HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG
KHOA AN TOÀN THÔNG TIN

BÁO CÁO BÀI THỰC HÀNH
HỌC PHẦN: THỰC TẬP CƠ SỞ
MÃ HỌC PHẦN: INT13147

BÀI THỰC HÀNH 2.2
TÌM HIỂU VÀ CÀI ĐẶT, CẤU HÌNH NIDS

Sinh viên thực hiện:

B22DCAT018 – Nguyễn Hoàng Anh

Giảng viên hướng dẫn: ThS. Ninh Thị Thu Trang

HỌC KỲ 2 NĂM HỌC 2024-2025

MỤC LỤC

DANH MỤC CÁC HÌNH VẼ

CHƯƠNG 1. TÌM HIỂU LÝ THUYẾT

1.1 Mục đích

Tìm hiểu và luyện tập việc cài đặt và vận hành các hệ thống phát hiện xâm nhập cho host (HIDS) và cho mạng (NIDS).

Luyện tập việc tạo và chỉnh sửa các luật phát hiện tấn công, xâm nhập cho các hệ thống phát hiện xâm nhập thông dụng.

1.2 Tìm hiểu lý thuyết

1.2.1 Tìm hiểu khái quát về các hệ thống phát hiện tấn công, xâm nhập, phân loại các hệ thống phát hiện xâm nhập, các kỹ thuật phát hiện xâm nhập.

1.2.1.1 Hệ thống phát hiện tấn công, xâm nhập

Hệ thống phát hiện xâm nhập (IDS - Intrusion Detection System) và hệ thống ngăn chặn xâm nhập (IPS - Intrusion Prevention System) là hai công nghệ quan trọng trong bảo mật mạng.

IDS (Intrusion Detection System): Giám sát và phát hiện các hoạt động đáng ngờ hoặc tấn công mạng nhưng không thực hiện chặn mà chỉ ghi nhận lại.

IPS (Intrusion Prevention System): Không chỉ phát hiện mà còn tự động ngăn chặn các tấn công bằng cách chặn gói tin hoặc cách ly hệ thống bị ảnh hưởng.

1.2.1.2 Phân loại các hệ thống phát hiện xâm nhập

Các hệ thống IDS/IPS có thể được phân loại theo nhiều tiêu chí khác nhau:

Phân loại theo cách triển khai

- HIDS (Host-based IDS) – Giám sát trên từng thiết bị/máy chủ.
- NIDS (Network-based IDS) – Giám sát toàn bộ mạng thông qua luồng dữ liệu.

Phân loại theo phương pháp phát hiện

- Phát hiện dựa trên chữ ký (Signature-based Detection): Sử dụng cơ sở dữ liệu chứa mẫu tấn công đã biết (ví dụ: tấn công SQL Injection, Cross-site Scripting). Nhanh, chính xác nhưng dễ bị bypass bởi các kỹ thuật tấn công mới.
- Phát hiện dựa trên hành vi (Anomaly-based Detection):Phân tích hành vi mạng và phát hiện các hoạt động bất thường và phát hiện được các mối đe dọa chưa từng thấy nhưng có thể gây báo động giả (False Positive).

- Phát hiện lai (Hybrid Detection): Kết hợp cả hai phương pháp trên để tăng độ chính xác.

1.2.2 Tìm hiểu về kiến trúc và tính năng của một số hệ thống phát hiện tấn công, xâm nhập, như Snort, Suricata, Zeek, OSSEC, Wazuh...

1.2.2.1 Snort

Loại: NIDS (Network-based IDS)

Phương pháp: Dựa trên chữ ký (Signature-based) và hành vi (Anomaly-based)

Kiến trúc: Snort hoạt động theo 4 giai đoạn chính:

- Packet Capture (Bắt gói tin) – Sử dụng thư viện PCAP để thu thập lưu lượng mạng.
- Preprocessing (Tiền xử lý) – Phân loại và chuẩn bị dữ liệu gói tin.
- Detection Engine (Bộ phát hiện) – So sánh dữ liệu với danh sách chữ ký và hành vi bất thường.
- Logging & Alerting (Ghi log và cảnh báo) – Lưu kết quả hoặc gửi cảnh báo đến quản trị viên

Tính năng: Phân tích lưu lượng mạng theo thời gian thực. Hỗ trợ quy tắc tùy chỉnh để phát hiện tấn công. Có thể hoạt động như IDS (chỉ phát hiện) hoặc IPS (ngăn chặn tấn công).

1.2.2.2 Suricata

Loại: NIDS/NIPS (Network-based IDS/IPS)

Phương pháp: Dựa trên chữ ký (Signature-based) và hành vi (Anomaly-based)

Kiến trúc: Suricata có kiến trúc đa luồng (multi-threaded), gồm các thành phần chính:

- Packet Acquisition – Sử dụng PCAP/DAG/AF_PACKET để lấy dữ liệu.
- Flow Engine – Phân tích luồng mạng, hỗ trợ IPv4, IPv6, TCP, UDP.
- Detection Engine – Sử dụng quy tắc Snort-compatible và phát hiện dựa trên hành vi.
- Logging & Output – Xuất log ra JSON, Syslog, Elasticsearch.

Tính năng: Hiệu suất cao nhờ kiến trúc đa luồng. Hỗ trợ phát hiện tấn công bằng cách kiểm tra nội dung gói tin. Tích hợp với IPS để chặn tấn công tự động.

1.2.2.3 Zeex

Loại: NIDS (Network-based IDS)

Phương pháp: Dựa trên hành vi (Anomaly-based)

Kiến trúc: Zeek có mô hình kiến trúc phân lớp:

- Event Engine – Bắt và xử lý gói tin.
- Policy Script Interpreter – Chạy các script để phân tích hành vi.
- Logging & Analysis – Ghi log chi tiết các sự kiện mạng.

Tính năng: Phân tích sâu về lưu lượng mạng (Deep Packet Inspection). Ghi log chi tiết về HTTP, DNS, FTP, SSH. Hỗ trợ script tùy chỉnh để phát hiện hành vi bất thường.

1.2.2.4 OSSEC

Loại: HIDS (Host-based IDS)

Phương pháp: Dựa trên chữ ký (Signature-based) và hành vi (Anomaly-based)

Kiến trúc: OSSEC có kiến trúc tập trung, gồm các thành phần chính:

- Agents – Cài đặt trên máy chủ/máy trạm để thu thập dữ liệu.
- Manager – Nhận và phân tích log từ các agent.
- Database – Lưu trữ log và cảnh báo.

Tính năng: Giám sát file, registry, log hệ thống. Tích hợp với SIEM để phân tích bảo mật. Kiểm tra tính toàn vẹn của file (File Integrity Monitoring).

1.2.2.5 Wazuh

Loại: HIDS + SIEM (Security Information & Event Management)

Phương pháp: Dựa trên hành vi và phân tích log

Kiến trúc: Wazuh có mô hình quản lý tập trung:

- Agents – Cài trên máy chủ/máy trạm để thu thập dữ liệu.
- Wazuh Manager – Phân tích dữ liệu từ agent.
- Dashboard (Kibana/Elasticsearch) – Hiển thị dữ liệu bảo mật.

Tính năng: Phát hiện tấn công nội bộ, mã độc, brute-force. Kiểm tra tuân thủ bảo mật (ISO 27001, PCI-DSS, GDPR). Tích hợp với Cloud Security để giám sát hệ thống đám mây.

CHƯƠNG 2. NỘI DUNG THỰC HÀNH

2.1 Chuẩn bị môi trường

01 máy tính (máy thật hoặc máy ảo) chạy Linux với RAM tối thiểu 2GB, 10GB đĩa cứng có kết nối mạng (LAN hoặc Internet).

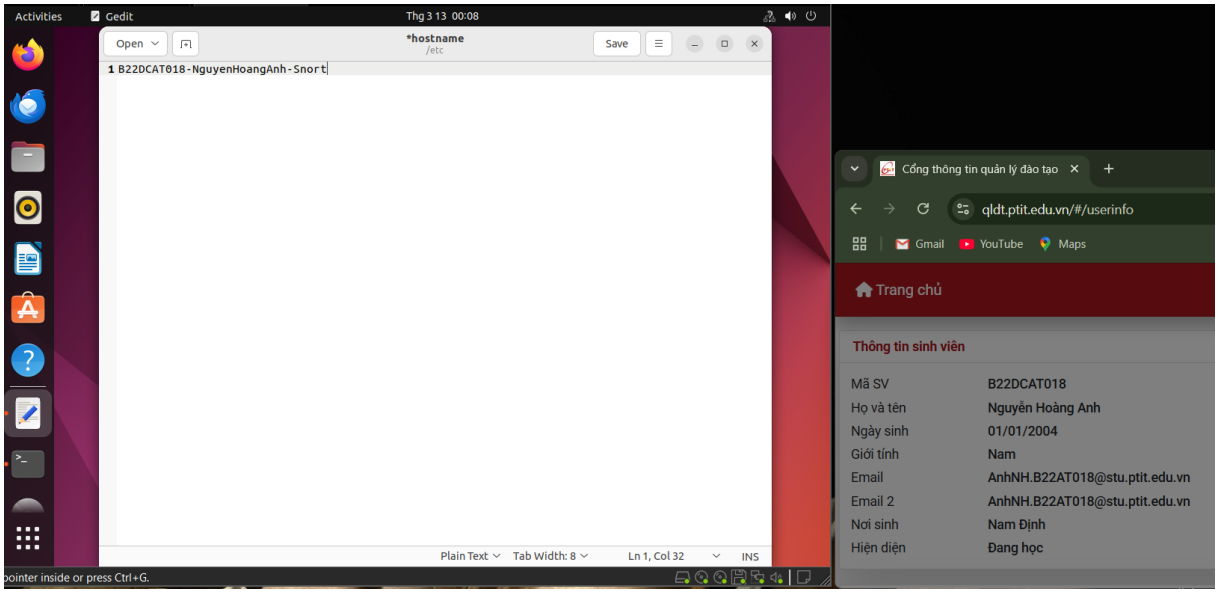
01 máy tính (máy thật hoặc máy ảo) chạy Kali Linux (bản 2021 trở lên)

Bộ phần mềm Snort tải tại https://www.snort.org/downloads

2.2 Các bước thực hiện

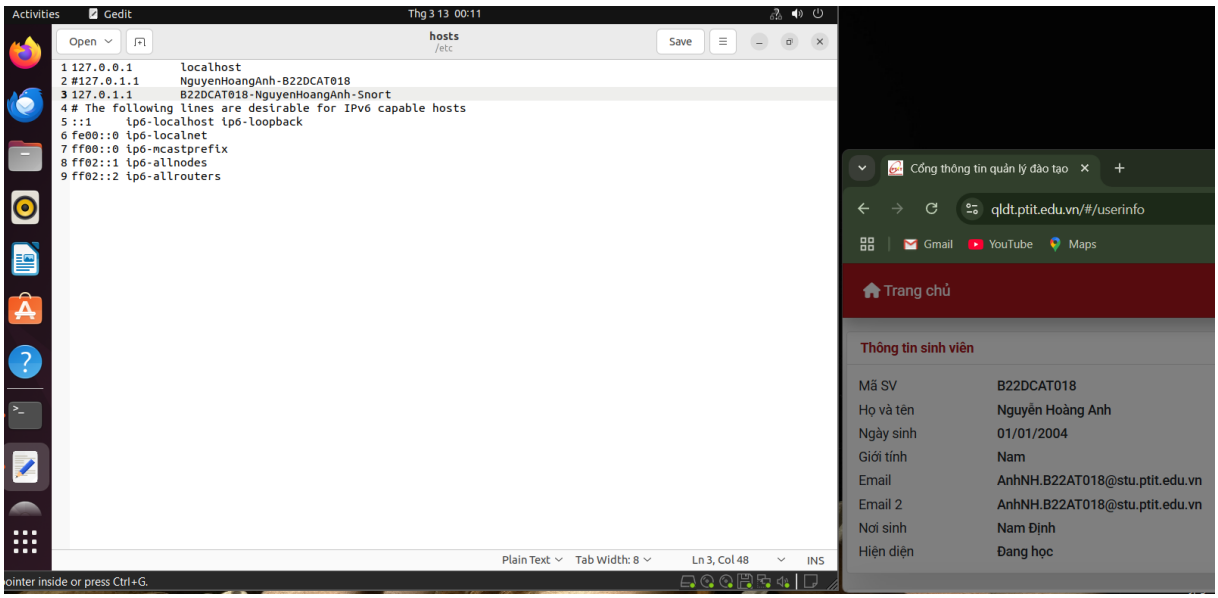
Bước 1: Chuẩn bị các máy tính như mô tả trong mục 2.2. Máy Kali Linux được đổi tên thành <Mã SV-Tên SV>-Kali và máy cài Snort thành <Mã SV-Tên SV>-Snort. Các máy có địa chỉ IP và kết nối mạng LAN:

Dùng gedit để đổi tên máy Linux thành: B22DCAT018-NguyenHoangAnh-Snort
sudo gedit /etc/hostname



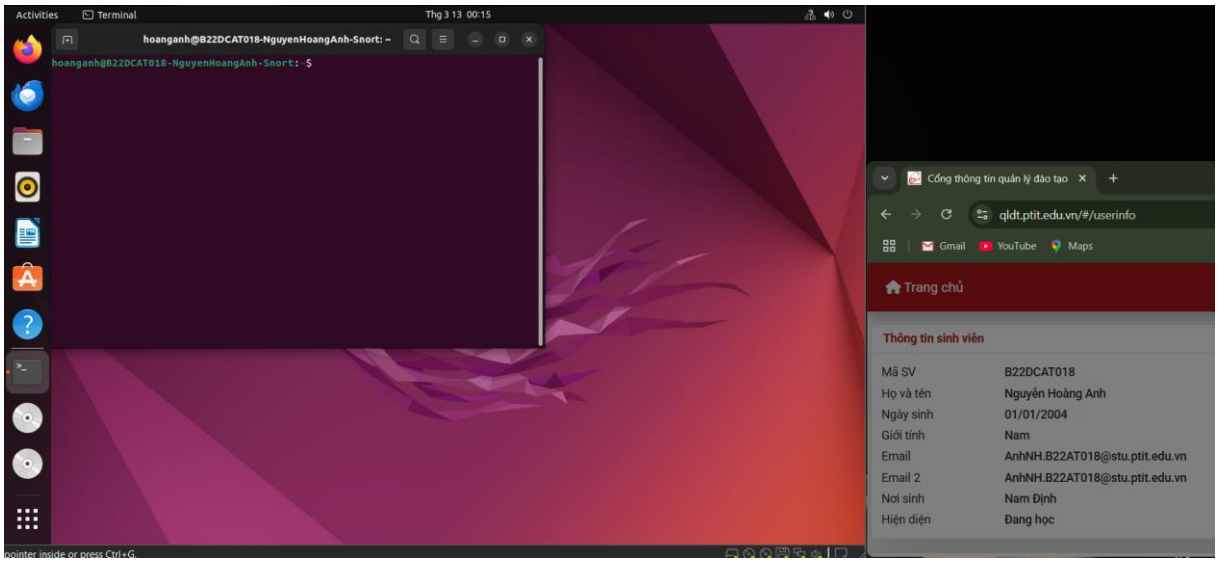
Hình 1 - đổi tên máy Linux (bước 1)

sudo gedit /etc/hosts



Hình 2 - đổi tên máy linux (bước 2)

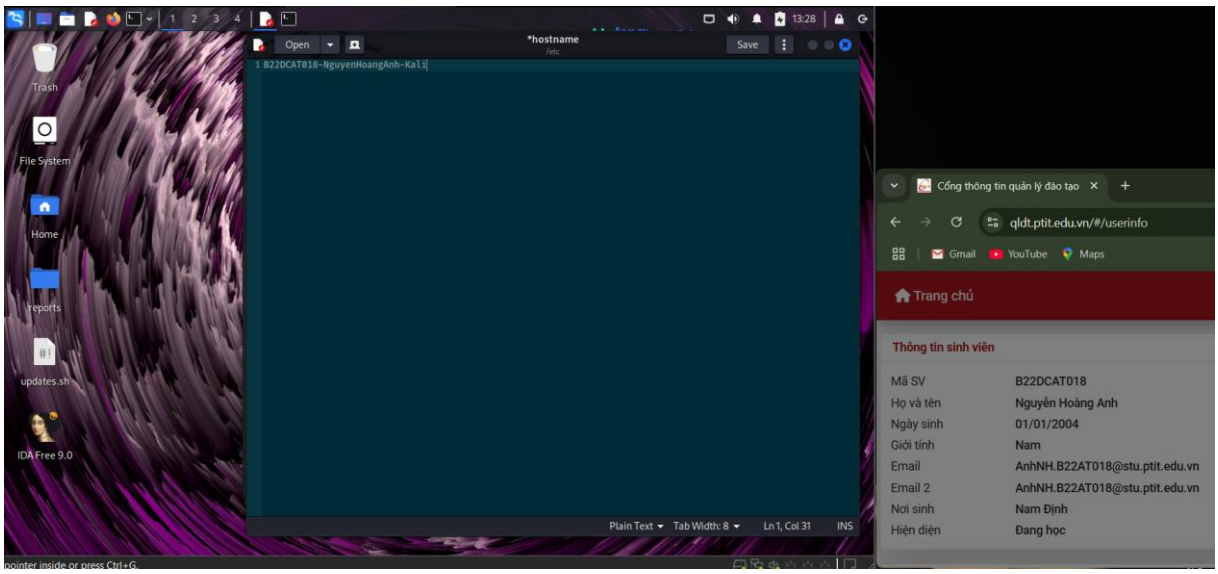
Khởi động lại máy để đổi tên thành công:



Hình 3 - đổi tên máy linux thành công

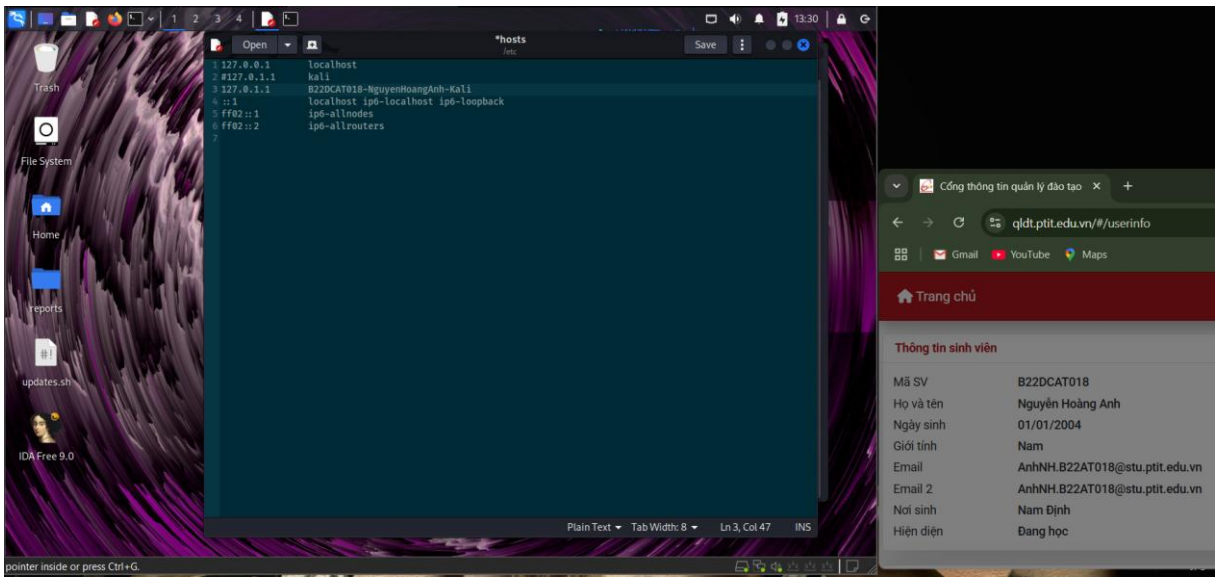
Đổi tên máy kali thành: B22DCAT018-NguyenHoangAnh-Kali

Sudo gedit /etc/hostname



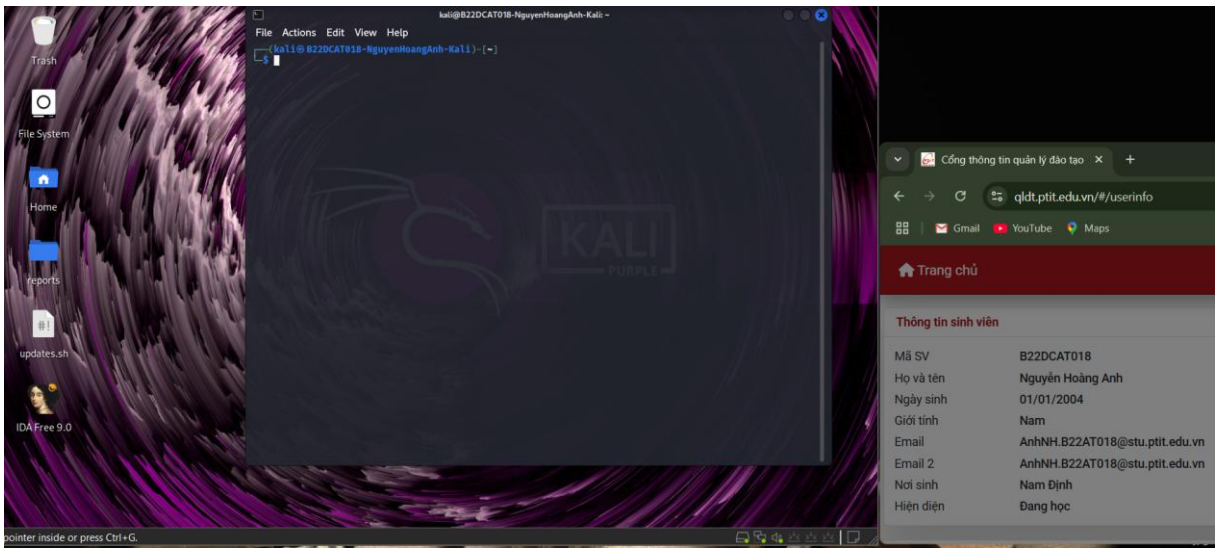
Hình 4 - đổi tên máy kali (bước 1)

Sudo gedit /etc/hosts



Hình 5 - đổi tên máy kali (bước 2)

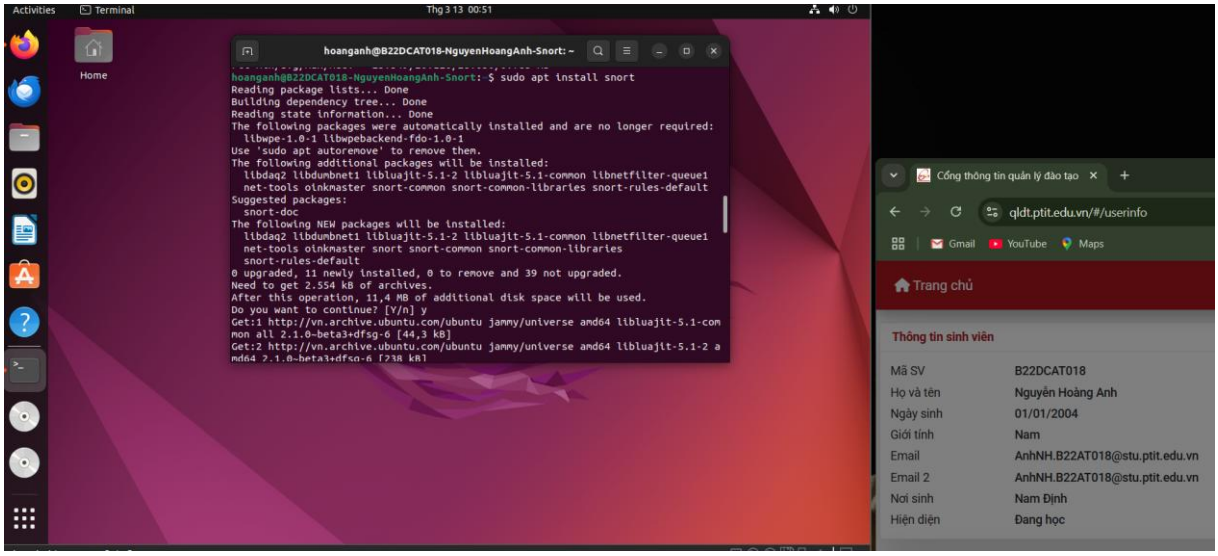
Đổi tên máy thành công



Hình 6 - đổi tên máy kali thành công

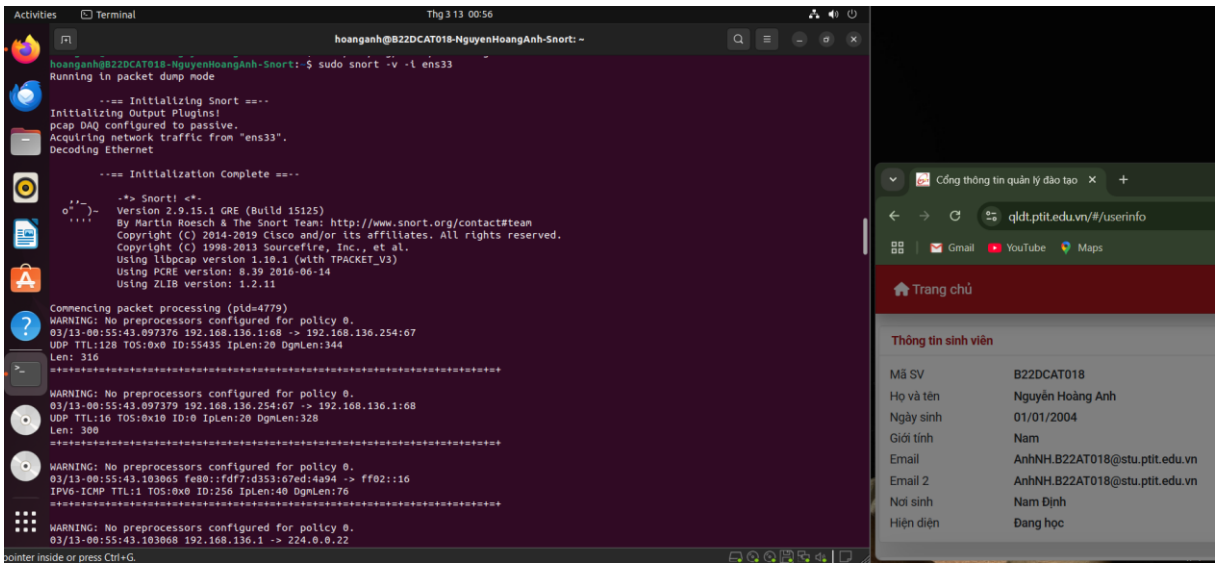
Bước 2: Tải, cài đặt Snort và chạy thử Snort. Kiểm tra log của Snort để đảm bảo Snort hoạt động bình thường.

Cài đặt snort: *sudo apt install snort*



Hình 7 - cài đặt snort

Chạy thử snort:



Hình 8 - chạy thử snort

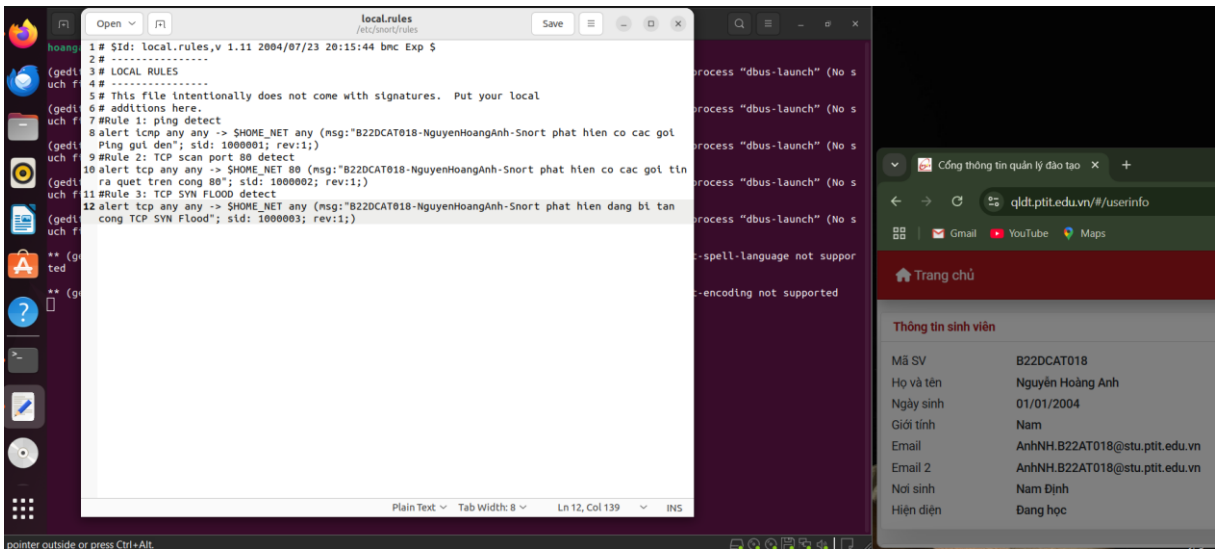
Bước 3: Tạo các luật Snort để phát hiện 3 dạng rà quét, tấn công hệ thống:

+ Phát hiện các gói tin ping từ bất kỳ một máy nào gửi đến máy chạy Snort. Hiển thị thông điệp khi phát hiện: "<Mã SV-Tên SV>-Snort phát hiện có các gói Ping gửi đến."

+ Phát hiện các gói tin rà quét từ bất kỳ một máy nào gửi đến máy chạy Snort trên cổng 80. Hiển thị thông điệp khi phát hiện: "<Mã SV-Tên SV>-Snort phát hiện có các gói tin rà quét trên cổng 80."

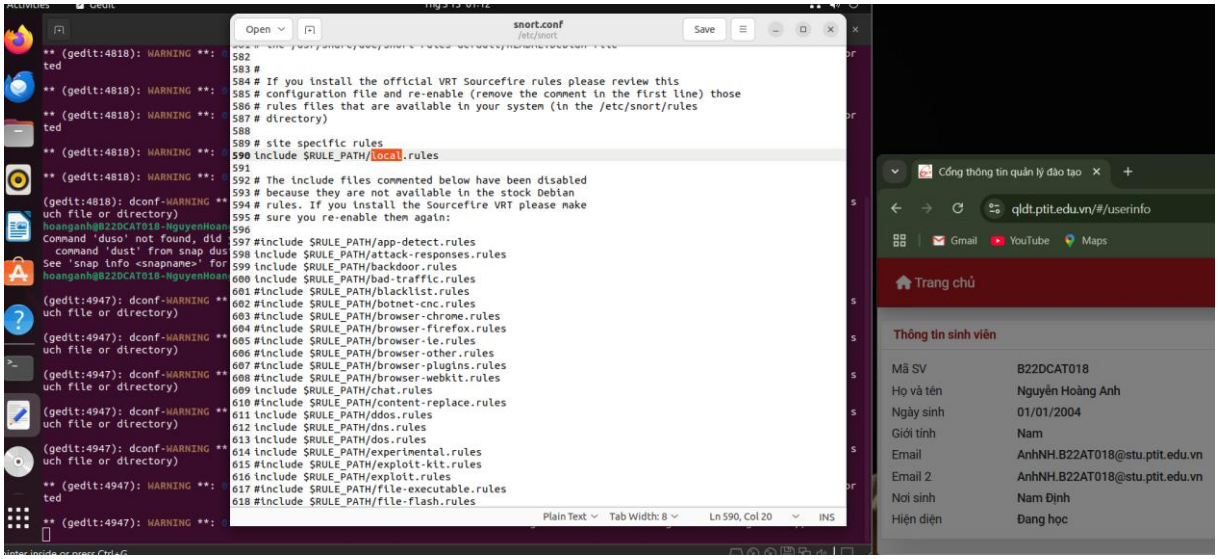
+ Phát hiện tấn công TCP SYN Flood từ bất kỳ một máy nào gửi đến máy chạy Snort. Hiển thị thông điệp khi phát hiện: "<Mã SV-Tên SV>-Snort phát hiện đang bị tấn công TCP SYN Flood."

Tạo 3 luật Snort trong gedit: *sudo gedit /etc/snort/rules/local.rules* theo yêu cầu đề bài:



Hình 9 - tạo 3 luật Snort

Đã áp dụng 3 rules trong *sudo gedit /etc/snort/snort.conf* (dòng 590)



Hình 10 - xác nhận áp dụng 3 luật vừa tạo

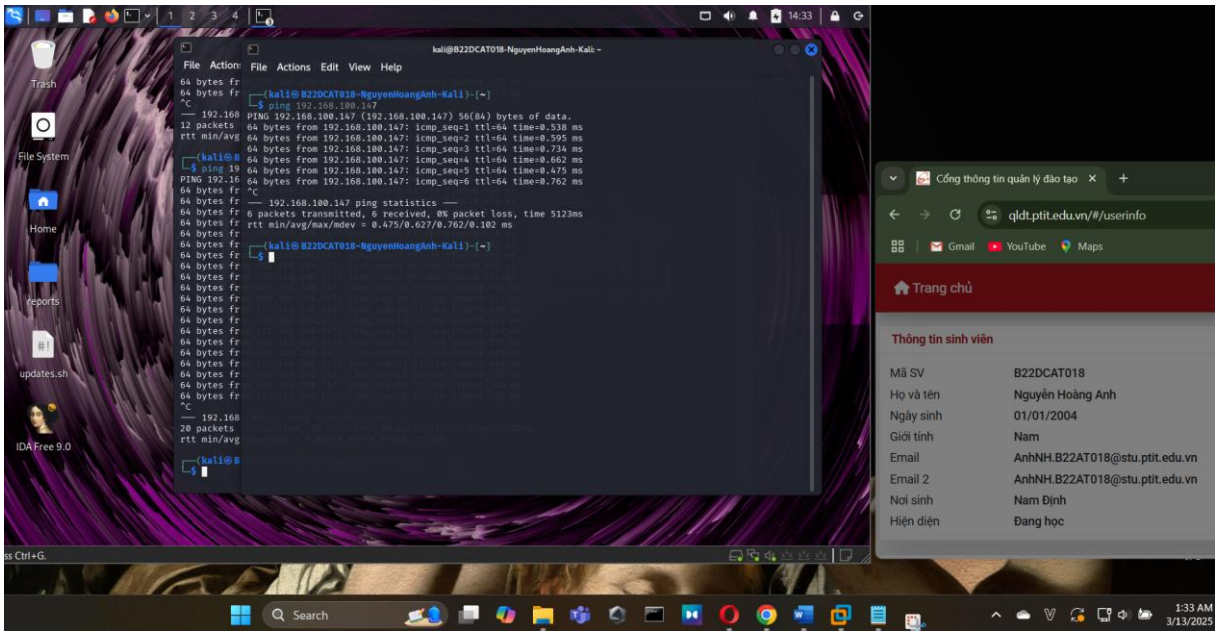
Bước 4: thực thi tấn công và phát hiện sử dụng Snort

+ Từ máy Kali, sử dụng lệnh ping để ping máy Snort. Trên máy Snort kiểm tra kết quả phát hiện trên giao diện terminal hoặc log của Snort.

+ Từ máy Kali, sử dụng công cụ nmap để rà quét máy Snort (dùng lệnh: nmap -sV -p80 -A <địa chỉ IP máy Snort>). Trên máy Snort kiểm tra kết quả phát hiện trên giao diện terminal hoặc log của Snort.

+ Từ máy Kali, sử dụng công cụ hping3 để tấn công TCP SYN Flood máy Snort (dùng lệnh: hping3 -c 15000 -d 120 -S -w 64 -p 80 --flood --rand-source <địa chỉ IP máy Snort>). Trên máy Snort kiểm tra kết quả phát hiện trên giao diện terminal hoặc log của Snort.

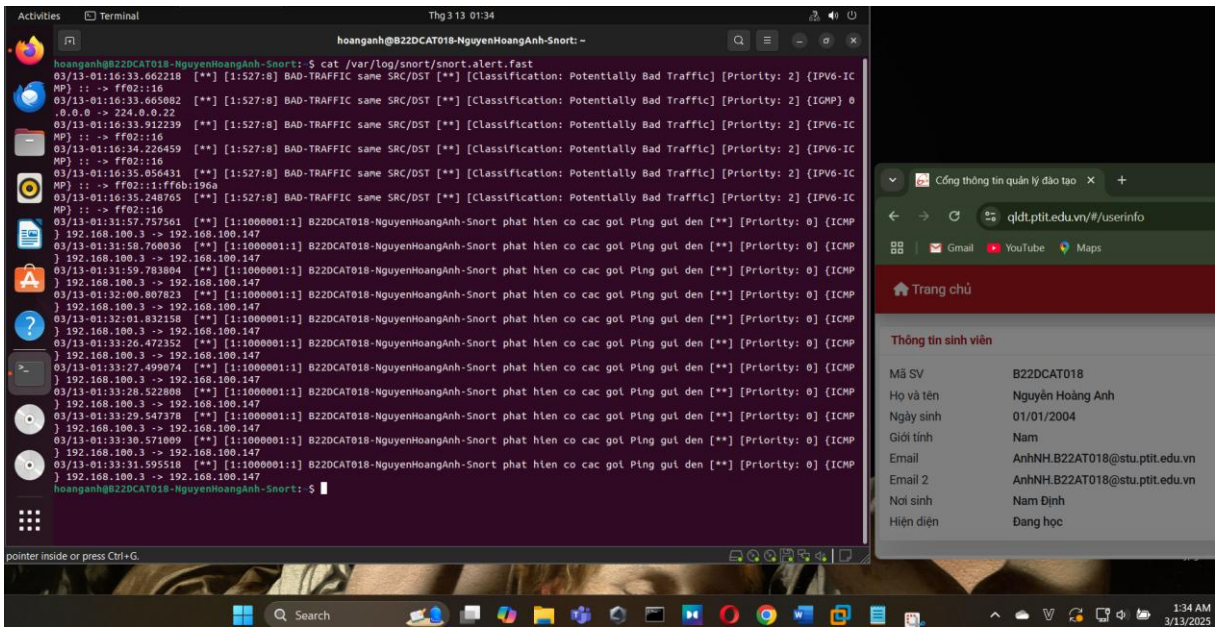
Từ máy kali ping đến máy snort



Hình 11 - ping từ máy kali đến máy snort

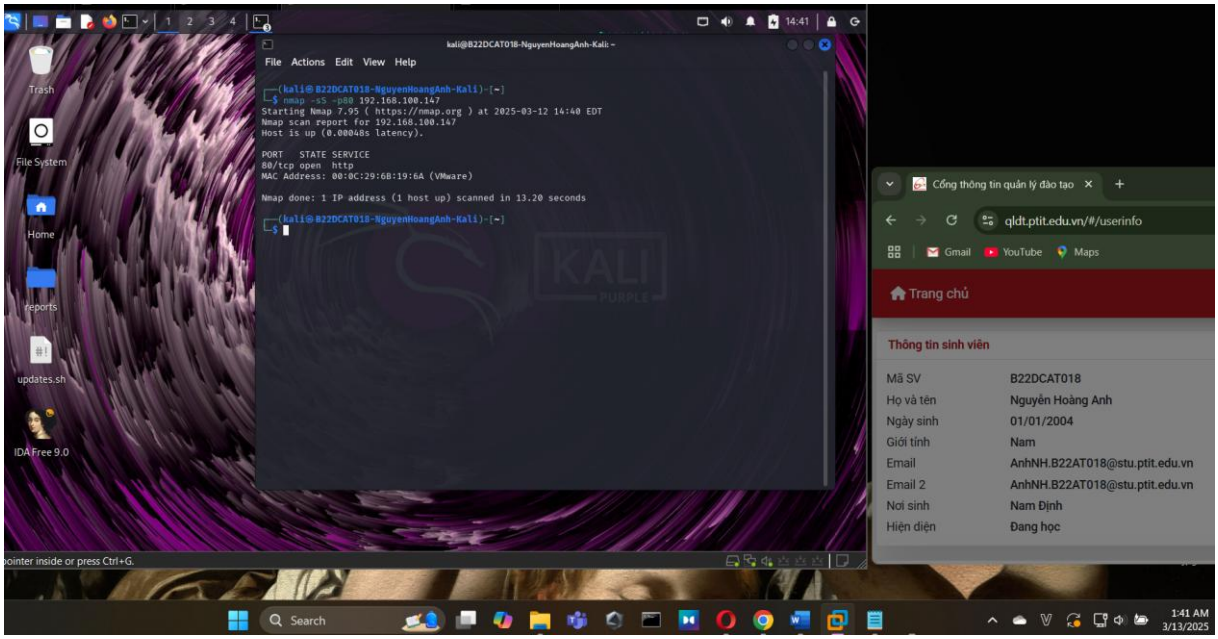
Máy snort phát hiện có máy ping đến bằng cách đọc file snort.alert.fast:

cat /var/log/snort/snort.alert.fast



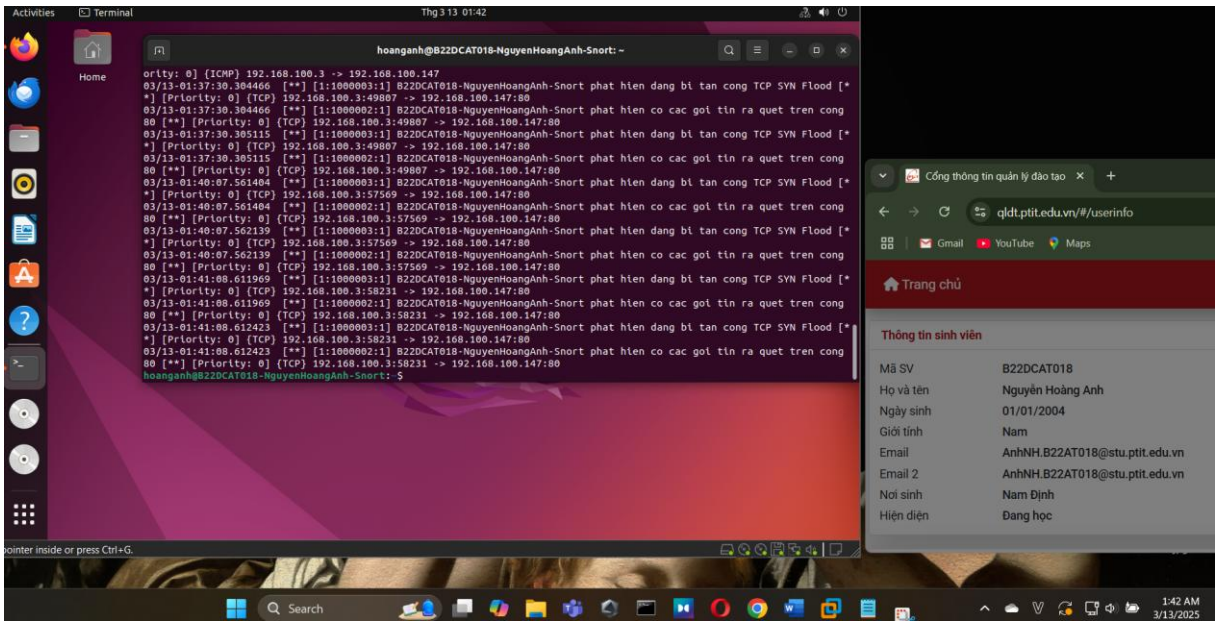
Hình 12 - phát hiện có máy ping đến

Sử dụng nmap để rà quét máy snort:



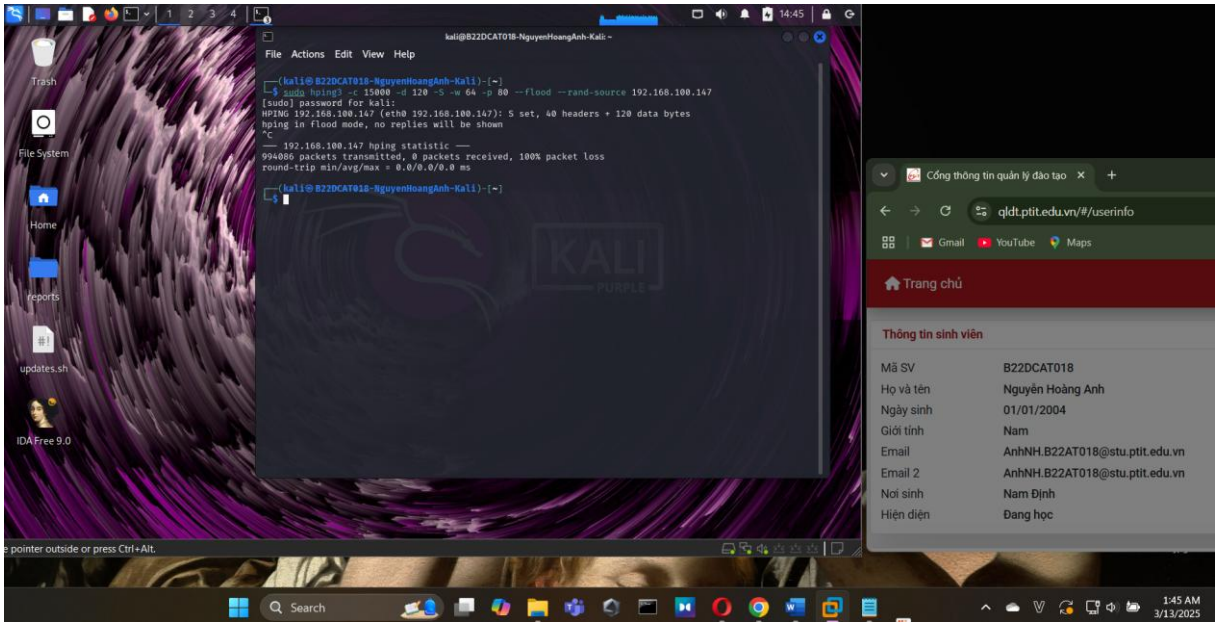
Hình 13 - rà quét máy snort bằng nmap

Phát hiện rà quét cổng trên máy snort



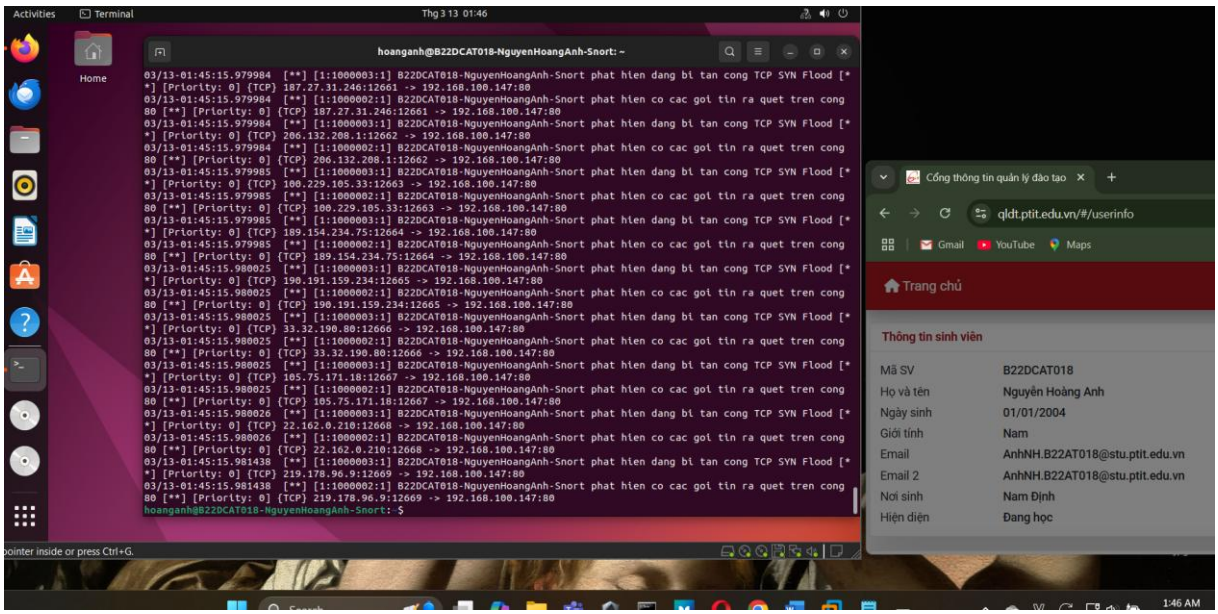
Hình 14 - phát hiện rà quét cổng trên máy snort

Trên máy kali, sử dụng công cụ hping3 để tấn công TCP SYN Flood máy Snort



Hình 15 - tấn công TCP SYN Flood đến máy snort

Trên máy snort, phát hiện bị tấn công TCP SYN Flood



Hình 16 - phát hiện tấn công TCP SYN Flood trên máy snort

TỔNG KẾT

Trong bài thực hành này, em đã tìm hiểu và triển khai hệ thống phát hiện xâm nhập mạng (NIDS) bằng Snort trên môi trường Linux. Quá trình thực hành bao gồm việc cài đặt, cấu hình và kiểm thử các luật phát hiện tấn công nhằm đánh giá khả năng giám sát và bảo vệ hệ thống trước các mối đe dọa từ mạng.

Thông qua các bước thực hiện, em đã:

- Cài đặt và cấu hình Snort thành công trên hệ thống mục tiêu.
- Viết và áp dụng các quy tắc Snort để phát hiện ba loại tấn công:
    - Ping request: phát hiện gói tin ICMP
    - Port scanning trên cổng 80: nhận diện hành vi rà quét dịch vụ web
    - TCP SYN Flood attack: phát hiện tấn công từ chối dịch vụ
- Thực hiện kiểm thử bằng cách gửi các cuộc tấn công từ Kali linux đến máy chạy snort, quan sát và xác nhận rằng snort đã ghi nhận chính xác các sự kiện tấn công trong log.

Kết quả thực hành cho thấy Snort là một công cụ mạnh mẽ trong việc giám sát an ninh mạng, giúp phát hiện kịp thời các cuộc tấn công tiềm ẩn. Bài thực hành cũng giúp em hiểu rõ hơn về cách xây dựng và tùy chỉnh các quy tắc IDS, cũng như cách phân tích log để phản hồi trước các mối đe dọa an ninh mạng.

Bài thực hành này đã giúp em có cái nhìn thực tế hơn về cách thức hoạt động của một hệ thống phát hiện xâm nhập, đồng thời nâng cao kỹ năng làm việc với các công cụ bảo mật trong môi trường thực tế.

TÀI LIỆU THAM KHẢO

1. Chương 5, Giáo trình Cơ sở an toàn thông tin, Học viện Công nghệ BVCT, 2020.
2. Suricata: https://suricata.io/documentation/
3. Snort: https://www.snort.org/#documents
4. OSSEC: https://www.ossec.net/docs/
5. Wazuh: https://documentation.wazuh.com/current/index.html