HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Khuất Thị Ngọc Ánh

PHƯƠNG PHÁP PHÁT HIỆN TẤN CÔNG WEB ỨNG DỤNG DỰA TRÊN KỸ THUẬT PHÂN TÍCH HÀNH VI

LUẬN VĂN THẠC SĨ KỸ THUẬT

(***Theo định hướng ứng dụng***)

HÀ NỘI - NĂM 2020

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

Khuất Thị Ngọc Ánh

PHƯƠNG PHÁP PHÁT HIỆN TẤN CÔNG WEB ỨNG DỤNG DỰA TRÊN KỸ THUẬT PHÂN TÍCH HÀNH VI

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. ĐỖ XUÂN CHỢ

HÀ NỘI - NĂM 2020

LỜI CAM ĐOAN

Tôi cam đoan đây là công trình nghiên cứu của riêng tôi. Nội dung của luận văn có tham khảo và sử dụng các tài liệu, thông tin được đăng tải trên những tạp chí khoa học và các trang web được liệt kê trong danh mục tài liệu tham khảo. Tất cả các tài liệu tham khảo đều có xuất xứ rõ ràng và được trích dẫn hợp pháp.

Tôi xin hoàn toàn chịu trách nhiệm và chịu mọi hình thức kỷ luật theo quy định cho lời cam đoan của mình.

Hà nội, ngày … tháng … năm 2020

Tác giả luận văn

Khuất Thị Ngọc Ánh

MỤC LỤC

DANH MỤC CÁC THUẬT NGỮ TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
OWASP	Open Web Application Security Project	Dự án mở về bảo mật ứng dụng Web
SQL	Structured Query Language	Ngôn ngữ truy vấn cấu trúc
HTTP	HyperText Transfer Protocol	Giao thức truyền tải siêu văn bản
HTTPS	Hyper Text Transfer Protocol Secure	Giao thức truyền tải siêu văn bản bảo mật
XSS	Cross-Site Scripting	Tấn công thực thi mã script
HTML	Hypertext Markup Language	Ngôn ngữ đánh dấu siêu văn bản
CSRF	Cross-Site Request Forgery	Tấn công giả mạo yêu cầu
ATP	Advanced Persistent Threat	Mối đe dọa nâng cao
IPS	Intrusion Prevention system	Hệ thống ngăn chặn xâm nhập
IDS	Intrusion detection system	Hệ thống phát hiện xâm nhập
NIDS	Network - Based IDS	Hệ thống phát hiện xâm nhập mạng
HIDS	Host - Based IDS	Hệ thống phát hiện xâm nhập dựa trên máy chủ
SSL	Secure Sockets Layer	Lớp socket bảo mật
TSL	transport layer security	Giao thức bảo mật tầng giao vận
WAF	Web Application Firewall	Giải pháp bảo mật trong hệ thống
IP	Internet Protocol	Giao thức Internet
VPN	Virtual Private Network	Mạng riêng ảo
DOS	Denial of Service	Tấn công từ chối dịch vụ
SVM	Support vector machine	Máy véc tơ hỗ trợ

DANH MỤC CÁC BẢNG

DANH MỤC CÁC HÌNH

MỞ ĐẦU

1. Tính cấp thiết của đề tài

Các nguy cơ mất an toàn thông tin trên thế giới nói chung và Việt Nam nói riêng liên tục ra tăng và phát triển về cả số lượng cũng như mức độ nguy hiểm của các cuộc tấn công. Theo ghi nhận của một số công ty bảo mật trên thế giới, trong vài năm trở lại đây Việt Nam luôn được coi là điểm nóng của mã độc và các cuộc tấn công website trái phép. Hàng loạt các cuộc tấn công website diễn ra với quy mô lớn vào các website của các doanh nghiệp, tổ chức chính phủ… đã gây mất an toàn thông tin và ảnh hưởng nghiêm trọng đến uy tín và doanh nghiệp, tổ chức chính phủ. Hiện nay, các cơ quan nhà nước, các tổ chức chính phủ đã và đang có nhiều biện pháp tích cực trong việc phòng chống và phát hiện tấn công website. Rất nhiều biện pháp đã được ứng dụng và triển khai trong thực tế. Tuy nhiên, các kỹ thuật tấn công website ngày càng được biến đổi tinh vi và phức tạp, đặc biệt là các truy cập thể hiện các hành vi bất thường của người dùng website rất dễ dàng để vượt qua được sự giám sát của các sản phẩm an toàn web.

Website của Trường Đại học Công nghệ Giao thông vận tải được sử dụng cho phép nhiều user bao gồm cả sinh viên, giảng viên và cán bộ công nhân viên chức sử dụng để làm việc và tra cứu thông tin. Hàng ngày có hàng trăm nghìn giao dịch, của người dùng truy cập vào website của trường nhằm khai thác và thực hiện mục đích của mình. Trong số các truy cập này đã có nhiều truy cập bất thường người của người dùng web được ghi nhận, gây mất an toàn thông tin và uy tín của nhà trường. Chính vì vậy, vấn đề phát hiện và ngăn chặn các truy nhập bất thường của người dùng web lên Website của Trường Đại học Công nghệ Giao thông vận tải đang rất được quan tâm hiện nay. Từ những lý do trên, học viên với sự giúp đỡ của TS. Đỗ Xuân Chợ lựa chọn đề tài: “Phương pháp phát hiện tấn công web ứng dụng kỹ thuật phân tích hành vi”.

2. Tổng quan vấn đề cần nghiên cứu

Hiện nay việc tăng trưởng và phát triển nhanh chóng của Internet dẫn đến nhu

cầu bảo mật và đảm bảo an toàn thông tin đang được các doanh nghiệp ngày càng chú trọng.

Theo Báo cáo an ninh website Q3/2018 của CyStack [15], trong quý 3 năm 2018 trên thế giới đã có 129.722 website bị tin tặc tấn công và chiếm quyền điều khiển. Như vậy, cứ mỗi phút trôi qua lại có một website bị tin tặc kiểm soát. Bằng việc chiếm quyền điều khiển website tin tặc có thể gây ra rất nhiều vấn đề rắc rối cho các chủ website: đánh cắp dữ liệu, cài đặt mã độc, phá hoại website, tạo trang lừa đảo (phishing), tống tiền…Theo thống kê, Việt Nam đứng thứ 19 (chiếm 0.9%) trong số các quốc gia có website bị tin tặc tấn công. Cụ thể trong quý 3 năm 2018 đã có 1.183 website của Việt Nam bị tin tặc tấn công và kiểm soát. Các website giới thiệu sản phẩm và dịch vụ của Doanh nghiệp là đối tượng bị tin tặc tấn công nhiều nhất, chiếm tới 71,51%. Vị trí thứ hai là các website Thương mại điện tử chiếm 13,86%. Các website có tên miền .gov.vn của chính phủ chiếm 1.9% trong danh sách với tổng số 23 website bị tấn công.

Ngoài việc sử dụng các phương pháp phòng chống tấn công truyền thống, xu hướng hiện nay là xử dụng trí tuệ nhân tạo, học máy để áp dụng trong lĩnh vực an toàn thông tin để phát hiện nhanh chóng và tăng độ chính xác. Có 2 hướng tiếp cận chính là dựa vào dấu hiệu và hành vi để phát hiện tấn công web nói chung và hành vi bất thường người dùng web nói riêng. Mỗi phương pháp đều có những ưu điểm và nhược điểm nhất định. Trong luận văn, tác giả sẽ đi sâu vào việc nghiên cứu về phương pháp phát hiện hành vi bất thường người dùng web dựa trên kỹ thuật phân tích hành vi. Để luận văn đạt được những kết quả trên, cần nghiên cứu và làm rõ các nội dung:

- Tìm hiểu một số lỗ hổng, điểm yếu và các cuộc tấn công lên web ứng dụng;
- Nghiên cứu và tìm hiểu về một số phương pháp và công nghệ phát hiện tấn công web ứng dụng;
- Nghiên cứu phương pháp phát hiện tấn công web bằng kỹ thuật phân tích hành vi trên cơ sở thuật toán học máy và hành vi người dùng.

3. Mục đích nghiên cứu

- Tìm hiểu về thuật toán phân loại học máy;
- Tìm hiểu về hành vi bất thường người dùng web;

- Nghiên cứu phương pháp phân loại hành vi bất thường của người dùng web dựa trên các thuật toán học máy.

4. Đối tượng và phạm vi nghiên cứu

- Đối tượng nghiên cứu: Dữ liệu Truy cập web, dữ liệu truy cập web ứng dụng của trường Đại học Công nghệ Giao thông vận tải.
- Phạm vi nghiên cứu: Hệ thống website và phương pháp phát hiện hành vi của người dùng web.

5. Phương pháp nghiên cứu

Dựa trên các thuật toán học máy có giám sát từ đó phân loại người dùng và xác định người dùng bất thường.

Cấu trúc nội dung luận văn gồm 3 chương với các nội dung như sau:

Chương 1: Nguy cơ mất an toàn thông tin web và biện pháp phòng chống

Nội dung chương 1 của luận văn sẽ trình bày về một số kỹ thuật tấn công website bao gồm: một số phương pháp tấn công, các công cụ hỗ trợ tấn công... Bên cạnh đó, trong chương 1 luận văn sẽ trình bày một số phương pháp và công cụ phòng chống tấn công web.

Chương 2: Phương pháp phát hiện tấn công trên web dựa trên kỹ thuật phân tích hành vi

Nội dung chương 2 của luận văn sẽ nghiên cứu về một số phương pháp phát hiện tấn công web bao gồm kỹ thuật phát hiện và các công cụ mã nguồn mở hỗ trợ phát hiện tấn công web. Ngoài ra, trong chương 2 sẽ trình bày về phương pháp phát hiện tấn công web dựa trên kỹ thuật phân tích hành vi.

Chương 3: Thực nghiệm và đánh giá

Nội dung chương 3 của luận văn sẽ thực hiện thực nghiệm phát hiện tấn công web dựa trên kỹ thuật phân tích hành vi trên cơ sở thuật toán và hành vi đã được lựa chọn và phân tích ở chương 2

Kết luận.

CHƯƠNG 1: NGUY CƠ MẤT AN TOÀN THÔNG TIN WEB VÀ BIỆN PHÁP PHÒNG CHỐNG

Tóm tắt chương: Chương 1 của luận văn trình bày về một số kỹ thuật tấn công website bao gồm: một số phương pháp tấn công, các công cụ hỗ trợ tấn công... Bên cạnh đó, trong chương 1 luận văn sẽ trình bày một số phương pháp và công cụ phòng chống tấn công web.

1.1. Top 10 lỗ hổng bảo mật ứng dụng web theo OWASP

Ngày nay nguy cơ mất an toàn thông tin ngày càng xảy ra nhiều và dẫn đến các hậu quả nghiêm trọng mà người quản trị website không thể lường trước được. Đặc biệt là đối với các cuộc tấn công web ngày càng tinh vi và khó lường. Chính vì vậy, trong mục này luận văn sẽ khảo sát các phương thức tấn công lỗ hổng bảo mật Website dựa trên khuyến nghị của OWASP (The Open Web Application Security Project- dự án mở về bảo mật ứng dụng Web) [12].

1.1.1. SQL injection

SQL injection là một kĩ thuật cho phép những kẻ tấn công lợi dụng lỗ hổng trong việc kiểm tra dữ liệu nhập trong các ứng dụng web và các thông báo lỗi của hệ quản trị cơ sở dữ liệu để "tiêm vào" (inject) và thi hành các câu lệnh SQL bất hợp pháp (không được người phát triển ứng dụng lường trước). Hậu quả của nó rất tai hại vì nó cho phép những kẻ tấn công có thể thực hiện các thao tác xóa, hiệu chỉnh,… do có toàn quyền trên cơ sở dữ liệu của ứng dụng, thậm chí là server mà ứng dụng đó đang chạy. Lỗi này thường xảy ra trên các ứng dụng web có dữ liệu được quản lí bằng các hệ quản trị cơ sở dữ liệu như SQL Server, MySQL, Oracle, DB2, Sysbase. Có 4 dạng tấn công kiểu SQL injection sau:

- Vượt qua kiểm tra lúc đăng nhập;
- Sử dụng câu lệnh SELECT;
- Sử dụng câu lệnh INSERT;

- Sử dụng các Stored-Procedures.

1.1.2. Broken Authentication And Session Management

Đây là kiểu tấn công lỗi xác thực và quản lý phiên làm việc (Broken Authentication And Session Management), bao gồm những đoạn chương trình kiểm tra danh tính và quản lý phiên làm việc của ngọời sử dụng thường hay được làm qua loa không đúng cách. Điều này giúp kẻ thâm nhập có thể ăn cắp mật mã, khóa, mã của các phiên làm việc {session token} hoặc tận dụng những lỗi khác để giả mạo danh tính các người dùng khác.

Quản lý xác thực và phiên bao gồm tất cả các khía cạnh xử lý xác thực và quản lý phiên làm việc. Xác thực là một khía cạnh quan trọng của quá trình này, nhưng ngay cả các cơ chế xác thực vững chắc cũng có thể bị suy yếu do chức năng quản lý có khe hở, bao gồm thay đổi mật khẩu, ghi nhớ mật khẩu, thay đổi tài khoản và nhiều chức năng khác. Vì các cuộc tấn công có thể xảy ra với nhiều ứng dụng web nên chức năng quản lý tài khoản yêu cầu xác thực lại ngay cả khi người sử dụng có phiên làm việc hợp lệ. Một phương pháp xác thực mạnh mẽ hơn là sử dụng phần mềm và phần cứng tuy nhiên phương pháp này rất tốn kém.

Các ứng dụng web thường phải thiết lập phiên để theo dõi các luồng yêu cầu từ người dùng, giao thức HTTP không hỗ trợ khả năng này vì vậy các ứng dụng web phải tự tạo ra nó. Thông thường môi trọờng ứng dụng web cung cấp khả năng phiên nhưng nhiều nhà phát triển thích tự họ tạo ra một thẻ phiên của riêng họ. Tuy nhiên, chức năng ứng dụng liên quan đến quản lý xác thực và phiên làm việc thường thực hiện một cách chính xác, điều này cho phép kẻ tấn công lấy được mật khẩu, khóa, thẻ phiên hoặc khai thác lỗ hổng để thực hiện các giả mạo danh tính người dùng.

1.1.3. Cross Site Scripting (XSS)

Kiểu tấn công thực thi mã script xấu Cross-Site Scripting (XSS) là một trong những kĩ thuật tấn công phổ biến nhất hiên nay, đồng thời nó cũng là một trong những vấn đề bảo mật quan trọng đối với các nhà phát triển web và cả những người sử dụng

web. Bất kì một website nào cho phép người sử dụng đăng thông tin mà không có sự kiểm tra chặt chẽ các đoạn mã nguy hiểm thì đều có thể tiềm ẩn các lỗi XSS.

Cross-Site Scripting hay còn được gọi tắt là XSS (thay vì gọi tắt là CSS để tránh nhầm lẫn với CSS-Cascading Style Sheet của HTML) là một kĩ thuật tấn công bằng cách chèn vào các website động (ASP, PHP, CGI, JSP ...) những thẻ HTML hay những đoạn mã script nguy hiểm có thể gây nguy hại cho những người sử dụng khác. Trong đó, những đoạn mã nguy hiểm được chèn vào hầu hết được viết bằng các Client-Site Script như JavaScript, JScript, DHTML và cũng có thể là cả các thẻ HTML. Kĩ thuật tấn công XSS đã nhanh chóng trở thành một trong những lỗi phổ biến nhất của Web Applications và mối đe dọa của chúng đối với người sử dụng ngày càng lớn.

1.1.4. Insecure Direct Object References

Kiểu tấn công đối tượng tham chiếu trực tiếp không an toàn (Insecure Direct Object References), xảy ra khi người phát triển để lộ một tham chiếu đến những đối tượng trong hệ thống như các tập tin, thư mục hay chìa khóa dữ liệu. Nếu chúng ta không có một hệ thống kiểm tra truy cập, kẻ tấn công có thể lợi dụng những tham chiếu này để truy cập dữ liệu một cách trái phép.

Việc phân quyền yếu cho phép người dùng có thể truy cập dữ liệu của người khác. Hacker có thể xác định được cấu trúc truy vấn gửi đến server và có thể nhanh chóng thu nhập dữ liệu như Credit Card, mã khách hàng, thông tin cá nhân.

1.1.5. Security Misconfiguration

Kiểu tấn công sai sót trong cấu hình bảo mật (Security Misconfiguration), như là một cơ chế an ninh tốt cần phải định nghĩa những hiệu chỉnh về an ninh và triển khai nó cho các ứng dụng, máy chủ ứng dụng, máy chủ web, máy chủ dữ liệu và các ứng dụng nền tảng.

Tất cả những thiết lập nên được định nghĩa, thực hiện và bảo trì bởi vì rất nhiều

hệ thống không được triển khai với thiết lập an toàn mặc định. Các hiệu chỉnh cũng bao gồm cập nhật phần mềm và những thư viện được sử dụng bởi ứng dụng.

1.1.6. Sensitive Data Exposure

Kiểu tấn công phơi bày các dữ liệu nhạy cảm (Sensitive Data Exposure), bao gồm nhiều ứng dụng web không bảo vệ dữ liệu nhạy cảm nhoị thẻ tín dụng, mã số thuế và những mã xác thực bí mật bằng các phoịơng thức mã hóa hay băm (hashing). Kẻ tấn công có thể ăn cắp hay thay đổi những dữ liệu nhạy cảm này và tiến hành hành vi trộm cắp, gian lận thẻ tín dụng, v.v...

1.1.7. Missing Function Level Access Control

Kiểu tấn công thiếu chức năng điều khiển truy cập (Missing Function Level Access Control) bao gồm gần như tất cả các ứng dụng web kiểm tra quyền truy cập cấp độ chức năng trước khi thực hiện chức năng mà có thể nhìn thấy trong giao diện ngoịời dùng. Tuy nhiên, các ứng dụng cần phải thực hiện kiểm tra kiểm soát truy cập tương tự trên máy chủ khi mỗi chức năng đoịợc truy cập. Nếu yêu cầu không được xác nhận, kẻ tấn công sẽ có thể giả mạo yêu cầu để truy cập vào chức năng trái phép.

1.1.8. Cross-Site Request Forgery (CSRF)

Kiểu tấn công giả mạo yêu cầu (CSRF) là kiểu tấn công này ép buộc trình duyệt web của một người dùng đã đăng nhập gửi những yêu cầu các HTTP giả bao gồm cookie của phiên truy cập và những thông tin tự động khác bao gồm thông tin đăng nhập đến một ứng dụng web. Điều này, cho phép kẻ tấn công buộc trình duyệt web tạo ra những yêu cầu đến ứng dụng web mà ứng dụng không thể biết đây là những yêu cầu giả mạo của kẻ tấn công.

1.1.9. Using Components with Known Vulnerabilities

Kiểu tấn công sử dụng thành phần đã tồn tại lỗ hổng (Using Components with Known Vulnerabilities) bao gồm các lổ hổng có thể có trong các thành phần (thành phần phát triển ứng dụng) như các thư viện, các framework, và mô-đun phần mềm

khác. Các thành phần này gần như luôn luôn chạy với quyền cao nhất trong hệ thống. Vì vậy, nếu bị khai thác, các thành phần này có thể gây mất dữ liệu nghiêm trọng.

Các ứng dụng sử dụng các thành phần tồn tại lỗ hổng có thể làm suy yếu phòng thủ của hệ thống, cho phép một loạt các cuộc tấn công và ảnh hưởng đến hệ thống.

1.1.10. Unvalidated Redirects and Forwards

Kiểu tấn công chuyển hướng và chuyển tiếp thiếu kiểm tra (Unvalidated Redirects and Forwards) là kiểu tấn công ứng dụng web thường chuyển hướng, chuyển tiếp người dùng đến những trang web, website khác và sử dụng những thông tin thiếu tin cậy để xác định trang đích đến. Nếu không được kiểm tra một cách cẩn thận, kẻ tấn công có thể lợi dụng để chuyển hướng nạn nhân đến các trang web lừa đảo hay trang web chứa phần mềm độc hại, hoặc chuyển tiếp để truy cập các trang trái phép.

1.2. Phương pháp phòng chống tấn công trên web

1.2.1. Các phương pháp phòng chống tấn công web phổ biến

❖ Phương pháp phòng chống tấn công SQL injection

SQL Injection attack [13] gây ra nhiều tác hại tùy thuộc vào môi trường và cách cấu hình hệ thống. Nếu ứng dụng sử dụng quyền dbo (quyền của người sở hữu CSDL - owner) khi thao tác dữ liệu, nó có thể xóa toàn bộ các bảng dữ liệu, tạo các bảng dữ liệu mới,… Nếu ứng dụng sử dụng quyền sa (quyền quản trị hệ thống), nó có thể điều khiển toàn bộ hệ quản trị CSDL và với quyền hạn rộng lớn như vậy nó có thể tạo ra các tài khoản người dùng bất hợp pháp để điều khiển hệ thống của bạn.

Để phòng tránh các nguy cơ có thể xảy ra, cần bảo vệ các câu truy vấn SQL là bằng cách kiểm soát chặt chẽ tất cả các dữ liệu nhập nhận được từ đối tượng Request (Request, Request.QueryString, Request.Form, Request.Cookies, and Request.ServerVariables).

Trong trường hợp dữ liệu nhập vào là chuỗi, lỗi xuất phát từ việc có dấu nháy đơn trong dữ liệu. Để tránh điều này, thay thế các dấu nháy đơn bằng hàm Replace để thay thế bằng 2 dấu nháy đơn:

```
p_strUsername = Replace(Request.Form("txtUsername"), "'", "''") p_strPassword = Replace(Request.Form("txtPassword"), "'", "''")
```

Trong trường hợp dữ liệu nhập vào là số, lỗi xuất phát từ việc thay thế một giá trị được tiên đoán là dữ liệu số bằng chuỗi chứa câu lệnh SQL bất hợp pháp. Để tránh điều này, đơn giản hãy kiểm tra dữ liệu có đúng kiểu hay không:

```
p_lngID = CLng(Request("ID"))
```

Như vậy, nếu người dùng truyền vào một chuỗi, hàm này sẽ trả về lỗi ngay lập tức.

Ngoài ra để tránh các nguy cơ từ SQL Injection attack, nên chú ý loại bỏ bất kì thông tin kĩ thuật nào chứa trong thông điệp chuyển xuống cho người dùng khi ứng dụng có lỗi. Các thông báo lỗi thông thường tiết lộ các chi tiết kĩ thuật có thể cho phép kẻ tấn công biết được điểm yếu của hệ thống. Cuối cùng, để giới hạn mức độ của SQL Injection attack, nên kiểm soát chặt chẽ và giới hạn quyền xử lí dữ liệu đến tài khoản người dùng mà ứng dụng web đang sử dụng. Các ứng dụng thông thường nên tránh dùng đến các quyền như dbo hay sa. Quyền càng bị hạn chế, thiệt hại càng ít.

❖ Phương pháp phòng chống tấn công Cross Site Scripting (XSS)

Tấn công XSS [13] được coi là một trong những loại nguy hiểm và rủi ro nhất, nên cần chuẩn bị các phương pháp ngăn ngừa. XSS là cuộc tấn công phổ vì vậy có nhiều cách để ngăn chặn nó.

Các phương pháp phòng ngừa chính được sử dụng phổ biến bao gồm:

- Data validation
- Filtering

- Escaping

Bước đầu tiên trong công tác phòng chống tấn công này là xác thực đầu vào. Mọi thứ, được nhập bởi người dùng phải được xác thực chính xác, bởi vì đầu vào của người dùng có thể tìm đường đến đầu ra. Xác thực dữ liệu có thể được đặt tên làm cơ sở để đảm bảo tính bảo mật của hệ thống. Xác thực không cho phép đầu vào không phù hợp. Vì vậy nó chỉ giúp giảm thiểu rủi ro, nhưng có thể không đủ để ngăn chặn lỗ hổng XSS có thể xảy ra.

Một phương pháp ngăn chặn tốt khác là lọc đầu vào của người dung bằng cách tìm kiếm các từ khóa nguy hiểm trong mục nhập của người dùng và xóa chúng hoặc thay thế chúng bằng các chuỗi trống. Những từ khóa đó có thể là: thẻ <script> </ script>; lệnh Javascript; đánh dấu HTML.

Lọc đầu vào là phương pháp đơn giản để thực hiện. Nó có thể được thực hiện theo nhiều cách khác nhau. Như: bởi các developers đã viết mã phía server; thư viện ngôn ngữ lập trình thích hợp đang được sử dụng.

Một phương pháp phòng ngừa khác có thể là ký tự Escape. Trong trường hợp này, các ký tự thích hợp đang được thay đổi bằng các mã đặc biệt.

Ví dụ: <ký tự Escape như & # 60.

❖ Phương pháp phòng chống tấn công Cross-Site Request Forgery (CSRF)

Dựa trên nguyên tắc của CSRF [13] "lừa trình duyệt của người dùng (hoặc người dùng) gửi các câu lệnh HTTP", các kĩ thuật phòng tránh sẽ tập trung vào việc tìm cách phân biệt và hạn chế các câu lệnh giả mạo.

Phía user: để phòng tránh trở thành nạn nhân của các cuộc tấn công CSRF, người dùng internet nên thực hiện một số lưu ý sau:

- Nên thoát khỏi các website quan trọng: tài khoản ngân hàng, thanh toán trực tuyến, các mạng xã hội, gmail, yahoo… khi đã thực hiện xong giao dịch hay các công việc cần làm. (Check - email, checkin…)

- Không nên click vào các đường dẫn mà bạn nhận được qua email, qua facebook… Khi bạn đưa chuột qua 1 đường dẫn, phía dưới bên trái của trình duyệt thường có địa chỉ website đích, bạn nên lưu ý để đến đúng trang mình muốn.
- Không lưu các thông tin về mật khẩu tại trình duyệt của mình (không nên chọn các phương thức "đăng nhập lần sau", "lưu mật khẩu"…
- Trong quá trình thực hiện giao dịch hay vào các website quan trọng không nên vào các website khác, có thể chứa các mã khai thác của kẻ tấn công.

Phía server: có nhiều lời khuyến cáo được đưa ra, tuy nhiên cho đến nay vẫn chưa có biện pháp nào có thể phòng chống triệt để CSRF. Sau đây là một vài kĩ thuật sử dụng.

Lựa chọn việc sử dụng GET VÀ POST: sử dụng GET và POST đúng cách. Dùng GET nếu thao tác là truy vấn dữ liệu. Dùng POST nếu các thao tác tạo ra sự thay đổi hệ thống (theo khuyến cáo của W3C tổ chức tạo ra chuẩn http) Nếu ứng dụng của bạn theo chuẩn RESTful, bạn có thể dùng thêm các HTTP verbs, như PATCH, PUT hay DELETE.

Sử dụng captcha, các thông báo xác nhận: captcha được sử dụng để nhận biết đối tượng đang thao tác với hệ thống là con người hay không? Các thao tác quan trọng như "đăng nhập" hay là "chuyển khoản","thanh toán" thường là hay sử dụng captcha. Tuy nhiên, việc sử dụng captcha có thể gây khó khăn cho một vài đối tượng người dùng và làm họ khó chịu. Các thông báo xác nhận cũng thường được sử dụng, ví dụ như việc hiển thị một thông báo xác nhận "bạn có muốn xóa hay k" cũng làm hạn chế các kĩ thuật Cả hai cách trên vẫn có thể bị vượt qua nếu kẻ tấn công có một kịch bản hoàn hảo và kết hợp với lỗi XSS.

Sử dụng token: tạo ra một token tương ứng với mỗi form, token này sẽ là duy nhất đối với mỗi form và thường thì hàm tạo ra token này sẽ nhận đối số là"SESSION" hoặc được lưu thông tin trong SESSION. Khi nhận lệnh HTTP POST về, hệ thống sẽ thực hiên so khớp giá trị token này để quyết định có thực hiện hay không. Mặc định trong Rails, khi tạo ứng dụng mới:

```
class ApplicationController < ActionController::Base
  protect_from_forgery with: :exception
end
```

Khi đó tất cả các form và Ajax request được tự động thêm sercurity token generate bởi Rails. Nếu security token không khớp, exception sẽ được ném ra.

Sử dụng cookie riêng biệt cho trang quản trị: Một cookie không thể dùng chung cho các domain khác nhau,chính vì vậy việc sử dụng "admin.site.com" thay vì sử dụng "site.com/admin" là an toàn hơn.

Kiểm tra REFERRER: kiểm tra xem các câu lệnh gửi đến hệ thống xuất phát từ đâu. Một ứng dụng web có thể hạn chế chỉ thực hiện các lệnh http gửi đến từ các trang đã được chứng thực. Tuy nhiên cách làm này có nhiều hạn chế và không thật sự hiệu quả.

Kiểm tra IP: một số hệ thống quan trọng chỉ cho truy cập từ những IP được thiết lập sẵn.

1.2.2. Một số phương pháp nâng cao bảo mật hệ thống máy chủ website

❖ Cập nhật các phiên bản cho website thường xuyên

Hacker ngày càng tinh vi, không có gì có thể bảo đảm chắc chắn rằng hacker không thể vượt qua tường rào bảo mật để xâm nhập vào website của chúng ta. Để nâng cao khả năng bảo vệ cho các dữ liệu website thì việc cập nhật phiên bản mới cho website thường xuyên là yếu tố bắt buộc của các nhà quản trị web [16].

Hiểu đơn giản rằng website của bạn đang được bảo mật an toàn trong phiên bản cũ, nhưng đó chỉ là trước khi hacker tìm ra được cách vượt qua tường bảo mật. Đến khi họ tìm ra được lỗ hổng bảo mật mà website của bạn vẫn chưa cập nhật phiên bản mới thì dĩ nhiên hacker sẽ dễ dàng vượt qua lớp bảo mật để tiếp cận dữ liệu web. Thế nhưng nếu lúc đó bạn đã cập nhật phiên bản mới thì phiên bản cũ tin tặc sẽ không thể lạm dụng được nữa. Chính vì vậy, cập nhật phiên bản thường xuyên chính là công việc quan trọng nhất giữ cho website của bạn tránh khỏi nguy cơ bị đánh cắp dữ liệu.

❖ **Quét virut và sao lưu (back up) dữ liệu thường xuyên**

Cũng giống như việc cập nhật phiên bản mới việc quét virut để bảo đảm an toàn website là một công việc cần thiết và thường xuyên. Quét virut giúp trang web của bạn nhanh chóng phát hiện và loại bỏ những mã độc có nguy cơ làm thông tin bị rò rỉ.

Bên cạnh đó, việc sao lưu dữ liệu thường xuyên không những giúp dữ liệu được lưu trữ an toàn mà còn giúp cho các hacker khó khăn trong việc truy cập vào chúng. Các công việc này không quá khó khăn mà lại không tốn thời gian, nhưng có thể vì thế mà các nhà quản trị mạng vẫn thường chủ quan và bỏ qua chúng. Hãy xem xét và cập nhật chúng nhanh nhất có thể để đảm bảo an toàn cho website của doanh nghiệp bạn.

❖ **Bảo mật website bằng SSL**

Một hình thức bảo mật bằng cách mã hóa các lưu lượng truy cập tương tác giữa trinh duyệt website và máy chủ, sau đó quản lý chúng an toàn, được gọi là bảo mật SSL (secure Sokets Layer) [18]. Tính năng này giúp hỗ trợ website nhạy cạm hơn với các lượt vi phạm bảo mật. Chúng góp phần ngăn chặn bên thứ ba xâm nhập vào các thông tin cá nhân như: thẻ tín dụng, tài khoản tài chính, mật khẩu truy cập…

Vì vậy, khi kích hoạt website hãy suy nghĩ đến việc cài đặt cho chúng thêm một lớp bảo mật bằng SSL để đảm bảo an toàn cho website của bạn.

❖ **Tường lửa ứng dụng web (WAF) – giải pháp bảo mật trong hệ thống**

Tường lửa ứng dụng web **(**WAF **–** Web Application Firewall) là một giải pháp nhằm giúp website tránh khỏi các lỗ hổng bảo mật. Nó được thiết kế dưới dạng phần cứng cài đặt trên máy chủ cung cấp các mô hình theo dõi thông tin được truyền dưới giao thức HTTP/HTPPS [16].

WAF có khả năng tự động hóa tiêu diệt virut, phân tích và cảnh báo nhà quản trị web những nguy cơ lỗ hổng bị xâm nhập, phòng chống các mã độc và các cuộc tấn công kỹ thuật khác. Nhờ đó chúng bảo vệ toàn diện trung tâm dữ liệu, các kết nối

loT đến đám mây và hệ thống chống thất thoát dữ liệu giúp công ty, doanh ngiệp bảo đảm an toàn các thông tin nhạy cảm ra bên ngoài. Đây là một phần không thể thiếu trong hệ thống bảo mật website của doanh nghiệp kinh doanh.

❖ **Sử dụng triệt để các Plugin hỗ trợ bảo mật website**

Khi bạn muốn tích hợp một số phần mềm bảo mật cho website nhưng chúng lại không tương thích với phiên bản bạn đang sử dụng thì plugin hỗ trợ là một giải pháp linh hoạt. Tương tự như vậy, sử dụng Plugin hỗ trợ bảo mật website chính là một cách bảo mật thông minh [16].

Tuy nhiên, đồ có giá trị bảo mật cao thì lại không bao giờ miễn phí. Ngoài một số Plugin do nhà cung cấp miễn phí thì bạn sẽ phải trả một khoản phí để có thể sử hữu những Plugin bảo mật này. Dĩ nhiên số tiền bỏ rà này không là gì so với việc thiệt hại do thất thoát dữ liệu từ website.

❖ **Giới hạn IP truy cập và giới hạn phân quyền đăng nhập**

Giới hạn quyền truy cập và các IP truy cập vào website là một sự phòng bị thông minh [16]. Khi một website có quá nhiều quản trị viên, hacker sẽ theo dõi và tiến hành hack một tài khoản của quản trị viên có bảo mật kém an toàn. Lúc này nếu như website có khả năng bảo mật kém thì việc lấy đi những thông tin quý giá rất dễ xảy ra. Trong trường hợp bị chơi xấu, hacker sẽ làm cho website đang hoạt động rất tốt trở nên vi phạm điều khoản Google. Và tất nhiên khi bot Google quét ra vi phạm điều khoản thì trang web này sẽ vĩnh viễn biến mất.

Kết luận chương 1

Trong chương 1, luận văn đã khảo sát về các nguy cơ mất an toàn thông tin Website cũng như tìm hiểu về kĩ thuật tấn công vào các lỗ hổng phổ biến hiện nay (Top 10 OWAPS). Từ đó đưa ra một số phương pháp phòng chống tấn công khi xây dựng Website.

Vấn đề phát hiện sớm các cuộc tấn công Website để có các biện pháp phòng ngừa hữu hiệu đóng một vai trò hết sức quan trọng. Chương tiếp theo, luận văn sẽ nghiên cứu các phương pháp phát hiện tấn công trên Website dựa trên kĩ thuật phân tích hành vi.

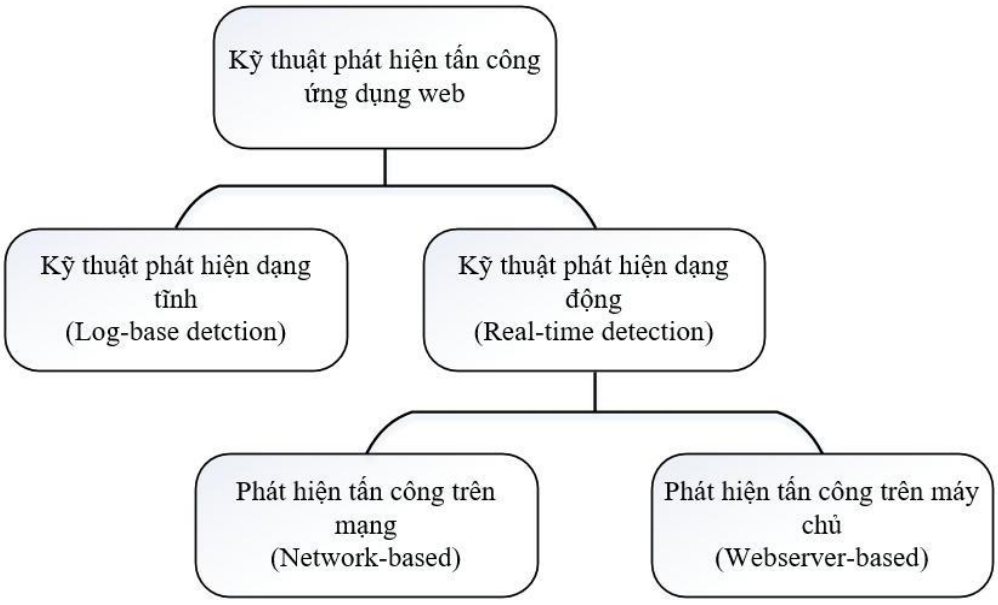
CHƯƠNG 2: PHƯƠNG PHÁP PHÁT HIỆN TẤN CÔNG TRÊN WEB DỰA TRÊN KỸ THUẬT PHÂN TÍCH HÀNH VI

Tóm tắt chương: Trong chương 2, luận văn sẽ nghiên cứu về một số phương pháp phát hiện tấn công web bao gồm kỹ thuật phát hiện và các công cụ mã nguồn mở hỗ trợ phát hiện tấn công web. Ngoài ra, trong chương 2 sẽ trình bày về phương pháp phát hiện tấn công web dựa trên kỹ thuật phân tích hành vi. Theo đó, kỹ thuật phân tích hành vi để phát hiện tấn công web bao gồm các quy trình: Nghiên cứu và trích xuất hành vi bất thường từ bộ dữ liệu về tấn công web (bộ dữ liệu CSIC 2010), lựa chọn và ứng dụng thuật toán học máy nhằm phân loại hành vi tấn công và hành vi bình thường lên web (trong luận văn này, học viên sẽ sử dụng một số thuật toán học máy có giám sát như: SVM, Random forest, Decision Tree).

2.1. Giới thiệu về phương pháp phát hiện tấn công web

2.1.1. Một số phương pháp phát hiện tấn công web

Thông thường có hai phương pháp tiếp cận phát hiện tấn công web là sử dụng kỹ thuật phát hiện dạng tĩnh và kỹ thuật phát hiện dạng động.



Hình 2.1: Phân loại phương pháp phát hiện tấn công web

❖ **Kỹ thuật phát hiện tĩnh**

Phát hiện xâm nhập dạng tĩnh [13] dựa trên nhật ký truy cập chi tiết của máy chủ web, nơi mà các thông tin về tất cả các yêu cầu truy cập từ phía người dùng được lưu lại. Ngoài ra, máy chủ web cũng có thể tạo ra các nhật ký truy cập theo các định dạng đặc biệt để kiểm soát dữ liệu được thu thập. Phát hiện xâm nhập dạng tĩnh được thực hiện chỉ sau khi giao dịch diễn ra. Vì vậy, việc ngăn chặn tấn công tại thời điểm đang bị tấn công là không thể.

❖ **Kỹ thuật phát hiện động**

Phát hiện xâm nhập dạng động (thời gian thực) [13] không chỉ có thể phát hiện xâm nhập, mà còn có thể phản ứng lại. Để xác định nhanh chóng và hiệu quả, việc lọc và phát hiện của hệ thống phát hiện xâm nhập hoạt động theo các mô hình:

Mô hình bảo mật Negative (Blacklist) thường được sử dụng nhiều hơn. Hệ thống chỉ cần xác định tấn công dựa vào một mẫu các tấn công nguy hiểm đã biết trước và cấu hình để loại bỏ nó.

Một mô hình bảo mật Positive (Whitelist): là một phương pháp tốt hơn để xây dựng các chính sách áp dụng trong hoạt động tường lửa. Trong lĩnh vực bảo mật ứng dụng web, một mô hình bảo mật Positive liệt kê tất cả các mô tả trong ứng dụng. Mỗi mô tả cần phải xác định như sau:

- Các phương thức yêu cầu được phép (ví dụ GET/POST hoặc chỉ POST)
- Content-Type được phép
- Content-Length được phép
- Các thông số được phép
- Những thông số là bắt buộc và là tùy chọn
- Kiểu dữ liệu tất cả các tham số (ví dụ văn bản hoặc số nguyên)
- Thông số bổ sung (nếu có).

Các lập trình viên, người quản trị hệ thống có nghĩa vụ phải thực hiện cung cấp danh sách các mô tả trên. Tuy nhiên, trong thực tế thông thường thì không thực hiện điều này. Sử dụng mô hình bảo mật Positive là tốt hơn nếu người quản trị hệ thống dành đủ thời gian để phát triển nó. Một khó khăn khác của phương pháp này là các ứng dụng thay đổi mô hình liên tục. Do đó, sẽ cần phải cập nhật mô hình mỗi khi một mô tả mới được thêm vào các ứng dụng hoặc thay đổi mới.

Mô hình bảo vệ dạng Rule-based và Anomaly-based:

Rule-based: mọi yêu cầu HTTP là đối tượng của một loạt các kiểm tra, trong đó mỗi kiểm tra bao gồm một hoặc nhiều quy tắc kiểm tra. Kết quả của quá trình kiểm tra là một yêu cầu có thỏa hay không? nếu thỏa kết quả là True, không thỏa là False. Sau đó, các yêu cầu HTTP có kết quả là True bị xem là mã độc và bị từ chối. Rule-based thường được sử dụng trong mô hình an toàn Negative.

Rule-based dễ xây dựng và sử dụng và có hiệu quả khi được sử dụng để bảo vệ chống lại các tấn công đã được biết đến hoặc xây dựng một chính sách tùy chỉnh. Tuy nhiên, do phải biết về các chi tiết, cụ thể của tất cả các mối đe dọa, nên phương pháp này phải dựa trên cơ sở dữ liệu các quy tắc (hoặc cơ sở dữ liệu mẫu tấn công). Các nhà cung cấp IDS duy trì, phát triển cơ sở dữ liệu quy tắc và phân phối thông qua chức năng cập nhật tự động của IDS.

Cách tiếp cận này ít có khả năng có thể để bảo vệ các ứng dụng tùy chỉnh hoặc để bảo vệ trước tấn công khai thác zero-day (khai thác các lỗ hổng chưa được biết đến công khai). Đối với các vấn đề này thì IDS theo hướng anomaly-base thực hiện tốt hơn.

Rule-based và Signature-based về cơ bản là giống nhau. Signature-based thường phát hiện xâm nhập bằng cách kiểm tra các chuỗi hoặc biểu thức trong lưu lượng dữ liệu với các mẫu có trùng khớp hay không. Trong khi đó, Rule-based cho phép thực hiện các phép toán logic phức tạp hơn, cũng như có thể kiểm tra đến một phần cụ thể của giao dịch web được đặt ra trong quy tắc.

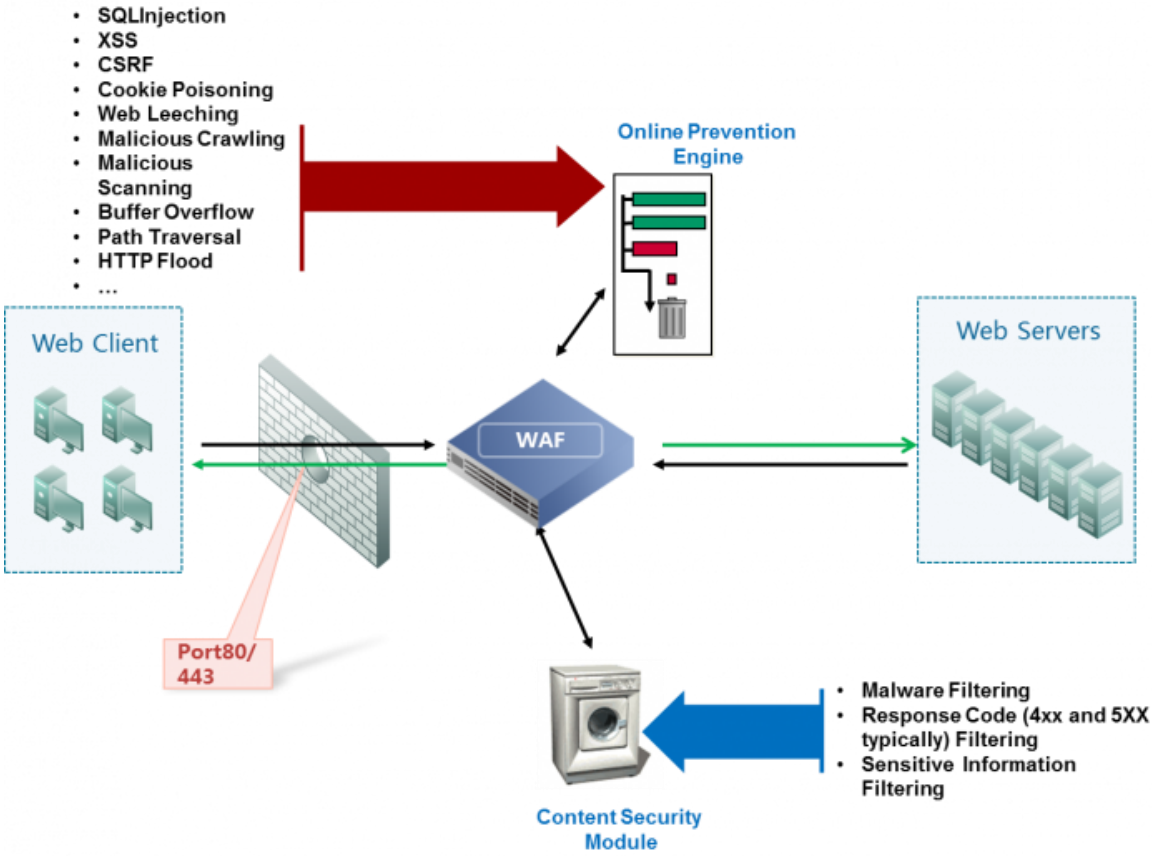
Anomaly-based: là hướng tiếp cận dựa trên các dấu hiệu bất thường với ý tưởng xây dựng một lớp bảo vệ và nó sẽ quan sát lưu lượng dữ liệu hợp pháp của ứng dụng và sau đó xây dựng một mô hình thống kê để đánh giá lưu lượng truy cập nhằm chống lại các tấn công. Về lý thuyết, một khi được đào tạo, một hệ thống Anomaly-based sẽ phát hiện bất cứ điều gì khác thường. Với phương pháp Anomaly-based thì không cần thiết sử dụng phương pháp Rule-based và khai thác zero-day không phải là một vấn đề đáng quan tâm. Tuy nhiên, hệ thống Anomaly-based rất khó để xây dựng và ít được sử dụng. Bởi vì người dùng không hiểu cách hệ thống Anomaly-based làm việc do đó không tin tưởng vào hệ thống như vậy, vì thế làm cho nó ít phổ biến.

2.1.2. Công cụ phát hiện tấn công web

2.1.2.1. Sử dụng tường lửa WAF

Tường lửa ứng dụng Web WAF (hay còn gọi là Web application firewall) được tạo ra để kiểm soát sự truy cập của các mạng không an toàn. Tất cả các dữ liệu được truyền vào hoặc gửi ra đều phải đi qua tường lửa và được kiểm soát theo chính sách bảo mật định sẵn [3].

WAF là giải pháp nhằm bảo vệ cho các ứng dụng mà bị những lỗi mã độc hay bảo mật vừa được đề cập ở trên. WAF là một thiết bị phần cứng hay phần mềm được thiết lập sẵn trên máy chủ để theo dõi các thông tin được truyền qua giao thức http/https khi người dùng truy cập vào máy chủ web của một web bất kì. WAF sẽ thực hiện các chính sách bảo mật dựa vào các dấu hiệu tấn công, các giao thức tiêu chuẩn và các lưu lượng truy cập ứng dụng web bất thường. Đây là điều mà các tường lửa thông thường khác không làm được.



Hình 2.2: Mô hình Web application firewal

(Nguồn: Tìm hiểu về Tường lửa ứng dụng Web WAF (Web Application Firewall - Internet)

Cách cài đặt WAF cũng như các ứng dụng firewall khác là sau tường lửa mạng và trước máy chủ ứng dụng web – hướng đi chủ yếu của các nguồn thông tin. Tuy nhiên, đôi khi cũng có ngoại lệ khi WAF chỉ được dùng để giám sát cổng đang mở trên máy chủ web. Cũng có một số trường hợp như cài đặt chương trình trực tiếp lên máy chủ và thực hiện các chức năng tương tự như các thiết bị WAF là giám sát các lưu động đến và ra khỏi ứng dụng web.

❖ Mô hình bảo mật của Web application firewall

Có 2 mô hình hoạt động chủ yếu của WAF, đó là Positive và Negative.

- Mô hình Positive: chỉ cho phép một lượng lưu hợp lệ đi qua, còn lại thì sẽ bị chặn.
- Mô hình Negative: cho phép tất cả lưu lượng đi qua nhưng sẽ chặn lại lưu lượng nào và ứng dụng này cho là nguy hại.

Trong một vài trường hợp thì WAF cung cấp cả hai mô hình trên nhưng thông thường chỉ cung cấp một trong hai mô hình, một điểm lưu ý là mô hình Postitive thì đòi hỏi nhiều cấu hình và tùy chỉnh, còn mô hình Negative chủ yếu dựa vào khả năng học hỏi và phân tích hành vi của lưu lượng mạng.

❖ **Mô hình hoạt động của WAF**

WAF hoạt động với một số mô hình riêng biệt, dưới đây là một trong những mô hình ví dụ:

- Layer 2 Brigde: đối với mô hình này, WAF có vai trò như mộ switch ở lớp 2. Mô hình này hỗ trợ mạng của bạn hoạt động với hiệu năng cao nhưng vẫn không làm thay đổi mạng, tuy nhiên với mô hình này WAF không thẻ cung cấp những dịch vụ cao cấp khác mà các mô hình khác mang lại.
- Host/Server Based: đối với mô hình này, WAF được cài đặt trực tiếp lên máy chủ web. Host based thì không cung cấp các tính năng như loại WAF network based. Tuy nhiên, mô hình này có thể khắc phục điểm yếu mà các mô hình network based có. Ngoài ra cũng làm tăng mức độ tải của máy chủ web.
- Reserve Proxy: đây là mô hình được sử dụng phổ biến khi triển khai WAF. Đối với mô hinh này, WAF sẽ theo dõi và giám sát tất cả nguồn thông tin đi vào ứng dụng web, thay vì cho các đỉa chỉ IP ngoài gửi yêu cầu trực tiếp đến máy chủ web thì trong mô hình này, WAF sẽ đứng ra làm trung gian gửi yêu cầu cho máy chủ web rồi trả lại kết quả cho địa chỉ IP kia.
- Transparent Proxy: mô hình này tương tự như Reserve Proxy, nhưng điểm khác biệt là WAF sẽ không đứng ra làm trung gian cũng như không cung cấp những dịch vụ như Reserve Proxy.

2.1.2.2. Sử dụng hệ thống phát hiện xâm nhập

Hệ thống phát hiện xâm nhập (IDS- Intrusion Detection System) [1] là một hệ thống nhằm phát hiện các hành động xâm nhập tấn công vào mạng. IDS phát hiện

dựa trên các dấu hiệu đặc biệt về các nguy cơ đã biết hay dựa trên so sánh lưu thông mạng hiện tại với basedline để tìm ra các dấu hiệu khác thường.

Phát hiện xâm nhập trái phép là một công việc đầy khó khăn do ảnh hưởng của sự tăng trưởng nhanh chóng các kết nối mạng, môi trường máy tính không đồng nhất, nhiều giao thức truyền thông và sự phân loại đáng kể của các ứng dụng thông dụng và độc quyền. Hầu hết các kỹ thuật IDS được xây dựng dựa trên sự khác biệt ứng xử của kẻ xâm nhập so với người dùng hợp lệ.

Một IDS có nhiệm vụ phân tích các gói tin mà Firewall cho phép đi qua, tìm kiếm các dấu hiệu đã biết mà không thể kiểm tra hoặc ngăn chặn bởi Firewall. Sau đó cung cấp thông tin và đưa ra các cảnh báo cho các quản trị viên.

IDS cung cấp thêm cho việc bảo vệ an toàn thông tin mạng một mức độ cao hơn. Nó được đánh giá giá trị giống như Firewall và VPN là ngăn ngừa các cuộc tấn công mà IDS cung cấp sự bảo vệ bằng cách trang bị cho bạn thông tin về cuộc tấn công. Bởi vậy, IDS có thể thỏa mãn nhu cầu về an toàn hệ thống của bạn bằng cách cảnh báo về khả năng các cuộc tấn công và đôi khi ngoài những thông báo chính xác thì chúng cũng đưa ra một số cảnh báo chưa đúng.

❖ Chức năng của IDS

Nhìn chung, IDS không tự động cấm các cuộc tấn công hoặc là ngăn chặn những kẻ khai thác một cách thành công, tuy nhiên , một sự phát hiện mới nhất của IDS đó là hệ thống ngăn chặn xâm phập đã có thể thực hiện nhiều vai trò hơn và có thể ngăn chặn các cuộc tấn công khi nó xảy ra.

Thực tế, IDS cho chúng ta biết rằng mạng đang bị nguy hiểm. Điều quan trọng để nhận ra đó là một vài cuộc tấn công vào mạng đã thành công nếu hệ thống không có IDS.

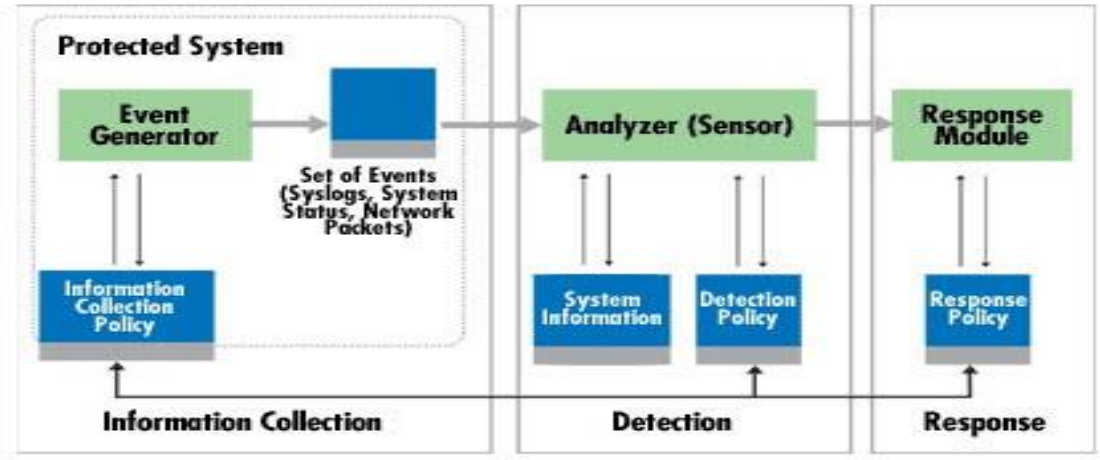
Hệ thống phát hiện xâm nhập cho phép các tổ chức bảo vệ hệ thống của họ khỏi những đe doạ với việc gia tăng kết nối mạng và sự tin cậy của hệ thống thông tin, bổ sung những điểm yếu của hệ thống khác... Một số ưu điểm của hệ thống IDS như:

- Bảo vệ tính toàn vẹn của dữ liệu, đảm bảo sự nhất quán của dữ liệu trong hệ thống. Các biện pháp đưa ra ngăn chặn được thay đổi bất hợp pháp hoặc phá hoại dữ liệu.
- Bảo vệ tính riêng tư, tức là đảm bảo cho người sử dụng khai thác tài nguyên của hệ thống theo đúng chức năng, nhiệm vụ đã được phân cấp, ngăn chặn được sự truy nhập thông tin bất hợp pháp.
- Bảo vệ tính bí mật, giữ cho thông tin không bị lộ ra ngoài. Bảo vệ tính khả dụng, tức la hệ thống luôn sẵn sàng thực hiện yêu cầu truy nhập thông tin của người dùng hợp pháp.
- Cung cấp thông tin về sự truy cập, đưa ra những chính sách đối phó, khôi phục, sửa chữa...

❖ **Kiến trúc của hệ thống phát hiện xâm nhập IDS**

Kiến trúc của hệ thống IDS bao gồm các thành phần chính: Thành phần thu thập gói tin (information collection), thành phần phân tích gói tin (Dectection), thành phần phản hồi (respontion) nếu gói tin đó được phát hiện là một tấn công của tin tặc. Trong 3 thành phần này thì thành phần phân tích gói tin là một thành phần quan trọng nhất và ở thành phần này bộ cảm biến đóng vai trò quyết định nên chúng ta đi xâu vào phân tích bộ cảm biến để hiểu rõ hơn kiến trúc của hệ thống phát hiện xâm nhập là như thế nào.

Bộ cảm biến được tích hợp với thành phần sưu tập dữ liệu – một bộ tạo sự kiện. Cách sưu tập này được xác định bởi chính sách tạo sự kiện để định nghĩa chế độ lọc thông tin sự kiện. Bộ tạo sự kiện (hệ điều hành, mạng, ứng dụng) cung cấp một số chính sách thích hợp cho các sự kiện, có thể là một bản ghi các sự kiện của hệ thống hoặc các gói mạng. Số chính sách này cùng với thông tin chính sách có thể được lưu trong hệ thống được bảo vệ hoặc bên ngoài. Trong trường hợp nào đó, ví dụ khi luồng dữ liệu sự kiện được truyền tải trực tiếp đến bộ phân tích mà không có sự lưu dữ liệu nào được thực hiện. Điều này cũng liên quan một chút nào đó đến các gói mạng.



Hình 2.3: Kiến trúc hệ thống IDS

(Nguồn: Nghiên cứu hệ thống phát hiện xâm nhập mạng trái phép IDS – Internet)

Vai trò của bộ cảm biến là dùng để lọc thông tin và loại bỏ dữ liệu không tương thích đạt được từ các sự kiện liên quan với hệ thống bảo vệ, vì vậy có thể phát hiện được các hành động nghi ngờ. Bộ phân tích sử dụng cơ sở dữ liệu chính sách phát hiện cho mục này. Ngoài ra còn có các thành phần: dấu hiệu tấn công, profile hành vi thông thường, các tham số cần thiết (ví dụ: các ngưỡng). Thêm vào đó, cơ sở dữ liệu giữa các tham số cấu hình, gồm có các chế độ truyền thông với module đáp trả. Bộ cảm biến cũng có cơ sở dữ liệu của riêng nó, gồm dữ liệu lưu về các xâm phạm phức tạp tiềm ẩn (tạo ra từ nhiều hành động khác nhau).

IDS có thể được sắp đặt tập trung (ví dụ như được tích hợp vào trong tường lửa) hoặc phân tán. Một IDS phân tán gồm nhiều IDS khác nhau trên một mạng lớn, tất cả chúng truyền thông với nhau. Nhiều hệ thống tinh vi đi theo nguyên lý cấu trúc một tác nhân, nơi các module nhỏ được tổ chức trên một host trong mạng được bảo vệ.

Vai trò của tác nhân là để kiểm tra và lọc tất cả các hành động bên trong vùng được bảo vệ và phụ thuộc vào phương pháp được đưa ra – tạo phân tích bước đầu và thậm chí đảm trách cả hành động đáp trả. Mạng các tác nhân hợp tác báo cáo đến máy chủ phân tích trung tâm là một trong những thành phần quan trọng của IDS.

DIDS có thể sử dụng nhiều công cụ phân tích tinh vi hơn, đặc biệt được trang bị sự phát hiện các tấn công phân tán. Các vai trò khác của tác nhân liên quan đến khả năng lưu động và tính roaming của nó trong các vị trí vật lý. Thêm vào đó, các tác nhân có thể đặc biệt dành cho việc phát hiện dấu hiệu tấn công đã biết nào đó. Đây là một hệ số quyết định khi nói đến nghĩa vụ bảo vệ liên quan đến các kiểu tấn công mới.

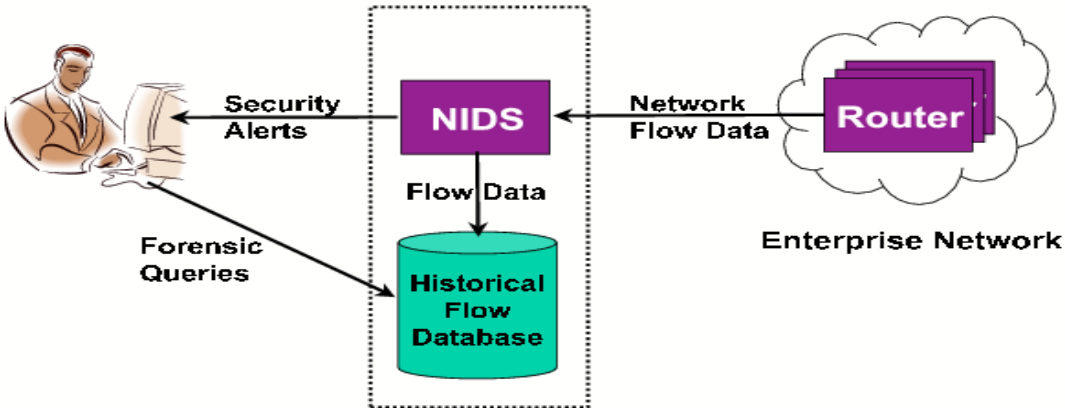
❖ **Cách thức làm việc của IDS**

Cách thức làm việc của phụ thuộc vào từng loại IDS. Ta sẽ xem xét cách thực làm việc của Network - Based IDS và Host - Based IDS, cùng với ưu và nhược điểm của mỗi loại.

- *Network - Based IDS*

Hệ thống IDS dựa trên mạng sử dụng bộ dò và bộ cảm biến cài đặt trên toàn mạng. Những bộ dò này theo dõi trên mạng nhằm tìm kiếm những lưu lượng trùng với mô tả sơ lược được định nghĩa hay là những dấu hiệu. Những bộ cảm biến thu nhận và phân tích lưu lượng trong thời gian thực. Khi nhận được mẫu lưu lượng hay dấu hiệu, bộ cảm biến gửi cảnh báo đến trạm quản trị và có thể được cấu hình nhằm tìm ra biện pháp ngăn chặn những xâm nhập xa hơn. NIDS là tập nhiều sensor được cài đặt ở toàn mạng để theo dõi những gói tin trong mạng so sánh với mạng được định nghĩa để phát hiện đó là tấn công hay không.

NIDS được đặt giữa kết nối hệ thống mạng bên trong và hệ thống mạng bên ngoài để giám sát toàn bộ lưu lượng vào ra. Có thể là một thiết bị phần cứng riêng biệt được thiết lập sẵn hay phần mềm cài đặt trên máy tinh, chủ yếu dùng để đo lưu lượng mạng được sử dụng.

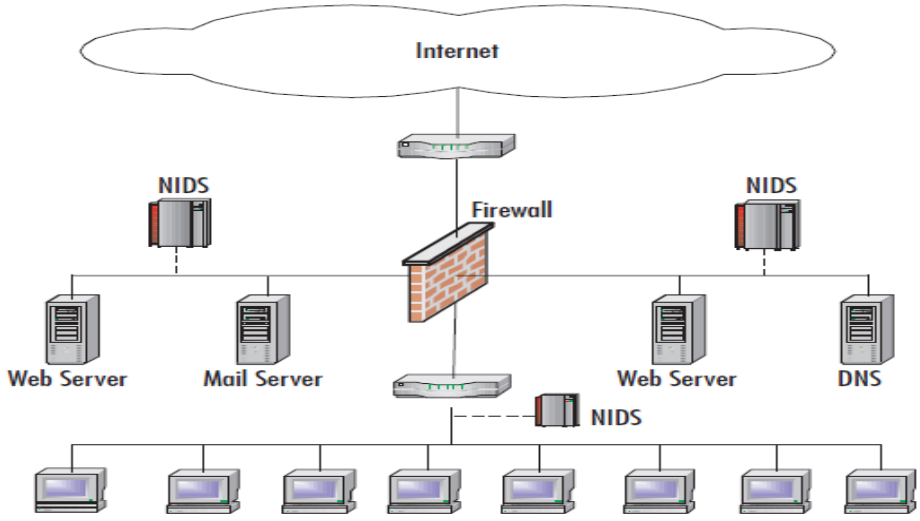


Hình 2.4: Mô hình NIDS

(*Nguồn: Nghiên cứu hệ thống phát hiện xâm nhập mạng trái phép IDS – Internet)*

NIDS giám sát toàn bộ mạng con của nó bằng cách lắng nghe tất cả các gói tin trên mạng con đó. (Nó thay đổi chế độ hoạt động của card mạng NIC vào trong chế độ Promisuous). Bình thường một NIC hoạt động ở chế độ Nonpromisuous nghĩa là nó chỉ nhận các gói tin mà có địa chỉ MAC trùng với địa chỉ của nó, các gói tin khác sẽ không nhận, hay không xử lý và bị loại bỏ. Để giám sát tất cả các truyền thông trong mạng con NIDS sẽ phải thiết lập chế độ hoạt động cho card mạng là Promisuous.

Ví dụ minh họa một mạng mà sử dụng 3 NIDS khác nhau:



Hình 2.5: Mô hình 3 NIDS

(*Nguồn: Nghiên cứu hệ thống phát hiện xâm nhập mạng trái phép IDS – Internet)*

Phân tích gói tin:

- NIDS kiểm tra tất cả các thành phần trong gói tin để tìm ra dấu hiệu của một cuộc tấn công trái phép bao gồm: các phần đầu(header) của gói tin và phần nội dung của gói tin (payload)
- Chống lại việc xóa dấu vết của Hacker: NIDS kiểm tra tất cả các luồng thông tin trên mạng một cách tức thời để phát hiện tấn công trong thời gian thực vì thế Hacker không thể xóa dấu vết của việc tấn công. Việc bắt dữ liệu không những tìm ra tấn công mà còn giúp cho việc xác định định danh của kẻ tấn công đó. Đây được coi là bằng chứng.
- Phát hiện và đáp ứng thời gian thực: NIDS phát hiện cuộc tấn công đang xảy ra trong thời gian thực và do đó tạo ra các phản ứng nhanh hơn.
- Đơn lập hệ điều hành: Network IDS thì không phụ thuộc vào hệ điều hành trong công việc phát hiện tấn công.

Ưu điểm của Network-Based IDSs:

- Quản lý được cả một network segment (gồm nhiều host)
- "Trong suốt" với người sử dụng lẫn kẻ tấn công
- Cài đặt và bảo trì đơn giản, không ảnh hưởng tới mạng
- Tránh tấn DOS ảnh hưởng tới một host nào đó
- Có khả năng xác định lỗi ở tầng Network (trong mô hình OSI)
- Độc lập với OS.

Hạn chế của Network-Based IDSs:

- Có thể xảy ra trường hợp báo động giả (false positive), tức không có intrusion mà NIDS báo là có intrusion
- Không thể phân tích các traffic đã được mã hóa (vd: SSL, SSH, IPSec…)
- NIDS đòi hỏi phải được cập nhật các signature mới nhất để thực sự an toàn

- Có độ trễ giữa thời điểm bị attack với thời điểm phát báo động. Khi báo động được phát ra, hệ thống có thể đã bị tổn hại
- Không cho biết việc attack có thành công hay không
- Giới hạn băng thông.

- *Host - Based IDS:*

Bằng cách cài đặt một phần mềm trên tất cả các máy tính chủ, IPS dựa trên máy chủ quan sát tất cả những hoạt động hệ thống, như các file log và những lưu lượng mạng thu thập được. Hệ thống dựa trên máy chủ cũng theo dõi OS, những cuộc gọi hệ thống, lịch sử sổ sách (audit log) và những thông điệp báo lỗi trên hệ thống máy chủ.

Trong khi những đầu dò của mạng có thể phát hiện một cuộc tấn công, thì chỉ có hệ thống dựa trên máy chủ mới có thể xác định xem cuộc tấn công có thành công hay không. Thêm nữa là hệ thống dựa trên máy chủ có thể ghi nhận những việc mà người tấn công đã làm trên máy chủ bị tấn công (compromised host).

Không phải tất cả các cuộc tấn công được thực hiện qua mạng. Bằng cách giành quyền truy cập ở mức vật lý (physical access) vào một hệ thống máy tính, kẻ xâm nhập có thể tấn công một hệ thống hay dữ liệu mà không cần phải tạo ra bất cứ lưu lượng mạng (network traffic) nào cả.

Hệ thống dựa trên máy chủ có thể phát hiện các cuộc tấn công mà không đi qua đường công cộng hay mạng được theo dõi, hay thực hiện từ cổng điều khiển (console), nhưng với một kẻ xâm nhập có hiểu biết, có kiến thức về hệ IDS thì hắn có thể nhanh chóng tắt tất cả các phần mềm phát hiện khi đã có quyền truy cập vật lý.

Một ưu điểm khác của IDS dựa trên máy chủ là nó có thể ngăn chặn các kiểu tấn công dùng sự phân mảnh hoặc TTL. Vì một host phải nhận và tái hợp các phân mảnh khi xử lí lưu lượng nên IDS dựa trên host có thể giám sát chuyện này. HIDS thường được cài đặt trên một máy tính nhất đinh.

Thay vì giám sát hoạt động của một network segment, HIDS chỉ giám sát các hoạt động trên một máy tính. HIDS thường được đặt trên các host xung yếu của tổ chức, và các server trong vùng DMZ - thường là mục tiêu bị tấn công đầu tiên. Nhiêm vụ chính của HIDS là giám sát các thay đổi trên hệ thống, bao gồm:

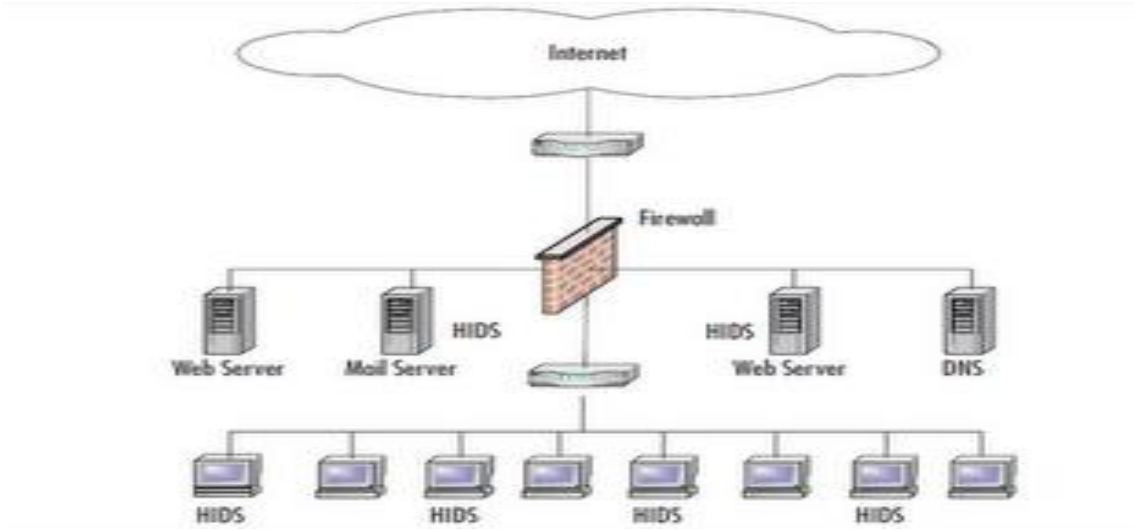
- Các tiến trình
- Các entry của Registry
- Mức độ sử dụng CPU
- Kiểm tra tính toàn vẹn và truy cập trên hệ thống file
- Một vài thông số khác.

Các thông số này khi vượt qua một ngưỡng định trước hoặc những thay đổi khả nghi trên hệ thống file sẽ gây ra báo động.

Dùng phần mềm để giám sát các tệp nhật ký của hệ thống. Ngay khi có bất kỳ thay đổi nào tới các tệp đó, Host - Based IDS sẽ so sánh thông tin với những gì được cấu hình trong chính sách, được thiết lập hiện tại và sau đó sinh ra các phản ứng lại với sự thay đổi đó. Một phương thức của Host - Based IDS giám sát các hoạt động trong thời gian thực. Một vài Host - Based IDS sẽ lắng nghe phát hiện tấn công mạng.

Host - Based IDS phát hiện sự thay đổi trên các tệp và trên hệ điều hành. Nó giám sát kích thước và tổng kiểm tra các tệp để đảm bảo rằng các tệp hệ thống không bị thay đổi. Ngoài ra nó có thể ngăn chặn các cuộc gọi hệ thống không hợp lệ mà đang cố gắng tìm kiếm các lỗ hổng của hệ thống.

Hình ảnh dưới đây minh họa cho việc sử dụng Host - Based IDS để bảo vệ máy chủ và máy trạm. Tập các nguyên tắc Host - Based IDS trên Mail server được tối ưu để bảo vệ cho các xâm nhập mail, trong khi đo các nguyên tắc cho web Server được tạo thích hợp để bảo vệ các xâm nhập web.



Hình 2.6: Mô hình HIDS

(*Nguồn: Nghiên cứu hệ thống phát hiện xâm nhập mạng trái phép IDS – Internet)*

Ưu điểm của Host - Based IDS:

Do Host - Based IDS được cài đặt trên một máy trạm cụ thể và dùng thông tin cung cấp bởi hệ điều hành (OS) nên nó có khả năng mà NIDS không có đó là:

- Kiểm tra tấn công: vì Host - Based IDS sử dụng các tệp nhật ký của hệ thống để phát hiện xâm nhập, những tệp này chứa những sự kiện đã xảy ra, do đó Host - Based IDS có thể biết được cuộc tấn công có thành công hay không thành công mà Network - Based IDS khó có thể biết được điều này.
- Giám sát các hành động đăng nhập và truy cập tệp tin: Host - Based IDS có thể giám sát các hành động của người dùng, cũng như hành động của thủ tục đăng nhập hoặc thoát ra được thực hiện, chúng sẽ ghi lại ở nhật ký đó dựa trên các chính sách hiện hành. Ngoài ra, Host - Based IDS có thể giám sát sự truy cập vào tệp tin và biết được thời điểm mở tệp tin đó.
- Giám sát các thành phần của hệ thống: Host - Based IDS cho ta khả năng giám sát các thành phần của hệ thống quan trọng như các thành phần hệ thống có thể thực thi, chẳng hạn như các file DLL và NT Registry. Nhưng file đó có thể gây ảnh

hưởng đến an toàn của hệ thống và mạng. Host - Based IDS có thể đưa ra các cảnh báo mỗi khi các file đó được thực thi.

- Phát hiện và phản ứng gần thời gian thực: hiện tại các Host - Based IDS có khả năng phát hiện và phản ứng ở gần thời gian thực, thay vì phải sử dụng một tiến trình để kiểm tra trạng thái và nội dung của nhật ký ở một thời gian xác định trước.

Hạn chế của Host - Based IDS:

- Thông tin từ HIDS là không đáng tin cậy ngay khi sự tấn công vào host này thành công
- HIDS không có khả năng phát hiện các cuộc dò quét mạng (Nmap, Netcat...)
- HIDS cần tài nguyên trên host để hoạt động
- HIDS có thể không hiệu quả khi bị tấn công DOS
- Đa số chạy trên hệ điều hành Windows. Tuy nhiên cũng đã có một số chạy được trên UNIX và những hệ điều hành khác
- Vì hệ thống IDS dựa trên máy chủ đòi hỏi phần mềm IDS phải được cài đặt trên tất cả các máy chủ nên đây có thể là điều khó khăn của những nhà quản trị khi nâng cấp phiên bản, bảo trì phần mêm, và cấu hình phần mêm trở thành công việc tốn thời gian và là những việc làm phức tạp.
- Phần mềm IDS phải được cài đặt trên mỗi hệ thống trên mạng nhằm cung cấp đầy đủ khả năng cảnh báo của mạng. Trong một môi trường hỗn tạp, điều này có thể là một vấn đề bởi vì phần mềm IDS phải tương ứng nhiều hệ điều hành khác nhau.

2.1.2.3. Công cụ phần mềm dò quét

Hiện nay có khá nhiều phần mềm miễn phí cũng như trả phí được phát hành. Một số ứng dụng phát hiện lỗ hổng bảo mật Website khá phổ biến đọợc liệt kê dưới đây.

- Google Safe Browsing Diagnostic: sẽ cho biết về tên miền và các đường dẫn liên quan đến địa chỉ web muốn kiểm tra.

- URL Void: có chức năng quét địa chỉ web bằng hơn 30 công cụ khác nhau.

- UnMask Parasites: quét web cho biết nó chứa các mối nguy hiểm nào ví dụ đường link, dòng lệnh đáng ngờ.

- PhishTank: cung cấp danh sách địa chỉ Web có nguy cơ phishing.

- UnShorten.it: đảm bảo an toàn cho người dùng khi truy cập vào các địa chỉ web rút gọn.

- Phần mềm Havij: phát hiện lỗ hổng cơ sở dữ liệu SQL với ngôn ngữ lập trình web là PHP.

- Phần mềm Rapid 7: có tính năng dò quét lỗ hổng toàn diện chuyên sâu trên cả phần cứng và phần mềm.

- Phần mềm Acunetix: là chương trình tự động kiểm tra các ứng dụng Web để tìm kiếm các lỗ hổng bảo mật như SQL Injection, hay Cross-Site Scripting,…

Các công cụ phát hiện tấn công web truyền thống như đã nêu trên có thể hoạt động tốt với các loại tấn công đã biết (có hành vi, có dấu hiệu nhận biết). Tuy nhiên lại chưa phát hiện được các loại tấn công mới, các loại tấn công chưa rõ hành vi. Vì thế giải pháp phát hiện dựa trên học máy đang được triển khai và bước đầu thu được nhiều kết quả tích cực. Ở phần tiếp theo, luận văn sẽ đề cập đến phương pháp này.

2.2. Phương pháp phát hiện hành vi bất thường người dùng web sử dụng học máy

Hành vi người dùng web là hành động, tương tác từ phía máy client của người dùng gửi yêu cầu đến hệ thống máy chủ web server. Một số hành vi của người dùng web như: đăng nhập, đăng xuất, tìm kiếm, tra cứu thông tin,…

Hành vi bất thường của người dùng web là hành động không xảy ra thường xuyên, chưa xuất hiện hoặc hiếm khi xảy ra trong lịch sử truy cập mà các máy chủ web đã lưu trữ lại từ trước đó. Hành vi bất thường người dùng web có thể xuất hiện

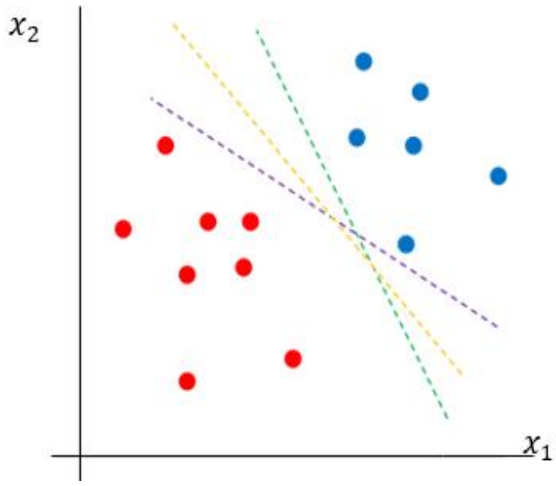
từ phía người dùng hoặc tin tặc tuy nhiên thông thường việc tấn công sẽ để lại nhiều dấu hiệu và hành vi bất thường hơn như: do quét, chèn kí tự đặc biệt

2.2.1. Một số thuật toán phát hiện tấn công web

2.1.1.1. Phương pháp học có giám sát sử dụng SVM (SVM- Support vector machine)

Support Vector Machine (SVM) [2] là một thuật toán thuộc nhóm Supervised Learning (Học có giám sát) dùng để phân chia dữ liệu (Classification) thành các nhóm riêng biệt.

Hình dung ta có bộ data gồm các điểm xanh và đỏ đặt trên cùng một mặt phẳng. Ta có thể tìm được đường thẳng để phân chia riêng biệt các bộ điểm xanh và đỏ như hình bên dưới.

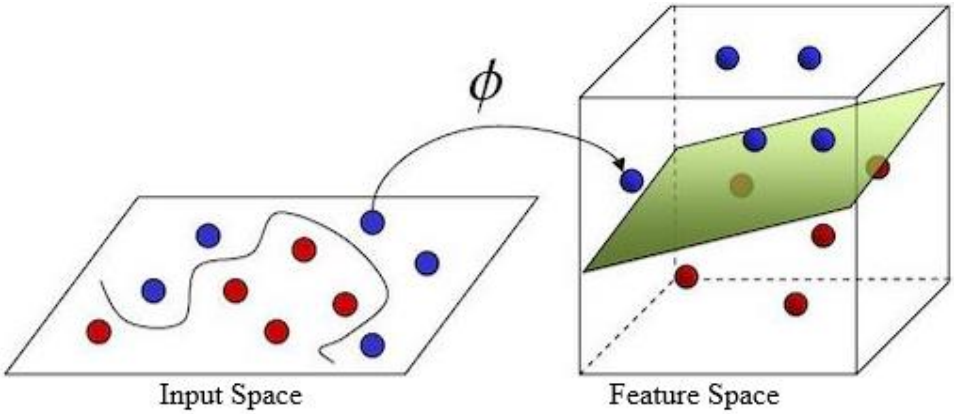


Hình 2.7: Mô tả hai bộ data trên cùng một mặt phẳng

(Nguồn: Tìm hiểu về Support Vector Machine (SVM) – Internet)

Với những bộ data phức tạp hơn mà không thể tìm được đường thẳng để phân chia thì ta cần dùng thuật toán để ánh xạ bộ data đó vào không gian nhiều chiều hơn (n chiều), từ đó tìm ra siêu mặt phẳng (hyperplane) để phân chia.

Ví dụ trong hình bên dưới là việc ánh xạ tập data từ không gian 2 chiều sang không gian 3 chiều.



Hình 2.8: Mô tả bộ data phức tạp trên không gian nhiều chiều

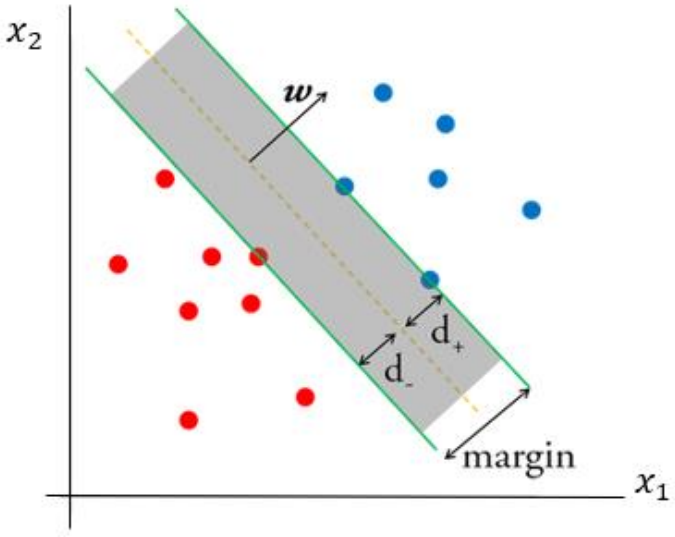
(Nguồn: Tìm hiểu về Support Vector Machine (SVM) – Internet)

Quay lại bài toán với không gian 2 chiều. Ở ví dụ trong Hình 2.7, ta thấy có thể tìm được rất nhiều các đường thẳng để phân chia 2 bộ điểm xanh, đỏ. Ta có thể thấy, đường tối ưu là đường tạo cho ta có cảm giác 2 lớp dữ liệu nằm cách xa nhau và cách xa đường đó nhất.

Tuy nhiên tính toán sự tối ưu bằng toán học, trong SVM sử dụng thuật ngữ Margin.

❖ Margin

Margin là khoảng cách giữa siêu phẳng (trong trường hợp không gian 2 chiều là đường thẳng) đến 2 điểm dữ liệu gần nhất tương ứng với 2 phân lớp.



Hình 2.9: Mô tả cách xác định margin

(Nguồn: Tìm hiểu về Support Vector Machine (SVM) – Internet)

SVM cố gắng tối ưu thuật toán bằng các tìm cách maximize giá trị margin này, từ đó tìm ra siêu phẳng đẹp nhất để phân 2 lớp dữ liệu.

❖ **Support Vectors**

Bài toán trở thành tìm ra 2 đường biên của 2 lớp dữ liệu (ở hình bên trên là 2 đường xanh lá cây) sao cho khoảng cách giữa 2 đường này là lớn nhất. Đường biên của lớp xanh sẽ đi qua một (hoặc một vài) điểm xanh. Đường biên của lớp đỏ sẽ đi qua một (hoặc một vài) điểm đỏ. Các điểm xanh, đỏ nằm trên 2 đường biên được gọi là các support vector, vì chúng có nhiệm vụ *support* để tìm ra siêu phẳng. Đó cũng là lý do của tên gọi thuật toán Support Vector Machine.

Trong bài toán không gian 2 chiều, ta giả sử đường thẳng phân chia cần tìm có phương trình là:

$$w_1 x_1 + w_2 x_2 + b = 0$$

Ta có thể dùng phương trình đường thẳng hay sử dụng là ax+by+c=0 để tính toán cho quen thuộc. Ở đây ta dùng các giá trị w_1, w_2, x_1, x_2 để sau này dễ dàng tổng quát lên không gian nhiều chiều.

Giả sử 2 đường thẳng đi qua các support vector của 2 lớp dữ liệu lần lượt là:

$$w_1 x_1 + w_2 x_2 + b = 1$$

$$w_1 x_1 + w_2 x_2 + b = -1$$

Vì sao lại là 1 và −1?

Giả sử nếu ta tìm được phương trình 3 đường thẳng tương ứng là:

$$2x_1 + 3x_2 + 5 = 0$$

$$2x_1 + 3x_2 + 9 = 0$$

$$2x_1 + 3x_2 + 1 = 0$$

Chia tất cả cho 4 để thu được phương trình như định nghĩa:

$$\frac{1}{2}x_1 + \frac{3}{4}x_2 + \frac{5}{4} = 0$$

$$\frac{1}{2}x_1 + \frac{3}{4}x_2 + \frac{5}{4} = 1$$

$$\frac{1}{2}x_1 + \frac{3}{4}x_2 + \frac{5}{4} = -1$$

Với không gian 2 chiều, margin giữa 2 đường thẳng được tính bằng công thức:

$$\text{margin} = \frac{2}{\sqrt{w_1^2 + w_2^2}}$$

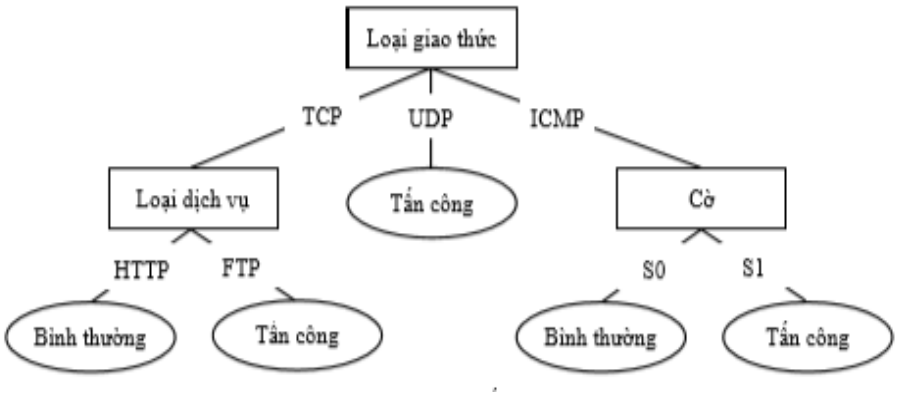
Với không gian nhiều chiều, tổng quát lên không gian nhiều chiều, cần tìm phương trình siêu phẳng có phương trình:

$$w^T x + b = 0$$

Margin sẽ được tính bằng công thức: margin = $\frac{2}{\|w\|}$

2.1.1.2. Decision Tree

Decision Tree- cây quyết định [2] là một mô hình được đánh giá cao trong việc phân lớp dữ liệu, nó bao gồm những ưu điểm như: xây dựng tương đối nhanh; đơn giản, dễ hiểu. Hơn nữa các cây có thể dễ dàng được chuyển đổi sang các câu lệnh SQL để có thể được sử dụng để truy nhập cơ sở dữ liệu một cách hiệu quả. Cuối cùng, việc phân lớp dựa trên cây quyết định đạt được sự tương tự và đôi khi là chính xác hơn so với các phương pháp phân lớp khác. Cây quyết định là biểu đồ phát triển có cấu trúc dạng cây, như mô tả trong hình vẽ sau:



Hình 2.10: Cây quyết định

Biểu đồ phát triển hình cây của cây quyết định được minh họa như ở hình 2.6, gồm:

- Gốc: là node trên cùng của cây;
- Node trong: biểu diễn một kiểm tra trên một thuộc tính đơn;
- Nhánh: biểu diễn các kết quả của kiểm tra trên node trong;
- Node lá: biểu diễn lớp.

Để phân lớp mẫu dữ liệu chưa biết, giá trị các thuộc tính của mẫu được đưa vào kiểm tra trên cây quyết định. Mỗi mẫu tương ứng có một đường đi từ gốc đến lá và lá biểu diễn dự đoán giá trị phân lớp của mẫu đó.

❖ Xây dựng cây quyết định

Quá trình xây dựng cây quyết định gồm hai giai đoạn:

- Giai đoạn thứ nhất phát triển cây quyết định: Giai đoạn này phát triển bắt đầu từ gốc, đến từng nhánh và phát triển quy nạp theo cách thức chia để trị cho tới khi đạt được cây quyết định với tất cả các lá được gán nhãn lớp.
- Giai đoạn thứ hai cắt, tỉa bớt các cành nhánh trên cây quyết định. Giai đoạn này nhằm mục đích đơn giản hóa và khái quát hóa từ đó làm tăng độ chính xác của cây quyết định bằng cách loại bỏ sự phụ thuộc vào mức độ lỗi (noise) của dữ liệu đào tạo mang tính chất thống kê, hay những sự biến đổi mà có thể là đặc tính riêng biệt của dữ liệu đào tạo. Giai đoạn này chỉ truy cập dữ liệu trên cây quyết

định đã được phát triển trong giai đoạn trước và quá trình thực nghiệm cho thấy giai đoạn này không tốn nhiều tài nguyên tính toán, như với phần lớn các thuật toán, giai đoạn này chiếm khoảng dưới 1% tổng thời gian xây dựng mô hình phân lớp. Do vậy, ở đây chúng ta chỉ tập trung vào nghiên cứu giai đoạn phát triển cây quyết định. Dưới đây là khung công việc của giai đoạn này:

1) Chọn thuộc tính "tốt" nhất bằng một độ đo đã định trước.

2) Phát triển cây bằng việc thêm các nhánh tương ứng với từng giá trị của thuộc tính đã chọn.

3) Sắp xếp, phân chia tập dữ liệu đào tạo tới node con.

4) Nếu các ví dụ được phân lớp rõ ràng thì dừng. Ngược lại: lặp lại bước 1 tới bước 4 cho từng node con.

❖ **Thuật toán xây dựng cây quyết định**

Phần lớn các thuật toán phân lớp dữ liệu dựa trên cây quyết định có mã giả như sau:

```
Make Tree (Training Data T)
{
        Partition(T)
}
Partition (Data S)
{
    if (all points in S are in the same class) then
    return
    for each attribute A do
            evaluate splits on attribute A;
    use best split found to partition S into S1, S2,..., Sk
    Partition(S1)
    Partition(S2) ...
    Partition(Sk)
```

}

Các thuật toán phân lớp như C4.5 (Quinlan, 1993), CDP (Agrawal và các tác giả khác, 1993), SLIQ (Mehta và các tác giả khác, 1996) và SPRINT (Shafer và các tác giả khác, 1996) đều sử dụng phương pháp của Hunt làm tư tưởng chủ đạo. Phương pháp này được Hunt và các đồng sự nghĩ ra vào những năm cuối thập kỷ 50 đầu thập kỷ 60.

Mô tả quy nạp phương pháp Hunt:

Giả sử xây dựng cây quyết định từ T là tập training data và các lớp được biểu diễn dưới dạng tập C = $\{C_1, C_2, \ldots, C_k\}$ Trường hợp 1: T chứa các case thuộc về một lớp đơn C_j, cây quyết định ứng với T là một lá tương ứng với lớp C_j Trường hợp 2: T chứa các case thuộc về nhiều lớp khác nhau trong tập C. Một kiểm tra được chọn trên một thuộc tính có nhiều giá trị $\{O_1, O_2, \ldots, O_n\}$. Trong nhiều ứng dụng n thường được chọn là 2, khi đó tạo ra cây quyết định nhị phân. Tập T được chia thành các tập con $T_1, T_2, \ldots, T_n$, với T_i chứa tất cả các case trong T mà có kết quả là O_i trong kiểm tra đã chọn. Cây quyết định ứng với T bao gồm một node biểu diễn kiểm tra được chọn, và mỗi nhánh tương ứng với mỗi kết quả có thể của kiểm tra đó. Cách thức xây dựng cây tương tự được áp dụng đệ quy cho từng tập con của tập training data. Trường hợp 3: T không chứa case nào. Cây quyết định ứng với T là một lá, nhưng lớp gắn với lá đó phải được xác định từ những thông tin khác ngoài T. Ví dụ C4.5 chọn giá trị phân lớp là lớp phổ biến nhất tại cha của node này.

2.2.1.3. Random Forest

Random Forests – rừng ngẫu nhiên [2] là thuật toán học có giám sát (supervised learning). Nó có thể được sử dụng cho cả phân lớp và hồi quy. Nó cũng là thuật toán linh hoạt và dễ sử dụng nhất. Một khu rừng bao gồm cây cối. Càng có nhiều cây thì rừng càng mạnh. Random forests tạo ra cây quyết định trên các mẫu dữ liệu được chọn ngẫu nhiên, được dự đoán từ mỗi cây và chọn giải pháp tốt nhất bằng cách bỏ phiếu. Nó cũng cung cấp một chỉ báo khá tốt về tầm quan trọng của tính năng.

Random forests có nhiều ứng dụng, chẳng hạn như công cụ đề xuất, phân loại hình ảnh và lựa chọn tính năng. Nó có thể được sử dụng để phân loại các ứng viên cho vay trung thành, xác định hoạt động gian lận và dự đoán các bệnh. Nó nằm ở cơ sở của thuật toán Boruta, chọn các tính năng quan trọng trong tập dữ liệu.

❖ **Thuật toán Random Forests**

Giả sử bạn muốn đi trên một chuyến đi và bạn muốn đi đến một nơi mà bạn sẽ thích. Vậy bạn sẽ làm gì để tìm một nơi mà bạn sẽ thích? Bạn có thể tìm kiếm trực tuyến, đọc các bài đánh giá trên blog và các cổng thông tin du lịch hoặc bạn cũng có thể hỏi bạn bè của mình.

Giả sử bạn đã quyết định hỏi bạn bè và nói chuyện với họ về trải nghiệm du lịch trong quá khứ của họ đến những nơi khác nhau. Bạn sẽ nhận được một số khuyến nghị từ tất cả các bạn. Bây giờ bạn phải tạo danh sách các địa điểm được đề xuất. Sau đó, bạn yêu cầu họ bỏ phiếu (hoặc chọn địa điểm tốt nhất cho chuyến đi) từ danh sách các địa điểm được đề xuất bạn đã thực hiện. Địa điểm có số phiếu bầu cao nhất sẽ là lựa chọn cuối cùng của bạn cho chuyến đi.

Trong quá trình quyết định ở trên, có hai phần. Trước tiên, hãy hỏi bạn bè về trải nghiệm du lịch cá nhân của họ và nhận được đề xuất từ nhiều nơi họ đã ghé thăm. Điều này cũng giống như sử dụng thuật toán cây quyết định. Ở đây, mỗi người trong số các bạn chọn những nơi mà họ đã ghé thăm cho đến nay. Phần thứ hai, sau khi thu thập tất cả các khuyến nghị, là thủ tục bỏ phiếu để chọn địa điểm tốt nhất trong danh sách các khuyến nghị. Toàn bộ quá trình nhận được khuyến nghị từ bạn bè và bỏ phiếu cho họ để tìm ra nơi tốt nhất được gọi là thuật toán rừng ngẫu nhiên.

Về mặt kỹ thuật, nó là một phương pháp tổng hợp (dựa trên cách tiếp cận phân chia và chinh phục) của các cây quyết định được tạo ra trên một tập dữ liệu được chia ngẫu nhiên. Bộ sưu tập phân loại cây quyết định này còn được gọi là rừng. Cây quyết định riêng lẻ được tạo ra bằng cách sử dụng chỉ báo chọn thuộc tính như tăng thông tin, tỷ lệ tăng và chỉ số Gini cho từng thuộc tính. Mỗi cây phụ thuộc vào một mẫu ngẫu nhiên độc lập.

Trong bài toán phân loại, mỗi phiếu bầu chọn và lớp phổ biến nhất được chọn là kết quả cuối cùng. Trong trường hợp hồi quy, mức trung bình của tất cả các kết quả đầu ra của cây được coi là kết quả cuối cùng. Nó đơn giản và mạnh mẽ hơn so với các thuật toán phân loại phi tuyến tính khác.

Phương thức hoạt động của thuật toán gồm 4 bước:

1. Chọn các mẫu ngẫu nhiên từ tập dữ liệu đã cho.
2. Thiết lập cây quyết định cho từng mẫu và nhận kết quả dự đoán từ mỗi quyết định cây.
3. Hãy bỏ phiếu cho mỗi kết quả dự đoán.
4. Chọn kết quả được dự đoán nhiều nhất là dự đoán cuối cùng.

Ưu điểm: Random forests được coi là một phương pháp chính xác và mạnh mẽ vì số cây quyết định tham gia vào quá trình này. Nó không bị vấn đề overfitting. Lý do chính là nó mất trung bình của tất cả các dự đoán, trong đó hủy bỏ những thành kiến. Thuật toán có thể được sử dụng trong cả hai vấn đề phân loại và hồi quy. Random forests cũng có thể xử lý các giá trị còn thiếu. Có hai cách để xử lý các giá trị này: sử dụng các giá trị trung bình để thay thế các biến liên tục và tính toán mức trung bình gần kề của các giá trị bị thiếu. Bạn có thể nhận được tầm quan trọng của tính năng tương đối, giúp chọn các tính năng đóng góp nhiều nhất cho trình phân loại.

Nhược điểm: Random forests chậm tạo dự đoán bởi vì nó có nhiều cây quyết định. Bất cứ khi nào nó đưa ra dự đoán, tất cả các cây trong rừng phải đưa ra dự đoán cho cùng một đầu vào cho trước và sau đó thực hiện bỏ phiếu trên đó. Toàn bộ quá trình này tốn thời gian. Mô hình khó hiểu hơn so với cây quyết định, nơi bạn có thể dễ dàng đưa ra quyết định bằng cách đi theo đường dẫn trong cây.

2.2.1.4. KNN

K-Nearest Neighbors algorithm (K-NN) [2] được sử dụng rất phổ biến trong lĩnh vực Data Mining. K-NN là phương pháp để phân lớp các đối tượng dựa vào

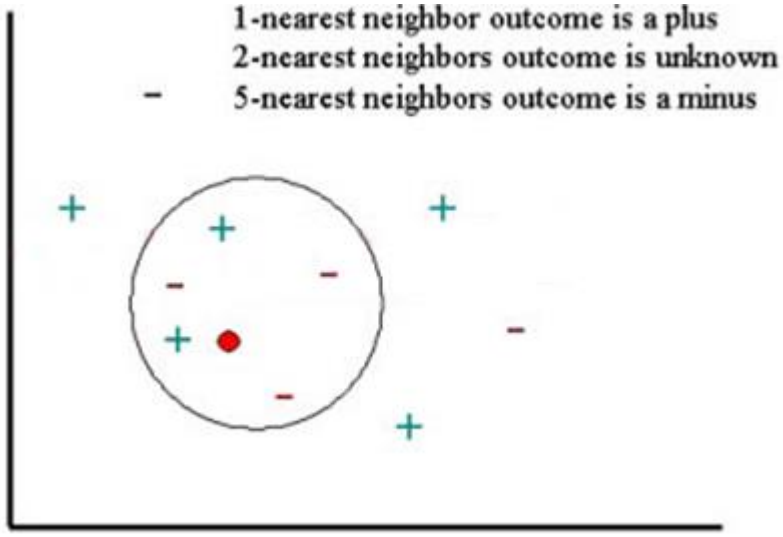
khoảng cách gần nhất giữa đối tượng cần xếp lớp (Query point) và tất cả các đối tượng trong Training Data.

Một đối tượng được phân lớp dựa vào K láng giềng của nó. K là số nguyên dương được xác định trước khi thực hiện thuật toán. Người ta thường dùng khoảng cách Euclidean để tính khoảng cách giữa các đối tượng.

Thuật toán K-NN được mô tả như sau:

1. Xác định giá trị tham số K (số láng giềng gần nhất);
2. Tính khoảng cách giữa đối tượng cần phân lớp (Query Point) với tất cả các đối tượng trong training data (thường sử dụng khoảng các Euclidean);
3. Sắp xếp khoảng cách theo thứ tự tăng dần và xác định K láng giềng gần nhất với Query Point;
4. Lấy tất cả các lớp của K láng giềng gần nhất đã xác định;
5. Dựa vào phần lớn lớp của láng giềng gần nhất để xác định lớp cho Query Point.

Ví dụ về K-NN dùng để phân lớp trong hình 2.11 ta có, training Data được mô tả bởi dấu (+) và dấu (-), đối tượng cần được xác định lớp cho nó (Query point) là hình tròn đỏ. Nhiệm vụ của chúng ta là ước lượng (hay dự đoán) lớp của Query point dựa vào việc lựa chọn số láng giềng gần nhất với nó. Nói cách khác chúng ta muốn biết liệu Query Point sẽ được phân vào lớp (+) hay lớp (-)



Hình 2.11: Mô tả K-NN dùng để phân lớp

(Nguồn: Thuật toán K láng giềng gần nhất -Internet)

Trong đó:

1-Nearest neighbor: kết quả là + (Query Point được xếp vào lớp dấu +)

2-Nearest neighbors: không xác định lớp cho Query Point vì số láng giềng gần nhất với nó là 2 trong đó 1 là lớp + và 1 là lớp – (không có lớp nào có số đối tượng nhiều hơn lớp kia)

5-Nearest neighbors: kết quả là - (Query Point được xếp vào lớp dấu – vì trong 5 láng giềng gần nhất với nó thì có 3 đối tượng thuộc lớp - nhiều hơn lớp + chỉ có 2 đối tượng).

2.2.2. Lựa chọn và trích xuất hành vi người dùng web

2.2.2.1. Mô tả bộ dữ liệu

Trong luận văn, tác giả trích xuất hành vi bất thường từ bộ dữ liệu về tấn công web CSIC 2010.

Bảng 2.1: Mô tả các trường dữ liệu trong bộ dữ liệu CSIC

Cột dữ liệu	**Mô tả**
index	Số thứ tự
method	Phương thức cho HTTP/1.1 như GET, HEAD, POST, PUT, …
url	Đường dẫn hay địa chỉ dùng để tham chiếu đến các tài nguyên trên mạng Internet
userAgent	Là một chuỗi nhận dạng của trình duyệt web khi gửi yêu cầu đến máy chủ web
cacheControl	Tối ưu tốc độ tải trang, tăng tính bảo mật
accept	Là kiểu dữ liệu mà sẽ nhận được từ response, response mà dại trả về khác kiểu thì sẽ bị ban ngay. Thường thấy nhất là các kiểu text/html, application/xhtml+xm
acceptEncoding	Khai báo kiểu mã hóa nội dung mà request chấp nhận
acceptCharset	Sử dụng để chỉ các bộ thiết lập ký tự nào được chấp nhận
acceptLanguage	Sử dụng để chỉ ngôn ngữ nào được chấp nhận
host	Địa chỉ IP máy chủ
contentLength	Chỉ dẫn kích cỡ của phần thân đối tượng, trong số thập phân của hệ 8, được gửi tới người nhận
contentType	Là kiểu thông tin mà server trả về cho client, nó phải phù hợp với cái accept mà client request tới

Cột dữ liệu	Mô tả
cookie	Chứa thông tin được mã hóa dùng để gửi lên server, giúp xác định phiên giữa client-server
payload	Chứa dữ liệu và các tham số của người dùng gửi lên

Thông thường trong bài toán phân tích hành vi người dùng để xác định bất thường, sẽ tập trung chủ yếu vào các trường dữ liệu người dùng nhập vào.

Đối với tập dữ liệu CSIC đã thu thập luận văn sẽ tập trung vào trường payload, url và cookie để xây dựng bộ feature.

2.2.2.2. Trích chọn thuộc tính sử dụng kỹ thuật TF-IDF (Term Frequency – Inverse Document Frequency)

❖ Ứng dụng N-Gram trong trích xuất kí tự và từ trong văn bản

N-Gram là mô hình ngôn ngữ thống kê cho phép gán (ước lượng) xác suất cho một chuỗi m phần tử (thường là từ) P($w_1 w_2 \dots w_m$) tức là cho phép dự đoán khả năng một chuỗi từ xuất hiện trong ngôn ngữ đó. Theo công thức Bayes:

$$\mathbf{P(AB) = P(B|A) * P(A)}$$

Trong đó:

- P(A): Xác suất xảy ra sự kiện A
- P(B): Xác suất xảy ra sự kiện B
- P(B|A): Xác suất (có điều kiện) xảy ra sự kiện B nếu biết rằng sự kiện A đã xảy ra.

Từ đó ta được:

$$P(w_1 w_2 \dots w_m) = P(w_1) * P(w_2|w_1) * P(w_3|w_1 w_2) * \dots * P(w_m|w_1 w_2 \dots w_{m-1})$$

Theo công thức này thì bài toán tính xác suất của mỗi chuỗi từ quy về bài toán tính xác suất của một từ với điều kiện biết các từ trước nó (có thể hiểu P($w_1 =$ P($w_1|start$) là xác suất để w_1 đứng đầu chuỗi hay nói cách khác người ta có thể đưa thêm ký hiệu đầu dòng start vào mỗi chuỗi).

Trong thực tế, dựa vào giả thuyết Markov người ta chỉ tính xác suất của một từ dựa vào nhiều nhất n từ xuất hiện liền trước nó, và thông thường n= 0,1,2,3. Vì vậy, nhiều người gọi mô hình ngôn ngữ là mô hình N-gram, trong đó n là số lượng từ (bao gồm cả từ cần tính và các từ ngữ cảnh phía trước).

- Với n = 1, unigram.

- Với n = 2, ta có khái niệm bigram.

- Với n = 3, ta có trigram.

Nhưng vì n càng lớn thì số trường hợp càng lớn nên thường người ta chỉ sử dụng với n = 1,2 hoặc đôi lúc là 3.

Theo công thức Bayes, mô hình ngôn ngữ cần phải có một lượng bộ nhớ vô cùng lớn để có thể lưu hết xác suất của tất cả các chuỗi độ dài nhỏ hơn m. Rõ ràng, điều này là không thể khi m là độ dài của các văn bản ngôn ngữ tự nhiên (m có thể tiến tới vô cùng). Để có thể tính được xác suất của văn bản với lượng bộ nhớ chấp nhận được, ta sử dụng xấp xỉ Markov bậc n:

P($w_1 w_2 \dots w_m$)= P(w_1) * P($w_2|w_1$) * P($w_3|w_1 w_2$) *…

* P($w_{m-1}|w_{m-n-1} w_{m-n} \dots w_{m-2}$)* P($w_m|$

$w_{m-n} w_{m-n+1} \dots w_{m-1}$)

Với công thức này, ta có thể xây dựng mô hình ngôn ngữ dựa trên việc thống kê các cụm có ít hơn n+1 từ. Các mô hình N-gram được hình dung thông qua ví dụ sau:

❖ TF-IDF

Term Frequency – Inverse Document Frequency (TF-IDF) là giải pháp đánh trọng số kết hợp tính chất quan trọng của một từ trong tài liệu chứa nó (TF- tần suất

xuất hiện của từ trong tài liệu) với tính phân biệt của từ trong tập tài liệu nguồn (IDF-nghịch đảo tần suất tài liệu). Đây là một kỹ thuật cơ bản và thường được sử dụng kết hợp với các thuật toán khác để xử lý văn bản. Mục đích của kỹ thuật này là tính trọng số của một từ, qua đó đánh giá mức độ quan trọng của từ đó trong văn bản. Trong đó:

- **TF** được tính theo công thức:

$$\mathbf{tf}(t,d) = \frac{\mathbf{f}(\mathbf{t},\mathbf{d})}{\mathbf{max}\{\mathbf{f}(\mathbf{w},\mathbf{d}) : \mathbf{w} \epsilon \mathbf{d}}$$

Trong đó:

- tf(t, d): tần suất xuất hiện của từ t trong văn bản d
- f(t, d): Số lần xuất hiện của từ t trong văn bản d
- max({f(w, d) : w ∈ d}): Số lần xuất hiện của từ có số lần xuất hiện nhiều nhất trong văn bản d

- **IDF** được tính theo công thức:

$$\mathbf{idf}(t,D) = \mathbf{log}\ \frac{|\mathbf{D}|}{|\{\mathbf{d} \epsilon \mathbf{D} : \mathbf{t} \epsilon \mathbf{d}\}|}$$

Trong đó:

- idf(t, D): giá trị idf của từ t trong tập văn bản
- |D|: Tổng số văn bản trong tập D
- |{d ∈ D : t ∈ d}|: thể hiện số văn bản trong tập D có chứa từ t.

- Giá trị **TF-IDF**:

$$\mathbf{tfidf(t, d, D) = tf(t, d) \times idf(t, D)}$$

Ví dụ trích chọn thuộc tính sử dụng kết hợp N-Gram và TF-IDF cho request người dùng: http://localhost:8080?id=abc';+drop+table+usuarios;. Thu được kết quả trong bảng 2.2.

Bảng 2.2: Kết quả trích chọn thuộc tính sử dụng kết hợp N-Gram và TF-IDF

	tfidf		tfidf		tfidf
tab	0.23375	**rop**	0.138057	**=ab**	0.138057
ble	0.23375	**st:**	0.138057	**?id**	0.138057
abl	0.23375	**t:8**	0.138057	**alh**	0.138057
abc	0.23375	**tp:**	0.138057	**bc'**	0.138057
';+	0.138057	**dro**	0.138057	**bct**	0.138057
lho	0.138057	**cal**	0.138057	**c';**	0.138057
e+a	0.138057	**cta**	0.138057	**ttp**	0.138057
hos	0.138057	**+ab**	0.138057		
htt	0.138057	**+dr**	0.138057		
id=	0.138057	**+ta**	0.138057		
le+	0.138057	**//l**	0.138057		
le;	0.138057	**/lo**	0.138057		
loc	0.138057	**80**	0.138057		
d=a	0.138057	**0?i**	0.138057		
oca	0.138057	**808**	0.138057		
op+	0.138057	**80?**	0.138057		
ost	0.138057	**://**	0.138057		
p+t	0.138057	**:80**	0.138057		
p:/	0.138057	**;+d**	0.138057		

Kết luận chương 2

Trong chương 2 luận văn đã giới thiệu tổng quát về các phương pháp phát hiện tấn công web và một số công cụ hỗ trợ phát hiện tấn công. Từ hạn chế của việc sử dụng các công cụ tấn công, luận văn đã đề xuất phương pháp phát hiện hành vi bất

thường của người dùng web sử dụng học máy thông qua các thuật toán: SVM, Random Forest, KNN. Luận văn sử dụng kĩ thuật trích chọn thuộc tính trong văn bản TF-IDF để lựa chọn và trích xuất hành vi người dùng đưa ra cảnh báo trước về các cuộc tấn công web cho người quản trị.

Trên cơ sở các kết quả đã đạt được của chương 2, trong chương tiếp theo luận văn sẽ tiến hành thực nghiệm phát hiện tấn công web dựa trên kỹ thuật phân tích hành vi trên cơ sở các thuật toán (SVM, Random Forest, KNN) và hành vi đã trích xuất-lựa chọn.

CHƯƠNG 3: THỰC NGHIỆM VÀ ĐÁNH GIÁ

Tóm tắt chương**:** Trong chương 3, luận văn sẽ thực hiện thực nghiệm phát hiện tấn công web dựa trên kỹ thuật phân tích hành vi trên cơ sở thuật toán và hành vi đã được lựa chọn và phân tích ở chương 2.

3.1. Một số yêu cầu cài đặt

3.1.1. Yêu cầu chung cho cài đặt thử nghiệm

- *Phần cứng:* Bộ xử lý 32bit (x86) hoặc 64bit (x64) có tốc độ 2 gigahertz (GHz) hoặc nhanh hơn; RAM 4GB trở lên; Đĩa cứng có dung lọợng trống 10 GB (64 bit).

- *Phần mềm:* Cài đặt trên hệ thống Windows/Linux (Centos 7.2); Công cụ lập trình: Phần mềm Python 2.7 trở lên hoặc phần mềm Pycham Professional 2020.1.

- *Dữ liệu:* Bộ dữ liệu tấn công CSIC 2010.

3.1.2. Giới thiệu chung về Python

Python là ngôn ngữ kịch bản hướng đối tượng (object-oriented scripting language). Không chỉ vậy, nó còn là một ngôn ngữ cấp cao có khả năng thông dịch (interpreted language) và có tính tương tác (interactive language) cao. Nhờ chức năng thông dịch mà trình thông dịch (Interpreter) của Python có thể xử lý lệnh tại thời điểm chạy chương trình (runtime). Nhờ đó mà ta không cần biên dịch chương trình trước khi thực hiện nó (tương tự như Perl và PHP).

Python là một ngôn ngữ lập trình đa mục đích, được sử dụng bởi hàng ngàn người để làm những việc từ kiểm thử vi mạch tại hãng Intel, sử dụng trong ứng dụng Instagram, cho tới xây dựng các video game với thư viện PyGame và có hàng trăm các thư viện của bên thứ ba (third-party). Có một số đặc điểm sau:

- *Đơn giản*: Python là một ngôn ngữ đơn giản và tối giản. Đọc một chương trình Python có cảm giác như đọc tiếng Anh, mặc dù ở dạng rút gọn. Tính tự nhiên của mã giả trong Python là một trong các điểm mạnh nhất của ngôn ngữ này. Điều

này giúp cho lập trình viên tập trung vào giải pháp giải quyết vấn đề hơn là việc tập trung vào ngôn ngữ.

- *Dễ học*: Python dễ học vì có cú pháp cực kỳ đơn giản.

- *Miễn phí và mã nguồn mở*: Python là một ví dụ của FLOSS (Free/Libré and Open Source Software). Vì vậy, chúng ta có thể tự do phân phối bản sao chép của phần mềm, cũng như mã nguồn, thay đổi hay sử dụng các thành phần phần mềm trong các chương trình mới. Một trong những lý do Python là ngôn ngữ mạnh vì nó được cộng đồng thường xuyên phát triển và nâng cấp

- *Ngôn ngữ bậc cao*: Khi sử dụng Python, chúng ta sẽ không bao giờ phải để ý đến các chi tiết mức thấp như quản lý bộ nhớ cho chương trình,…

- *Khả năng bỏ túi*: Do tính tự nhiên mã mở của Python, Python cũng xây dựng chạy trên nhiều nền tảng khác nhau. Có thể sử dụng Python trên GNU/Linux, Windows, FreeBSD, Macintosh, Solaris, OS/2, Amiga, AROS, AS/400, BeOS, OS/390, z/OS, Palm OS, QNX, VMS, Psion, Acorn RISC OS, VxWorks, PlayStation, Sharp Zaurus, Windows CE và PocketPC. Ngoài ra còn có thể dùng một nền tảng như Kivy để tạo các trò chơi trên máy tính dành cho iPhone, iPad, và Android.

- *Diễn dịch*: Khi một chương trình được viết bằng ngôn ngữ biên dịch (như C hoặc C++) thì nó được chuyển đổi từ mã ngôn ngữ (C/C++) thành ngôn ngữ mà máy tính có hiểu được bằng cách dùng 1 trình biên dịch với các chức năng khác nhau. Trái lại, Python không cần biên dịch ra nhị phân. Chương trình viết bằng Python chạy trực tiếp từ mã nguồn. Cụ thể, Python sẽ chuyển mã nguồn thành một dạng trung gian gọi là bytecode, sau đó dịch dạng trung gian thành ngôn ngữ mà máy tính có thể hiểu được

- *Hướng đối tượng*: Python là ngôn ngữ hỗ trợ cho lập trình hướng đối tượng lẫn cả lập trình thủ tục. Nếu so sánh với C++ hoặc Java, Python rất mạnh nhưng lại cực kỳ đơn giản để thực hiện lập trình hướng đối tượng.

- *Tính mở rộng*: Nếu chúng ta cần một đoạn mã chạy nhanh hoặc một vài thuật toán đóng, có thể lập trình ở C/C++ và sau đó sử dụng nó cho chương trình Python. Python cho phép tích hợp các chương trình ở các ngôn ngữ khác

- *Thư viện mở rộng*: Thư viện tiêu chuẩn Python thì rất lớn. Thư viện giúp chúng ta làm nhiều thứ khác nhau liên quan đến biểu thực chính quy, gieo tài liệu, tiến trình/tiểu trình, database, trình duyệt web, CGI, FTP, email, XML, XML-RPC, HTML, tập tin WAV, mã hóa, GUI, và các phần khác. Tất cả thứ này đều sẵn có khi cài đặt Python.

Từ những khảo sát trên, luận văn đã lựa chọn Python làm ngôn ngữ để tiến hành cài đặt các thử nghiệm.

3.1.3. Giới thiệu về bộ dữ liệu CSIC

Bộ dữ liệu HTTP CSIC 2010 chứa hàng ngàn yêu cầu web được tạo tự động. Nó có thể được sử dụng để thử nghiệm các hệ thống bảo vệ tấn công web. Bộ dữ liệu HTTP CSIC được phát triển tại "Viện bảo mật thông tin" của CSIC (Hội đồng nghiên cứu quốc gia Tây Ban Nha) [14].

Bộ dữ liệu được tạo tự động và chứa 36.000 yêu cầu bình thường và hơn 25.000 yêu cầu dị thường. Các yêu cầu HTTP được gắn nhãn là bình thường hoặc bất thường và bộ dữ liệu bao gồm các cuộc tấn công như SQL, tràn bộ đệm, thu thập thông tin, tiết lộ tệp, tiêm CRLF, XSS, bao gồm phía máy chủ, giả mạo tham số, v.v.

Dữ liệu được tạo theo các bước:

Đầu tiên, dữ liệu thực được thu thập cho tất cả các tham số của ứng dụng web. Tất cả dữ liệu (tên, họ, địa chỉ, v.v.) được trích xuất từ cơ sở dữ liệu thực. Các giá trị này được lưu trữ trong hai cơ sở dữ liệu: một cho các giá trị bình thường và một cho các giá trị dị thường. Ngoài ra, tất cả các trang có sẵn công khai của ứng dụng web được liệt kê.

Tiếp theo, các yêu cầu bình thường và bất thường được tạo ra cho mọi trang web. Trong trường hợp các yêu cầu bình thường có tham số, các giá trị tham số được

điền đầy đủ với dữ liệu được lấy từ cơ sở dữ liệu bình thường. Quá trình này là tương tự cho các yêu cầu bất thường, trong đó các giá trị của các tham số được lấy từ cơ sở dữ liệu bất thường.

Ba loại hành vi bất thường đã được xem xét:

1) Các cuộc tấn công tĩnh cố gắng yêu cầu các tài nguyên bị ẩn (hoặc không tồn tại). Các yêu cầu này bao gồm các tệp lỗi thời, ID phiên trong ghi lại URL, tệp cấu hình, tệp mặc định, v.v.

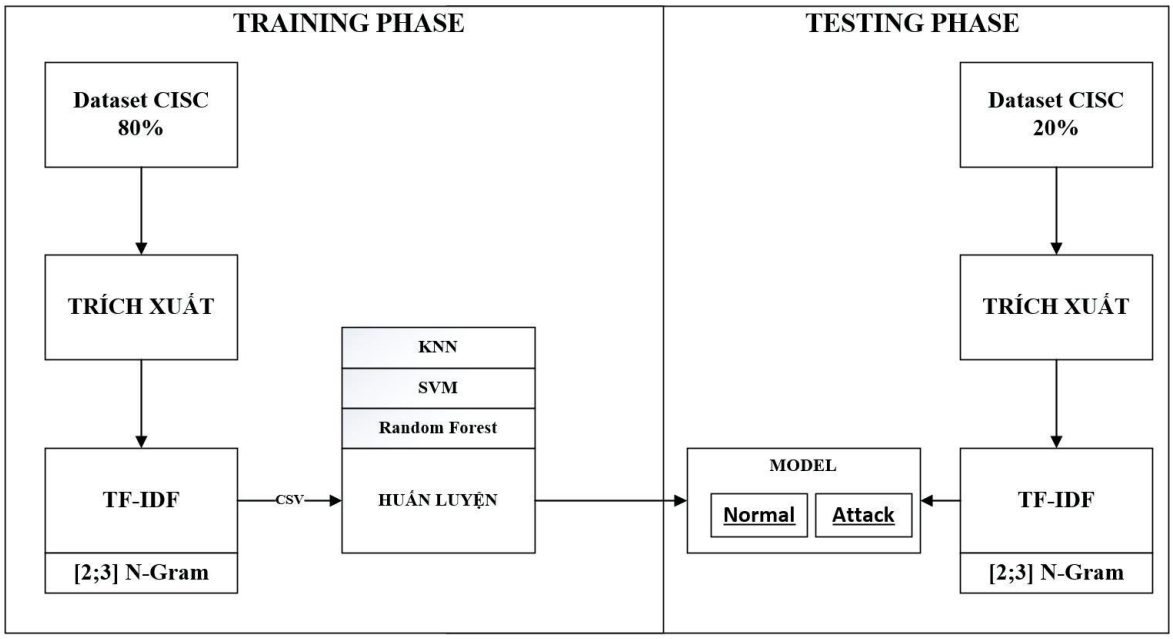
2) Các cuộc tấn công động sửa đổi các đối số yêu cầu hợp lệ: SQL SQL, CRLF, kịch bản chéo trang, tràn bộ đệm, v.v.

3) Vô tình yêu cầu bất hợp pháp. Các yêu cầu này không có mục đích xấu, tuy nhiên chúng không tuân theo hành vi thông thường của ứng dụng web và không có cấu trúc giống như các giá trị tham số bình thường (ví dụ: số điện thoại gồm các chữ cái).

3.2. Kịch bản thực nghiệm

Bộ dữ liệu CSIC đầu vào sẽ được chia thành nhiều tập khác nhau để kiểm nhiệm mô hình. Quá trình xây dựng mô hình bao gồm hai giai đoạn chính:

- Giai đoạn 1: Huấn luyện mô hình (Training phase)
- Giai đoạn 2: Kiểm thử mô hình (Testing phase).



Hình 3.1: Quá trình xây dựng mô hình

❖ **Giai đoạn huấn luyện mô hình (bao gồm 3 bước chính):**

- Bước 1: Bộ dữ liệu các request bình thường từ người dùng trong tập dữ liệu CSIC. Tại bước này, thực hiện tính toán sự xuất hiện của các ký tự quan trọng mới và lưu chúng trong cơ sở dữ liệu.
- Bước 2: Mô-đun không gian vectơ được sử dụng để chuyển đổi dữ liệu chuỗi thành các vectơ. Sử dụng kĩ thuật trích chọn dữ liệu tác giả đã giới thiệu và mô tả cách tính ở chương 2.
- Bước 3: Mô-đun xử lý dữ liệu sử dụng thuật toán học máy (lần lượt thay thế các thuật toán khác nhau để xác định mô hình tối ưu nhất cho bài toán: KNN, SVM, Random Forest).

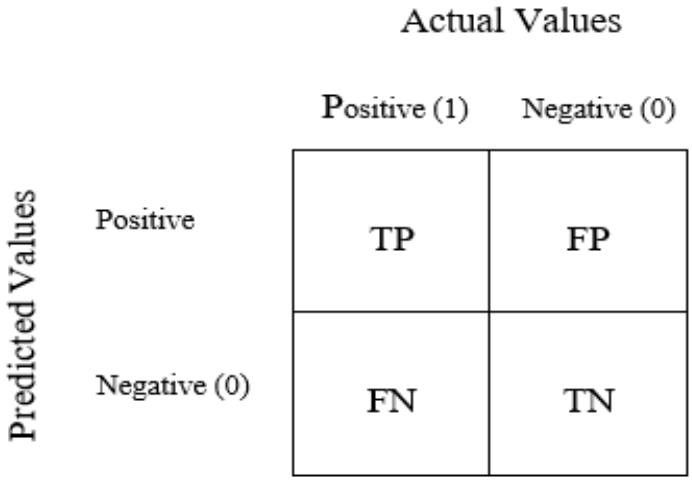
❖ **Giai đoạn kiểm thử mô hình:**

- Bước 1: Phần dữ liệu thử nghiệm được tiến hành loại bỏ nhãn dữ liệu.
- Bước 2: Thực hiện quá trình trích xuất đặc trưng dữ liệu tương tự bước 2 ở giai đoạn 1.

- Bước 3: Thử nghiệm các mô hình ứng với các thuật toán học máy đã được xây dựng ở giai đoạn 1. Tác giả lựa chọn phương pháp đánh giá độ chính xác bằng cách sử dụng ma trận độ đo (confusion matrix) được mô tả như sau:

Confusion Matrix là một phương pháp đánh giá kết quả của những bài toán phân loại với việc xem xét cả những chỉ số về độ chính xác và độ bao quát của các dự đoán cho từng lớp. Một confusion matrix gồm 4 chỉ số sau đối với mỗi lớp phân loại:

- **TP (True Positive)**: mẫu mang nhãn dương được phân lớp đúng vào lớp dương
- **TN (True Negative)**: mẫu mang nhãn âm được phân lớp đúng vào lớp âm.
- **FP (False Positive - Type 1 Error)**: mẫu mang nhãn âm bị phân lớp sai vào lớp dương.
- **FN (False Negative - Type 2 Error)**: mẫu mang nhãn dương bị phân lớp sai vào lớp âm.



Hình 3.2: Ma trận độ đo (Conusion matrix)

Ký hiệu TP là True Positive; TN là True Negative; FP là False Positive và FN là False Negative. Thực hiện phép đo Precision – Recall, trong đó, Precision là tỉ lệ số điểm TP trong những điểm được phân loại Positive, còn Recall là tỉ lệ số điểm TP trong số điểm thực sự là Positive. Công thức như sau:

$$\text{precision} = \frac{TP}{TP+FP}$$

$$\text{recall} = \frac{TP}{TP+FN}$$

Ta thấy rằng, Precision và Recall phủ càng cao thì càng tốt. Nhưng trong thực tế, hai giá này không thể đạt được cực đại cùng một lúc và thông thường phải tìm kiếm sự cân bằng. Thước đo $F1_{score}$ là trung bình hài hòa giữa Precision và Recall. Nó có xu hướng bằng không nếu hai giá trị này có xu hướng bằng không.

$$F1_{score} = 2 * \frac{precision \times recall}{precision + recall}$$

❖ **Kịch bản thực nghiệm chi tiết:**

Trong mô hình này đã sử dụng bộ dữ liệu bao gồm 25065 liên kết bất hợp pháp của một số loại tấn công (XSS, SQL injection) và 36000 liên kết hợp pháp. Bộ dữ liệu từ một số nguồn dữ liệu của các công cụ bảo vệ hệ thống như tệp nhật ký của hệ thống phát hiện và ngăn chặn xâm nhập, yêu cầu HTTP (phương thức GET, phương thức POST) của tường lửa ứng dụng Web.

Các bộ dữ liệu ban đầu đã được thực hiện phân chia thành hai phần riêng biệt với 80% các liên kết để đào tạo và 20% các liên kết để thử nghiệm. Trong quá trình thử nghiệm thêm một số phương pháp học máy để so sánh phương pháp đã đề xuất.

3.3. Một số kết quả thực nghiệm

Thực hiện thử nghiệm trên dữ liêu bao gồm:

- 36000 request bình thường;
- 25065 request bất thường;
- Tỉ lệ chia dữ liệu Traning/Testing là 8/2;
- Số lớp dữ liệu cần phần là 2 lớp: Bình thường/Bất thường.

Từ việc thực hiện phân chia dữ liệu đầu vào của tập bình thường thành các đoạn với tỉ lệ như trên, ta được bảng kết quả:

Bảng 3.1: Kết quả thực nghiệm xây dựng bộ phân lớp bình thường/bất thường theo kịch bản

	KNN				SVM				Rừng ngẫu nhiên			
	F1_Score	**Confusion Matrix**	**Precision**	**Recall**	**F1_Score**	**Confusion Matrix**	**Precision**	**Recall**	**F1_Score**	**Confusion Matrix**	**Precision**	**Recall**
N=2	0.967577	[6983 206] [262 4762]	0.971345	0.963837	0.983698	[7090 99] [136 4888]	0.986229	0.981179	0.985959	[7092 97] [105 4919]	0.986507	0.985411
N=3	0.978954	[7047 142] [161 4863]	0.980248	0.977664	**0.997219**	[7171 18] [22 5002]	0.997496	0.996941	0.984335	[7069 120] [105 4919]	0.983308	0.985364
N=4	0.986447	[7133 56] [140 4884]	0.99221	0.980751	0.994289	[7138 51] [31 4993]	0.992906	0.995676	0.978658	[7061 128] [178 4846]	0.982083	0.975257

Nhận xét: Kết quả sau khi chạy với 3 thuật toán học máy ta thu được mô hình tốt nhất với thuật toán SVM và Ngram = 3.

Phát hiện bất thường từ hành vi người dùng web là một vấn đề khó trong phòng chống tấn công ứng dụng web. Thuật toán phân loại được đề xuất để phát hiện các liên kết bất hợp pháp dựa trên ứng dụng phương pháp học máy với việc trích chọn các đặc trưng thuộc tính dữ liệu của người dùng. Thuật toán phát hiện liên kết bất hợp pháp phân tích các liên kết theo một chuỗi các bước để xác định xem liên kết đó là hợp pháp hay độc hại. Mặc dù thuật toán đề xuất cải thiện độ chính xác phân loại của các liên kết bất hợp pháp, nhưng với sự gia tăng số lượng tham số có trong các yêu cầu, độ chính xác phân loại sẽ giảm. Do đó, trong thời gian tới, cần tìm sự kết hợp của các phương pháp phát hiện bất thường dựa trên học sâu nhằm cải thiện độ chính xác phân loại không chỉ của các liên kết đáng ngờ mới đặc trưng các loại tấn công chưa được định danh.

Kết luận chương 3

Trong chương 3 luận văn đã xây dựng ba kịch bản thử nghiệm phân loại hành vi người dùng web. Với mỗi kịch bản đã xây dựng được mô hình học máy như: KNN, SVN, Random Forest.

Các kết quả thử nghiệm ban đầu cho thấy giải pháp phát hiện tấn công web ứng dụng dựa trên kỹ thuật phân tích hành vi đề xuất có tính khả thi cao và phù hợp với các yêu cầu đề ra.

KẾT LUẬN

1. Những đóng góp của luận văn

Với mục tiêu nghiên cứu các phương pháp phát hiện tấn công web ứng dụng dựa trên kỹ thuật phân tích hành vi và thử nghiệm, luận văn đã đi sâu nghiên cứu các vấn đề xung quanh đề tài nghiên cứu, các thuật toán học máy phát hiện tấn công web để ứng dụng vào phát hiện hành vi bất thường của người dùng.

Những kết quả chính đã đạt được trong luận văn:

- Khảo sát một số nguy cơ mất an toàn thông tin thông qua các kĩ thuật tấn công web, đưa ra các phương pháp phòng chống tấn công web phổ biến cũng như đưa ra một số phương pháp nhằm nâng cao bảo mật hệ thống.
- Tìm hiểu phương pháp phát hiện tấn công web dựa trên kỹ thuật phân tích hành vi. Thực hiện trích xuất hành vi bất thường từ bộ dữ liệu về tấn công web (bộ dữ liệu CSIC 2010) sử dụng kĩ thuật trích chọn TF-IDF kết hợp N-Gram.
- Lựa chọn và ứng dụng thuật toán học máy nhằm phân loại hành vi tấn công và hành vi bình thường lên web, sử dụng các thuật toán học máy có giám sát: KNN, SVM, Random forest.
- Thực nghiệm xây dựng mô hình bộ phân lớp bình thường/bất thường theo từng kịch bản để đưa ra mô hình tốt nhất khi sử dụng N-Gram với n=3.

2. Hướng phát triển của luận văn

Một số hướng phát triển tiếp theo của luận văn:

- Mặc dù thuật toán đề xuất cải thiện độ chính xác phân loại của các liên kết bất hợp pháp, nhưng với sự gia tăng số lượng tham số có trong các yêu cầu, độ chính xác phân loại sẽ giảm. Do đó, cần tìm sự kết hợp của các phương pháp phát hiện bất thường dựa trên học sâu nhằm cải thiện độ chính xác phân loại không chỉ của các liên kết đáng ngờ mới đặc trưng các loại tấn công chưa được định danh.

- Thực hiện nghiên cứu phương pháp phát hiện tấn công web dựa trên kỹ thuật phân tích hồ sơ hành vi.

DANH MỤC CÁC TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] GSTS Nguyễn Thúc Hải – "Mạng máy tính và các hệ thống mở", NXB Giáo dục, 1989.

[2] Vũ Hữu Tiệp (2016-2020) – "Machine Learning cơ bản".

Tiếng Anh

[3] Trustware SpiderLabs, "ModSecurity: Open Source Web Application Firewall," https://www.modsecurity.org/.

[4] Ying Dong, Yuqing Zhang, Hua Ma et al., "An adaptive system for detecting malicious queries in web attacks," *Science China Information Sciences*, vol. 61, no. 3, Article ID 032114, 2018.

[5] C. Torrano Gimenez, H. T. Nguyen, G. Alvarez, K. Franke, "Combining expert knowledge with automatic feature extraction for reliable web attack detection," *Security and Communication Networks*, vol. 8, no.16, pp. 2750–2767, 2015.

[6] Neline van Ginkel, Willem De Groef, Fabio Massacci, Frank Piessens, "A Server-Side JavaScript Security Architecture for Secure Integration of Third-Party Libraries," *Security and Communication Networks*, vol. 2019, no. 6, pp. 1-21, 2019.

[7] Muhammad Hilmi Kamarudin, Carsten Maple, Tim Watson, Nader Sohrabi Safa, "A New Unified Intrusion Anomaly Detection in Identifying Unseen Web Attacks," *Security and Communication Networks*, vol. 2017, no. 1, pp. 1- 18, 2017.

[8] Hu Y, Li B, Ye W, Yuan G. "A Human-Machine Collaborative Detection Model for Identifying Web Attacks," in *Proceedings of the International Conference on Collaborative Computing: Networking, Applications and Worksharing, CollaborateCom 2017*, pp.109-119, Shanghai, China, 1-3 December 2017.

[9] WenChuan Yang, Wen Zuo, BaoJiang Cui, "Detecting Malicious URLs via a Keyword-based Convolutional Gated-recurrent-unit Neural Network," *IEEE Access*, Volume 7, no. 2019, pp. 29891 – 29900, 2019.

[10] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, T. Berners-Lee, "Hypertext Transfer Protocol -- HTTP/1.1", 1999, https://tools.ietf.org/html/rfc2616#section-5.

[11] ModSecurity Core Rule Set Project, "OWASP ModSecurity Core Rule Set," 2016, https://coreruleset.org/.

[12] K. K. Mookhey, "Evasion and Detection of Web Application Attacks," *Black Hat USA 2004*, https://www.blackhat.com/presentations/bh-usa-04/bh-us-04-mookhey/bh-us-04-mookhey-up.ppt.

Trang web:

[13] (2016). What are Web Application Vulnerabilities?. Available: https://www.rapid7.com/fundamentals/web-application-vulnerabilities/. Truy cập ngày 15/2/2020.

[14] (2010). HTTP DATA SET CSIC 2010, External Data Source [Online]. Available: https://www.impactcybertrust.org/dataset_view?idDataset=940. Truy cập ngày 15/2/2020.

[15] (2018). Hơn 120.000 website bị tin tặc tấn công trong quý 3 năm 2018. Available: https://cystack.net/vi/resource/website-bi-tin-tac-tan-cong-quy-3-2018/. Truy cập ngày 18/03/2020.

[16] (2020). Quản trị website là gì? Người quản trị cần có những kỹ năng nào?. Available: https://wtstats.info/quan-tri-website/. Truy cập ngày 19/05/2020.