HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Tô Thanh Tú

GIẢI PHÁP CẢNH BÁO KIỂU TẤN CÔNG AN NINH MẠNG DEFACE VÀ HIỆN THỰC

LUẬN VĂN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

TP.HỒ CHÍ MINH - NĂM 2023

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Tô Thanh Tú

GIẢI PHÁP CẢNH BÁO KIỂU TẤN CÔNG AN NINH MẠNG DEFACE VÀ HIỆN THỰC

CHUYÊN NGÀNH: HỆ THỐNG THÔNG TIN

MÃ SỐ: 8.48.01.04

LUẬN VĂN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC

TS. NGUYỄN ĐỨC THÁI

TP.HỒ CHÍ MINH - NĂM 2023

LỜI CAM ĐOAN

Tôi cam đoan luận văn **"Giải pháp cảnh báo kiểu tấn công an ninh mạng deface và hiện thực"** là công trình nghiên cứu của chính tôi.

Tôi cam đoan các kết quả, các số liệu được nêu ra trong luận văn là trung thực, chưa có tác giả nào nghiên cứu hoặc công bố trước đây. Các công trình, đề tài được dùng làm tài liệu tham khảo, làm cơ sở hoàn thiện luận văn này được trích dẫn đúng theo quy định.

TP. Hồ Chí Minh, ngày 28 tháng 02 năm 2023

Học viên thực hiện luận văn

Tô Thanh Tú

LỜI CẢM ƠN

Trước hết, tôi xin chân thành cảm ơn Ban Giám đốc, Phòng Đào tạo Sau đại học và Quý Thầy, Cô tại Học viện Công nghệ Bưu chính Viễn thông Cơ sở TP.Hồ Chí Minh đã tạo mọi điều kiện thuận lợi giúp tôi trong thời gian học tập và hoàn thành luận văn.

Tôi xin chân thành cảm ơn **Thầy TS.Nguyễn Đức Thái** đã tận tình giúp đỡ, tạo điều kiện thuận lợi cho tôi trong suốt quá trình thực hiện luận văn.

Tôi đã sử dụng những kiến thức được học tập, được truyền đạt để hoàn thành luận văn. Trong quá trình nghiên cứu, thực hiện luận văn, tôi đã nhận được nhiều sự động viên, hướng dẫn hết sức quý báu của quý Thầy, Cô. Tuy nhiên do kết quả nghiên cứu vẫn có hạn chế, thiếu sót. Tôi mong tiếp tục nhận được sự góp ý của quý Thầy, Cô để ngày càng hoàn thiện hơn.

Trân trọng cảm ơn!

TP. Hồ Chí Minh, ngày 28 tháng 02 năm 2023

Học viên thực hiện luận văn

Tô Thanh Tú

MỤC LỤC

DANH SÁCH BẢNG

DANH SÁCH HÌNH VẼ

DANH MỤC CÁC THUẬT NGỮ, CHỮ VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
CSS	Cascading Style Sheets	
DOS	Denial Of Service	Tấn công từ chối dịch vụ
HTML	Hypertext Markup Language	Ngôn ngữ đánh dấu siêu văn bản
PSNR	Peak Signal To Noise Ratio	Tỷ số tín hiệu cực đại trên nhiễu
SMS	Short Message Service	Dịch vụ tin nhắn ngắn
SQL	Structured Query Language	ngôn ngữ truy vấn có cấu trúc
XSS	Cross Site Scripting	

MỞ ĐẦU

Từ khi Internet lần đầu xuất hiện tại Việt Nam, đến nay Internet là đã phát triển rộng khắp đến mọi nơi và trở nên phổ biến. Cùng với đó, lĩnh vực công nghệ thông tin cũng đã phát triển mạnh mẽ và trở thành ngành khoa học đóng vai trò thúc đẩy kinh tế xã hội phát triển. Nhiều cơ hội được tạo ra, nhiều tri thức của nhân loại được tiếp cận do lực lượng lao động từng bước được nâng cao trình độ lao động, hiệu suất làm việc.

Bên cạnh những thuận lợi, các lợi ích mang lại, công nghệ thông tin cũng tạo ra nhiều thách thức. Trong các thách thức về an toàn và an ninh thông tin thì an ninh mạng có vai trò hết sức quan trọng. Các hình thức, tần suất và mức độ tấn công, phá hoại trên mạng Internet đang ngày càng trở lên tinh vi, phức tạp và gây ra nhiều hậu quả nghiêm trọng. Do vậy, đòi hỏi vai trò của người quản trị mạng phải đảm bảo an toàn hệ thống trở nên hết sức cần thiết.

Thông qua việc tìm hiểu về một số kiểu tấn công phổ biến hiện nay, đề tài sẽ tập trung tìm hiểu vào hình thức tấn công an ninh mạng deface (còn được gọi là tấn công làm thay đổi nội dung website). Một loại tấn công phổ biến nhưng có ảnh hưởng nghiêm trọng đối với nhà quản trị website và các tổ chức, doanh nghiệp.

CHƯƠNG 1: GIỚI THIỆU ĐỀ TÀI

1.1 Tính cấp thiết của đề tài

Hiện nay, tôi đang công tác tại Sở Thông tin và Truyền thông tỉnh Tây Ninh, đây là một cơ quan phụ trách về hoạt động ứng dụng Công nghệ thông tin trên địa bàn tỉnh. Việc phát hiện các cuộc tấn công an ninh mạng, trong đó tấn công an ninh mạng deface là một trong những mối quan tâm nhằm kịp thời phòng, chống lại việc bị up các thông tin sai lệch, các hình ảnh phản động, bôi nhọ lãnh đạo của Đảng, Nhà nước trên website của Sở, ban, ngành của tỉnh. Đến nay, chưa có công cụ nhằm phát hiện cuộc tấn công nêu trên. Và đó là lý do tôi chọn đề tài này.

Từ các kiến thức tìm hiểu được, tôi mong muốn góp một phần nhỏ vào việc nghiên cứu, xác định các dấu hiệu phát hiện cuộc tấn công an ninh mạng deface. Từ đó, xây dựng hệ thống có khả năng kịp thời cảnh báo đến người quản trị hệ thống, giúp cho các website của tỉnh được vận hành ổn định.

1.2 Mục tiêu của đề tài

Tìm hiểu cách thức hoạt động, trình bày của một website, các kỹ thuật tấn công và bảo mật website.

Xác định các dấu hiệu nhằm phát hiện một cuộc tấn công an ninh mạng deface:

- Dấu hiệu phát hiện sự thay đổi tính toàn vẹn: Ứng dụng hàm băm để kiểm tra sự thay đổi của mã nguồn của website trên máy chủ web;

- Dấu hiệu phát hiện sự thay đổi dữ liệu;

- Dấu phát hiện tấn công làm tê liệt website;

Đề xuất xây dựng một hệ thống giám sát website có 02 tính năng như sau:

- Đặt tại máy chủ website nhằm phát hiện kịp thời cuộc tấn công an ninh mạng deface. Dự kiến áp dụng đối với các website của các Sở, ban, ngành trên bàn tỉnh Tây

Ninh. Các thông tin cảnh báo sẽ được xử lý để quyết định có gửi thông báo đến người quản trị hay không, tần suất cảnh báo phải phù hợp. Hình thức cảnh báo: Sử dụng email và tin nhắn SMS để thông báo.

- Đặt bên ngoài máy chủ của website để tiếp tục thực hiện chức năng cảnh báo ngay cả khi website bị kẻ gian chiếm quyền điều khiển hoặc phá hủy.

1.3 Phương pháp thực hiện đề tài

- Phương pháp luận: Dựa trên bài báo, đề xuất nhằm phát hiện, chống tấn công làm thay đổi nội dung đã được trình bày, công bố trước đó.

- Phương pháp thống kê: Phương pháp này được áp dụng để tổng hợp, chọn lọc những thông tin, dữ liệu theo đúng yêu cầu đặt ra.

- Phương pháp thực nghiệm: Xây dựng hệ thống và thực nghiệm thuật toán đã đề xuất.

1.4 Cấu trúc luận văn

Luận văn được trình bày 6 Chương, cụ thể như sau:

Chương 1. Giới thiệu đề tài

Giới thiệu tổng quan, lý do lựa chọn đề tài.

Chương 2. Những công trình liên quan

Chương này sẽ trình bày một số công trình, kết quả nghiên cứu các biện pháp nhằm phát hiện một trang website đã bị tấn công deface đã được thực hiện hoặc đề xuất.

Chương 3. Nền tảng lý thuyết liên quan đến tấn công an ninh mạng deface

Chương này trình bày những một số biện pháp tấn công an ninh mạng, khái niệm chung về tấn công an ninh mạng deface và thực trạng việc tấn an ninh mạng deface hiện nay.

Khái niệm về hàm băm và việc ứng dụng hàm băm để phát hiện sự tương đồng giữa 02 tài liệu.

Chương 4. Đề xuất biện pháp nhằm phát hiện cuộc tấn công an ninh mạng deface

Trên cơ sở nền tảng lý thuyết và các công trình nghiên cứu trước, tác giả đề xuất một số biện pháp cơ bản nhằm phát hiện cuộc tấn công an ninh mạng deface.

Chương 5. Xây dựng hệ thống giám sát và cảnh báo cuộc tấn công an ninh mạng deface

Chương này đề xuất hệ thống thực hiện các biện pháp đề xuất để đánh giá.

Chương 6. Kết luận

Chương này sẽ tổng hợp kết quả đạt được của luận văn, các tồn tại còn có và đề xuất hướng nghiên cứu, xử lý trong thời gian tới.

CHƯƠNG 2: NHỮNG CÔNG TRÌNH LIÊN QUAN

Chương này sẽ trình bày một số công trình, các biện pháp của các nhà khoa học đã thực hiện hoặc đề xuất nhằm phát hiện một trang website đã bị tấn công deface.

Hai tác giả Andrew Cooks và Martin S Olivier (2004) đề xuất giải pháp hạn chế bị tấn công an ninh mạng deface bằng chiến thuật chỉ đọc tại bài báo [1]"curtailing web defacement using a read-only strategy". Giải pháp này cố gắng cải thiện tính toàn vẹn (integrity) của website bằng cách đảm bảo nội dung và cấu hình hệ thống không bị thay đổi trên máy chủ web: Thường xuyên khởi động lại hệ thống đặt lại nó về trạng thái đáng tin cậy.

Tại bài báo [2]"Anti Web Site Defacement System (AWDS)" đăng bởi tác giả Mazin S.Al-Hakeem (2010) đã đề xuất một hệ thống có khả năng phát hiện và khôi phục website bằng cách áp dụng thuật toán Rabin's Fingerprinting cải tiến.

Năm 2010, ba nhà nghiên cứu Bartoli, A.; Davanzo, G. và Medvet, E. tại bài báo [3]"A Framework for Large-Scale Detection of Web Site Defacements" đã đề xuất một biện pháp nhằm phát hiện cuộc tấn công deface ở quy mô lớn dựa vào kỹ thuật phát hiện sự bất thường (anamoly detection technique): Mạng lưới (network) trong thời gian huấn luyện sẽ xác định trạng thái lưu lượng đang bình thường chuyển sang trạng thái bị tấn công để bật cảnh báo ứng với một sự kiện cụ thể. Tuy nhiên đề xuất này còn nhiều giới hạn.

Năm 2011, ba tác giả G. Davanzo, E. Medvet và A. Bartoli đã đề xuất sử dụng kỹ thuật học máy để phát hiện cuộc tấn công an ninh mạng deface [4] "Anomaly detection techniques for a web defacement monitoring service": các website cần sát giám được xây dựng thành các hồ sơ (profile), dựa trên các kỹ thuật học máy và đưa ra cảnh báo khi nội dung website không phù hợp với hồ sơ đã có.

Trong bài báo [5] "An approach to Reveal Website Defacement" của ba tác giả Rajiv Kumar Gurjwa, Divya Rishi Sahu, Deepak Singh Tomar đã trình bày biện

pháp nhằm phát hiện tấn công an ninh mạng deface dựa vào việc sử dụng mã CRC 32, kỹ thuật hàm băm (hashing), tỷ suất nhiễu so với tín hiệu (Peak Signal to Noise Ratio - PSNR), cấu trúc tương đồng (Structural Similarity - SSIM) vào năm 2013.

Để phát hiện Website bị tấn công an ninh mạng deface, Ramniwas, Nikhil, và Deepak (2014) đề xuất phân tích tập tin ghi nhận sự thay đổi của website [6]: Các thông tin thay đổi sẽ được ghi nhận, xử lý và phân tích nhằm xác định website bị tấn công. Phương pháp này không được xem là hiệu quả đối với trường hợp tập tin nhật ký bị can thiệp trái phép. Khi này sẽ không kịp thời phát hiện website đang bị tấn công.

Ebot Enaw và Djoursoubo Pagou Prosper (2014) đề xuất sử dụng trí tuệ nhân tạo để phát hiện cuộc tấn công an ninh mạng tại bài báo [7]“A conceptual approach to detect web defacement through Artificial Intelligence”.

Một phương pháp nhằm phát hiện tấn công an ninh mạng deface thông qua ứng dụng hàm băm vào năm 2015 do Rashmi và Shahzia đề xuất. Xây dựng một môđun tích hợp vào máy chủ web để phát hiện cuộc tấn công trong các website. hệ thống này còn có khả năng tự thay đổi cấu hình để trang web bị tấn công không hiển thị. Đây là một phương pháp mới so với các phương pháp đã biết trước đây. Tuy nhiên nó cũng chỉ phù hợp cho các trang tĩnh [8].

Các tác giả Francesco Bergadano, Fabio Carretto, Fabio Cogno và Dario Ragno (2019) đề xuất phát hiện cuộc tấn công an ninh mạng deface thông qua biện pháp máy học đối thủ thụ động (Passive Adversaries)[9].

Trên Tạp chí khoa học Đại học Đà Lạt (Tập 8, Số 2, năm 2018), các tác giả Trần Đắc Tốt, Đặng Lê Nam, Phạm Nguyễn Huy Phương đề xuất xây dựng [11]“hệ thống cảnh báo tấn công thay đổi giao diện website” bằng việc kết hợp giám sát trong mạng nội bộ (Local Area Network - LAN) và giám sát từ xa. Ở từng thời điểm xác định, hệ thống sẽ tiến hành giám sát máy chủ, cơ sở dữ liệu, mã nguồn của website để phát hiện sự thay đổi bất hợp pháp.

CHƯƠNG 3: NỀN TẢNG LÝ THUYẾT LIÊN QUAN ĐẾN TẤN CÔNG AN NINH MẠNG DEFACE

Chương này trình bày những một số biện pháp tấn công an ninh mạng, khái niệm chung về tấn công an ninh mạng deface và thực trạng việc tấn an ninh mạng deface hiện nay.

Khái niệm về hàm băm và việc ứng dụng hàm băm để phát hiện sự tương đồng giữa 02 tài liệu.

3.1 Tổng quan về an ninh mạng

3.1.1 Tìm hiểu về an toàn thông tin và an ninh mạng

An ninh mạng là sự bảo đảm hoạt động trên không gian mạng không gây phương hại đến an ninh quốc gia, trật tự, an toàn xã hội, quyền và lợi ích hợp pháp của cơ quan, tổ chức, cá nhân. Nói cách khác, an ninh mạng là các biện pháp nhằm bảo vệ hệ thống điện tử khỏi bị xâm nhập hoặc tấn công. Một phần của an ninh mạng là xác định dữ liệu quan trọng là gì, ở đâu, mức độ rủi ro và công nghệ cần được khai triển để bảo vệ dữ liệu đó.

An toàn thông tin mạng là sự bảo vệ thông tin, hệ thống thông tin tránh bị truy nhập, sử dụng, tiết lộ, gián đoạn, sửa đổi hoặc phá hoại trái phép nhằm bảo đảm tính nguyên vẹn, tính bảo mật và tính khả dụng của thông tin.

Với sự phát triển của công nghệ thông tin, sự phổ biến của thiết bị di động và Internet, có nhiều cách để truy cập, khai thác tri thức và dữ liệu.. Dẫn đến các phương pháp bảo vệ dữ liệu cũng phải thay đổi để bắt kịp. Một máy tính không có kết nối mạng Internet, chỉ yêu cầu cài đặt phần mềm phòng chống mã độc. Tuy nhiên, đối với một máy tính có kết nối mạng thì yêu cầu về đảm bảo an toàn, an ninh thông tin cũng nhiều hơn: Các dư liệu trên máy tính đó có thể bị truy cập, bị thay đổi thậm chí bị xóa.

3.1.2 Sự cần thiết phải bảo vệ an toàn thông tin

Khi tham gia trên môi trường mạng Internet, các cá nhân, doanh nghiệp hay tổ chức nào cũng có những yếu tố, tài liệu quan trọng cần phải được bảo vệ như:

- Tài nguyên: Nguồn lực con người, hệ thống và đường truyền.

- Dữ liệu.

- Danh tiếng của công ty.

Cá cá nhân, tổ chức, nhất là doanh nghiệp sẽ hứng chịu nhiều thiệt hại khi gặp các sự cố về an toàn, an ninh mạng như là:

- Tổn thất về chi phí.

- Tổn thất về thời gian.

- Hệ thống hoạt động trì trệ, kém hiệu quả.

- Tổn thất đến danh dự, uy tín của doanh nghiệp.

- Mất cơ hội kinh doanh.

3.2 Một số lỗ hổng an ninh trên môi trường mạng

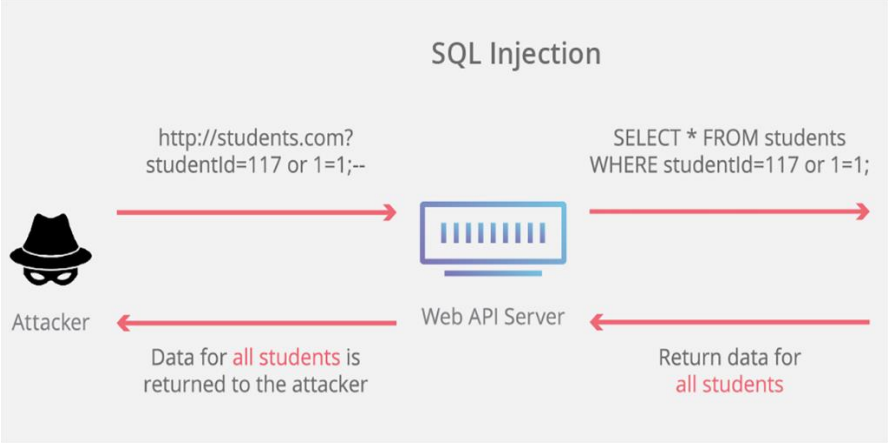
Trên môi trường mạng, lỗ hổng an ninh hay còn gọi là lỗ hổng bảo mật có thể được hiểu là các điểm yếu của hệ thống mà kẻ gian có thể khai thác, tấn công nhằm xâm nhập, điều khiển trái phép hệ thống. Các lỗ hổng bảo mật có thể xuất hiện ở dịch vụ, tiện ích như web, email, file... hoặc ở ngay trong các ứng dụng, ngay trên hệ điều hành.

3.3 Một số kỹ thuật tấn công và bảo mật website

3.3.1 Kỹ thuật tấn công SQL Injection

Tấn công SQL Injection là kiểu tấn công hệ thống bằng cách đưa vào các câu vấn tin SQL và thực thi bất hợp pháp. Bằng việc khai thác tính thiếu bảo mật, không xử lý dữ liệu do người dùng nhập vào. Hậu quả là thông tin, cơ sở dữ liệu sẽ bị lộ lọt hoặc bị điều chỉnh bởi bị kẻ gian khai thác.

Ngày nay, kỹ thuật tấn công SQL Injection là một trong những kiểu tấn công mạng khá phổ biến và hiệu quả, mặc dù đã được công bố, phát hiện từ rất lâu. Hậu quả của kỹ thuật này là nghiêm trọng do dữ liệu trên hệ thống có thể bị chỉnh sửa hoặc thậm chí bị xóa hẳn.



Hình 3.1: Mô hình tấn công SQL Injection

3.3.1.1 Hậu quả khi bị tấn công SQL Injection

Các cuộc tấn công SQL Injection có thể gây ra những hậu quả vô cùng nghiêm trọng với các tổ chức, doanh nghiệp là nạn nhân của chúng:

- Làm rò rỉ dữ liệu: Khi tấn công SQLi thành công, tin tặc có thể truy cập vào cơ sở dữ liệu, sau đó chỉnh sửa, xóa hoặc đánh cắp chúng. Thiệt hại mà doanh nghiệp phải chịu sẽ phụ thuộc vào mức độ quan trọng của dữ liệu bị rò rỉ.

- Ảnh hưởng đến lợi ích của khách hàng ở các dịch vụ khác: Do thói quen sử dụng một mật khẩu cho nhiều tài khoản của khách hàng. Như vậy, chỉ cần mật khẩu của một tài khoản bị lộ thì các tài khoản khác cũng gặp rủi ro mất an toàn.

- Làm giảm uy tín của doanh nghiệp: Uy tín và hình ảnh của doanh nghiệp sẽ bị ảnh hưởng nghiêm trọng sau khi thông tin về sự cố bị phát ra ngoài. Khách hàng rời bỏ hoặc không sử dụng dịch vụ doanh nghiệp cung cấp. Thay vào đó sẽ tìm, sử dụng các sản phẩm, dịch vụ của doanh nghiệp khác hoặc đối thủ. Giảm doanh thu là hậu quả tất yếu mà ai cũng có thể nhìn thấy.

3.3.1.2 Các hình thức tấn công SQL Injection

- Tấn công SQL Injection bằng hình thức tham số:

Giả sử chúng ta có một biểu để tìm kiếm các sản phẩm theo mã định danh (ID) trên trang web. Đoạn mã để tìm kiếm sản phẩm sẽ như sau:

```
$prod_id = $_GET["prod_id"];
$sql = "SELECT * FROM Products WHERE product_id = " . $prod_id;
```

Dự kiến, khi người dùng nhập vào một số bất kỳ là mã định danh sản phẩm, giả sử là “20”, thì đoạn vấn tin SQL sẽ có dạng như sau:

```
SELECT * FROM Products WHERE product_id = 20
```

Tuy nhiên, nếu người dùng nhập vào một mã định danh có dạng chuỗi là “20; DROP TABLE Products;” thì đoạn vấn tin SQL sẽ có dạng như sau:

```
SELECT * FROM Products WHERE product_id = 20; DROP TABLE Products;
```

Kết quả khi thực thi đoạn vẫn tin này thì bảng “Products” sẽ bị xóa khỏi cơ sở dữ liệu.

- Tấn công SQL Injection bằng hình thức sử dụng điều kiện luôn đúng:

Hình thức này cho phép dữ liệu luôn lấy được cho dù người dùng có nhập bất cứ thông tin nào. Xét một website đăng nhập, đang yêu cầu người dùng phải cung cấp thông tin đăng nhập như sau:

```
$username = $_POST["username"];
$password = $_POST["password"];

$sql = "SELECT * FROM Users WHERE username = \"" . $username . "\" AND
password = \"" . $password . "\"";
```

Khi người dùng nhập vào tên đăng nhập là “admin” và mật khẩu là “12345678” thì đoạn vấn tin hoạt động bình thường và cho phép đăng nhập nếu đúng thông tin trên cơ sở dữ liệu

```
SELECT * FROM Users WHERE username = "admin" AND password = "12345678"
```

Tuy nhiên, khi người dùng nhập vào tên người dùng là “ invalid_user" OR "1"="1 ” và mật khẩu là “ invalid_pass" OR "1"="1 ” thì đoạn vấn tin sẽ là

```
SELECT * FROM Users WHERE username = "invalid_user" OR "1"="1" AND
password = "invalid_pass" OR "1"="1"
```

Do "1"="1" luôn trả kết quả đúng nên bất kể người dùng cung cấp thông tin đăng nhập là gì thì cũng đăng nhập được vào hệ thống.

3.3.1.3 Cách phòng chống tấn công SQL Injection

Một số biện pháp dưới đây dùng để hạn chế tối đa rủi ro bị tấn công SQL Injection gồm:

- Dữ liệu nhập vào do người dùng cung cấp phải được kiểm tra, xử lý: Dữ liệu luôn phải được xác thực, phải được xử lý trước các câu lệnh SQL được thực thi. Hạn

chế sử dụng các từ khóa SELECT, UNION khi nhập dữ liệu hoặc dùng các bộ lọc thích hợp.

- Không thực hiện cộng chuỗi để tạo SQL: Hãy sử dụng parameter thay vì cộng chuỗi. Nếu câu lệnh SQL đầu vào có dấu hiệu khác thường, SQL Engine (modun chuyển đổi các câu lệnh SQL) sẽ phát hiện lỗi bất thường này. Doanh nghiệp không cần dùng code để check.

- Sao lưu dữ liệu thường xuyên: Dữ liệu cần được sao lưu thường xuyên để trong trường hợp xấu nhất là bị tin tặc xóa thì doanh nghiệp vẫn có thể khôi phục.

- Không hiển thị các mã lỗi, các thông báo liên quan lỗi: Dựa vào thông tin của các message lỗi, tin tặc có thể để tìm ra cấu trúc database. Vì vậy, khi có lỗi, trang web chỉ nên hiển thị thông báo lỗi (không hiển thị thông tin nhạy cảm về lỗi) để tránh bị tin tặc lợi dụng.

- Phân quyền truy cập rõ ràng: Việc cho phép mọi tài khoản đều được truy cập vào cơ sở dữ liệu tiềm ẩn nhiều rủi ro. Hãy chỉ định một số tài khoản nhất định có quyền kết nối với cơ sở dữ liệu. Việc này sẽ giúp hạn chế các lệnh SQL được thực thi tự động trên server.

3.3.2 Tấn công XSS (Cross Site Scripting)

3.3.2.1 Khái niệm về XSS

XSS là viết tắt của Cross-Site Scripting, đây là một kĩ thuật tấn công mà kẻ tấn công sẽ chèn vào các website động những đoạn mã có thể thực thi một câu lệnh bất hợp pháp. Kỹ thuật tấn công khá phổ biến và có hiệu quả cao tương tự như kỹ thuật tấn công SQL Injection. Các đoạn mã đó thường được viết bằng JavaScript, DHTML, JScript và cũng có thể là cả các thẻ HTML. Kỹ thuật này nhận được nhiều sự quan tâm của nhà quản trị web, người dùng và kẻ gian.

Mục đích chính của cuộc tấn công này là ăn cắp dữ liệu nhận dạng của người dùng như: session tokens, cookies của người khác và các thông tin khác. Như chúng ta biết, cookie giúp người sử dụng thuận tiện hơn để đăng nhập tự động. Khi cookie bị đánh cắp, kẻ gian có thể sử dụng để đăng nhập vào hệ thống một cách trái phép. Đây là một trong những lý do tại sao kiểu tấn công này được coi là một trong những kiểu tấn công nguy hiểm, gây ra nhiều thiệt hại nhất. Những trang web nổi tiếng như ebay.com, fbi.gov, yahoo.com cũng từng bị xâm nhập bởi kỹ thuật tấn công XSS.

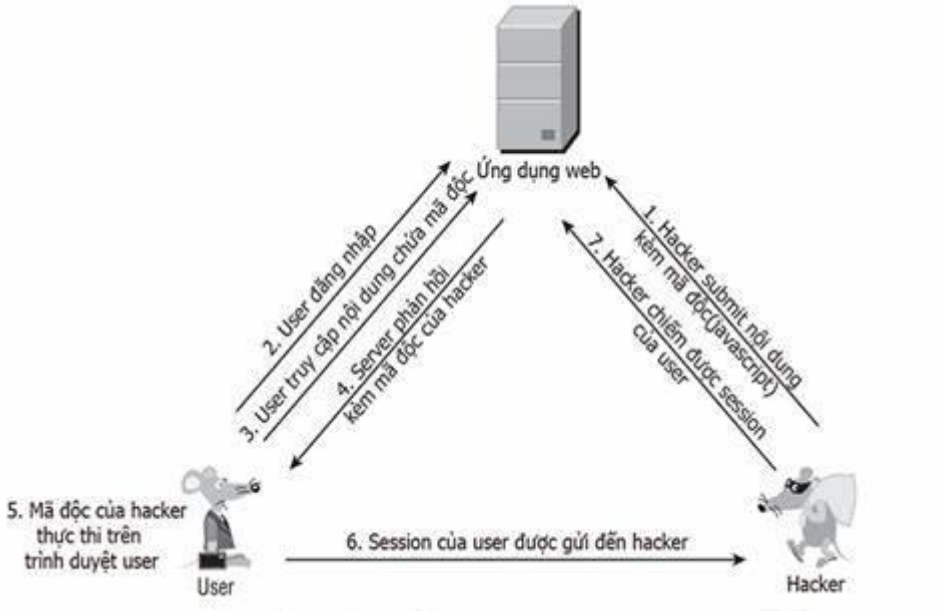
Tấn công XSS đang được thực hiện ở phía client. Nó có thể được thực hiện với các ngôn ngữ lập trình phía client khác nhau. Tuy nhiên, thường xuyên nhất cuộc tấn công này được thực hiện với Javascript và HTML.

Tấn công XSS nói chung được chia làm 3 loại chính là Phản xạ (Reflected), Lưu trữ (Stored) và DOM based:

- Tấn công XSS phản xạ (Reflected XSS): Là hình thức tấn công XSS được sử dụng nhiều nhất trong chiếm phiên làm việc của người dùng mạng. Qua đó, hacker đánh cắp các dữ liệu người dùng, chiếm quyền truy cập và hoạt động của họ trên website thông qua việc chia sẻ địa chỉ URL chứa mã độc và chờ nạn nhân 'cắn câu'. Hình thức tấn công này thường nhắm vào một số ít nạn nhân.

- Tấn công XSS lưu trữ (Stored XSS): Không giống như Reflected XSS, Stored XSS nhắm đến khá nhiều nạn nhân cùng lúc. Với hình thức tấn công này, hacker chèn các mã độc vào database thông qua những dữ liệu đầu vào như form, input, textarea…Khi người dùng mạng truy cập website và tiến hành những thao tác liên quan đến các dữ liệu đã lưu, mã độc lập tức hoạt động trên trình duyệt của người dùng.

- Tấn công XSS DOM Based: Dạng tấn công XSS thường gặp cuối cùng đó là DOM Based XSS. Kỹ thuật này dựa vào sự thay đổi các cấu trúc DOM (do thay đổi HTML) .



Hình 3.2: Kịch bản tấn công XSS

3.3.2.2 Biện pháp phòng ngừa, chống tấn công XSS

Như mô tả đã nêu, kỹ thuật tấn công XSS là khá nguy hiểm nhưng để ngăn ngừa cũng không quá khó khăn. Có nhiều cách có thể giải quyết vấn đề này như sau:

- Chỉ chấp nhận những dữ liệu hợp lệ.

- Từ chối nhận các dữ liệu hỏng.

- Liên tục kiểm tra và thanh lọc dữ liệu.

Trong thực tế, do bộ lọc không phù hợp, không bao quát hết nên có nhiều trường hợp phải chấp nhận tất cả mọi dữ liệu đầu vào. Vì vậy cần phải có những cách riêng để giải quyết: Trước khi hiển thị ra website, chúng ta dùng cách mã hóa các ký tự đặt biệt. Đối với nhiều ngôn ngữ Web Application (PHP, ASP...) đây là một biện pháp hiệu quả và có thể áp dụng.

3.3.3 Tấn công từ chối dịch vụ DOS (Denial of Service)

3.3.3.1 Khái niệm về tấn công từ chối dịch vụ DOS

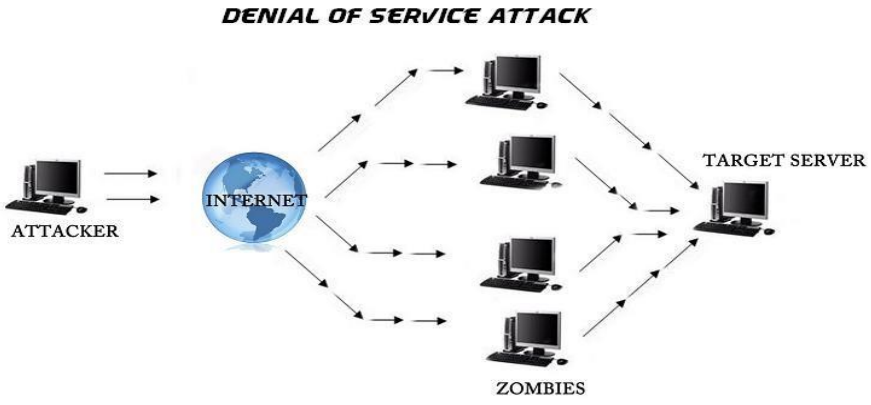
Đây là một kiểu tấn công nhằm làm cho hệ thống không thể sử dụng hoặc làm cho hệ thống phản ứng chậm hơn một cách đáng kể so với trạng thái bình thường. Nguyên nhân là các tài nguyên của hệ thống bị sử dụng quá tải. Hậu quả của cuộc tấn công là các dịch vụ do hệ thống đó cung cấp sẽ bị gián đoạn, không thể được truy cập, khai thác, gây tổn thất cho doanh nghiệp:

- Tiêu tốn tài nguyên tính toán như băng thông, dung lượng đĩa cứng hoặc thời gian xử lý.

- Phá vỡ các thông tin cấu hình (như thông tin định tuyến); các trạng thái thông tin (việc tự động reset lại các phiên làm việc Transmission Control Protocol).

- Phá vỡ các thành phần vật lý của mạng máy tính.

- Thông tin liên lạc bị gián đoạn, bị tắc nghẽn.



Hình 3.3: Mô hình tấn công DOS

3.3.3.2 Các biện pháp phòng chống tổng quát

Để phòng, chống một cuộc tấn công DOS tiềm ẩn thường chúng ta cần quan tâm các giai đoạn như sau:

- Giai đoạn ngăn ngừa: Tối thiểu hóa lượng Agent, tìm và vô hiệu hóa các Handler.

- Giai đoạn đối đầu: Khi hệ thống đang bị cuộc tấn công DOS, cần ngăn chặn hoặc chuyển hướng cuộc tấn công.

- Giai đoạn sau khi tấn công: Triển khai các bước thu thập, phân tích dữ liệu để xác định nguồn tấn công, chứng cứ, kinh nghiệp.

3.4 Về tấn công an ninh mạng deface

Tấn công Deface (Website Defacement) là hình thức tấn công làm thay đổi giao diện trực quan của một trang web. Đây là hành động của những hacker chuyên bẻ khóa hệ thống. Chúng đột nhập vào máy chủ web và thay thế trang web được host bằng giao diện trang web của riêng chúng. Hình thức tấn công Deface phổ biến nhất là sử dụng SQL Injection để đăng nhập vào tài khoản admin.

Mục đích của cuộc tấn công thay đổi nội dung là:

- Mục đích xấu: Đăng các nội dung bôi xấu nạn nhân hoặc lan truyền các quan điểm chống phá nhà nước…

- Mục đích cá nhân: Chứng tỏ năng lực của bản thân hoặc của nhóm hacker.

Các hậu quả của tấn công an ninh mạng deface gồm [8]:

- Thay đổi một phần hoặc toàn bộ nội dung của trang web.

- Thay đổi mã nguồn của trang web.

- Chuyển hướng của trang web.

- Hủy hoặc xóa toàn bộ trang web.

3.4.1 Nguyên nhân website bị tấn công an ninh mạng deface

Một website thường bị tấn công là do còn tồn tại các lỗ hổng bảo mật nghiêm trọng cho phép kẻ gian có thể khai thác. Bên cạnh đó còn có các nguyên nhân như sau:

- Mật khẩu của tài khoản quản trị yếu, dễ đoán: Độ dài ký tự ngắn, không sử dụng ký tự viết hoa, ký tự đặc biệt, chưa có biện pháp chống tấn công bruce force. Mật khẩu của tài khoản quản trị không được thay đổi định kỳ.

- Cài đặt các module, plugin, extension,... trong các mã nguồn mở hiện nay (thường là các website joomla, wordpress,...).

- Để lộ mật khẩu quản trị....

3.4.2 Dấu hiệu nhận biết website bị tấn công an ninh mạng deface

Thông thường, khi các trang mặc định như: home.html, trangchu.html, default.html… bị thay đổi nghĩa là website đã bị tấn công deface. Tuy nhiên, nếu tin tặc không thay đổi nội dung của những file trên thì hơi khó phát hiện. Trong trường hợp này, doanh nghiệp có thể nhận được cảnh báo từ nhà cung cấp hosting.

3.4.3 Tình hình về tấn công an ninh mạng deface

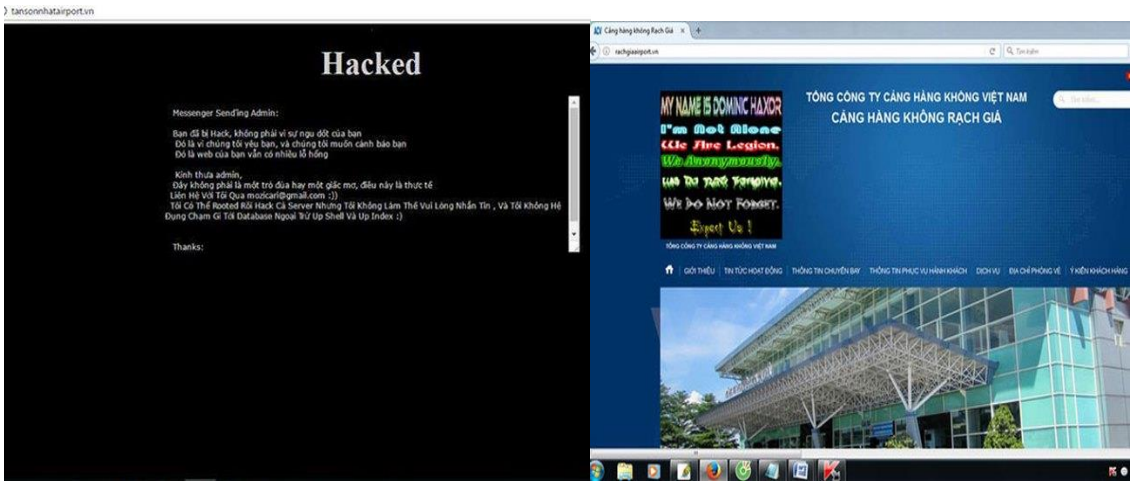
Theo Bộ thông tin và Truyền thông thống kê cho thấy trong tháng 9/2021, tại Việt Nam đã xuất hiện 1.074 cuộc tấn công an ninh mạng, giảm 6,45% so với tháng trước đó. Qua phân loại, có 192 cuộc Phishing, 743 cuộc Malware và 139 cuộc tấn công an ninh mạng Deface.

Tính đến Quý III/2021, cơ quan này đã ghi nhận 3.241 cuộc tấn công mạng (547 cuộc Phishing, 579 cuộc Deface, 2.115 cuộc Malware), tăng 57,03% so với cùng kỳ Quý III/2020, tăng 97,14% so với Quý II/2021.

Lũy kế 9 tháng của năm 2021, con số này là 6.156 cuộc tấn công mạng (1.404 cuộc Phishing, 1.109 cuộc Deface, 3.643 cuộc Malware), tăng 30,15% so với cùng kỳ 9 tháng đầu năm 2020.

Trong dịp Tết Nguyên đán Nhâm Dần 2022 (tính từ ngày 29/1 đến 5/2), các hệ thống cảnh báo, giám sát an toàn không gian mạng đã ghi nhận và hướng dẫn cơ quan, tổ chức khắc phục hơn 240 sự cố tấn công mạng vào các hệ thống thông tin tại Việt Nam gồm: 180 cuộc Phishing (74%), 59 cuộc tấn công cài mã độc và 01 cuộc tấn công an ninh mạng deface.

Đến tháng 1/2022, Bộ thông tin và Truyền thông tiếp tục phát hiện 1.383 cuộc tấn công mạng tại Việt Nam. Báo cáo của Bộ thống kê có 197 cuộc tấn công giả mạo (phishing), 125 cuộc tấn công an ninh mạng deface, 1.061 cuộc mã độc (malware), tăng 10,29% so với tháng 12/2021.



Hình 3.4: Website của sân bay Tân Sơn Nhất và Rạch Giá bị tấn công thay đổi nội dung ngày 08 và 09/03/2017

Hình 3.5: Website của Sở Khoa học Công nghệ Bà Rịa - Vũng Tàu bị tấn công thay đổi nội dung ngày 05/02/2017

Một số website khác của tỉnh Bà Rịa - Vũng Tàu bị hacker tấn công vào ngày 05/02/2017 như là:

- http://dkqm.sotttt.baria-vungtau.gov.vn/NcPro.html

- http://motcuacapxa.baria-vungtau.gov.vn/NcPro.html

- http://sokhcn.baria-vungtau.gov.vn/NcPro.html

- http://dost.baria-vungtau.gov.vn/NcPro.html

- http://daotaonghe.soldtbxh.baria-vungtau.gov.vn/NcPro.html

3.5 Hàm băm

3.5.1 Khái niệm hàm băm

Hàm băm là hàm chuyển đổi một thông điệp có độ dài bất kỳ thành một dãy ký tự (hoặc dãy bit) nhưng có độ dài cố định. Tuy nhiên, khi thực thi hàm băm trong thực tế, thông điệp vào của các hệ thống thường có được tách ra thành các thông điệp

nhỏ hơn, có chiều dài xác định. Kết quả biến đổi của hàm băm sẽ tạo ra dãy có n (n>0) ký tự hoặc bit và được gọi là giá trị băm hay mã băm.

Giá trị băm của một tài liệu có thể xem là dấu hiệu nhận dạng (hay còn gọi là dấu vân tay) của tài liệu đó. Nhờ có nó ta có thể kiểm tra rằng một văn bản là toàn vẹn, chính xác và không bị sửa đổi trong quá trình truyền tải hoặc kiểm tra tính ổn định.

3.5.2 Tính chất và yêu cầu của hàm băm

Một số tính chất cơ bản của hàm băm là:

- Tính tất định: Cùng một thông điệp đầu vào luôn tạo ra cùng một giá trị băm. Nói cách khác, khi có sự thay đổi nào trong thông điệp đầu vào cũng sẽ tạo ra một giá trị băm khác so với giá trị băm trước đó.

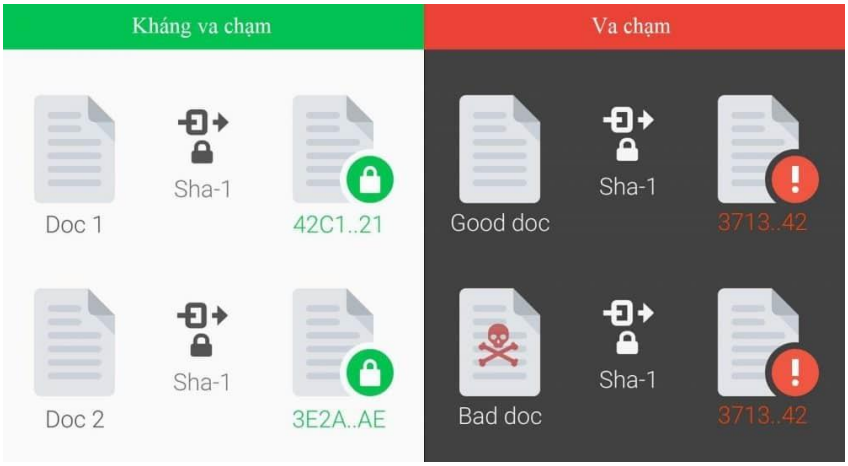
- Tính một chiều: Khi nhận được giá trị băm thì không thể (hoặc khó) có thể suy ra được thông điệp đầu vào (hay còn gọi là tiền ảnh). Tính một chiều này là tính chất quan trọng nhất của hàm băm, làm cho hàm băm được xem là công cụ cơ bản của mật mã hiện đại.

- Tính hiệu quả: Là khả năng tính toán nhanh chóng giá trị băm của bất kỳ thông điệp nào.

Yêu cầu đối với một hàm băm tốt, an toàn:

- Tính toán nhanh.

- Tính kháng va chạm: Không thể tính toán để tìm được hai đầu vào khác nhau x và x’ bất kỳ mà có cùng đầu ra, tức là thỏa mãn h(x) = h(x’). Hoặc xác suất trùng mã băm nhỏ.



Hình 3.6: Tính kháng va chạm của hàm băm

3.5.3 Một số hàm băm phổ biến

Tính đến nay, trong nhiều ứng dụng khác nhau (quân sự, truyền tin, xác thực...) đã có nhiều hàm băm được tạo ra.

Thuật toán	Kích thước đầu ra (*output size*)	Kích thước trạng thái trong (*Internal state size*)	Kích thước khối (*Block size*)	Độ dài (*Length size*)	Kích thước word (*Word size*)	Xung đột (*Collision*)
HAVAL	256/224/192/160/128	256	1024	64	32	Có
MD2	128	384	128	Không	8	khả năng lớn
MD4	128	128	512	64	32	Có
MD5	128	144	122	88	88	Có
PANAMA	256	8736	256	No	32	Có lỗi
RIPEMD	128	128	512	64	32	Có
RIPEMD-128/256	128/256	128/256	512	64	32	Không
RIPEMD-160/320	160/320	160/320	512	64	32	Không
SHA-0	160	160	512	64	32	Không
SHA-1	160	160	512	64	32	Có lỗi
SHA-256/224	256/224	256	512	64	32	Không
SHA-512/384	512/384	512	1024	128	64	Không
Tiger(2)-192/160/128	192/160/128	192	512	64	64	Không
VEST-4/8 (hash mode)	160/256	256/384	8	80/128	1	Không[1]
VEST-16/32 (hash mode)	320/512	512/768	8	160/256	1	Không
WHIRLPOOL	512	512	512	256	8	Không

Hình 3.7: Một số hàm băm phổ biến

- Hàm băm MD5: Hàm này do Ronald Rivest công bố vào năm 1991 (thay cho hàm băm MD4) và được định làm tiêu chuẩn là RFC 1321 năm 1992. MD5 tạo ra một bản tóm tắt có kích thước 128 bit (16 byte). Tuy nhiên, đến đầu những năm 2000 thì hàm băm MD5 trở lên không an toàn trước sức mạnh tính toán của các hệ

thống tính toán thế hệ mới. Với sức mạnh tính toán và sự phát triển của công nghệ thám mã thời gian gần đây, chúng ta có thể tính toán các va chạm trong MD5 với độ phức tạp 221 phép toán chỉ trong vòng vài giây khiến thuật toán không phù hợp với hầu hết các trường hợp sử dụng trong thực tế.

- Hàm băm SHA-1: SHA-1 được phát triển như một phần của dự án Capstone của Chính phủ Hoa Kỳ. Phiên bản đầu tiên, thường được gọi là SHA-0 được xuất bản năm 1993 với tiêu đề Secure Hash Standard, FIPS PUB 180, bởi NIST (Viện Tiêu chuẩn và Công nghệ Quốc gia Hoa Kỳ). Nó đã bị NSA rút lại ngay sau khi xuất bản và được thay thế bởi phiên bản sửa đổi, được xuất bản năm 1995 trong FIPS PUB 180-1 và thường được đặt tên là SHA-1. SHA-1 tạo ra bản tóm tắt có kích thước 160 bit (20 byte). Các va chạm chống lại thuật toán SHA-1 đầy đủ có thể được tạo ra bằng cách sử dụng kỹ thuật tấn công phá vỡ. Cho đến nay, hàm băm này được coi là không đủ an toàn, không phải là một hàm băm tốt.

- Hàm băm RIPEMD-160: Viết tắt của từ RACE Integrity Primitives Evaluation Message Digest là họ hàm băm được phát triển tại Leuven (Bỉ) bởi ba nhà mật mã học Hans Dobbertin, Antoon Bosselaers và Bart Preneel của nhóm nghiên cứu COSIC thuộc đại học Katholieke Universiteit Leuven. RIPEMD lần đầu tiên được công bố vào năm 1996 dựa trên các nguyên tắc thiết kế được sử dụng trong MD4. RIPEMD-160 tạo ra một bản tóm tắt gồm 160 bit (20 byte) và có hiệu năng tương tự như SHA-1 nhưng ít được phổ biến hơn. Và cho đến nay RIPEMD-160 chưa bị phá vỡ.

- Hàm băm SHA-2: Được công bố đầu tiên vào năm 2001, là một tập hợp các hàm băm mật mã được thiết kế bởi Cơ quan an ninh quốc gia Hoa Kỳ (NSA). Chúng được xây dựng bằng cấu trúc Merkle–Damgård, chức năng nén một chiều của nó được xây dựng bằng cấu trúc Davies–Meyer từ một hệ mật mã khối chuyên dụng. Hàm băm SHA-2 thực ra bao gồm hai thuật toán băm là SHA-256 và SHA-512. SHA-224 là một biến thể của SHA-256 với các giá trị khởi tạo và đầu ra bị cắt bỏ khác nhau. SHA-384 và SHA-512/224 và SHA-512/256 ít được biết đến hơn biến thể của

SHA-512. SHA-512 an toàn hơn SHA-256 và thường nhanh hơn SHA-256 trên các máy 64 bit như AMD64. Do có nhiều phiên bản thuật toán khác nhau do đó kích thước đầu ra của họ SHA-2 cũng khác nhau tùy theo thuật toán. Phần mở rộng của tên phía sau tiền tố "SHA" chính là độ dài của thông điệp băm đầu ra. Ví dụ với SHA-224 thì kích thước đầu ra là 224 bit (28 byte), SHA-256 tạo ra 32 byte, SHA-384 tạo ra 48 byte và cuối cùng là SHA- 512 tạo ra 64 byte. Bitcoin sử dụng hàm băm SHA-256 là một phiên bản trong họ SHA-2 này.

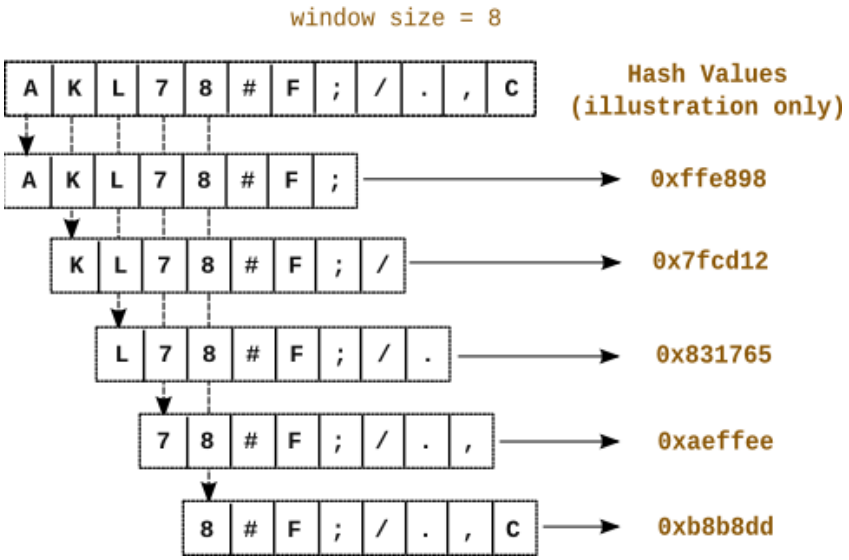
- Hàm băm SHA-3: SHA-3 được NIST phát hành vào ngày 5 tháng 8 năm 2015. Đây có lẽ là tiêu chuẩn hàm băm mới nhất cho đến hiện nay. SHA-3 là một tập con của họ nguyên thủy mật mã rộng hơn là Keccak. Thuật toán Keccak được đưa ra bởi Guido Bertoni, Joan Daemen, Michael Peeters và Gilles Van Assche. Keccak dựa trên cấu trúc bọt biển (sponge). Cấu trúc này cũng có thể được sử dụng để xây dựng các nguyên thủy mã hóa khác như các hệ mật mã dòng. SHA-3 cũng có các kích cỡ đầu ra tương tự như SHA-2 bao gồm: 224, 256, 384 và 512 bit.

- Hàm băm BLAKE2: Một phiên bản cải tiến của BLAKE có tên BLAKE2 đã được công bố vào ngày 21 tháng 12 năm 2012. BLAKE được phát triển bởi Jean-Philippe Aumasson, Samuel Neves, Zooko Wilcox-O'Hearn và Christian Winnerlein với mục tiêu thay thế các thuật toán băm phổ biến như MD5 và SHA-1. Khi chạy trên các kiến trúc 64 bit x64 và ARM, BLAKE2b cho tốc độ nhanh hơn SHA-3, SHA-2, SHA-1 và MD5. Mặc dù BLAKE và BLAKE2 chưa được tiêu chuẩn hóa như SHA-3, nhưng nó đã được sử dụng trong nhiều giao thức bao gồm hàm băm mật khẩu Argon2 do hiệu quả cao mà nó mang lại cho các dòng CPU hiện đại. Do BLAKE cũng là ứng cử viên cho tiêu chuẩn SHA-3, vì vậy, BLAKE và BLAKE2 đều có các kích thước đầu ra giống như SHA-3 và có thể tùy chọn khi sử dụng trong thực tế.

3.5.4 Thuật toán Rabin-Karp

Thuật toán Rabin-Karp là một thuật toán được sử dụng để tìm kiếm hoặc đối sánh các mẫu trong đoạn văn bản. Phương pháp của thuật toán này là dựa vào việc so

sánh các giá trị băm của mẫu hoặc chuỗi cần tìm: Hai chuỗi sẽ được xem là giống nhau nếu có cùng một giá trị băm. Thuật toán này rất hiệu quả trong việc so sánh hoặc tìm nhiều mẫu trong một chuỗi: Chuỗi con tiếp theo được tính toán dựa vào giá trị băm của chuỗi trước đó. Thuật toán Rabin-Karp tạo ra một giá trị băm từ chuỗi con trong các trang web bởi tính nhanh và dễ để thực thi.



Hình 3.8: Thuật toán Rabin-Karp

3.5.5 Thuật toán Rabin-Karp cải tiến

Đầu vào: Tài liệu (trang web công khai)

Đầu ra: Thực hiện băm để lấy dấu vân tay tài liệu.

Bước 1: Bắt đầu.

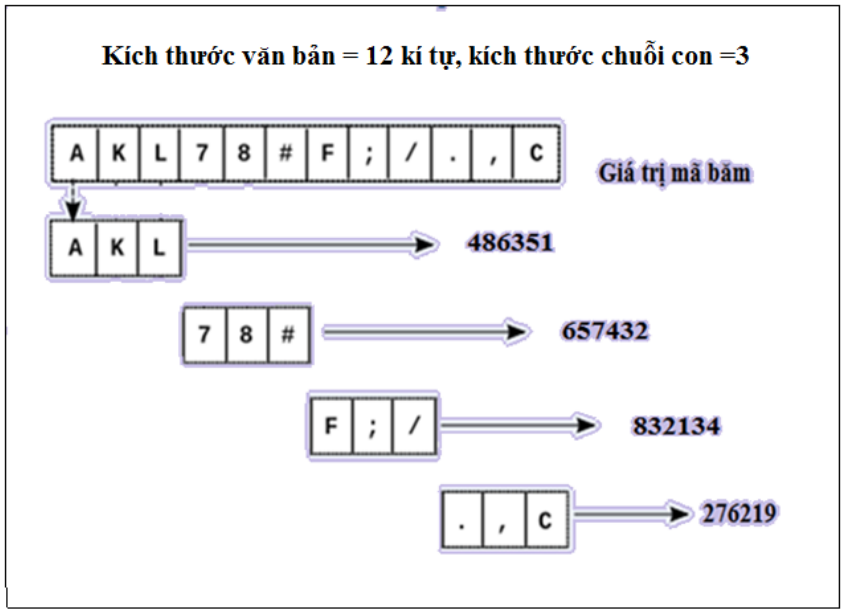
Bước 2: Xóa các ký đặc biệt, luôn xuất hiện theo quy ước của ngôn ngữ lập trình.

Bước 3: Chia văn bản nhận được thành các chuỗi con có n ký tự.

Bước 4: Tính mã băm cho các chuỗi con.

Bước 5: Lưu lại các giá trị băm.

Bước 6: Kết thúc.



Hình 3.9: Thuật toán Rabin-Karp cải tiến

3.6 Thuật toán đối sánh chuỗi

Đối sánh chuỗi là việc so sánh một hoặc vài chuỗi (thường được gọi là mẫu hoặc pattern) với toàn bộ văn bản để tìm ra nơi và số lần xuất hiện của chuỗi đó trong văn bản.

3.6.1 Phân loại thuật toán đối sánh chuỗi

- Theo thứ tự đối sánh:

+ Từ trái sang phải

+ Từ phải sang trái

+ Đối sánh tại vị trí cụ thể

+ Không theo thứ tự nhất định

- Theo số lượng pattern:

+ Đối sánh chuỗi đơn pattern

+ Đối sánh đa chuỗi pattern

- Theo độ sai khác đối sánh:

+ Đối sánh chuỗi chính xác

+ Đối sánh chuỗi gần đúng

- Theo sự thay đổi của pattern và văn bản

+ Pattern thay đổi, văn bản cố định

+ Pattern cố định, văn bản thay đổi

+ Pattern thay đổi, văn bản thay đổi

3.6.2 Dấu vân tay tài liệu (Document Fingerprint)

Trong khoa học máy tính, dấu vân tay nhận dạng duy nhất dữ liệu gốc cho tất cả các mục đích thực tiễn giống như là việc nhận dạng duy nhất dấu vân tay người trong thực tế. Dấu vân của tài liệu là tập hợp các mã được sinh ra từ các khóa nội dung của tài liệu đó. Mỗi mã đó được gọi là một giá trị băm.

3.7 Ứng dụng thuật toán Rabin Karp để so sánh tìm độ tương đồng của 02 tài liệu

3.7.1 Các bước tiền xử lý trước khi thực hiện băm tài liệu

Có 04 bước như sau [11]:

Bước 1-Mã hóa (Tokenizing): Chuyển tài liệu cần băm thành các nhóm ký tự và từ viết hoa sang viết thường.

Bước 2-Loại bỏ các từ thường xuất hiện (Stopword Removal): Bởi vì, với, và, hoặc,...

Bước 3-Đưa các từ có tiền tố (prefix) hoặc hậu tố (suffix) trở lại các từ gốc.

Bước 4-Thực hiện băm tài liệu.

3.7.2 Ví dụ tính mã băm của chuỗi "MEDAN"

Chọn hệ số K-Gram=5; hệ số mũ B (basis)=7; Chuỗi A = MEDAN

A (1) = 77; A(2) = 69; A(3) = 68; A(4) = 65; A(5) = 78

Giá trị băm = $(77 \times 7^5) + (69 \times 7^4) + (68 \times 72^3) + (65 \times 7^2) + (78 \times 7^1)$

= 1.486.863

3.7.3 Ví dụ tính mã băm của một tài liệu

- Giả sử chọn hệ số K-Gram=5, tính giá trị băm của 5 ký tự đầu tiên. Sau đó tính giá trị băm của 5 ký tự kế tiếp: bỏ ký tự đầu tiên và thêm một ký tự kế tiếp. Ví dụ, văn bản là "123456789", ta sẽ tính bảng băm của chuỗi như sau:

Hash (12345)

Hash (23456)

Hash (34567)

Hash (45678)

Hash (56789)

- Giả sử ta có 02 bảng giá trị băm như hình dưới đây của 02 tài liệu [11]:

Bảng 3.1: Bảng giá trị băm của 02 tài liệu

19875	16830	23124	17433	20546
21489	26753	13498	23846	16528
21848	28447	29994	10301	13009
18832	27217	23157	25854	22492
14952	14337	29348	19978	28809
13485	14188	13131	21215	12053
25669	13809	26508	19455	25356
29964	17723	2663	17445	11803
19477	27142	24814	15155	26266
28432	19007	21896	16625	20681

28432	26406	28424	13930	19187
18049	10867	18516	26753	19975
10152	13053	24120	21896	18351
12605	25101	21215	20750	15513
22949	26006	25045	25932	10695
13254	21504	20286	22492	10615
25565	29941	17403	23018	22666
19744	19769	19877	29535	13139
25669	16830	14297	20916	24640
16960	20681	13131	13009	18947

Mỗi bảng này có 50 giá trị băm, trong đó có 10 giá trị băm là giống nhau. Khi này mức độ tương đồng của 02 tài liệu được tính bằng công thức sau:

$$P = \frac{2\ x\ SH}{THA + THB}\ x\ 100$$

Với P: mức độ tương đồng.

SH (Identical Hash): Số giá trị băm giống nhau.

THA (Total Hash in DocumentA): Tổng số giá trị băm của tài liệu A

THB (Total Hash in DocumentB): Tổng số giá trị băm của tài liệu B

Khi này,

$$P = \frac{2\ x\ 10}{50 + 50}\ x\ 100$$

$$= \frac{20}{100}\ x\ 100$$

$$= 20\%$$

3.7.4 Ví dụ về tính mã băm của 02 chuỗi với hệ số K-Gram=5, hệ số B=7

Chuỗi 1= “MEDAN**CDE**MABCC”

Chuỗi 2= “MEDAN**ZXY**MABCC”

Ta có 02 bảng băm như sau

Bảng 3.2: Bảng giá trị băm của 02 chuỗi với hệ số K-Gram=5, hệ số B=7

1	2	3	4	5	6	7	8	9
1486863	1349537	1329454	1306529	1499057	1317232	1338603	1370558	**1476594**

1	2	3	4	5	6	7	8	9
1486863	1349698	1330721	1315538	1562120	1758673	1722763	1706698	**1476594**

Mỗi bảng băm có 9 phần tử, trong đó 2 phần tử có giá trị giống nhau. Vậy, mức độ tương đồng là:

P = (2 x 2) / (9+9) x 100% = 22,2%

3.7.5 Ví dụ về tính mã băm của 02 chuỗi với hệ số K-Gram=3, hệ số B=7

Chuỗi 1= “MEDAN**CDE**MABCC”

Chuỗi 2= “MEDAN**ZXY**MABCC”

Ta có 02 bảng băm như sau

Bảng 3.3: Bảng giá trị băm của 02 chuỗi với hệ số K-Gram=3, hệ số B=7

1	2	3	4	5	6	7	8	9	10	11	12
4256	**3857**	**3787**	**3731**	4291	3759	3815	3920	**4228**	**3647**	**3703**	**3752**

1	2	3	4	5	6	7	8	9	10	11	12
4256	**3857**	**3787**	**3731**	4452	5026	4935	4900	**4228**	**3647**	**3703**	**3752**

Mỗi bảng băm có 12 phần tử, trong đó 8 phần tử có giá trị giống nhau. Vậy, mức độ tương đồng là:

P = (2 x 8) / (12+12) x 100% = 66,6%

CHƯƠNG 4: ĐỀ XUẤT BIỆN PHÁP NHẰM PHÁT HIỆN CUỘC TẤN CÔNG AN NINH MẠNG DEFACE

Một website bao gồm các thành phần chính như là Domain (tên miền), Host, Source (mã nguồn) và Database (cơ sở dữ liệu). Tương ứng với mỗi thành phần là biện pháp tấn công khác nhau. Có thể kể ra như là Kiểm soát truy cập website, chiếm hữu phiên làm việc, lợi dụng các thiếu sót trong việc kiểm tra dữ liệu nhập hợp lệ hay để lộ thông tin, từ chối dịch vụ… Trên cơ sở đó, học viên đề xuất một số biện pháp để phát hiện sự thay đổi của website như sau:

4.1 Biện pháp giám sát việc thay đổi nội dung của website

Như tên gọi của cuộc tấn công này, mục tiêu của tin tặc nhằm thay đổi nội dung trình bày tại trang chủ của website để thông báo hệ thống đã bị xâm nhập. Đây là cơ chế giám sát website từ xa, hướng từ bên ngoài:

Bước 1 - Khai báo thông tin của website: Để phát hiện sự thay đổi nội dung trong một website, trước hết nội dung trang chủ của website cần giám sát phải được thu thập.

Bước 2 - Lấy tài liệu CSS của website: Sau một khoảng thời gian được định sẵn, nội dung trang chủ của website (mã nguồn HTML) được lấy về và tiến hành và so sánh với nội dung đã được lưu trữ trước đó.

Bước 3 - So sánh nội dung của website với nội dung đã lưu trước đó: Kiểm tra độ dài của văn bản bằng cách so sánh độ dài nội dung văn bản vừa lấy được với nội dung văn bản đã được lưu trước đó. Nếu tỉ lệ giữa độ dài văn bản mới lấy so với độ dài đã lưu vượt qua một hằng số ngưỡng đã cho trước (ví dụ là 50%, hoặc do người dùng chỉ định) thì đưa ra cảnh báo. Nếu nhỏ hơn ngưỡng thì tiếp tục kiểm tra cụm từ bắt buộc phải có có trong website như là: copyright, tên công ty, cơ quan, địa chỉ, số điện thoại, người chịu trách nhiệm nội dung…nếu bất cứ cụm từ nào mà không tìm thấy thì đưa ra cảnh báo.

Kiểm tra cụm từ không bao giờ cho phép, bị cấm xuất hiện trong trang website. Đây là các cụm từ mà tin tặc thường để lại sau khi tấn công thành công như là: Hacked by, hacked on…

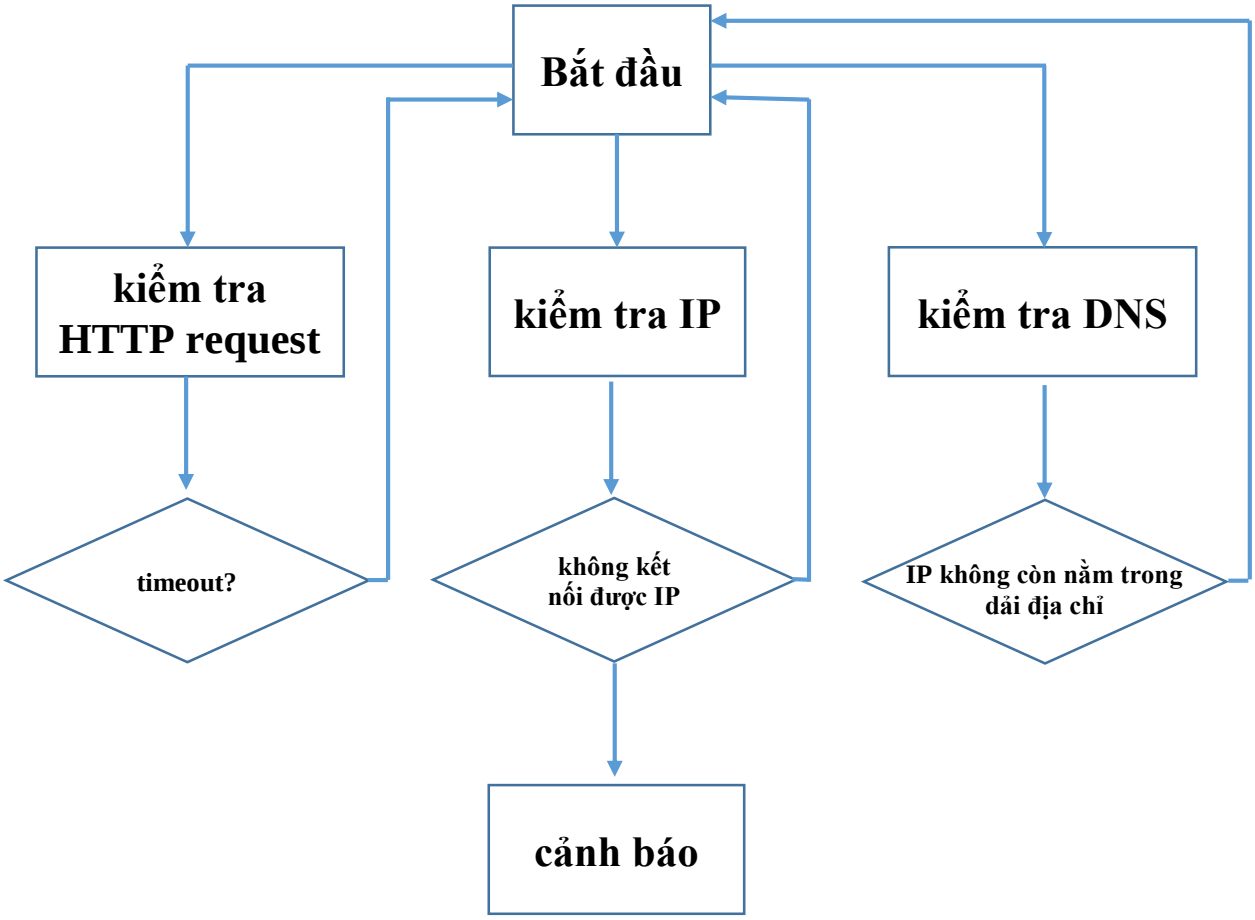
4.2 Biện pháp giám sát tình trạng hoạt động của website

Để biết tình trạng hoạt động hay không hoạt động của một Website, ta dựa vào ba tham số để kiểm tra là DNS, HTTP Request và IP:

- Bước 1 - Kiểm tra DNS: Kiểm tra xem địa chỉ IP đang xét có nằm trong danh sách địa chỉ IP mà DNS phân giải hay không. Nếu phát hiện thấy địa chỉ IP đang xét còn trong nằm trong danh sách là bình thường, ngược lại thì cảnh báo.

- Bước 2 - Kiểm tra HTTP Request: Khi ta gửi yêu cầu mà Website trả về trạng thái hồi đáp (response) là 200 nghĩa là bình thường, ngược lại thì đã xảy ra vấn đề trong kết nối với Webserver nên cần cảnh báo.

- Bước 3 - Kiểm tra IP Public: Ta gửi gói tin truy vấn ICMP (Internet Control Message Protocol) dạng “echo-request” đến cho máy đích và lắng nghe gói tin hồi đáp ICMP “echo-response”. Nếu gói tin trả về là “thành công” thì kết nối đến máy chủ web được xem là bình thường. Ngược lại thì hệ thống sẽ đưa ra cảnh báo.



Hình 4.1: Biện pháp giám sát tình trạng hoạt động, không hoạt động của website

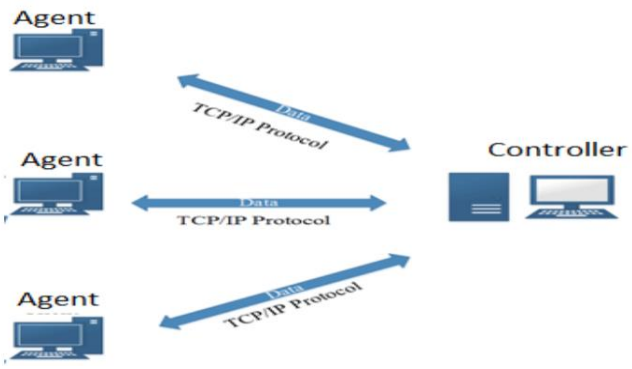
4.3 Biện pháp phát hiện sự thay đổi tính toàn vẹn

Biện pháp này nhằm phát hiện sự thay đổi mã nguồn, giám sát thư mục hoặc giám sát tập tin của website: Một Website muốn hoạt động tốt và an toàn thì cần phải bảo vệ thư mục chứa mã nguồn của website trên máy chủ web. Mọi thư mục, tập tin cần bảo vệ liên tục. Nguyên nhân là do chỉ cần một thay đổi nhỏ trên tập tin cấu hình có thể dẫn đến hậu quả là website hoạt động sai lệch hoặc các tập tin bị chèn thêm hoặc xóa nội dung. Thậm chí có nhiều tập tin độc hại được các kẻ tấn công đưa lên và được thay đổi liên tục nhằm qua mặt các phần mềm chống virus. Ở đây, phương pháp này áp dụng hàm băm và đối sánh chuỗi để kiểm tra sự thay đổi, thêm hoặc xóa tập tin.

4.4 Biện pháp phát hiện cuộc tấn công làm tê liệt website

Khi bị xâm nhập, kẻ tấn công có thể kiểm soát được website hoặc máy chủ Web, khi đã chiếm được quyền quản trị thì kẻ tấn công sẽ vô hiệu hóa các ứng dụng có liên quan nhằm che đậy hành vi, dễ dàng can thiệp sâu vào hệ thống. Ở một số trường hợp được bảo mật tốt thì khi không xâm nhập được, kẻ gian có thể tìm cách đánh sập hoặc làm tê liệt hoạt động của website hoặc cả webserver. Kỹ thuật được sử dụng phổ biến nhất là kỹ thuật tấn công từ chối dịch vụ (Denial of Service attack - DDoS/DoS) mà sơ khai nhất là hình thức DoS (Denial of Service). Tiếp đến là tấn công từ chối dịch vụ phân tán (Distributed Denial of Service - DDoS) và tấn công theo phương pháp phản xạ phân tán (Distributed Reflection Denial of Service - DRDoS).

Để phát hiện trường hợp kẻ tấn công phá hủy chương trình giám sát hay bị tấn công từ chối dịch vụ làm tê liệt máy chủ web. Học viên đề xuất sử dụng mô hình Agent – Controller. Agent chính là một ứng dụng được cài đặt trên các máy chủ. Controller là trung tâm giám sát được cài đặt trên một máy độc lập với máy chủ web, Controller này sẽ tiếp nhận các thông tin từ Agent gửi về. Với mô hình này giữa Agent và Controller sẽ duy trì kênh liên lạc riêng sử dụng Advanced Encryption Standard (AES) để mã hóa các thông điệp trao đổi và định kỳ Controller sẽ gửi thông tin liên lạc nếu mất thông tin liên lạc trong một khoảng thời gian nhất định thì sẽ cảnh báo.



Hình 4.2: Mô hình Agent-Controller

CHƯƠNG 5: XÂY DỰNG HỆ THỐNG GIÁM SÁT VÀ CẢNH BÁO CUỘC TẤN CÔNG AN NINH MẠNG DEFACE

5.1 Các yêu cầu đối với hệ thống đề xuất

Như đã tìm hiểu nêu trên, hậu quả của cuộc tấn công an ninh mạng deface gây ra hậu quả rất nghiêm trọng. Học viên đề xuất xây dựng một hệ thống có khả năng giám sát và cảnh báo khi có cuộc tấn công an ninh mạng deface với các yêu cầu như sau:

- Kịp thời phát hiện và cảnh báo khi có sự thay đổi bất thường của nội dung của website: Có thể kiểm tra sự thay đổi toàn bộ hoặc một phần của website.

- Kịp thời phát hiện và cảnh báo: Khi có sự thay đổi tính toàn vẹn; khi website không hoạt động và khi website phá hủy hoặc bị làm tê liệt.

- Giám sát liên tục hoặc giám sát trong một khung giờ cố định.

- Có thể giám sát nhiều website cùng lúc.

- Có lưu trữ kết quả giám sát để phục vụ việc tra cứu, đánh giá tình hình.

- Thực hiện cảnh báo đến máy tính, gửi tin nhắn và thư điện tử được chỉ định.

- Hệ thống hoạt động ổn định, tin cậy.

- Có giao diện đơn giản, thân thiện.

5.2 Mô tả hệ thống được đề xuất

Hệ thống đề xuất sẽ gồm các giao diện như sau:

- Giao diện Chính: Đây là giao diện đầu tiên hiển thị và cho phép người sử dụng đi đến các giao diện khác của hệ thống.

- Giao diện Tham số hệ thống: Giao diện này cho phép người sử dụng thiết lập các tham số của hệ thống như tần suất giám sát, số điện thoại, email để gửi cảnh báo...

- Giao diện Kết quả giám sát: Giao diện này cung cấp thông tin kết quả của việc giám sát theo thời gian của hệ thống.

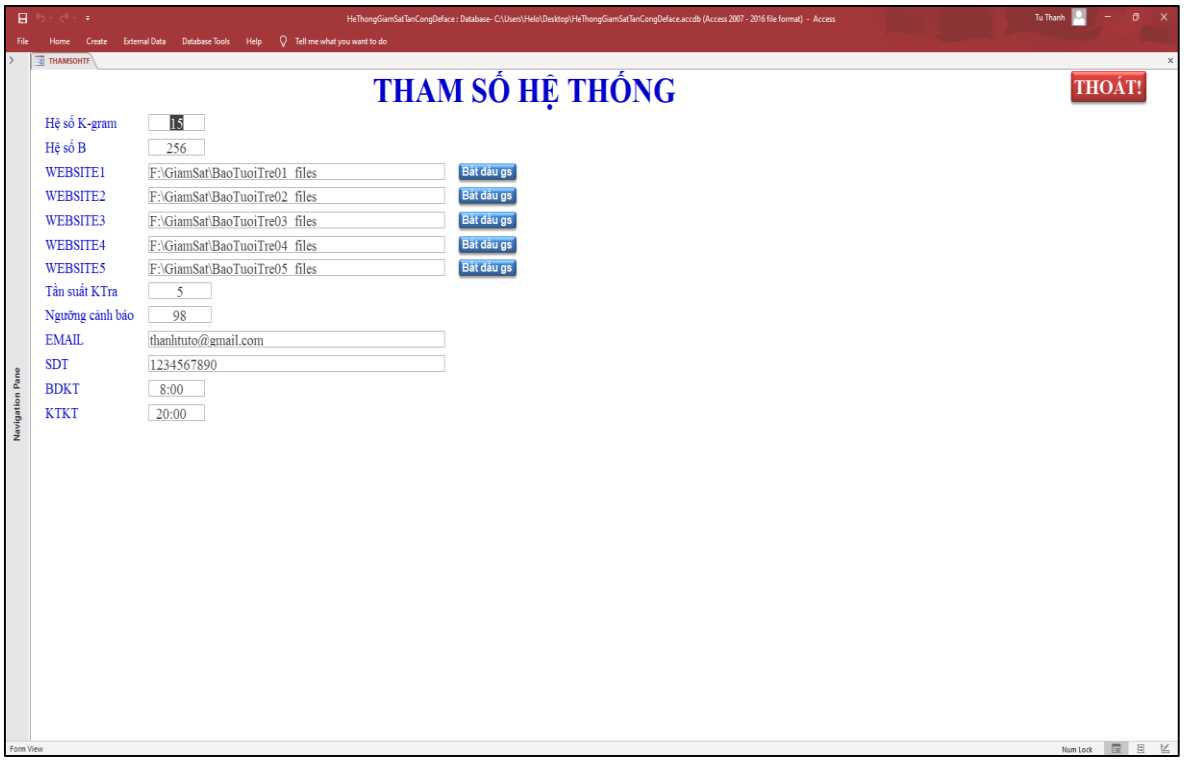
Hình 5.1: Giao diện Chính của hệ thống

Gồm các chức năng:

- Bắt đầu hoặc ngừng giám sát.

- Cho phép thiết lập các tham số của hệ thống.

- Cho phép xem kết quả giám sát.



Hình 5.2: Giao diện Tham số của hệ thống

Gồm các chức năng:

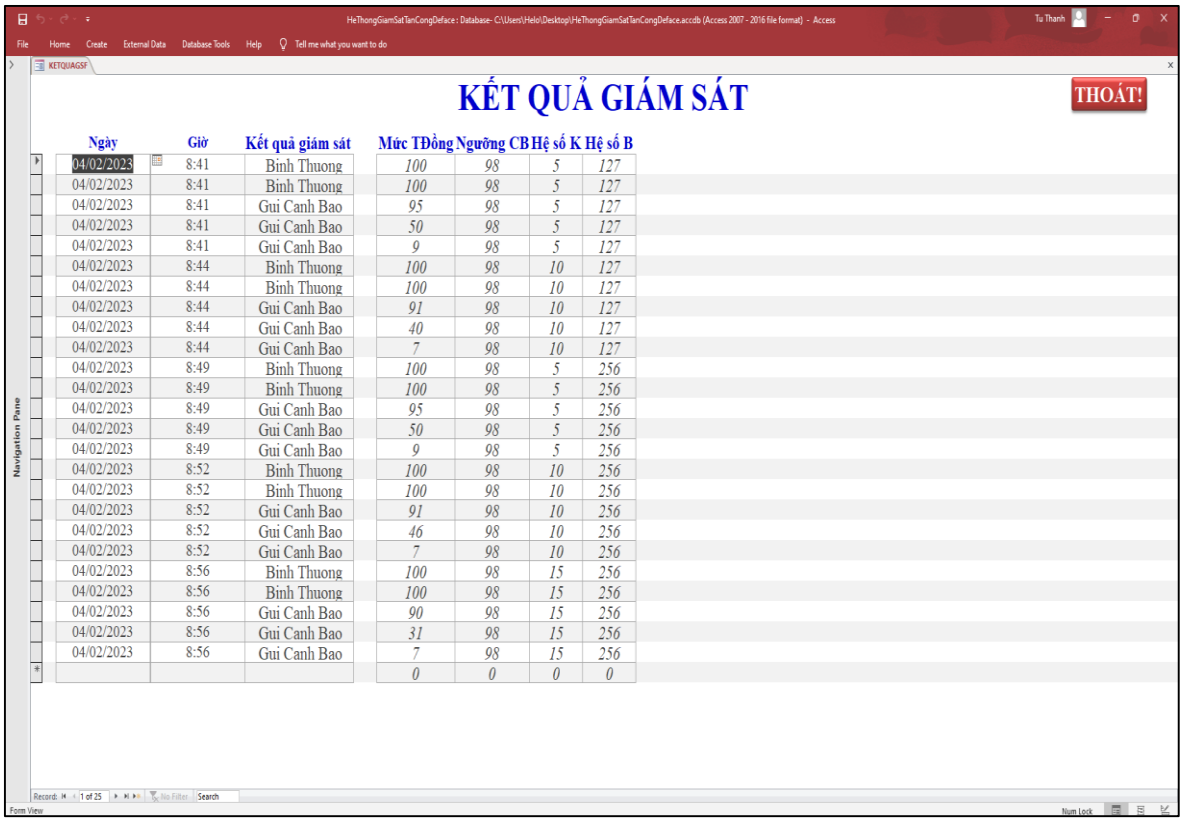
- Thiết lập các tham số của thuật toán Rabin-Karp.

- Chỉ định các website được giám sát cùng lúc.

- Tần suất thực việc giám sát (phút).

- Thiết lập ngưỡng cảnh báo: Khi độ tương đồng của 02 trạng thái giám sát dưới ngưỡng này thì thực hiện cảnh báo.

- Chỉ định các máy tính, số điện thoại, email để gửi cảnh báo



Ngày	Giờ	Kết quả giám sát	Mức TĐồng	Ngưỡng CB	Hệ số K	Hệ số B
04/02/2023	8:41	Binh Thuong	100	98	5	127
04/02/2023	8:41	Binh Thuong	100	98	5	127
04/02/2023	8:41	Gui Canh Bao	95	98	5	127
04/02/2023	8:41	Gui Canh Bao	50	98	5	127
04/02/2023	8:41	Gui Canh Bao	9	98	5	127
04/02/2023	8:44	Binh Thuong	100	98	10	127
04/02/2023	8:44	Binh Thuong	100	98	10	127
04/02/2023	8:44	Gui Canh Bao	91	98	10	127
04/02/2023	8:44	Gui Canh Bao	40	98	10	127
04/02/2023	8:44	Gui Canh Bao	7	98	10	127
04/02/2023	8:49	Binh Thuong	100	98	5	256
04/02/2023	8:49	Binh Thuong	100	98	5	256
04/02/2023	8:49	Gui Canh Bao	95	98	5	256
04/02/2023	8:49	Gui Canh Bao	50	98	5	256
04/02/2023	8:49	Gui Canh Bao	9	98	5	256
04/02/2023	8:52	Binh Thuong	100	98	10	256
04/02/2023	8:52	Binh Thuong	100	98	10	256
04/02/2023	8:52	Gui Canh Bao	91	98	10	256
04/02/2023	8:52	Gui Canh Bao	46	98	10	256
04/02/2023	8:52	Gui Canh Bao	7	98	10	256
04/02/2023	8:56	Binh Thuong	100	98	15	256
04/02/2023	8:56	Binh Thuong	100	98	15	256
04/02/2023	8:56	Gui Canh Bao	90	98	15	256
04/02/2023	8:56	Gui Canh Bao	31	98	15	256
04/02/2023	8:56	Gui Canh Bao	7	98	15	256
			0	0	0	0

Hình 5.3: Giao diện Kết quả giám sát

Gồm các chức năng:

- Xem lại ngày, giờ và kết quả giám sát của một website.

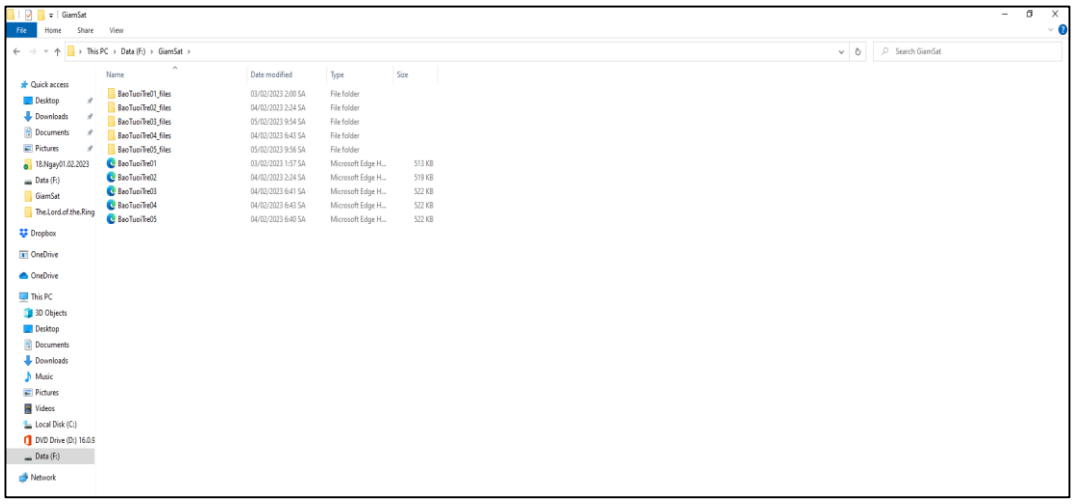
- Lọc kết quả giám sát theo khoảng ngày chỉ định.

5.3 Xây dựng hệ thống

Nhằm thực hiện hệ thống đã đề xuất, học viên sử dụng ngôn ngữ VBA(Visual Basic For Applications) trong phần mềm Microsoft Access để hiện thực hóa thuật toán Hàm băm Rabin-Karp và việc sử dụng hàm băm để phát hiện cuộc tấn công an ninh mạng deface đã nêu trên.

Trong thư mục “BaoTuoiTre” có 05 file CSS gồm cms_ads.css, font.css, tto_new_style_2023.css, font.2017.11.28.1.min.css và tuoitrenew.home.19012023v1.min.css. Trong đó, file font.css là file có nội dung ít nhất (chỉ có 22 dòng lệnh). Để kiểm tra tính đúng đắn của các thuật toán, học viên thực hiện giám sát chỉ riêng đối với file “font.css” trước khi thực hiện giám sát với toàn bộ các file css trong thư mục.

Học viên giả sử hệ thống sẽ thực hiện giám sát file “font.css” trên websibe của Báo Tuổi trẻ điện tử ở 05 trạng thái được lưu tại 05 thư mục “Giamsat” tại ổ F: của máy tính như hình:



Hình 5.4: Thư mục Giamsat

- Trạng thái 1: Đây là trạng thái đầu tiên của websibe của Báo Tuổi trẻ.

- Trạng thái 2: Giả sử trạng thái của Báo Tuổi trẻ vẫn không đổi (file “font.css” không đổi nội dung so với trạng thái 1).

- Trạng thái 3: Giả sử trạng thái của Báo Tuổi trẻ có thay đổi do file “font.css” thay đổi nội dung là các kiểu định dạng từ “font-style: **normal**;” chuyển thành “font-style: **italic**;”

- Trạng thái 4: Giả sử trạng thái của Báo Tuổi trẻ có thay đổi do file “font.css” do nội dung bị xóa một phần.

```
@font-face {
   font-family: 'wb4';
   font-style: normal;
   font-weight: 700;
   src: local('Roboto Medium'), local('Roboto-Medium'), url(../fonts/wb3.woff) format('woff');}
```

- Trạng thái 5: Giả sử trạng thái của Báo Tuổi trẻ có thay đổi do file “font.css” do nội dung là các phần định dạng chữ font-face bị xóa hết.

5.3.1 Hàm tính giá trị băm của chuỗi ký tự

```
Public Function TINHGIATRIBAMCHUOIKYTU(chuoi As String, b As Integer) As Long
'Tinh_gia_tri_bam_chuoi_ky_tu
'Hàm tính giá tri bam cua chuoi ky tu
'b là he so co so (base)=256
'songuyento = 997

    Dim i As Integer
    Dim sokytu As Integer
    Dim p As Long
    Dim songuyento As Integer

    p = 0
    songuyento = 997
    sokytu = Len(chuoi)

    For i = 1 To sokytu
        p = (b * p + Asc(Mid(chuoi, i, 1))) Mod songuyento
    Next i

    TINHGIATRIBAMCHUOIKYTU = p
End Function
```

5.3.2 Hàm tính bảng băm của một file text

```
Public Function TINHBANGBAMFILETEXT(FText As String, BANGGTB As Integer)
'Tinh_bang_bam_file_text
'Hàm tính cac gia tri bam cua file text
'HSOK (K-gram) là so ky tu cua chuoi muon tinh
'BANGGTB:bang gia tri bam = 1 hoac 2
'-------------------------------------
    Dim db As DAO.Database
    Dim rs1 As DAO.Recordset
    Dim strSQL As String

    Dim i As Integer
    Dim sokytu As Integer
    Dim p As Long
    Dim chuoiK As String
    Dim sochuoi As Integer
    Dim HSOB As Integer
    Dim HSOK As Integer

'---------------------------------
'1.Lay gia tri hsoB
    Set db = CurrentDb()
    strSQL = " SELECT THAMSOHT.IDTHS, THAMSOHT.HSOK, THAMSOHT.HSOB, THAMSOHT.WEBSITE1, THAMSOHT.WEBSITE2 " & _
             " FROM THAMSOHT " & _
             " WHERE (((THAMSOHT.IDTHS)=1)); "
    Set rs1 = db.OpenRecordset(strSQL, dbOpenDynaset)

    HSOB = rs1!HSOB
    HSOK = rs1!HSOK
    rs1.Close
'---------------------------------
'2.Xoa cac record trong Bang gia tri bam 1 hoac 2
    If (BANGGTB = 1) Then
        strSQL = " Delete BANGGTB1.GTB " & _
                 " FROM BANGGTB1;"
    Else
        strSQL = " Delete BANGGTB2.GTB " & _
                 " FROM BANGGTB2;"
    End If
    DoCmd.SetWarnings False
    DoCmd.RunSQL strSQL
```

```
'---------------------------------
'3.Them moi cac record trong Bang gia tri bam 1 hoac 2
    If (BANGGTB = 1) Then
        strSQL = " SELECT BANGGTB1.GTB " & _
                 " FROM BANGGTB1;"
    Else
        strSQL = " SELECT BANGGTB2.GTB " & _
                 " FROM BANGGTB2;"
    End If
    Set rs1 = db.OpenRecordset(strSQL, dbOpenDynaset)
'-------------------------------------
'4.Tinh cac gia tri bam va luu vao bang
    p = 0
    sokytu = Len(FText)
    sochuoi = ((sokytu - (sokytu Mod HSOK)) / HSOK) 'tinh so chuoi cua file

    For i = 1 To sochuoi
        chuoiK = Mid(FText, (i - 1) * HSOK + 1, HSOK)
        p = TINHGIATRIBAMCHUOIKYTU(chuoiK, HSOB)

        rs1.AddNew
        rs1!GTB = p
        rs1.Update
    Next i

rs1.Close
db.Close
End Function
```

5.3.3 Hàm tính mức độ tương đồng của hai tài liệu

```
Public Function TinhSuTuongDong02BangGTB() As Integer
'Tinh_su_tuong_02_bang_Gia_Tri_Bam
    Dim db As DAO.Database
    Dim rs1, rs2, rs3 As DAO.Recordset
    Dim strSQL1, strSQL2, strSQL3 As String
    Dim SH As Integer       'SH (Identical Hash): so giá tri bam giong nhau.
    Dim TyleTD As Integer   'ty le tuong dong
    Dim a, b As Integer
    Dim sophantuBangGTB1, sophantuBangGTB2 As Integer

    SH = 0
    TyleTD = 0
    sophantuBangGTB1 = 0
    sophantuBangGTB2 = 0

    strSQL1 = " SELECT BANGGTB1.GTB " & _
              " FROM BANGGTB1 " & _
              " GROUP BY BANGGTB1.GTB; "

    strSQL2 = " SELECT BANGGTB2.GTB " & _
              " FROM BANGGTB2 " & _
              " GROUP BY BANGGTB2.GTB; "

    Set db = CurrentDb()
    Set rs1 = db.OpenRecordset(strSQL1, dbOpenDynaset)
    Set rs2 = db.OpenRecordset(strSQL2, dbOpenDynaset)
```

```
'--------------------------
'1. Kiem tra bang bam 1 va bang bam 2 co phan tu nao ko
    If (rs1.BOF = True) And (rs1.EOF = True) Then
        MsgBox "BANG GIA TRI BAM 1 KHONG CO GIA TRI NAO!"
        TinhSuTuongDong02BangGTB = TyleTD
        Exit Function
    End If
    If (rs2.BOF = True) And (rs2.EOF = True) Then
        MsgBox "BANG GIA TRI BAM 2 KHONG CO GIA TRI NAO!"
        TinhSuTuongDong02BangGTB = TyleTD
        Exit Function
    End If
rs1.Close
rs2.Close
'--------------------------
'2. Lay tong phan tu cua tung bang
    strSQL1 = " SELECT Count(BANGGTB1.GTB) AS CountOfGTB " & _
              " FROM BANGGTB1; "
    strSQL2 = " SELECT Count(BANGGTB2.GTB) AS CountOfGTB " & _
              " FROM BANGGTB2; "

    Set rs1 = db.OpenRecordset(strSQL1, dbOpenDynaset)
    Set rs2 = db.OpenRecordset(strSQL2, dbOpenDynaset)

    sophantuBangGTB1 = rs1!CountOfGTB
    sophantuBangGTB2 = rs2!CountOfGTB
rs1.Close
rs2.Close
'--------------------------
    strSQL1 = " SELECT BANGGTB1.GTB " & _
              " FROM BANGGTB1 " & _
              " GROUP BY BANGGTB1.GTB; "

    strSQL2 = " SELECT BANGGTB2.GTB " & _
              " FROM BANGGTB2 " & _
              " GROUP BY BANGGTB2.GTB; "

    Set rs1 = db.OpenRecordset(strSQL1, dbOpenDynaset)
    Set rs2 = db.OpenRecordset(strSQL2, dbOpenDynaset)
```

```
'3. Xet tung gia tri bam cua bang bam 1 voi tat ca gia tri bam cua bang bam 2
    rs1.MoveFirst
    Do Until rs1.EOF
        rs2.MoveFirst
        Do Until rs2.EOF
            If rs1!GTB = rs2!GTB Then
                strSQL3 = " SELECT BANGGTB1.GTB, Count(BANGGTB1.GTB) AS CountOfGTB " & _
                          " FROM BANGGTB1 " & _
                          " GROUP BY BANGGTB1.GTB " & _
                          " HAVING (((BANGGTB1.GTB)= " & rs1!GTB & "  )); "

                Set rs3 = db.OpenRecordset(strSQL3, dbOpenDynaset)
                If (rs3.BOF = True) And (rs3.EOF = True) Then
                    a = 0
                Else
                    a = rs3!CountOfGTB
                End If
                rs3.Close

                strSQL3 = " SELECT BANGGTB2.GTB, Count(BANGGTB2.GTB) AS CountOfGTB " & _
                          " FROM BANGGTB2 " & _
                          " GROUP BY BANGGTB2.GTB " & _
                          " HAVING (((BANGGTB2.GTB)= " & rs2!GTB & "  )); "

                Set rs3 = db.OpenRecordset(strSQL3, dbOpenDynaset)

                If (rs3.BOF = True) And (rs3.EOF = True) Then
                    b = 0
                Else
                    b = rs3!CountOfGTB
                End If
                rs3.Close

                If (a < b) Then
                    SH = SH + a
                Else
                    SH = SH + b
                End If
            End If
        rs2.MoveNext
        Loop
    rs1.MoveNext
    Loop
rs1.Close
rs2.Close
db.Close
'---------------------------
'3. Tinh ty le tuong dong
    TyleTD = (2 * SH) / (sophantuBangGTB1 + sophantuBangGTB2) * 100
    TinhSuTuongDong02BangGTB = TyleTD
'    MsgBox SH & "--" & sophantuBangGTB1 & "--" & sophantuBangGTB2 & "--" & TyleTD
End Function
```

5.4 Kết quả thực nghiệm hệ thống và nhận xét

Để thực nghiệm hệ thống đã xây dựng, học viên đã giám sát 05 trạng thái giả thuyết với các tham số của thuật toán Rabin-Karp như sau:

- K=5 và B=127.

- K=10 và B=127.

- K=5 và B=256.

- K=10 và B=256.

- K=15 và B=256.

Bảng 5.1: Bảng kết quả thực nghiệm đối với file font.css

NGAY	GIO	KQGS	HESOK	HESOB	NGUONGCB	P
04/02/2023	8:41	Binh Thuong	5	127	98	100
04/02/2023	8:41	Binh Thuong	5	127	98	100
04/02/2023	8:41	Gui Canh Bao	5	127	98	95
04/02/2023	8:41	Gui Canh Bao	5	127	98	50
04/02/2023	8:41	Gui Canh Bao	5	127	98	9
04/02/2023	8:44	Binh Thuong	10	127	98	100
04/02/2023	8:44	Binh Thuong	10	127	98	100
04/02/2023	8:44	Gui Canh Bao	10	127	98	91
04/02/2023	8:44	Gui Canh Bao	10	127	98	40
04/02/2023	8:44	Gui Canh Bao	10	127	98	7
04/02/2023	8:49	Binh Thuong	5	256	98	100
04/02/2023	8:49	Binh Thuong	5	256	98	100
04/02/2023	8:49	Gui Canh Bao	5	256	98	95
04/02/2023	8:49	Gui Canh Bao	5	256	98	50
04/02/2023	8:49	Gui Canh Bao	5	256	98	9
04/02/2023	8:52	Binh Thuong	10	256	98	100
04/02/2023	8:52	Binh Thuong	10	256	98	100
04/02/2023	8:52	Gui Canh Bao	10	256	98	91
04/02/2023	8:52	Gui Canh Bao	10	256	98	46
04/02/2023	8:52	Gui Canh Bao	10	256	98	7
04/02/2023	8:56	Binh Thuong	15	256	98	100
04/02/2023	8:56	Binh Thuong	15	256	98	100
04/02/2023	8:56	Gui Canh Bao	15	256	98	90
04/02/2023	8:56	Gui Canh Bao	15	256	98	31
04/02/2023	8:56	Gui Canh Bao	15	256	98	7

Bảng 4.2: Bảng kết quả thực nghiệm đối với file font.css trong thư mục có 05 file css

NGAY	GIO	KQGS	HESOK	HESOB	NGUONGCB	P
15/02/2023	17:54	Binh Thuong	15	256	98	100
15/02/2023	17:54	Binh Thuong	15	256	98	100
15/02/2023	17:54	Binh Thuong	15	256	98	99
15/02/2023	17:54	Gui Canh Bao	15	256	98	93
15/02/2023	17:54	Binh Thuong	15	256	98	100
15/02/2023	18:05	Binh Thuong	15	256	98	100
15/02/2023	18:05	Binh Thuong	15	256	98	100
15/02/2023	18:05	Binh Thuong	15	256	98	99
15/02/2023	18:05	Gui Canh Bao	15	256	98	93
15/02/2023	18:05	Binh Thuong	15	256	98	100
15/02/2023	18:31	Binh Thuong	15	256	98	100
15/02/2023	18:31	Binh Thuong	15	256	98	100
15/02/2023	18:31	Binh Thuong	15	256	98	99
15/02/2023	18:31	Gui Canh Bao	15	256	98	93
15/02/2023	18:31	Binh Thuong	15	256	98	100

Đối với 25 kết quả đầu tiên, khi chạy thực nghiệm giám sát chỉ đối với sự thay đổi của file font.css cho thấy khi file này càng thay đổi nhiều thì mức độ tương đồng của các trạng thái càng giảm. Điều này cho thấy thuật toán được học viên xây dựng trong hệ thống là chính xác.

Các kết quả tiếp theo là việc chạy thực nghiệm giám sát chỉ đối với sự thay đổi của file font.css trong trường hợp thư mục có đủ 05 file css. Khi này, sự thay đổi của file này dù nhiều thì mức độ tương đồng của các trạng thái vẫn rất lớn, giảm rất ít. Phương pháp đề xuất này chỉ phù hợp với các website có tần suất thay đổi nội dung ít như website của sở ban ngành, các trường đại học.

Trong hệ thống do ba tác Trần Đắc Tốt, Đặng Lê Nam, Phạm Nguyễn Huy Phương đã sử dụng hàm băm MD5 để phát hiện sự thay đổi của website do có tốc độ mã hóa cao nhất [10]. Các tác cũng nhận thấy khi giải pháp được đề xuất chưa phù hợp khi có sự thay đổi lớn trong nội dung của file css. Do vậy, giải pháp sử dụng hàm

băm Rabin-Karp học viên đề xuất cũng có độ tin cậy còn thấp, cần phải cải tiến, bổ sung thêm một số kỹ thuật khác để tăng độ tin cậy.

Dựa vào kết quả thực nghiệm trên, ta nhận thấy:

- Khi hai trạng thái giám sát của website không đổi thì hệ thống đề xuất sẽ không đưa ra cảnh báo: Mức tương đồng của 02 trạng thái =100%.

- Khi trạng thái giám sát của website có thay đổi thì hệ thống có phát hiện sự thay đổi: Mức tương đồng của hai trạng thái <100%.

- Khi cố định hệ số cơ sở B của thuật toán Rabin-Karp thì mức tương đồng của hai trạng thái sẽ giảm khi K càng lớn.

CHƯƠNG 6: KẾT LUẬN

1. Kết quả nghiên cứu của đề tài

Tấn công an ninh mạng deface là một kiểu tấn công phổ biến và gây ra nhiều hậu quả nghiêm trọng đối với bất kỳ tổ chức, cá nhân nhất là đối với các doanh nghiệp. Do vậy việc kịp thời phát hiện website bị tấn công để đưa ra các biện pháp phòng chống là cần thiết để hạn chế tối đa các hậu quả không mong muốn. Thông qua đề tài và nhờ sự hướng dẫn tận tình của Thầy Nguyễn Đức Thái học viên đã tìm hiểu thêm các kiến thức về website và an ninh mạng. Những kiến thức này sẽ giải quyết các vấn đề cần thiết, giúp ích cho học viên tại cơ quan đang công tác.

- Thông qua đề tài, học viên hiểu được cách mà trình duyệt web trình bày một trang web; cách thức một website hoạt động và các vấn đề về an ninh mạng, những lỗ hổng bảo mật, những kỹ thuật tấn công để xâm nhập, khống chế website và cũng như cách phòng chống.

- Hiểu được thuật toán Rabin-Karp và ứng dụng của thuật toán nhằm phát hiện cuộc tấn công an ninh mạng deface.

- Giải quyết một phần các yêu cầu đặt ra.

2. Hạn chế của luận văn

Tuy nhiên do giới hạn về thời gian và có nhiều kiến thức cần phải nghiên cứu nên luận văn còn tồn tại các hạn chế sau:

- Chưa xây dựng được hệ thống giám sát đối với các website trên môi trường thực tế mà chỉ dừng ở môi trường giả thuyết.

- Chưa xây dựng được hệ thống giám sát đặt ngoài website cần giám sát để đánh giá website có còn hoạt động hay không.

- Qua kết quả thực nghiệm, hệ thống được đề xuất cần phải xác định ngưỡng cảnh báo đối với từng trường hợp website cụ thể để xác định ngưỡng cảnh báo phù hợp. Ở góc độ rộng hơn, thì với giải pháp chỉ giám sát sự thay đổi của file css thì chưa đảm bảo tính chính xác, khả năng tin cậy đủ để áp dụng vào thực tế. Để đảm bảo tính khả thi và hiệu quả của ứng dụng, cần bổ sung thêm nhiều kỹ thuật khác nhằm phát hiện các hình thức tấn công deface đa dạng và phức tạp hơn.

3. Kiến nghị và hướng nghiên cứu tiếp theo

Đề tài cần được nghiên cứu, cải tiến trong thời gian tới như sau:

- Xây dựng hệ thống giám sát đối với các website trên môi trường thực tế và hệ thống giám sát đặt ngoài website.

- Xây dựng nhiều kích bản về cuộc tấn công an ninh mạng deface để xác định các tham số hệ thống nhằm làm cho hệ thống hoạt động hiệu quả, đáng tin cậy.

- Nghiên cứu, ứng dụng các kỹ thuật khác nhằm tăng cường khả năng phát hiện cuộc tấn công an mạng deface.

DANH MỤC TÀI LIỆU THAM KHẢO

[1] Cooks, A.; Olivier, M. (2004) Curtailing web defacement using a read-only strategy. In Proceedings of the 4th Annual Information Security South Africa Conference, Midrand, South Africa, 30 June–2 July 2004.

[2] Mazin S. Al-Hakeem (2010), "Anti Web Site Defacement System (AWDS)", Conference Paper, February 2010.

[3] Alberto Bartoli, Giorgio Davanzo and Eric Medvet (2010),"A Framework for Large-Scale Detection of Web Site Defacements", ACM Transactions on Internet Technology Volume 10, Issue 3, October 2010.

[4] Davanzo, G.; Medvet, E.; Bartoli, A. (2011), "Anomaly detection techniques for a web defacement monitoring service", Expert Syst. Appl. 2011, 38, 12521–12530.

[5] Rajiv Kumar Gurjwar, Divya Rishi Sahu, Deepak Singh Tomar (2013), "An Approach to Reveal Website Defacement", (IJCSIS) International Journal of Computer Science and Information Security, Vol. 11, No. 6.

[6] Ramniwas, K., Nikhil, K. S., & Deepak, S. T. (2014), "A novel approach to detect Webpage tampering", International Journal of Computer Science and Information Technologies, 5(3), 4604-4607.

[7] Enaw, E.E.; Prosper, D.P. (2014), "A conceptual approach to detect web defacement through Artificial Intelligence". Int. J. Adv. Comput. Technol. 2014, 3, 77–83.

[8] Rashmi, K. V., Shahzia, S. (2015). Implementation of Web defacement detection technique. International Journal of Innovations in Engineering and Technology, 6(1), 134-140.

[9] Francesco Bergadano 1, Fabio Carretto, Fabio Cogno and Dario Ragno (2019), "Defacement Detection with Passive Adversaries", Algorithms 2019, 12, 150.

[10] Trần Đắc Tốt, Đặng Lê Nam, Phạm Nguyễn Huy Phương (2018), "Hệ thống cảnh báo tấn công thay đổi giao diện website", Tạp chí Khoa học Đại học Đà Lạt.

[11] Ranti Eka Putri, A. Siahaan (2017), "Examination of Document Similarity Using Rabin-Karp Algorithm", International Journal of Recent Trends in Engineering & Research (IJRTER) Volume 03, Issue 08; August – 2017.

BẢN CAM ĐOAN

Tôi cam đoan đã thực hiện việc kiểm tra mức độ tương đồng nội dung luận văn/luận án qua phần mềm Kiểm tra tài liệu (https://kiemtratailieu.vn) một cách trung thực và đạt kết quả mức độ tương đồng **15 %** toàn bộ nội dung luận văn/luận án. Bản luận văn/luận án kiểm tra qua phần mềm là bản cứng luận văn/luận án đã nộp bảo vệ trước hội đồng. Nếu sai sót tôi xin chịu các hình thức kỷ luật theo quy định hiện hành của Học viện.

TP. Hồ Chí Minh, ngày 28 tháng 02 năm 2023

Học viên thực hiện luận văn

Tô Thanh Tú

KiểmTraTàiLiệu

BÁO CÁO KIỂM TRA TRÙNG LẶP

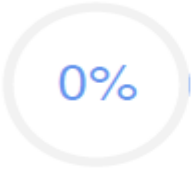
Thông tin tài liệu

Tên tài liệu:	01.ToThanhTu_LuanVan_26.02.2023
Tác giả:	Thanh Tú Tô
Điểm trùng lặp:	15
Thời gian tải lên:	08:44 28/02/2023
Thời gian sinh báo cáo:	08:47 28/02/2023
Các trang kiểm tra:	64/64 trang

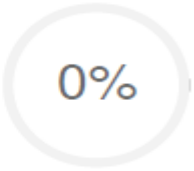
Kết quả kiểm tra trùng lặp

Có 15% nội dung trùng lặp

Có 85% nội dung không trùng lặp



Có 0% nội dung người dùng loại trừ



Có 0% nội dung hệ thống bỏ qua

Nguồn trùng lặp tiêu biểu

123docz.net tckh.dlu.edu.vn tailieu.vn

Học viên	Người hướng dẫn khoa học
Tô Thanh Tú	**TS.Nguyễn Đức Thái**