết tắt của Cross-site scripting. Đây là một hình thức tấn công bằng mã độc phổ biến. Các hacker sẽ lợi dụng lỗ hổng trong bảo mật web để chèn các mã script, sau đó gửi cho người dùng để truy cập và mạo danh người dùng.



Hình 2

Mục đích của việc này chính là ăn cắp dữ liệu nhận dạng của người dùng như session tokens, cookies và các thông tin khác. Khi đăng nhập vào được các tài khoản website, hacker có thể truy cập vào bất cứ dữ liệu nào và toàn quyền kiểm soát tất cả các chức năng và dữ liệu của ứng dụng. (Nguyễn Hưng, 2022)

Tấn công XSS là hình thức tấn công đơn giản nhưng đặc biệt nguy hiểm. Đây cũng là kỹ thuật được hacker sử dụng phổ biến nhất trong thời điểm hiện nay.

* + - 1. **Cách hoạt động của XSS**

Tấn công XSS nghĩa là gửi chèn lệnh và script độc hại, những mã đôc hại này thường được viết bằng các ngôn ngữ lập trình phía client như JavaScript, VBScript, Flash, HTML,… Tuy nhiên, cách tấn công này thường sử dụng JavaScript và HTML.

XSS hoạt động bằng cách điều khiển một trang web dễ bị tấn công để trả về JavaScript độc hại cho người dùng. Khi mã độc thực thi bên trong trình duyệt, hacker hoàn toàn có thể xâm nhập vào tài khoản và sử dụng dữ liệu của người dùng. (Nguyễn Hưng, 2022)

* + - 1. **Các loại tấn công XSS**

Có ba loại tấn công XSS chính là: Reflected XSS: Là hình thức tấn công được sử dụng nhiều nhất. Đây là nơi mã script độc hại đến từ HTTP request. Từ đó, hacker đánh cắp dữ liệu của người dùng, chiếm quyền truy cập và hoạt động của họ trên website thông qua việc chia sẻ URL chứa mã độc. Hình thức này thường nhắm đến ít nạn nhân.

Stored XSS: Khác với Reflected XSS, hình thức này nhắm đến nhiều nạn nhân cùng lúc. Đây là nơi script độc hại đến từ cơ sở dữ liệu của trang web. Hacker chèn các mã độc vào cơ sở dữ liệu thông qua các dữ liệu đầu vào như input, form,… Khi người dùng truy cập và tiến hành những thao tác liên quan đến dữ liệu đã lưu thì mã độc sẽ lập tức hoạt động trên trình duyệt.

DOM-based XSS: Là nơi lỗ hổng bảo mật tồn tại trong mã phía client chứ không phải mã phía server. Hình thức này dùng để khai thác XSS dựa vào việc thay đổi HTML của tài liệu, hay nói cách khác là thay đổi cấu trúc DOM. (Nguyễn Hưng, 2022)

* + - 1. **Tác hại của tấn công XSS**

Hacker khai thác lỗ hổng XSS trên nhiều trang web thường có thể:

• Mạo danh hoặc giả dạng người dùng của nạn nhân.

• Thực hiện bất kỳ hành động nào mà người dùng có thể thực hiện.

• Đọc bất kỳ dữ liệu nào mà người dùng có thể truy cập.

• Nắm bắt thông tin đăng nhập của người dùng.

• Thực hiện thay đổi bề mặt ảo của trang web.

• Chèn trojan độc hại vào trang web.

* + - 1. **Cách kiểm tra tấn công XSS**

Để kiểm thử tấn công XSS, kiểm thử hộp đen có thể được thực hiện. Điều này có nghĩa là chúng ta có thể test mà không cần xem xét code. Tuy nhiên, xem lại code luôn là một việc luôn được khuyến khích nên làm và cũng mang lại kết quả đáng tin cậy hơn. Khi bắt đầu kiểm tra, người kiểm tra nên xem xét phần nào của trang web dễ bị tấn công XSS. Nên liệt kê trong bất kỳ tài liệu thử nghiệm nào để chắc chắn không có gì bị bỏ sót. Sau đó, người kiểm tra nên lập kế hoạch cho những trường nhập code hoặc script cần được kiểm tra. Kết quả kiểm tra sẽ cho chúng ta biết ứng dụng đó dễ bị tấn công hay không. Để kiểm tra khả năng bị tấn công, cần phải kiểm tra xem các script đã nhập đang được phản hồi như thế nào và những script đó có được thực thi hay không,… Ví dụ: Người kiểm tra có thể cố nhập script . Nếu script này đang được thực thi, thì có khả năng rất lớn là XSS có thể xảy ra. Ngoài ra, trong khi kiểm tra theo cách thủ công về khả năng tấn công Cross Site Scripting, cần nhớ là cũng nên thử các dấu ngoặc được mã hóa. Ví dụ: người kiểm tra có thể cố gắng nhập tập lệnh trình duyệt như: %3cscript%3ealert(document.cookie)%3c/script%3e

Một số người cố gắng bảo vệ các trang web và hệ thống khỏi các cuộc tấn công khác nhau bằng cách thay đổi dấu ngoặc thành hai dấu ngoặc.

Tiêu đề test

. Nếu lỗ hổng bảo mật này xuất hiện trong ứng dụng web, một văn bản được chỉ định sẽ được chèn vào thẻ

. Cố gắng chuyển một số mã thông qua yêu cầu HTTP vì đây cũng là một phương pháp để kiểm tra xem liệu cuộc tấn công này có khả thi hay không. (Nguyễn Hưng, 2022)

* + - 1. **Cách ngăn chặn tấn công XSS**

Kiểu tấn công này là một trong những kiểu tấn công có nhiều nguy hiểm và rủi ro nhất. Tuy nhiên, vẫn có rất nhiều cách để ngăn chặn tấn công này. Phương pháp ngăn chặn tấn công XSS bao gồm:

• Data validation.

• Filtering.

• Escaping.

Bước đầu tiên trong việc ngăn chặn cuộc tấn công này là Input validation (Xác thực đầu vào). Mọi thứ do người dùng nhập phải được xác thực chính xác vì thông tin input của người dùng có thể tìm đường đến output. Data validation (Xác thực dữ liệu) có thể được đặt tên là cơ sở để đảm bảo tính bảo mật của hệ thống.

➔ Việc này chỉ giúp giảm thiểu rủi ro, và có thể không đủ để ngăn chặn tấn công XSS có thể xảy ra. Một phương pháp ngăn chặn tốt hơn là lọc input filtering (thông tin đầu vào) của người dùng. Ý tưởng của bộ lọc là tìm kiếm các từ khóa gây rủi ro trong thông tin input của người dùng và xóa chúng hoặc thay thế chúng bằng các chuỗi trống. Những từ khóa đó có thể là:

1) tags.

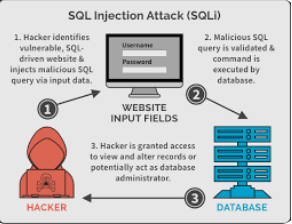
2) Javascript commands.

3) HTML markup.

Một phương pháp ngăn chặn có thể thực hiện là phương pháp characters escaping. Trong phương pháp này, các ký tự thích hợp đang được thay đổi bằng các code đặc biệt. Ví dụ: Ký tự escaping có thể trông giống như <. (Nguyễn Hưng, 2022)

* + 1. **SQL Injection**
       1. **Khái niệm về SQL và SQL Injection**

SQL được viết tắt từ Structured Query Language, là ngôn ngữ truy vấn có cấu trúc, xây dựng chế độ xem dữ liệu có thể tùy chỉnh cho mỗi người dùng. Ngôn ngữ truy vấn có cấu trúc này được sử dụng để thực thi các lệnh, chẳng hạn như truy xuất dữ liệu, cập nhật và xóa bản ghi. SQL Injection là một kỹ thuật lợi dụng những lỗ hổng về câu truy vấn của các ứng dụng.



Hình 3

Được thực hiện bằng cách chèn thêm một đoạn SQL để làm sai lệnh đi câu truy vấn ban đầu, từ đó có thể khai thác dữ liệu từ database. SQL injection có thể cho phép những kẻ tấn công thực hiện các thao tác như một người quản trị web, trên cơ sở dữ liệu của ứng dụng. (topdev, n.d.)

Các phần dễ bị tấn công bao gồm:

* + - * + Form đăng nhập
        + Form tìm kiếm
        + Form nhận xét
        + Bất kì trường lưu hoặc trường đầu vào của dữ liệu
        + Liên kết của website
      1. **Tác hại của SQL Injection**

Việc tấn công SQL Injection mang lại hệ quả vô cùng lớn, có thể kể tới như:

• Lấy cắp thông tin trong tài khoản cá nhân

• Sao chép dữ liệu web hoặc hệ thống quản trị

• Thay đổi dữ liệu được gửi tới người dùng web

• Thay đổi quyền quản trị viên, chiếm hoàn toàn hệ thống điều hành

• Sửa đổi cấu trúc cơ sở dữ liệu

* + - 1. **Phương pháp phát hiện, ngăn chặn tấn công SQL Injection**

• Sử dụng các thủ tục lưu trữ, Xác thực dữ liệu đầu vào của người dùng.

• Ẩn thông tin của các thông báo lỗi.

• Hạn chế đặc quyền

• Phát hiện và ngăn chặn dựa trên chữ ký “signature”.

• Phát hiện và ngăn chặn dựa trên bất thường.

• Phân tích mã.

* + 1. **Mã hóa base64**
* Base64 là một sơ đồ binary-to-text(nhị phân thành văn bản). Nó biểu diễn dữ liệu nhị phân ở định dạng chuỗi ASCII có thể in được bằng cách dịch nó thành biểu diễn cơ số 64.
* encode Base64 thường được sử dụng khi có nhu cầu truyền dữ liệu nhị phân qua phương tiện không xử lý chính xác dữ liệu nhị phân và được thiết kế để xử lý dữ liệu văn bản chỉ thuộc bộ mã US-ASCII 7-bit.



Hình 5

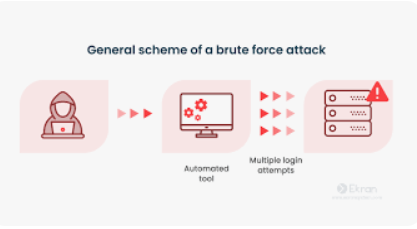
Một số mục đích sử dụng Base64 có thể bao gồm:

* Mã hóa dữ liệu truyền: Base64 có thể sử dụng để mã hóa dữ liệu truyền trong các giao thức mạng như email hoặc HTTP, tránh việc dữ liệu bị trộn với giao thức.
* Lưu trữ dữ liệu: Base64 có thể sử dụng để lưu trữ dữ liệu bất kỳ như hình ảnh, tập tin âm thanh hoặc video dưới dạng chuỗi văn bản.
* Xác thực dữ liệu: Base64 có thể sử dụng để tạo chữ ký số cho các tập tin để xác thực dữ liệu gửi hoặc nhận.
* Mã hóa tài khoản và mật khẩu: Base64 có thể sử dụng để mã hóa tài khoản và mật khẩu trong các hệ thống đăng nhập.
  + 1. **Mã hóa Bcrypt**
* Bcrypt là một thuật toán mã hóa mật khẩu được sử dụng để bảo vệ mật khẩu người dùng trong các hệ thống thông tin và ứng dụng. Nó được thiết kế để chống lại các cuộc tấn công dò mật khẩu thông qua việc sử dụng kỹ thuật “chậm” và “đắt đỏ” (slow and expensive). Bcrypt sử dụng hàm băm (hash function) có thời gian tính toán chậm để mã hóa mật khẩu. Quá trình mã hóa này được thực hiện thông qua việc lặp đi lặp lại một số lần (số lần lặp lại được gọi là “factors” hoặc “cost”). Bằng cách thực hiện nhiều lần lặp, bcrypt tạo ra một chuỗi mã băm dài và khó đoán, gây khó khăn cho kẻ tấn công khi tìm ra mật khẩu ban đầu từ mã băm.



Hình 6

* Bcrypt cung cấp một phương thức đơn giản để sử dụng trong các ngôn ngữ lập trình khác nhau. Thông thường, khi lưu trữ mật khẩu trong cơ sở dữ liệu, ta sẽ sử dụng Bcrypt để băm mật khẩu người dùng trước khi lưu trữ nó. Khi người dùng đăng nhập, hệ thống sẽ băm mật khẩu đã nhập và so sánh với giá trị băm trong cơ sở dữ liệu để xác nhận tính hợp lệ của mật khẩu. Với các tính năng an toàn và độ tin cậy của nó, bcrypt đã trở thành một trong những lựa chọn hàng đầu để bảo vệ mật khẩu trong các ứng dụng và hệ thống hiện đại.
  + 1. **Brute Force Attack**
       1. **Khái niệm**
* Hãy tưởng tượng hacker nắm trong tay một danh sách rất lớn các username và mật khẩu phổ biến hay được sử dụng. Sau đó họ gửi liên tục các truy vấn đăng nhập vào file wp-login.php của bạn và nếu tài khoản nào sai, nó sẽ bỏ qua và thử tiếp tài khoản khác. Cứ lần lượt như vậy, sau đó lại “trộn” mật khẩu đến khi nào đăng nhập được thì thôi. Đó là brute force attack.



Hình 7

* Bạn có thể hiểu phương thức này là một cách để dò ra mật khẩu và tài khoản của người quản trị cao nhất.
* Hình thức tấn công này dễ phòng chống nhưng lại rất dễ bị dính nếu bạn chủ quan trong việc đặt mật khẩu username của mình. Thường thì bạn sẽ dễ bị tấn công kiểu này khi:
* Đặt username là admin, administrator hoặc tương tự.
* Mật khẩu không an toàn, dễ đoán ra, sử dụng phổ biến.
* Không bảo mật đường dẫn đăng nhập. Không thay đổi mật khẩu thường xuyên.
* Như vậy, các vấn đề liên quan đến bảo mật tài khoản đăng nhập sẽ đều giúp cho hacker sử dụng brute force attack để tấn công.
  + - 1. **Cách phòng chống**

Để phòng chống brute force attack thì bạn cần làm các công việc sau:

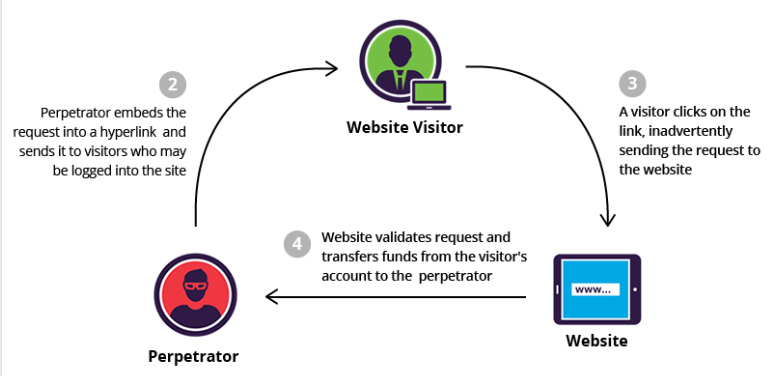
* Đặt tên đăng nhập khó đoán ra.
* Mật khẩu dài, mạnh, có ký tự đặc biệt và không liên quan đến các thông tin cá nhân.
* Hạn chế số lần đăng nhập sai.
* Bảo mật đường dẫn đăng nhập.
* Thường xuyên thay đổi mật khẩu.

Login Attemps – Đơn giản là hạn chế số lần đăng nhập sai.

Hoặc nếu Vậy thì nếu bạn cần chống brute force attack, mình khuyến khích bạn kết hợp các plugin sau:

* Better WP Security – Có tính năng ẩn đường dẫn đăng nhập và hạn chế số lần đăng nhập sai.
* Login Security Solution – Bắt buộc sử dụng mật khẩu mạnh, bắt đổi mật khẩu định kỳ, hạn chế số lần đăng nhập.
* BruteProtect – Chặn các IP xấu hay các truy vấn kiểu brute force có trong dữ liệu của riêng họ.
* Limit bạn muốn an toàn hơn thì sử dụng thêm plugin KeyCaptcha để tạo mã kiểm tra bằng cách xếp ảnh, như thế web bạn sẽ không phải mất công xử lý truy vấn nữa.
* Hãy chắc chắn rằng website bạn đã được bảo vệ trước Brute Force Attack để hạn chế khả năng bị lấy cắp tài khoản quản trị nhé.
  + 1. **CSRF**
       1. **Khái niệm**

Cross-Site Request Forgery, cũng được biết đến với các tên gọi one-click attack, session riding, XSRF, là một phương thức tấn công lừa người dùng cuối thực hiện những hành động không mong muốn trên một ứng dụng web. Phương thức tấn công này sẽ “mượn tay” người dùng gửi những request định sẵn đến server, trong khi họ vẫn đang trong phiên đăng nhập.



Hình 4

* + - 1. **Cách hoạt động**

A có quyền thực hiện một quyền X thông qua một link HTTP. B bằng cách nào đó biết được link này, nhưng không có quyền thực thi. Anh ta sẽ lừa A click vào link soạn sẵn của mình, khi đó, link HTTP sẽ được thực thi một cách hợp pháp dưới quyền của A

* + - 1. **Tác hại của tấn công CSRF**

Tiết lộ thông tin cá nhân: Kẻ tấn công có thể sử dụng tấn công CSRF để truy cập và đánh cắp thông tin cá nhân của người dùng, như tên đăng nhập, mật khẩu hoặc thông tin tài khoản ngân hàng.

Mất dữ liệu quan trọng: Kẻ tấn công có thể sử dụng tấn công CSRF để xóa, chỉnh sửa hoặc thêm dữ liệu trên trang web mà người dùng đã đăng nhập. Điều này có thể dẫn đến mất dữ liệu quan trọng và ảnh hưởng đến hoạt động kinh doanh của các tổ chức.

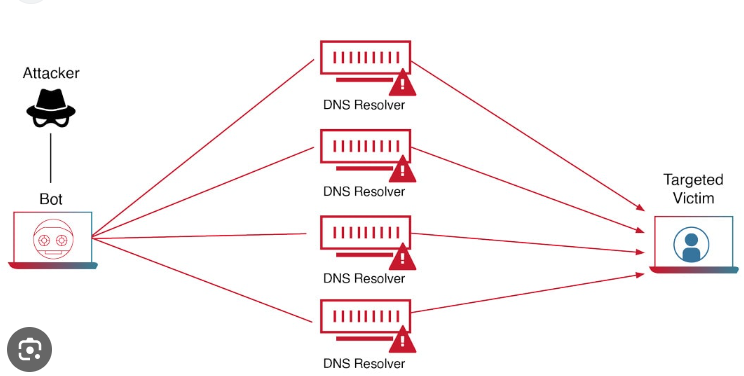
Mất niềm tin của khách hàng: Nếu một trang web bị tấn công CSRF và dẫn đến các vấn đề an ninh, khách hàng có thể mất niềm tin vào trang web và không muốn sử dụng nó nữa. Điều này có thể ảnh hưởng đến doanh thu của trang web và danh tiếng của nó.

Lây nhiễm malware: Kẻ tấn công có thể sử dụng tấn công CSRF để chèn mã độc vào trang web mà người dùng đã đăng nhập. Nếu người dùng truy cập vào trang web đó, họ có thể bị lây nhiễm malware hoặc virus, gây hại cho máy tính hoặc thiết bị của họ.

* + - 1. **Phương pháp ngăn chặn**

Sử dụng token csrf.Khi một yêu cầu được gửi đến trang web, token này được kiểm tra để đảm bảo rằng yêu cầu đó được gửi từ người dùng chính xác và không phải là một yêu cầu giả mạo.

* + 1. **SQL Injection in the DBMS (Lỗi chèn mã SQL)**
       1. **Khái niệm**
          - SQL Injection là một kỹ thuật lợi dụng những lỗ hổng về câu truy vấn của các ứng dụng. Được thực hiện bằng cách chèn thêm một đoạn SQL để làm sai lệnh đi câu truy vấn ban đầu, từ đó có thể khai thác dữ liệu từ database. SQL injection có thể cho phép những kẻ tấn công thực hiện các thao tác như một người quản trị web, trên cơ sở dữ liệu của ứng dụng.
       2. **Lỗi chèn mã SQL có thể giúp tin tặc** 
          - Vượt qua các khâu xác thực người dùng
          - Chèn, sửa đổi, hoặc xóa dữ liệu
          - Đánh cắp các thông tin trong CSDL
          - Chiếm quyền điều khiển hệ thống máy chủ CSDL
    2. **DDOS**
       1. **Khái niệm DDOS**
* DDoS là viết tắt của Distributed Denial of Service, có nghĩa là từ chối dịch vụ phân tán, là hoạt động chấm dứt hoặc gián đoạn các dịch vụ tại máy nạn nhân. Tấn công DDOS huy động số lượng lớn các máy bị hacker lợi dụng để tấn công máy nạn nhân vào cùng một thời điểm.



Hình 8

* Trên Internet tấn công DDOS là một dạng tấn công nhiều máy tính tới một đích, nó gây ra từ chối các yêu cầu hợp lệ của các user bình thường bằng cách tạo ra những gói tin cực nhiều đến một đích cụ thể, nó có thể gây tình trạng tương tự như hệ thống bị shutdown.
* Khi một mạng máy tính bị Hacker tấn công nó sẽ chiếm một lượng lớn tài nguyên trên server như dung lượng ổ cứng, bộ nhớ , CPU, băng thông… Lượng tài nguyên này sẽ phụ thuộc vào khả năng huy động tấn công của mỗi Hacker. Khi đó Server không thể đáp ứng hết những yêu cầu của những người sử dụng hợp pháp và từ đó Server có thể nhanh chóng bị ngừng hoạt động.
  + - 1. **Các đặc tính của DDOS**
* DDoS được tấn công từ hệ thống máy tính cực lớn trên Internet, và thường dựa vào các dịch vụ có sẵn trên các máy tính trong mạng botnet.
* Các dịch vụ tấn công được điều khiển từ những “primary victim” trong khi đó các máy tính bị chiếm quyền sử dụng trong mạng Bot được sử dụng để tấn công thường được gọi là ” secondary victims”. - Đây là dạng tấn công rất khó có thể phát hiện bởi vì những tấn công này được sinh ra từ nhiều địa chỉ IP trên Internet.
  + - 1. **Phân loại các kiểu tấn công DDOS**
* Trong phần này chúng em sẽ trình bày về các kiểu tấn công DDOS hay gặp phải. Các loại tấn công DDoS có rất nhiều biến thể, nên việc phân loại cũng có rất nhiều cách khác nhau. Ở trong phạm vi đề tài này, em sẽ dựa trên việc tấn công nhắm vào các thành phần phần khác nhau của kết nối mạng để phân chia ra các loại tấn công DDOS. Một kết nối mạng bao gồm nhiều thành phần hoặc “tầng” khác nhau.
* Cụ thể như mô hình OSI bên dưới, là một khung khái niệm được sử dụng để mô tả kết nối mạng trong 7 lớp riêng biệt.
  + - 1. **Các hình thức tấn công DDOS phổ biến hiện nay**

Trong phần này chúng em dự định trình bày về các hình thức tấn công DDOS phổ biến hiện nay.

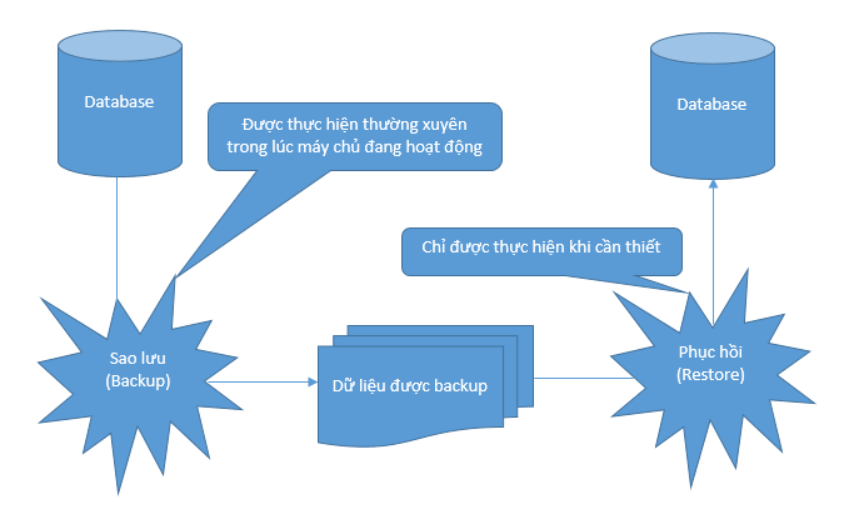
* SYN Flood: Là hình thức tấn công nhắm vào những điểm yếu của chuỗi kết nối TCP, bằng cách dựa vào các kết nối chưa hoàn chỉnh. Hacker có thể gửi nhiều yêu cầu kết nối TCP SYN đến server để server sử dụng tài nguyên của mình phục vụ cho những yêu câu kết nối TCP SYN không hợp pháp của hacker. Để khi có các truy cập hợp pháp thì máy server sẽ không còn đủ tài nguyên để phục vụ nữa.
* UDP Flood: UDP Flood là cuộc tấn công DDoS gây ra hiện tượng Flooding với mục tiêu là các gói Giao thức dữ liệu người dùng (UDP). Mục tiêu là làm ngập ngẫu nhiên các cổng trên một máy chủ từ xa. Điều này khiến máy chủ liên tục kiểm tra ứng dụng đang lắng nghe tại cổng đó, và khi không thấy ứng dụng, máy chủ sẽ trả lời bằng gói ICMP Destination Unreachable. Quá trình này làm tiêu hao tài nguyên của máy chủ, cuối cùng có thể dẫn đến không thể truy cập được.
* HTTP Flood: Trong một cuộc tấn công của HTTP Flood, kẻ tấn công sẽ khai thác các yêu cầu HTTP GET hoặc POST hợp pháp để có thể dễ dàng tấn công máy chủ web hoặc ứng dụng. HTTP Flood dùng hàng loạt botnet kết hợp với hàng nghìn máy tính. Các máy tính này bị kiểm soát bởi những phần mềm độc hại. Tuy chúng làm tiêu tốn ít băng thông nhưng lại buộc máy chủ dùng tối đa nguồn tài nguyên để xử lý.
* Smurf Attack: Đây là kiểu tấn công sử dụng chương trình độc hại Smurf để lợi dụng IP cùng giao thức ICMP. Đối tượng tấn công lấy IP nguồn để ping các ICMP đến nhiều Broadcast của các mạng. Từ đó, chúng khiến địa chỉ IP sẽ nhận hàng loạt phản hồi gói ICMP. Điều này làm mạng chậm và không đáp ứng được các yêu cầu dịch vụ khác.
* NTP Amplification: Đây là kiểu tấn công bằng cách sử dụng những gói dữ liệu mà đối tượng tấn công đã khai thác được từ máy chủ Network Time Protocol (NTP), làm cho máy chủ đích hoặc hệ thống mạng quá tải do các gói UDP bị khuếch đại.
  + - 1. **Cách nhận biết cuộc tấn công DDOS**
* Phần khó nhất của một cuộc tấn công DDoS là sẽ không có cảnh báo cụ thể. Một số nhóm hacker lớn sẽ gửi thư đe dọa, nhưng đa phần những kẻ tấn công sẽ gửi lệnh tấn công bất ngờ mà không có bất kỳ cảnh báo nào cả. Ban đầu, người dùng có thể không nghĩ đó là một cuộc tấn công DDoS mà thay vào đó cho rằng máy tính của mình chỉ đang gặp những vấn đề cơ bản. Dù đã tiến hành kiểm tra máy tính và thực hiện các cuộc kiểm tra cơ bản, nhưng người dùng sẽ chỉ thấy một lượng lớn lưu lượng truy cập mạng với tài nguyên đã được sử dụng tối đa. Thông thường các server của website đang gặp phải một cuộc tấn công DDoS sẽ có những dấu hiệu như: mặc dù mạng Internet đang ổn định và truy cập các website khác vẫn diễn ra bình thường nhưng mạng của mình hoặc mạng của hệ thống bị chậm một cách bất thường khi truy cập vào website đó. Hoặc người dùng có thể kiểm tra xem email của bản thân có đang phải nhận nhiều thư rác hay không. Và việc không thể truy cập vào một mục của website hay không thể truy cập vào nhiều website cũng là dấu hiệu của một cuộc tấn công DDoS.
  + - 1. **Các cách phòng chống tấn công DDOS**
* Trong phần này chúng em sẽ trình bày về các cách phòng chống tấn công DDOS. Rõ ràng rằng có rất nhiều loại tấn công DDoS. Và cũng có rất nhiều cách để phòng chống một cuộc tấn công DDoS mà người dùng có thể tự trang bị cho bản thân. Nhóm em sẽ liệt kê một vài cách giúp phòng chống và giảm thiểu tấn công DDoS cho website như sau:
* Sử dụng dịch vụ hosting cao cấp: Việc sử dụng các nguồn hosting cao cấp sẽ giúp máy chủ có thể kịp thời ngăn chặn các cuộc tấn công DDoS. Bởi vì dịch vụ hosting càng cao cấp thì khả năng bảo mật của nó cũng sẽ cao, từ đó sẽ giúp giảm đáng kể các cuộc tấn công mạng vào website.
* Theo dõi lưu lượng truy cập: Khi các tổ chức biết một cuộc tấn công DDoS đang diễn ra, họ có thể thực hiện nhiều hành động khác nhau để bảo vệ cơ sở hạ tầng của mình. Khi cuộc tấn công xảy ra đầu tiên họ sẽ ngăn các gói dữ liệu độc hại tiếp cận máy chủ bằng cách "định tuyến rỗng", việc này sẽ làm giảm và chuyển hướng các yêu cầu Flooding dưới sự chỉ đạo của mạng botnet. Trong một số trường hợp, tất cả lưu lượng truy cập được chuyển hướng đến một "bộ lọc" để sắp xếp các yêu cầu hợp pháp từ những yêu cầu độc hại một cách kỹ lưỡng hơn. Tuy nhiên, nhiều biện pháp an ninh mạng phụ thuộc vào băng thông có thể bị choáng ngợp bởi các cuộc tấn công quy mô lớn.
  + 1. **Sao lưu và backup dữ liệu**

Sao lưu và phục hồi dữ liệu trong sql server là một trong những thao tác quan trọng mà người quản trị cơ sở dữ liệu phải thực hiện.

Nếu như thao tác sao lưu (Backup database) được thực hiện để lưu dữ liệu và được thực hiện thường xuyên thì thao tác phục hồi dữ liệu (Restore database) chỉ được thực hiện khi nào máy chủ bị sự cố như hư ổ cứng hoặc dữ liệu bị mất do người dùng vô tình hoặc cố ý xoá,…

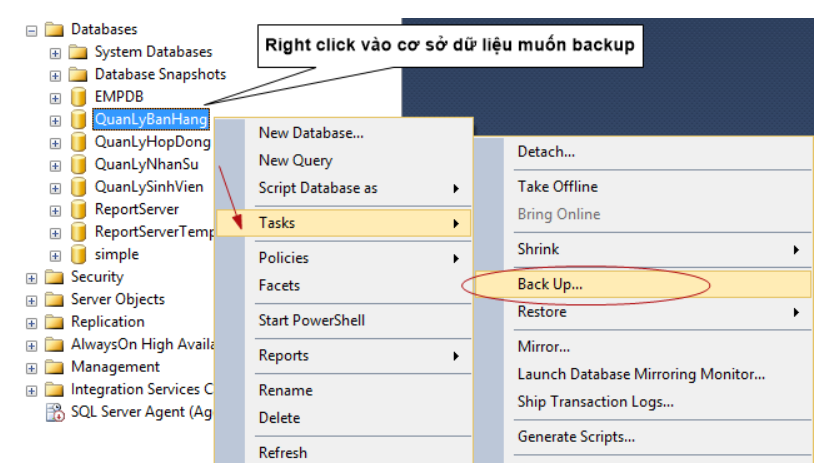
Ngoài ra thao tác phục hổi dữ liệu cũng được thực hiện để sao chép database từ máy chủ này sang máy chủ khác.

**Loại 1: Full backup**

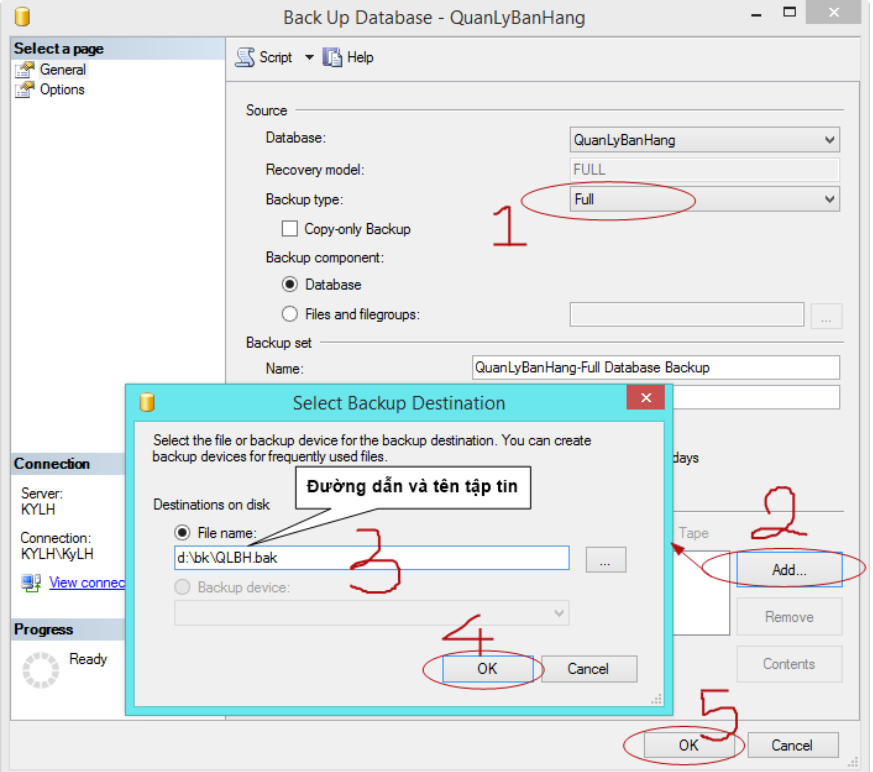


Hình 9

Thực hiện full backup bằng giao diện



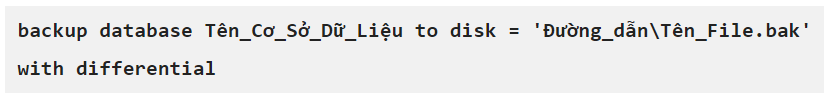
**Hình 10**



**Hình 11**

**Loại 2: Different backup**

**Cú pháp:**



**Loại 3 log backup**



* + 1. **Default and Weak Passwords**
       - * Nhiều hệ quản trị dữ liệu sử dụng tài khoản quản trị với mật khẩu ngầm định đơn giản hoặc rỗng
    2. **Excessive User & Group Privileges**
       1. **Khái niệm**
          - Quyền lợi người dùng quá mức xảy ra khi người dùng cá nhân có nhiều quyền hơn là cần thiết để thực hiện các nhiệm vụ cụ thể của họ.
          - Quyền lợi nhóm quá mức đề cập đến tình trạng mà toàn bộ nhóm người dùng được cấp quyền truy cập nhiều hơn so với cần thiết.
       2. **Rủi ro an ninh**
          - Truy cập trái phép: Người dùng có quyền lợi quá mức có thể truy cập dữ liệu nhạy cảm một cách vô tình hoặc có ý định xấu.
          - Rủi ro mất dữ liệu: Quyền lợi quá mức tăng cường rủi ro mất dữ liệu, vì người dùng có thể truy cập nhiều dữ liệu hơn là cần thiết.
          - Malware và Tận dụng lỗ hổng: Phần mềm độc hại hoặc kẻ tấn công có thể tận dụng quyền lợi quá mức để thực hiện các hoạt động không được phép.
       3. **Cách phòng chống**
          - Nguyên tắc ít quyền lợi nhất: Gán mức quyền lợi tối thiểu cần thiết cho người dùng và nhóm để thực hiện nhiệm vụ của họ.
          - Kiểm tra định kỳ: Thực hiện kiểm tra và đánh giá định kỳ về quyền lợi người dùng và nhóm để xác định và sửa chữa quyền lợi quá mức.
          - Kiểm soát dựa trên vai trò (RBAC): Triển khai RBAC để gán quyền dựa trên vai trò công việc, giúp quản lý và kiểm soát quyền truy cập.
    3. **Privilege Escalation**
       1. **Khái niệm**
          - Privilege Escalation là một kỹ thuật tấn công mạng trong đó kẻ tấn công có được quyền truy cập trái phép vào các đặc quyền cao hơn bằng cách tận dụng các lỗi, điểm yếu và lỗ hổng bảo mật trong hệ thống của tổ chức. Đó là nỗ lực nâng cao quyền truy cập bằng cách khai thác lỗi, lỗi hệ thống, hành vi của con người, giám sát cấu hình hoặc kiểm soát truy cập yếu. Trong hầu hết các trường hợp, nỗ lực tấn công thâm nhập đầu tiên là không đủ để đạt được mức truy cập dữ liệu cần thiết. Sau đó, những kẻ tấn công sử dụng biện pháp leo thang đặc quyền để có quyền truy cập sâu hơn vào mạng, tài sản và thông tin nhạy cảm.
          - Các cuộc tấn công leo thang đặc quyền được thực hiện nhằm gây nguy hiểm cho hoạt động kinh doanh bằng cách lấy cắp dữ liệu và tạo ra các cửa hậu. Mục tiêu của việc leo thang đặc quyền là giành quyền kiểm soát hoàn toàn hệ thống hoặc mạng với mục đích xấu là vi phạm an ninh, đánh cắp dữ liệu, v.v. Các tác nhân đe dọa thực hiện các cuộc tấn công này có thể là tin tặc bên ngoài hoặc nội bộ bắt đầu bằng cách thực hiện một cuộc tấn công kỹ thuật xã hội như lừa đảo để có quyền truy cập vào mạng và hệ thống máy tính thông qua hành vi trộm cắp thông tin xác thực.
       2. **Cách thức tấn công**
          - Tạo 1 thủ tục myproc chứa mã cấp quyền cho người dùng hiện tại thành Database Admin(DBA).
          - Tạo hàm myfn chứa mã để tạo 1 trigger trong system schema. Trigger chứa lệnh gọi thực hiện myproc.
          - Tiến hành khai thác lỗi trong mô đun MDSYS.RESET\_INPROG\_INDEX, yêu cầu MDSYS thực hiện myfn để tạo trigger.
          - Sử dụng người dùng có quyền PUBLIC để chạy 1 lệnh SQL để kích hoạt việc thực hiện trigger đã tạo. Hệ thống sẽ thực hiện trigger và trigger sẽ gọi thực hiện myproc cấp quyền DBA cho người dùng.
       3. **Biện pháp phòng chống**
          - Thiết lập đặc quyền tối thiểu để hạn chế quyền truy cập vào hệ thống, dữ liệu nhạy cảm.
          - Khuyến khích người dùng sử dụng mật khẩu mạnh và độc đáo, khó đoán hoặc bẻ khóa hơn.
          - Vá lỗi và cập nhật phần mềm thường xuyên
    4. **Logic Attacks**
       1. **Giới thiệu**
          - Logic attacks là dạng tấn công nhằm vào logic xử lý của website và thường không dựa vào các kỹ thuật biết trước.
          - Dạng tấn công này thường không tồn tại các công cụ, kỹ thuật rà quét để phát hiện các lỗi hoặc lỗ hổng trong thiết kế hoặc logic xử lý của hệ thống.
       2. **Các dạng tấn công logic attacks**
          - **Lạm dụng lưu trình xử lý:** kẻ tấn công có thể thử website rất kỹ để tìm ra lỗi trong lưu trình xử lý và lạm dụng.
          - **Khai thác các lỗi trong chính sách và áp dụng.**
          - **Các mẫu thiết kế không an toàn:**

Các hành vi mơ hồ không xác định và bất ngờ.

Kiểm tra cấp quyền không đầy đủ.

Lọc đữ liệu không đầy đủ.

Trộn lẫn dữ liệu và mã.

Chuẩn hóa sai và cú pháp đồng nghĩa.

Tin tưởng vào ben máy khách

* + - * + **Các lỗi cài đặt hệ mã hóa:** việc cài đặt phát sinh một số lỗ hổng bảo mật do các yêu cầu đối với các tham số và bản thân thuật toán mã hóa không thể đảm bảo chặt chẽ.
      1. **Các biện pháp phòng chống**
         * Yêu cầu viết tài liệu đảm bảo rằng các tài liệu mô tả các khâu cần được thục hiện đầy đủ.
         * Tạo các test case toàn diện về: chức năng, lưu đồ xử lý và các bộ lọc, kiểm tra định dạng, kích thước và nội dung dữ liệu.
         * Ánh xạ chính sách - kiểm soát.
         * Lập trình an toàn: mã cần được viết rõ ràng, đủ chú thích, dễ đọc, dễ bảo trì và sửa đổi.
         * Xác minh máy khách: kiểm tra việc chuyển trạng thái và kiểm tra quy trình xử lý giao dịch.
         * Thực hiện theo các hướng dẫn mã hóa

Sử dụng các giải thuật mã hóa đã được chuẩn hóa và cài đặt dựa trên các thư viện đã được test kỹ càng.

Sử dụng hàm băm HMAC để phát hiện sửa đổi dữ liệu.

Không thông báo lỗi giải mã cho client do kẻ tấn công có thể lợi dụng để dung cho việc phân tích phá mã.

Giảm bớt tần suất trình duyệt truy nhập dữ liệu nhạy cảm.

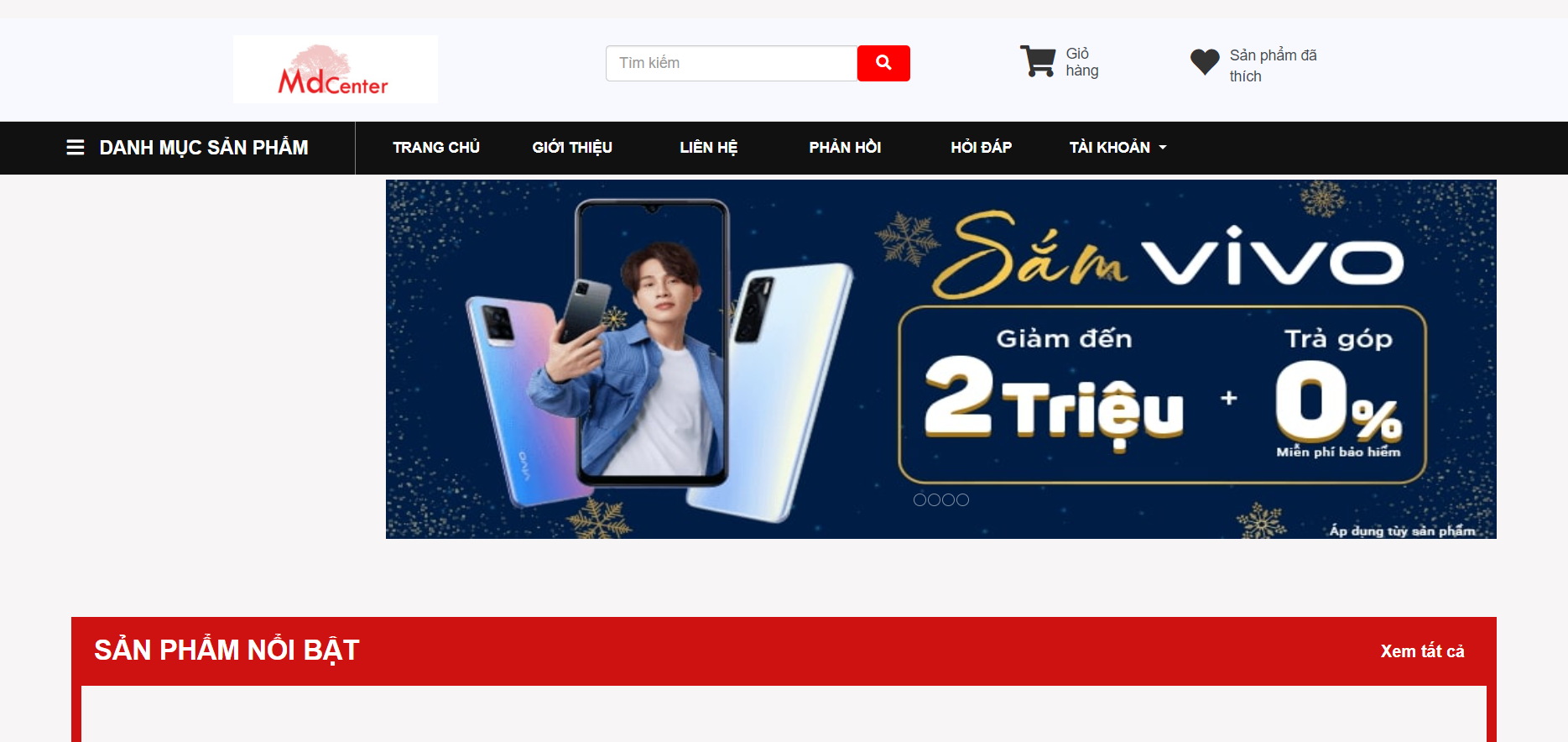
Nhận dạng các điểm truy nhập vào dữ liệu nhạy cảm không mã hóa.

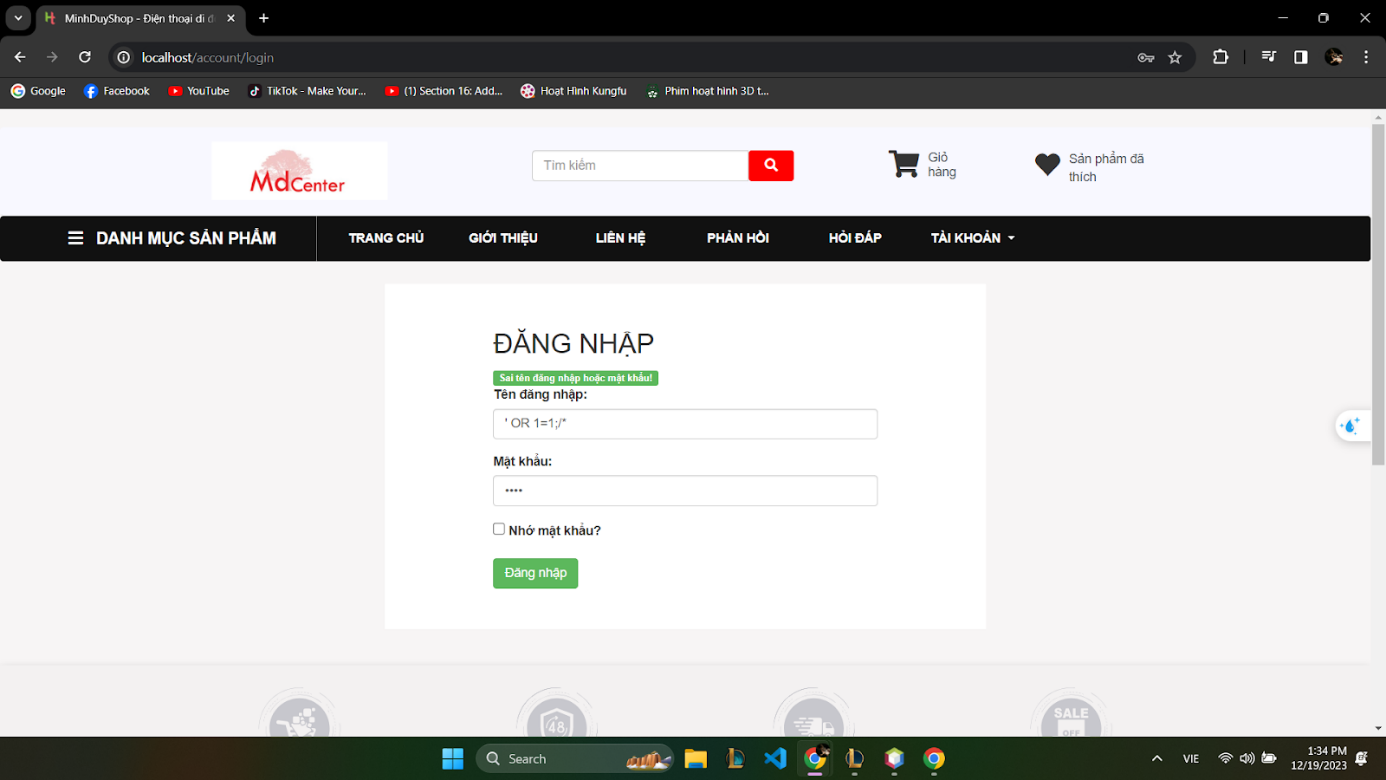
Đồng thời, cần ghi log truy nhập với nhóm người dùng có quyền truy nhập vào dữ liệu nhạy cảm không mã hóa.

Tạo số ngẫu nhiên thực sự ngẫu nhiên.

1. **Chương 2: Chương trình DEMO và thực nghiệm**

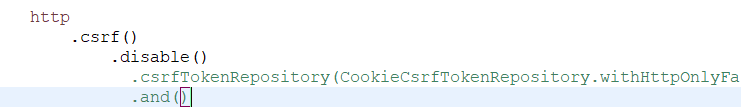
- Chương trình được DEMO trên web bán điện thoại di động

**Hình 12**

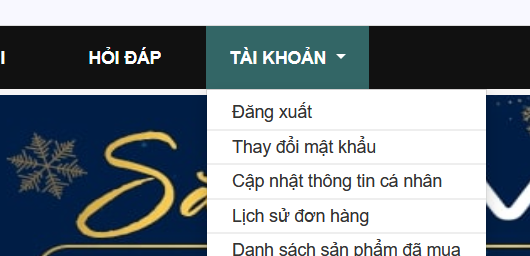
****

**Hình 13**

Demo CSRF

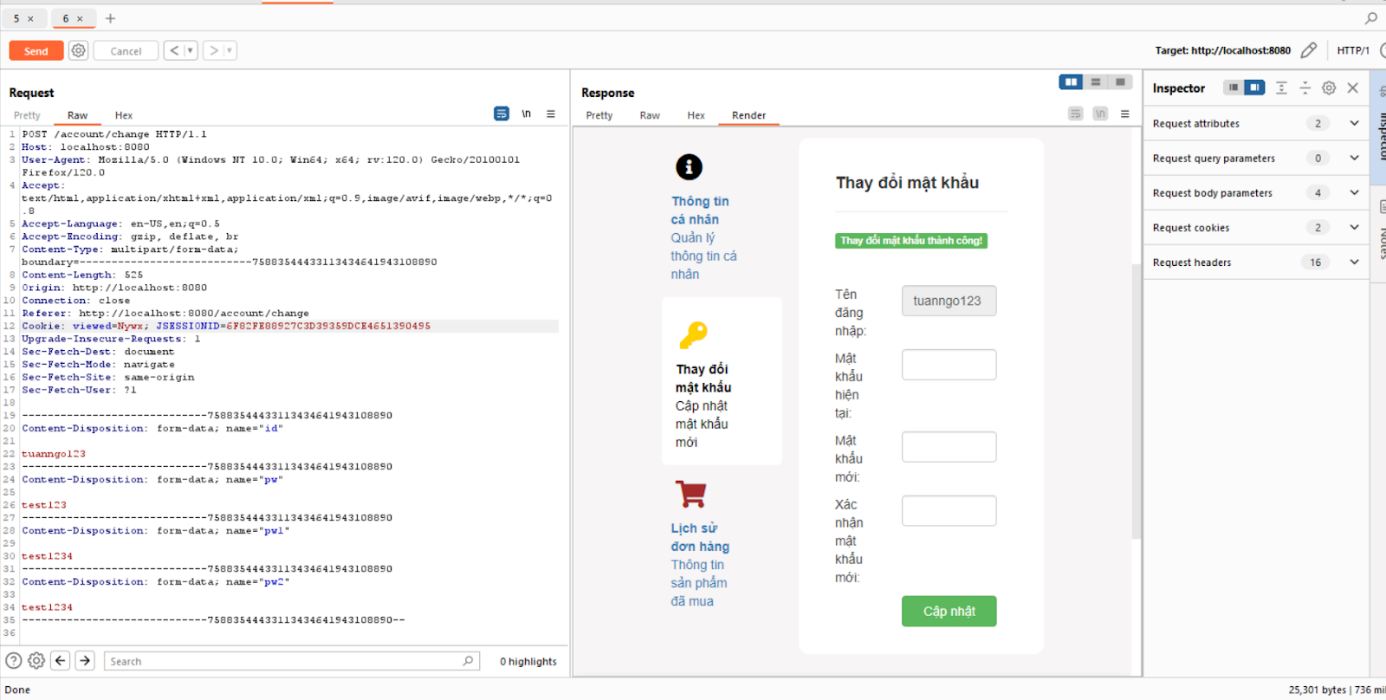
****  
**Hình 14**

* Disable csrf token để tiến hành thử nghiệm
* Sử dụng burp suite bắt request, chỉnh sửa nhằm tấn công

****

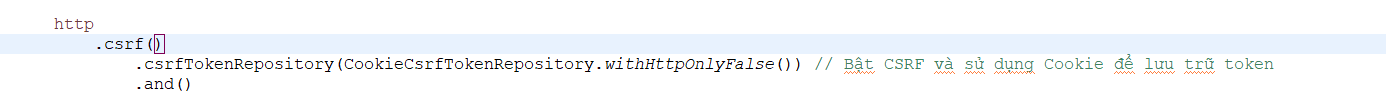
**Hình 15**

**Thực hiện thay đổi mật khẩu**

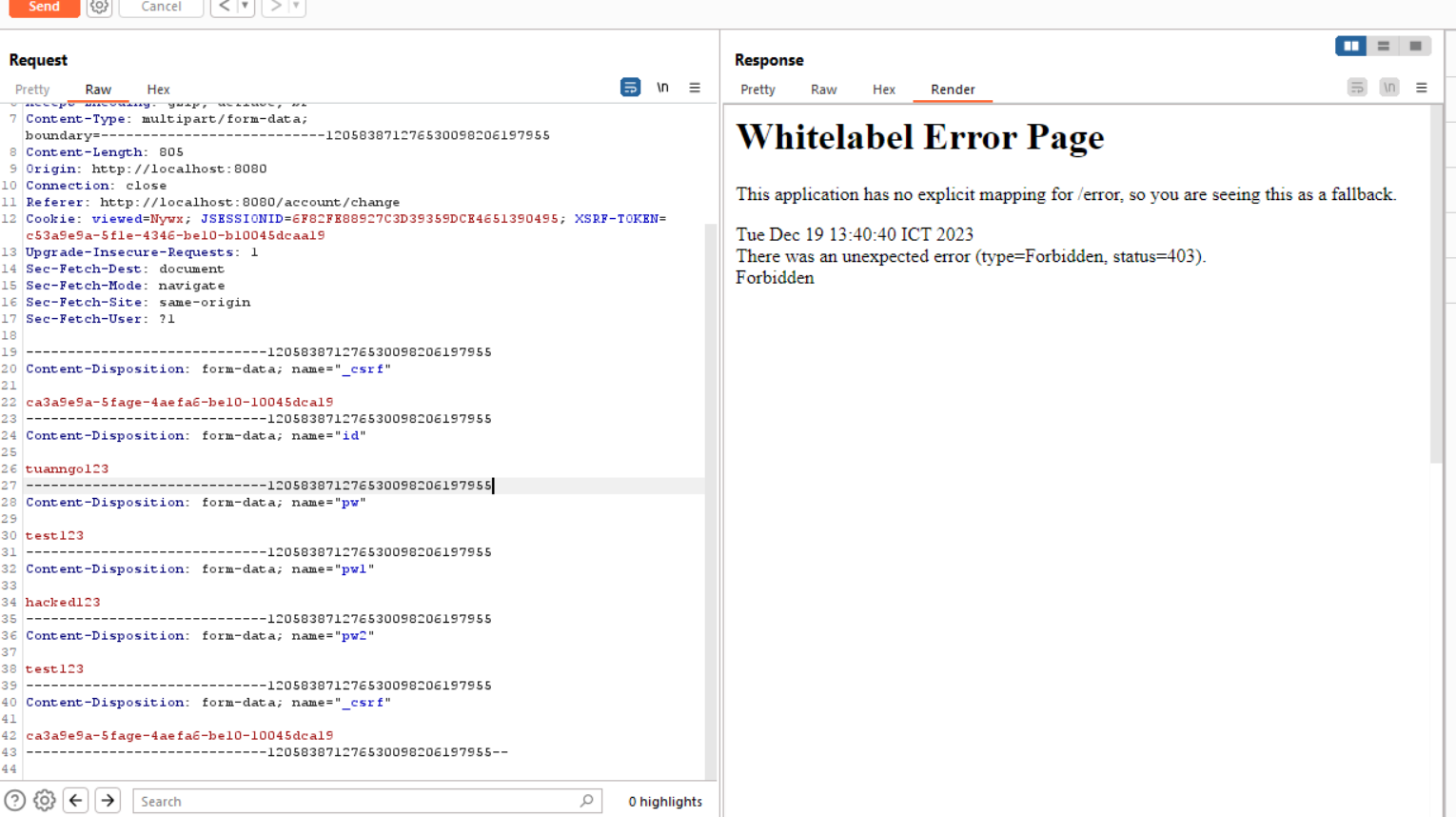
****

**Hình 16**

* Khi chưa có token csrf, việc thay đổi mật khẩu và brute force là hoàn toàn có thể diễn ra bởi không có một trường nào có thể kiểm tra request đó có đến từ trang web hay không.
* Enable csrf token tiến hành đổi mật khẩu lại thông qua burp suite

****

**Hình 17**

****

**Hình 18**

1. **Chương 3: Tổng kết**
   1. **Các lỗi đã khắc phục**

Qua phần tìm hiểu lý thuyết cùng với phần thực hành demo tại web bán điện thoại nhóm nắm được và giải quyết các lỗ hỏng cơ bản và tìm cách khắc phục, phòng chống như là:

* Chèn mã HTML và cross-Site scripting(XSS), DDOS, Brute Force Attack. CSRF
* Tấn công chèn mã SQL
* Tấn công vào cơ chế xát thực tài khoảng
* Tấn công vào trình duyệt web sự riêng tư của người dùng
* Xát thực người dùng và quyền truy cập, bảo mật phiên làm việc, bảo mật máy chủ web, bảo mật cơ sở dữ liệu web, bảo mật trình duyệt
* Bảo mật CSDL dưới một số dạng tấn công phổ biến, sử dụng mã hóa trong CSDL
* Backup data bảo mật để tránh thông tin thông bị mất.

**3.2 Các lỗi còn chưa khắc phục được**

* Một số lỗi bảo mật cơ sở dữ liệu còn chưa khắc phục được
* Chưa sử dụng được một số phần miềm bảo vệ hệ thống cơ sở dữ liệu

# **TÀI LIỆU THAM KHẢO**

[1] Denny Cherry (2012), securing SQL Server: Protecting Database from Attackers, Syngess.

[2] Michael E.Whitman, Herbert J.Mattord (2012), Principles of information security,4th edition, Course Tecnology, Cengage Learning.

[3] MiKe Shema (2012), Hacking web apps: Detecting and Preventing Web Application Security Proplems, Elsevier Inc.

[4]Roberta Bragg, Mark Rhodes-Ousley and keith Strasberg (2013), Network Security: The Complete Reference, MeGraw-Hill Osborne Media.

**BẢNG PHÂN CÔNG NHIỆM VỤ**

|  |  |  |
| --- | --- | --- |
| STT | HỌ & TÊN | Nhiệm vụ |
| 1 | Đỗ Diệu Hiền | xát thực thông tin dk, backup lưu trữ thông tin data, mã hóa password |
| 2 | Ngô Đức Tuấn | Xss,csrf |
| 3 | Phan Xuân Tấn | Excessive User & Group Privileges |
| 4 | Trần Quang Việt | buffer overflows, Excessive User & Group Privileges |
| 5 | Trần Trung Kiên | sql injection với brute force |