HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

NGUYỄN CÔNG HIẾU

NGHIÊN CỨU XÂY DỰNG GIẢI PHÁP PHÁT HIỆN XÂM NHẬP VÀ ỨNG DỤNG CHO HỌC VIỆN THANH THIẾU NIÊN VIỆT NAM

LUẬN VĂN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

HÀ NỘI – 2022

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

NGUYỄN CÔNG HIẾU

NGHIÊN CỨU XÂY DỰNG GIẢI PHÁP PHÁT HIỆN XÂM NHẬP VÀ ỨNG DỤNG CHO HỌC VIỆN THANH THIẾU NIÊN VIỆT NAM

CHUYÊN NGÀNH: KHOA HỌC MÁY TÍNH

MÃ SỐ: 8.48.01.01

LUẬN VĂN THẠC SĨ KỸ THUẬT

(Theo định hướng ứng dụng)

NGƯỜI HƯỚNG DẪN KHOA HỌC:

TS. DƯƠNG TRẦN ĐỨC

HÀ NỘI – 2022

LỜI CẢM ƠN

Đề tài " Nghiên cứu xây dựng giải pháp phát hiện xâm nhập và ứng dụng cho Học viện Thanh thiếu niên Việt Nam" là nội dung mà em đã nghiên cứu và làm luận văn tốt nghiệp sau thời gian theo học tại Khoa Sau Đại học trường Học viện Bưu Chính Viễn Thông. Trong quá trình nghiên cứu và hoàn thành luận văn, em đã nhận được rất nhiều sự quan tâm, giúp đỡ của các thầy cô giáo, đồng nghiệp, gia đình và bạn bè.

Để có thể hoàn thiện được luận văn thạc sĩ của mình, trước tiên, em xin bày tỏ lòng biết ơn sâu sắc nhất tới thầy - TS.Dương Trần Đức. Sự gần gũi và nhiệt tình hướng dẫn của thầy là nguồn động lực rất lớn đối với em trong suốt thời gian thực hiện luận văn.

Em cũng xin gửi lời cảm ơn chân thành nhất tới tất cả các thầy cô Học viện Công nghệ Bưu chính viễn thông đã nhiệt tình giảng dạy, cung cấp, hướng dẫn cho em những kiến thức, kinh nghiệm trong suốt quá trình học tập.

Đồng thời em cũng xin gửi lời cảm ơn đến người thân trong gia đình, các bạn học viên, đồng nghiệp nơi tôi công tác đã giúp đỡ, động viên, tạo điều kiện tốt nhất cho tôi trong suốt khóa học tại Học viện Công nghệ Bưu chính viễn thông để tôi có thể hoàn thiện tốt luận văn thạc sĩ của mình.

Tôi xin chân thành cảm ơn !

Hà Nội, ngày tháng năm 2022

Học viên

Nguyễn Công Hiếu

LỜI CAM ĐOAN

Tôi cam đoan rằng, luận văn "Nghiên cứu xây dựng giải pháp phát hiện xâm nhập và ứng dụng cho Học viện thanh thiếu niên Việt Nam" là bài nghiên cứu của chính tôi. Ngoại trừ những tài liệu tham khảo được trích dẫn trong luận văn này, tôi cam kết là không có nghiên cứu nào của người khác được sử dụng trong luận văn mà không được trích dẫn theo quy định. Trong toàn bộ nội dung của luận văn, những điều được trình bày hoặc là của cá nhân tôi hoặc là được tổng hợp từ nhiều nguồn tài liệu khác nhau. Các số liệu, kết quả nêu trong luận văn là đúng sự thực và chưa được ai công bố trong bất kỳ công trình hay luận văn nào khác. Luận văn này chưa bao giờ được nộp để nhận bất kỳ bằng cấp nào tại các cơ sở giáo dục khác./.

Hà Nội, ngày tháng năm 2022

Học viên

Nguyễn Công Hiếu

MỤC LỤC

DANH MỤC VIẾT TẮT

Viết tắt	Tiếng Anh	Tiếng Việt
ADE	Adverse Drug Event	Công cụ phát hiện dị thường
CGI	Computer-Generated Imagery	Công nghệ mô phỏng hình ảnh bằng máy tính
CIS	Center for Internet Security	Trung Tâm An Ninh Internet
CPU	Central Processing Unit	Bộ xử lý trung tâm
DDOS	Distributed Denial of Service	Tấn công từ chối dịch vụ phân tán
DNS	Domain Name Servers	Hệ thống phân giải tên miền
DOS	Denial of Service	Tấn công từ chối dịch vụ
FIM	Federated Identity Manager	Hệ thống quản lý nhận dạng
FTP	File Transfer Protocol	Giao thức truyền tải tập tin
GNU/GPL	GNU General Public License	Giấy phép phần mềm tự do
HIDS	Host Intrusion Detection System	Hệ thống phát hiện xâm nhập host
HTTP	Hypertext Transfer Protocol	Giao thức Truyền tải Siêu Văn Bản
ICMP	Internet Control Message Protocol	Giao thức Thông điệp Điều khiển Internet
IDS	Intrusion Detection System	Hệ thống phát hiện xâm nhập
IP	Internet Protocol	Địa chỉ giao thức Internet
LAN	Local Area Network	Mạng cục bộ
ML	Machine Learning	Phương pháp học máy
NIC	Network Interface Card	Card giao tiếp mạng
NIDS	Network Intrusion Detection System	Hệ thống phát hiện xâm nhập mạng

PCI-DSS	Payment Card Industry Data Security Standard	Bộ tiêu chuẩn bảo mật dữ liệu thẻ thanh toán
RAM	Random Access Memory	Bộ nhớ truy xuất ngẫu nhiên
SEM	security event management	quản lý sự kiện bảo mật
SIEM	Security Information and Event Management	Quản lý thông tin và sự kiện bảo mật
SMB	Server Message Block	Hệ thống tệp Internet chung
SNMP	Simple Network Management Protocol	Giao thức giám sát mạng đơn giản
SSH	Secure Shell	Giao thức SSH
SSL	Secure Sockets Layer	Chứng chỉ socket bảo mật
TCP	Transmission Control Protocol	Giao thức TCP
UDP	User Datagram Protocol	Giao thức UCP
VPN	Virtual Private Network	Mạng riêng ảo
WMI	Windows Management Instrumentation	Thiết bị quản lý Windows
XSS	Cross-site scripting	Tấn công script độc hại

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG BIỂU

MỞ ĐẦU

1. Lý do chọn đề tài

Cùng với sự phát triển của mạng Internet, mạng World Wide Web toàn cầu và các dịch vụ trên nền Internet, các dạng tấn công, xâm nhập vào các hệ thống mạng, máy chủ và thiết bị đầu cuối của người dùng cũng phát triển ở mức đáng lo ngại. Các dạng tội phạm trên không gian mạng trở nên rất phổ biến và luôn đứng đầu danh sách truy nã của Cục Điều tra liên bang Mỹ (FBI) trong những năm gần đây [1]. Về mặt địa lý, Việt Nam trong những năm gần đây luôn nằm trong top 10 nước là đích bị tấn công nhiều nhất. Các dạng mã độc và tấn công, khai thác cũng tăng vọt trên các nền tảng di động và IoT. Hãng F-Secure ước tính số lượng tấn công, xâm nhập vào các thiết bị IoT tăng gấp 3 lần trong 6 tháng đầu năm 2019 [2]. Cũng trong khoảng thời gian này, số lượng dạng tấn công không liên quan đến file (fileless attacks) - một dạng tấn công tinh vi và nguy hiểm mới được phát hiện trong thời gian gần đây tăng 256%. Trong mô hình này, hệ thống mạng thường được bảo vệ bằng lớp bảo vệ thứ nhất, gồm tường lửa, các biện pháp kiểm soát truy nhập, xác thực, mã hóa,... Lớp bảo vệ hệ thống thứ hai là host và hệ thống phát hiện, ngăn chặn tấn công mạng. Các hệ thống phát hiện xâm nhập mạng (NIDS – Network-based Intrusion Detection System) được sử dụng để giám sát và bảo vệ cả mạng, hoặc một phân đoạn mạng. Các hệ thống phát hiện xâm nhập host (HIDS – Host-based Intrusion Detection System) được sử dụng để bảo vệ một máy (host), hoặc một ứng dụng, hoặc dịch vụ cụ thể.

HIDS và NIDS hiện đều đang được sử dụng rộng rãi và mỗi loại có những ưu điểm, nhược điểm riêng [1]. Ưu điểm của HIDS là có khả năng phát hiện chính xác các xâm nhập và các hành vi lạm dụng trên từng máy cụ thể do HIDS được cài đặt trên từng máy để giám sát các sự kiện xảy ra trong hệ thống. Hạn chế của HIDS là phải triển khai trên từng máy và điều này có thể phát sinh chi phí lớn cho cài đặt và bảo trì với các hệ thống mạng lớn. Ngược lại, ưu điểm của NIDS là có khả năng giám sát phát hiện các dạng xâm nhập cho cả mạng, hoặc phân đoạn mạng do nó thường được triển khai tại cổng mạng và sử dụng lưu lượng mạng gồm các gói tin

đi và đến làm nguồn dữ liệu. Hạn chế của NIDS là gặp nhiều khó khăn khi phải giám sát cổng mạng có lưu lượng lớn, hoặc lưu lượng bị mã hóa và các dạng xâm nhập trên các máy không phát sinh lưu lượng qua cổng mạng. Luận văn này chọn NIDS để xây dựng giải pháp phát hiện xâm nhập do miền bảo vệ rộng và chi phí triển khai thấp.

Trong số các hệ thống phát hiện tấn công, xâm nhập đang được sử dụng trên thực tế, Snort [2] thuộc nhóm NIDS, là hệ thống mã mở phát hiện tấn công, xâm nhập mã mở được sử dụng rộng rãi nhờ khả năng phát hiện tốt với tập luật dựng sẵn gồm khoảng 3000 luật, hỗ trợ đa nền tảng. Tuy vậy, việc quản lý các sự kiện giám sát và kết quả phát hiện còn tương đối hạn chế cho Snort chỉ hỗ trợ giao diện quản trị cơ bản. Trong khi đó, bộ công cụ quản lý log ELK [5] hỗ trợ xử lý, tìm kiếm và lưu trữ các sự kiện log khá hiệu quả với giao diện thân thiện và dễ sử dụng. Từ đó, luận văn này với đề tài "Nghiên cứu xây dựng giải pháp phát hiện xâm nhập và ứng dụng cho Học viện thanh thiếu niên Việt Nam" có mục tiêu là tập trung nghiên cứu và xây dựng giải pháp phát hiện xâm nhập dựa sử dụng hệ thống phát hiện xâm nhập mạng Snort và bộ công cụ quản lý log ELK có khả năng phát hiện xâm nhập hiệu quả với giao diện quản trị thân thiện.

2. Tổng quan vấn đề nghiên cứu

Các hệ thống phát hiện và ngăn chặn tấn công, xâm nhập (IDS/IPS) đã được quan tâm nghiên cứu, phát triển và ứng dụng khá rộng rãi trên thực tế. Nhiệm vụ chính của các hệ thống phát hiện và ngăn chặn tấn công, xâm nhập bao gồm [1]:

- Theo dõi các hành vi trên một hệ thống để nhận dạng các dấu hiệu của tấn công, xâm nhập hoặc lưu lượng mạng.

- Ghi log các hành vi phát hiện các tấn công, xâm nhập cho phân tích bổ sung sau này.

- Dừng hoặc ngăn chặn các hành vi tấn công, xâm nhập (với IPS).

- Khi phát hiện các hành vi tấn công xâm nhập sẽ gửi thông báo, cảnh báo cho người quản trị để được phát hiện kịp thời.

Có nhiều hệ thống phát hiện và ngăn chặn tấn công, xâm nhập mã mở và thương mại đã được ứng dụng trên thực tế, chẳng hạn như Snort, Suricata, Zeek, Bro, IBM Qradar,… Những hệ thống này có tính năng, miền bảo vệ và chi phí cài đặt, vận hành rất khác nhau. Các hệ thống mã mở, miễn phí, như Snort, Suricata thường có khả năng phát hiện tốt, nhưng khó cài đặt và bảo trì, giao diện quản trị khá hạn chế. Ngược lại, các hệ thống thương mại như IBM Qradar có tính năng phong phú, giao diện quản trị mạnh và thân thiện, nhưng chi phí cài đặt và vận hành lớn. Do vậy, việc khảo sát, đánh giá để lựa chọn hệ thống phù hợp với yêu cầu của hệ thống mạng cụ thể là cần thiết.

Luận văn này trước hết tập trung nghiên cứu, khảo sát một số hệ thống phát hiện xâm nhập hiện có trên thị trường. Trên cơ sở kết quả khảo sát sẽ lựa chọn hệ thống phát hiện xâm nhập thích hợp kết hợp với khả năng quản lý log hiệu quả của hệ thống ELK cho triển khai nhằm tăng cường an toàn cho hệ thống mạng của Học viện thanh thiếu niên Việt Nam.

3. Mục đích nghiên cứu

Luận văn nghiên cứu, khảo sát các hệ thống phát hiện xâm nhập, hệ thống quản lý log và triển khai ứng dụng một giải pháp phát hiện xâm nhập phù hợp nhằm tăng cường an toàn cho hệ thống mạng của Học viện thanh thiếu niên Việt Nam.

4. Đối tượng và phạm vi nghiên cứu

- Đối tượng nghiên cứu là các hình thức tấn công, xâm nhập và các phương pháp, hệ thống phát hiện xâm nhập.
- Phạm vi nghiên cứu: Giới hạn trong mạng LAN gồm các máy chủ và các máy trạm chạy hệ điều hành Microsoft Windows hoặc Ubuntu Linux.

5. Phương pháp nghiên cứu

Luận văn sử dụng kết hợp các phương pháp nghiên cứu sau:

- Nghiên cứu lý thuyết: Nghiên cứu các phương pháp và kỹ thuật phát hiện xâm nhập. Khảo sát một số hệ thống phát hiện xâm nhập hiện có.

- Thực nghiệm: Xây dựng và triển khai thử nghiệm mô hình giải pháp phát hiện xâm nhập dựa trên Snort và ELK cho hệ thống mạng Học viện thanh thiếu niên Việt Nam.

CHƯƠNG 1. TỔNG QUAN VỀ PHÁT HIỆN XÂM NHẬP

1.1. KHÁI QUÁT VỀ TẤN CÔNG, XÂM NHẬP

1.1.1. Đối tượng bị tấn công, xâm nhập

Có thể là bất cứ ai có thông tin bí mật được lưu trữ trên môi trường mạng. Các chủ thể này có thể là doanh nghiệp tư nhân, tổ chức chính phủ hoặc phi chính phủ. Cũng có thể là các cá nhân trong thông tin riêng tư của họ. Tuy nhiên, đối tượng phổ biến nhất của các vụ tấn công mạng là doanh nghiệp. Khi đe dọa đến các thông tin nội bộ ảnh hưởng đến hiệu quả hoạt động của doanh nghiệp. Bởi mục tiêu chính của hacker là lợi nhuận và tìm kiếm các lợi ích vật chất.

Hacker có thể tấn công thông qua mạng nội bộ (như con người, máy tính, thiết bị…).

Đối với yếu tố con người, hacker có thể tiếp cận thông qua:

- Phần mềm độc hại
- Lỗi phần mềm hoặc phần cứng
- Kẻ tấn công ở bên trong
- Mất trộm các thiết bị
- Kẻ tấn công ở bên ngoài
- Tai họa thiên nhiên
- Gián điệp công nghiệp
- Tấn công phá hoại

1.1.2. Mục đích của tấn công, xâm nhập

Thường những cuộc tấn công, xâm nhập mà các kẻ xấu hay hacker thực hiện nhằm mục đích kiếm tiền, tìm kiếm, phát triển công cụ tấn công để đem lại lợi ích cho cá nhân hay tổ chức của họ, Vì trong một hệ thống mạng máy tính luôn tồn tại các điểm yếu (Weakness), các hacker có thể dựa vào các khiếm khuyết này để tấn công và xâm nhập vào hệ thống.

Số lượng các cuộc tấn công, xâm nhập internet đang tăng dần đều theo từng năm. Trong quá khứ đã có những cuộc tấn công mạng tầm vĩ mô gây ảnh hưởng tê

liệt hàng triệu hệ thống trên toàn thế giới, bao gồm cả những công ty hàng đầu từ thông tin cá nhân, bí mật doanh nghiệp, đòi tiền chuộc, tống tiền v.v…

Ở Việt Nam hiện nay, trong bối cảnh dịch bệnh vẫn còn phức tạp, nhiều hệ thống thông tin quan trọng vẫn trở thành mục tiêu tấn công của tin tặc. Những nhóm tội phạm này lợi dụng tình hình dịch diễn biến phức tạp tấn công mạng vào các cơ quan chức năng bằng cách gửi tài liệu giả mạo để phát tán mã độc hay tấn công có chủ đích.

Theo Cục an ninh mạng (Bộ Công an) cho thấy Việt Nam trong thời gian vừa qua phải chịu nhiều đợt tấn công nhằm vào các hệ thống thông tin của quốc gia, xuyên tạc, phát tán thông tin sai sự thật để chiếm đoạt tài sản, lừa đảo. Các thiết bị dễ bị tấn công nhất thường là điện thoại di động, IoT.

1.2. GIỚI THIỆU MỘT SỐ DẠNG TẤN CÔNG, XÂM NHẬP THƯỜNG GẶP

Có nhiều kiểu tấn công, xâm nhập mạng khác nhau và được phân thành một số loại chính sau: tấn công chiếm quyền "root", kiểu thăm dò, tấn công từ chối dịch vụ, tấn công điều khiển từ xa. Tấn công, xâm nhập mạng gây ra hậu quả vô cùng nặng nề cho các tổ chức, doanh nghiệp là nạn nhân của chúng. Dưới đây là sáu hình thức tấn công mạng phổ biến nhất hiện nay

1.2.1. Hình thức tấn công mạng bằng phần mềm độc hại (MalwareAttack)

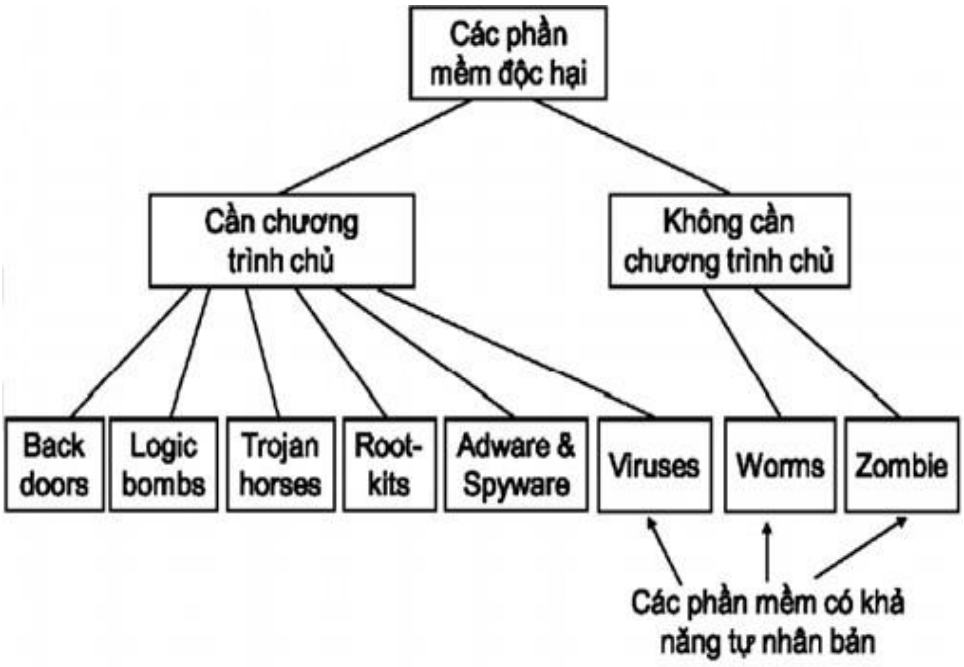
Tấn công Malware là một trong những hình thức tấn công qua mạng phổ biến nhất hiện nay. Malware bao gồm:

- Spyware (phần mềm gián điệp)
- Ransomware (mã độc tống tiền)
- Virus
- Worm (phần mềm độc hại lây lan với tốc độ nhanh)

Thông thường, Hacker sẽ tiến hành tấn công người dùng thông qua các lỗ hổng bảo mật. Hoặc lừa người dùng Click vào một đường Link hoặc Email

(Phishing) để cài phần mềm độc hại tự động vào máy tính. Một khi được cài đặt thành công, Malware sẽ gây ra những hậu quả nghiêm trọng:

- Chặn các truy cập vào hệ thống mạng và dữ liệu quan trọng (Ransomware).
- Cài đặt thêm phần mềm độc hại khác vào máy tính người dùng.
- Đánh cắp dữ liệu (Spyware).
- Phá hoại phần cứng, phần mềm, làm hệ thống bị tê liệt, không thể hoạt động.



Hình 1. Các dạng phần mềm độc hại

1.2.2. Hình thức tấn công giả mạo (Phishing Attack)

Phishing Attack là tấn công mà trong đó tin tặc giả mạo thành một cá nhân hoặc tổ chức uy tín để lấy lòng tin của người dùng. Hacker sẽ giả mạo là ví điện tử, ngân hàng, trang giao dịch trực tuyến hoặc các công ty thẻ tín dụng để lừa người dùng chia sẻ các thông tin cá nhân như: mật khẩu giao dịch, thẻ tín dụng, tài khoản & mật khẩu đăng nhập và các thông tin quan trọng khác. Đây là thủ đoạn tấn công thường dùng và thường là hoạt động mở đầu cho chuỗi các hành động tiếp theo của tin tặc, từ đó, chúng đánh cắp các dữ liệu nhạy cảm như tài khoản ngân hàng, thẻ tín dụng…

Các cuộc tấn công giả mạo thường được thực hiện qua email. Người dùng sẽ nhận được email giả mạo một tổ chức/ cá nhân uy tín với thông điệp vô cùng khẩn thiết. Thông điệp này yêu cầu người dùng click vào đường link tin tặc tạo ra. Khi click vào, người dùng sẽ được chuyển đến một website giả mạo có giao diện y như website thật và được yêu cầu đăng nhập. Khi đó, tin tặc sẽ có được thông tin đăng nhập và dữ liệu nhạy cảm khác của người dùng.

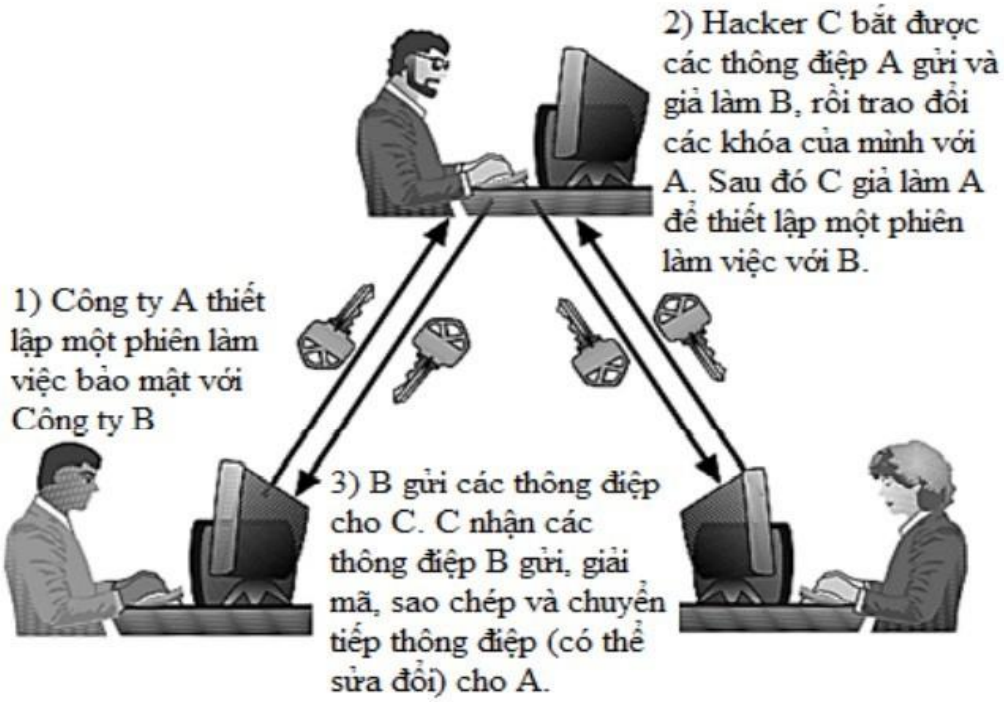
Tấn công Phishing/Social Engineering diễn ra rất phổ biến, rất nguy hiểm do khó bị phát hiện, khó phòng tránh hữu hiệu bằng biện pháp kỹ thuật; chúng tấn công vào sự chủ quan, bất cẩn, hiểu biết hạn chế hay tình trạng thiếu kiểm soát của người dùng, do đó, tin tặc hoàn toàn có thể thành công ngay cả với những hệ thống bảo mật cao.

Ngoài ra còn có các hình thức tấn công giả mạo như SmiShing (cũng tương tự như lừa đảo qua email nhưng nó lừa người dùng qua tin nhắn văn bản, nhiều người nhận ra được email lừa đảo tuy nhiên ít người nghi ngờ về tin nhắn SM, điều này làm tăng khả năng lừa đảo thành công) hay Spear Phishing (nó nhắm đến một cá nhân cụ thể, bao gồm một chuỗi email được thiết kế để thu hút họ hành động) hoặc Whaling (cũng nhắm vào một cá nhân hoặc tổ chức, đó thường là người có nhiều thứ để mất, chẳng hạn như CEO, người nổi tiếng, nhân vật chính trị hoặc các gia đình giàu có).

1.2.3. Hình thức tấn công trung gian (Man in the middle attack)

Tấn công trung gian là hình thức tin tặc xen vào giữa phiên giao dịch hay giao tiếp giữa hai đối tượng. Khi đã xâm nhập thành công, chúng có thể theo dõi được mọi hành vi của người dùng. Tệ hơn, chúng có thể đánh cắp được toàn bộ dữ liệu trong phiên giao dịch đó. Tấn công trung gian dễ xảy ra khi nạn nhân truy cập vào một mạng wifi không an toàn. Hình 1.3 minh họa mô hình tấn công kiểu người đứng giữa trong một phiên truyền file ở dạng rõ (plaintext) sử dụng giao thức FTP giữa máy khách (Client) và máy chủ (Server).

Hình 2: Mô hình tấn công kiểu người ở giữa



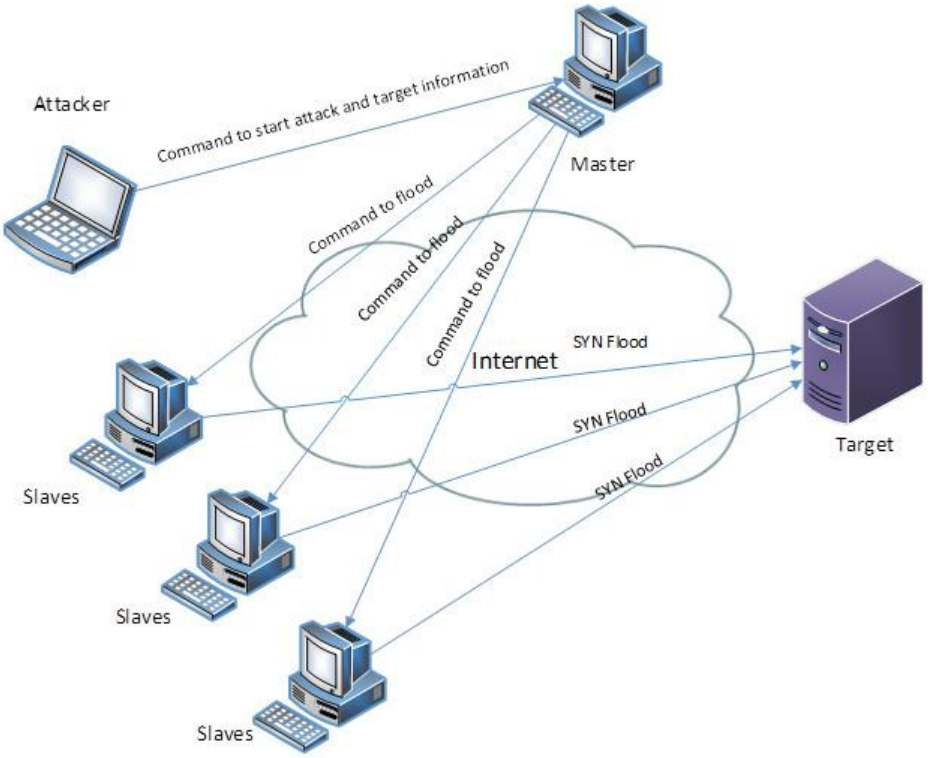
Hình 3: Kịch bản tấn công kiểu người ở giữa

1.2.4. Hình thức tấn công từ chối dịch vụ (DoS & DDoS)

a. Tấn công từ chối dịch vụ (DoS)

Dos – Denial Of Service được dịch ra là từ chối dịch vụ, đây là một hình thức tấn công khá phổ biến khiến cho máy tính mục tiêu không thể xử lý kịp các tác vụ dẫn đến quả tải. Hay nói cách khác mục đích của Dos là làm sập một máy chủ hoặc mạng khiến người dùng không thể truy cập vào máy chủ/mạng đó. Hacker thực hiện điều này bằng cách tuồn ồ ạt traffic hoặc gửi tin có thể kích hoạt sự cố đến hệ thống mạng, máy chủ,…

b. Tấn công từ chối dịch vụ phân tán (DDoS)



Hình 4: Tấn công từ chối dịch vụ phân tán

Ddos – Distributed Denial Of Service được dịch là từ chối dịch vụ phân tán, hình thức này là một dạng tấn công nỡ lực làm sập một dịch vụ trực tuyến bằng cách làm tràn ngập với traffic từ nhiều nguồn. Ddos khiến cho người bị tấn công không thể sử dụng một dịch vụ nào đó, không thể kết nối với dịch vụ internet nào đó hoặc làm ngưng hoạt động của một chiếc máy tính, một mang LAN nội bộ hoặc thậm chí cả một hệ thống mạng.

Tấn công Ddos mạnh hơn Dos ở chỗ là hình thức này có thể phân tán được từ nhiều dải IP khác nhau, khiến người bị tấn công khó phát hiện để ngăn chặn. Kẻ tấn công có thể sử dụng máy tính của bạn để tấn công vào các máy tính khác bằng cách lợi dụng những lỗ hổng của bảo mật để giành lấy quyền điều khiển máy tính của bạn.

Ba loại tấn công cơ bản của Ddos:

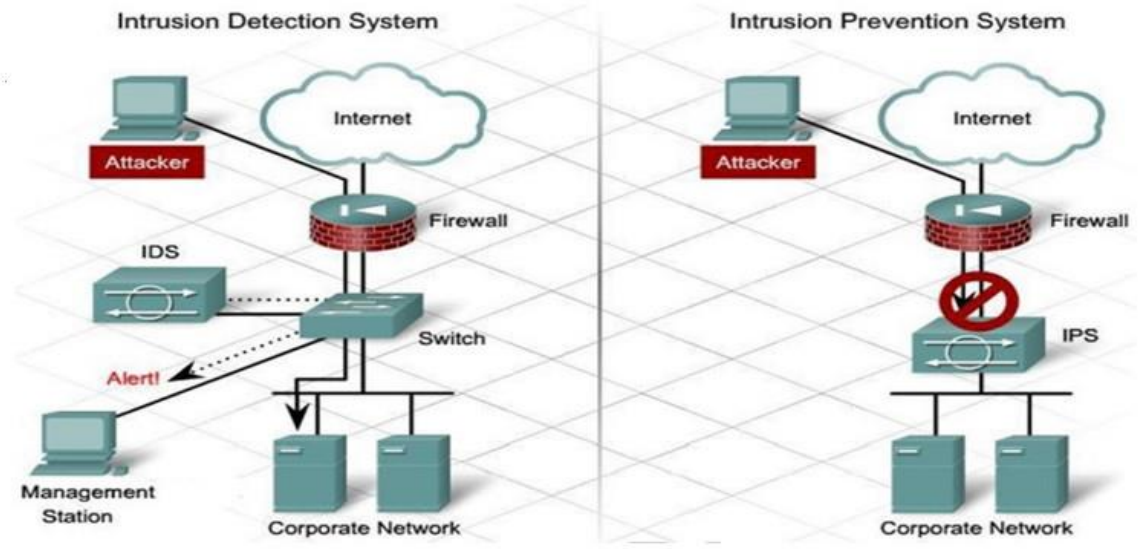
- Volume-based: Lưu lượng truy cập cao để làm tràn băng thông
- Protocol: Khai thác các tài nguyên máy chủ

- Application: Tập trung vào các ứng dụng web

1.3. PHÁT HIỆN XÂM NHẬP

1.3.1. Khái quát về phát hiện xâm nhập

Là quá trình giám sát các sự kiện xảy ra trong mạng máy tính hoặc một hệ thống máy tính và phân tích để tìm ra các dấu hiệu của sự cố. Các sự cố ở đây có thể là các vi phạm hoặc các mối đe dọa sắp xảy ra vi phạm chính sách bảo mật, chính sách sử dụng được chấp nhận hoặc các phương pháp bảo mật tiêu chuẩn. Ví dụ như một đối tượng tấn công truy cập trái phép vào hệ thống hoặc người dùng cố gắng giành thêm các đặc quyền mà họ không được phép. Hệ thống phát hiện xâm nhập (IDS) là phần mềm hoặc thiết bị phần cứng tự động hóa quá trình phát hiện xâm nhập.



Hình 5: Vị trí các hệ thống IDS và IPS trong sơ đồ mạng

1.3.2. Phân loại các hệ thống phát hiện xâm nhập

a. Hệ thống phát hiện xâm nhập là gì?

Hệ thống phòng chống xâm nhập (Intrusion prevention system, viết tắt là IPS) là một số biện pháp an ninh mạng quan trọng nhất mà mạng có thể có. IPS được biết đến như một hệ thống kiểm soát, vì nó không chỉ phát hiện các mối đe dọa tiềm ẩn đối với hệ thống mạng và cơ sở hạ tầng của nó, mà còn tìm cách chủ động chặn

bất kỳ kết nối nào có thể là mối đe dọa. Điều này khác với các biện pháp bảo vệ thụ động như hệ thống phát hiện xâm nhập.

Công nghệ IPS là gì?

Hệ thống phòng chống xâm nhập luôn luôn giám sát bất thường của mạng, đặc biệt là ở các gói riêng lẻ, để tìm kiếm bất kỳ cuộc tấn công nguy hiểm nào có thể xảy ra. Nó thu thập thông tin về các gói tin này và báo cáo cho quản trị viên hệ thống, nhưng nó cũng thực hiện những động thái phòng ngừa của riêng mình. Nếu phát hiện bất thường hoặc loại tấn công khác thì IPS sẽ chặn các gói đó truy cập vào mạng.

IPS có thể thực hiện đóng lỗ hổng bảo mật của hệ thống có thể bị khai thác liên tục, đóng các điểm truy cập vào mạng, cũng như cấu hình tường lửa thứ cấp để phát hiện những loại tấn công này trong tương lai, bổ sung các lớp bảo mật cho hệ thống phòng thủ.

Hệ thống phòng chống xâm nhập liên tục giám sát lưu lượng mạng

IPS có thể ngăn chặn những loại tấn công nào?

Các hệ thống phòng chống xâm nhập có thể bảo vệ tìm kiếm và chống lại nhiều loại tấn công nguy hiểm. Chúng có khả năng phát hiện và chặn các cuộc tấn công tấn công từ chối dịch vụ phân tán (DDoS), từ chối dịch vụ (DoS), virus máy tính, worm, bộ công cụ exploit và những loại phần mềm độc hại khác.

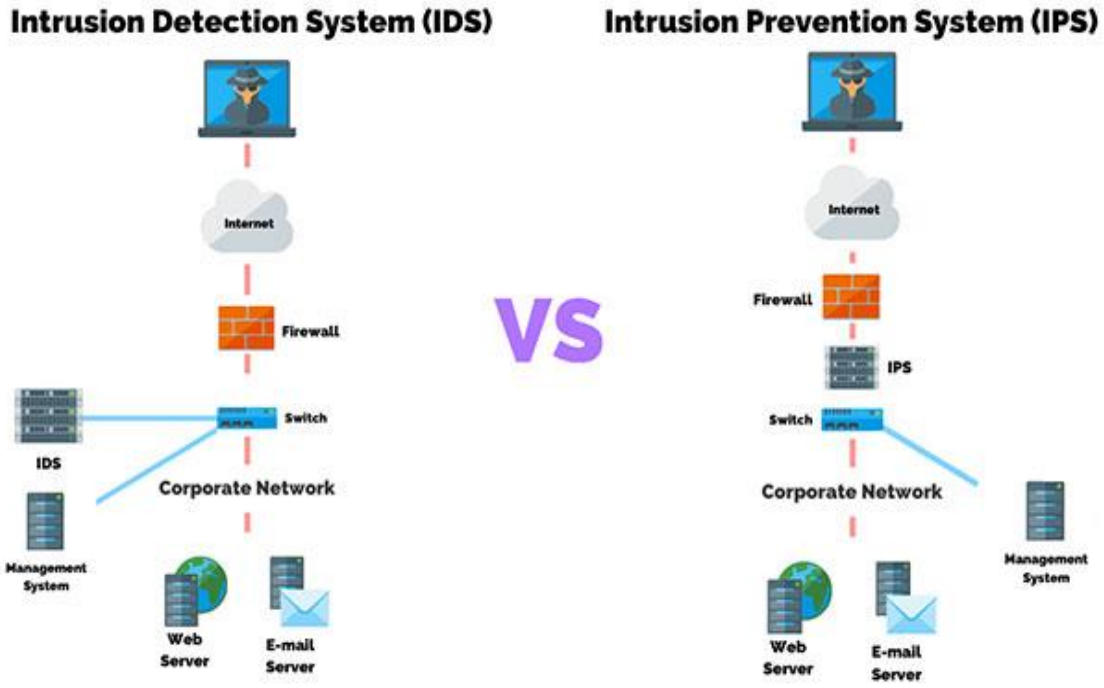
IPS sẽ làm gì nếu nó phát hiện ra một cuộc tấn công?

Một hệ thống ngăn chặn xâm nhập có thể phát hiện nhiều cuộc tấn công khác nhau bằng cách phân tích các gói và tìm kiếm những chữ ký phần mềm độc hại cụ thể, mặc dù nó cũng có thể tận dụng khả năng theo dõi hành vi để tìm kiếm hoạt động bất thường trên mạng, cũng như giám sát bất kỳ giao thức và chính sách bảo mật cấp quản trị nào, cũng như liệu chúng có bị vi phạm hay không.

Nếu bất kỳ phương pháp nào trong số này phát hiện ra một cuộc tấn công tiềm ẩn, IPS có thể ngay lập tức chấm dứt kết nối đến. Địa chỉ IP vi phạm sau đó có thể bị chặn nếu IPS được cấu hình để làm như vậy hoặc người dùng liên kết với nó bị cấm truy cập lại vào mạng và bất kỳ tài nguyên nào được kết nối.

IPS cũng có thể thay đổi cài đặt tường lửa cục bộ để phát hiện lại các cuộc tấn công như vậy và thậm chí có thể loại bỏ mọi tàn tích của cuộc tấn công bằng cách loại bỏ những header bị ảnh hưởng bởi phần mềm độc hại, file đính kèm bị nhiễm virus, cũng như những liên kết độc hại khỏi file và email server.

IDS và IPS có gì khác nhau?



Hình 6: So sánh IDS và IPS

Hệ thống phát hiện xâm nhập (IDS) và hệ thống ngăn chặn xâm nhập (IPS) đều có thể liên quan đến bảo mật

Hệ thống phát hiện xâm nhập (IDS) và hệ thống ngăn chặn xâm nhập (IPS) đều có thể liên quan đến bảo mật, nhưng chúng có các mục tiêu và phương tiện hoàn toàn khác nhau.

Có nhiều loại IDS cũng như IPS và tất cả chúng đều hoạt động hơi khác nhau một chút. Đối với IDS, có các hệ thống phát hiện xâm nhập mạng (NIDS), đặt tại các điểm chiến lược trong mạng để phát hiện những cuộc tấn công tiềm ẩn khi chúng đang diễn ra trong mạng. HIDS hay hệ thống phát hiện xâm nhập máy chủ chạy trên các hệ thống và thiết bị riêng lẻ, chỉ giám sát hoạt động trên mạng đi và đến hệ thống cụ thể đó.

Trong cả hai trường hợp, IDS phát hiện ra một cuộc tấn công tiềm ẩn sẽ thông báo cho quản trị viên hệ thống.

Hệ thống IPS sẽ đóng một vai trò tương tự như IDS – và có thể được sử dụng kết hợp để giám sát mạng tốt hơn – nhưng sẽ đóng vai trò tích cực hơn trong việc bảo vệ mạng. IPS cũng sẽ thông báo cho quản trị viên nếu phát hiện các cuộc tấn công, nhưng chúng cũng sẽ thực hiện những hành động trừng phạt đối với bất kỳ hệ thống, tài khoản cá nhân hoặc lỗ hổng tường lửa nào để đảm bảo rằng cuộc tấn công đã bị chặn và mọi file liên quan bị xóa khỏi mạng.

Như tên gọi cho thấy, các hệ thống phát hiện xâm nhập được thiết kế để cho bạn biết liệu có khả năng và khi nào một cuộc tấn công xảy ra, để bạn có thể xử lý vấn đề theo cách thủ công. Hệ thống ngăn chặn xâm nhập được thiết kế để chủ động bảo vệ hệ thống của bạn khỏi các cuộc tấn công và ngăn chặn những cuộc tấn công trong tương lai thông qua việc điều chỉnh các thông số mạng.

Chức năng chính của IDS

- IDS cho phép bạn tăng cường bảo mật cho các thiết bị mạng và dữ liệu mạng có giá trị bằng cách giám sát lưu lượng mạng đáng ngờ và thông báo đến bạn. Mạng của bạn cần bảo mật mạnh mẽ để bảo vệ thông tin hiện có và truyền dữ liệu mạng bên trong và bên ngoài. Các cuộc tấn công mạng ngày càng tinh vi và thường xuyên, vì vậy điều quan trọng là phải có một hệ thống phát hiện xâm nhập toàn diện và hiệu quả.
- Hệ thống phát hiện xâm nhập giúp tổ chức dữ liệu mạng quan trọng. Mạng của bạn tạo ra hàng tấn thông tin mỗi ngày thông qua các hoạt động thường xuyên và hệ thống phát hiện xâm nhập có thể giúp bạn phân biệt hoạt động nào quan trọng hơn. Một hệ thống phát hiện xâm nhập có thể giúp bạn không phải tìm kiếm thông tin quan trọng của hàng nghìn nhật ký hệ thống. Điều này có thể giúp bạn tiết kiệm thời gian, giảm công sức thủ công và giảm thiểu lỗi của con người khi phát hiện xâm nhập.
- Các hệ thống ngăn chặn xâm nhập được xây dựng để phát hiện, sắp xếp và cảnh báo chuyên sâu về lưu lượng mạng vào/ra, xác định chính xác thông tin

quan trọng nhất. Bằng cách lọc thông qua lưu lượng mạng, hệ thống phát hiện xâm nhập có thể giúp xác định mức độ tuân thủ của mạng và các thiết bị của nó.

- IDS được tạo ra để tối ưu hóa việc phát hiện và ngăn chặn xâm nhập bằng cách lọc qua luồng lưu lượng. Điều này có thể giúp bạn tiết kiệm thời gian, năng lượng và tài nguyên trong khi phát hiện hoạt động đáng ngờ trước khi nó biến thành một mối đe dọa toàn diện. IDS cũng cung cấp khả năng hiển thị cao hơn đối với lưu lượng mạng, có thể giúp bạn chống lại các hoạt động độc hại, xác định trạng thái tuân thủ và cải thiện hiệu suất mạng tổng thể. IDS của bạn càng nắm bắt và hiểu được hoạt động độc hại trên mạng của bạn, thì IDS càng có thể thích ứng với các cuộc tấn công ngày càng tinh vi.

Hoạt động của IDS

Sau khi thu thập dữ liệu, một IDS được thiết kế để quan sát lưu lượng mạng và match với các mẫu lưu lượngvới các cuộc tấn công đã biết. Thông qua phương pháp này (đôi khi được gọi là tương quan mẫu hay Pattern Correlation), một hệ thống ngăn chặn xâm nhập có thể xác định xem hoạt động bất thường có phải là một cuộc tấn công mạng hay không.

Một khi hoạt động đáng ngờ hoặc độc hại được phát hiện, hệ thống phát hiện xâm nhập sẽ gửi báo động đến các kỹ thuật viên hoặc quản trị viên CNTT được chỉ định. Báo động IDS cho phép bạn nhanh chóng bắt đầu khắc phục sự cố và xác định các nguồn gốc của vấn đề hoặc phát hiện và ngăn chặn các tác nhân gây hại trong quá trình theo dõi của chúng.

Các hệ thống phát hiện xâm nhập chủ yếu sử dụng hai phương pháp phát hiện xâm nhập chính: phát hiện xâm nhập dựa trên chữ ký (signature-based intrusion detection) và phát hiện xâm nhập dựa trên sự bất thường (Anomaly-based intrusion detection).

Phát hiện xâm nhập dựa trên chữ ký được thiết kế để phát hiện các mối đe dọa có thể xảy ra bằng cách so sánh lưu lượng mạng nhất định và dữ liệu nhật ký với các mẫu tấn công hiện có. Những mẫu này được gọi là chuỗi - sequences (do đó có

tên) và có thể bao gồm byte sequences, được gọi là chuỗi lệnh độc hại. Tính năng phát hiện dựa trên chữ ký cho phép bạn phát hiện và xác định chính xác các cuộc tấn công đã biết.

Phát hiện xâm nhập dựa trên sự bất thường thì ngược lại — nó được thiết kế để xác định chính xác các cuộc tấn công không xác định, chẳng hạn như phần mềm độc hại mới và thích ứng với chúng một cách nhanh chóng bằng cách sử dụng máy học. Các kỹ thuật máy học cho phép IDS tạo ra các đường cơ sở của hoạt động đáng tin cậy (được gọi là mô hình tin cậy). Sau đó so sánh hành vi mới với các mô hình tin cậy đã được xác minh, nhưng cảnh báo giả vẫn có thể xảy ra khi sử dụng IDS dựa trên bất thường, vì lưu lượng mạng hợp pháp chưa xác định trước đây có thể bị xác định sai là hoạt động độc hại.

Hybrid intrusion detection systems hay Hệ thống phát hiện xâm nhập, là sự kết hợp sử dụng tính năng phát hiện xâm nhập dựa trên chữ ký và dựa trên sự bất thường để tăng phạm vi hệ thống ngăn chặn xâm nhập của bạn. Điều này cho phép bạn xác định càng nhiều mối đe dọa càng tốt. Một hệ thống phát hiện xâm nhập toàn diện (IDS) có thể hiểu các kỹ thuật trốn tránh mà tội phạm mạng sử dụng để đánh lừa hệ thống ngăn chặn xâm nhập nghĩ rằng không có một cuộc tấn công nào đang diễn ra.

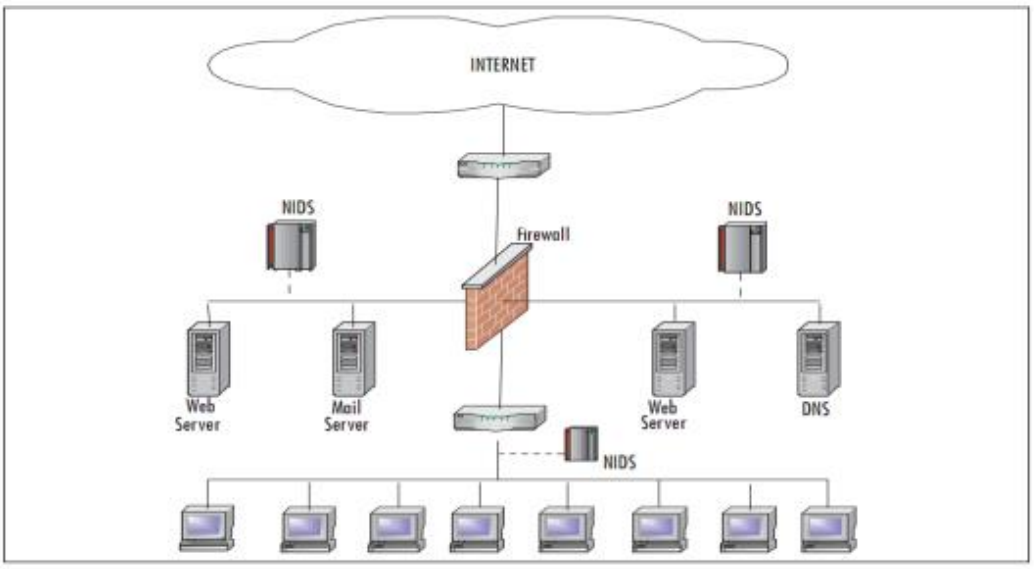
b. Hệ thống phát hiện xâm nhập bao gồm các loại nào?

Có nhiều loại IDS khác nhau, mỗi loại có một chức năng và nhiệm vụ riêng tuy nhiên hai loại phổ biến nhất là NIDS và HIDS

- **NIDS (Network-Based IDS)**

Hệ thống phát hiện xâm nhập mạng (NIDS) được thiết lập tại một điểm được lên kế hoạch trong mạng để kiểm tra lưu lượng truy cập từ tất cả các thiết bị trên mạng. NIDS có thể là các hệ thống dựa trên phần cứng hoặc phần mềm, tùy thuộc vào nhà sản xuất hệ thống, có thể gắn vào các phương tiện mạng khác nhau như Ethernet, FDDI và các phương tiện khác. Thông thường, NIDS có hai giao diện mạng, một được sử dụng để nghe các cuộc trò chuyện mạng ở chế độ hỗn hợp và một được sử dụng để kiểm soát và báo cáo. Trong khi có nhiều nhà cung cấp NIDS,

tất cả các hệ thống đều có xu hướng hoạt động theo một trong hai cách; NIDS là hệ thống dựa trên chữ ký hoặc dựa trên sự bất thường. Cả hai đều là cơ chế phân tách lưu lượng truy cập lành tính khỏi những người anh em độc hại của nó. Các vấn đề tiềm ẩn với NIDS bao gồm quá tải dữ liệu mạng tốc độ cao, khó điều chỉnh, mã hóa và thời gian trễ phát triển chữ ký.



Hình 7: Mô hình NIDS

Ưu điểm của NIDS:

- Quản lý được cả một network segment (gồm nhiều host)
- Trong suốt với người sử dụng lẫn kẻ tấn công
- Bảo trì và cài đặt đơn giản không mất nhiều thời gian, không gây ảnh hưởng gì tới mạng
- Tránh DOS ảnh hưởng tới một host nào đó
- Có khả năng xác định lỗi ở tầng Network
- Độc lập với OS (Operating System)

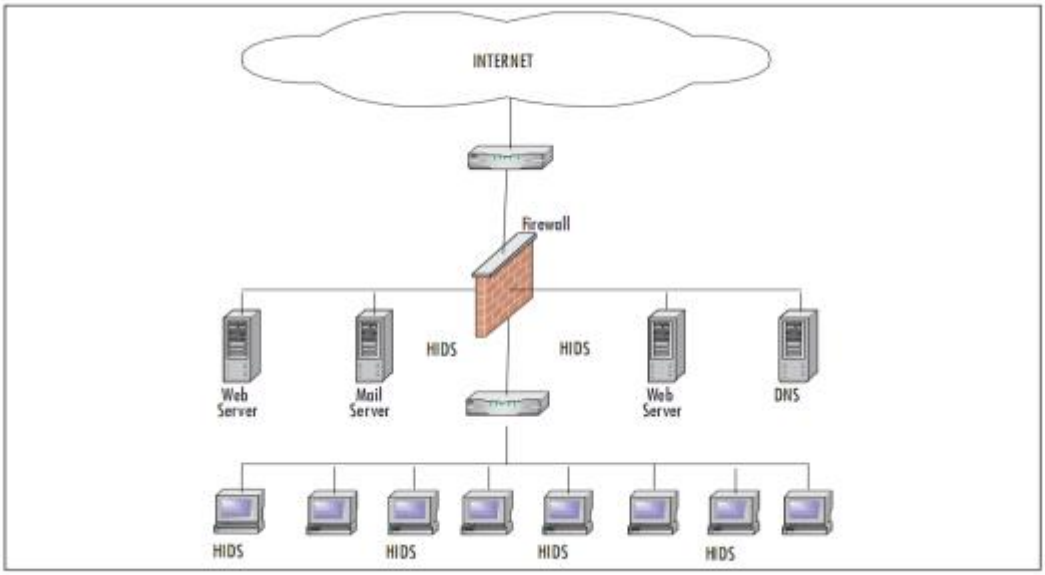
Nhược điểm của NIDS:

- Có thể xảy ra trường hợp báo động giả
- Không phân tích được các dữ liệu đã mã hóa (VD: SSL, SSH, IPSec...)
- NIDS yêu cầu phải được cập nhật signature mới nhất để được an toàn

- Phát đi thông báo chậm trễ giữa thời điểm bị tấn công với thời điểm phát báo động. Dẫn đến hệ thống có thể đã bị tổn hại.
- Không thông báo việc tấn công có thành công hay thất bại
- Hạn chế lớn nhất là giới hạn băng thông. Bộ dò mạng phải nhận hết các lưu lượng mạng sau đó sắp xếp lại những lưu lượng đó rồi phân tích chúng. Khi tốc độ mạng tăng thì khả năng của đầu dò cũng phải tỉ lệ thuận.

➢ **HIDS (Hot-Based IDS)**

HIDS được cài đặt cục bộ trên một máy tính và do đó linh hoạt hơn nhiều so với NIDS. HIDS có thể được cài đặt trên nhiều loại máy tính cụ thể như máy chủ, máy trạm, máy tính xách tay. HIDS cho phép bạn làm việc linh hoạt trong các phân đoạn mạng mà NIDS không thể. Lưu lượng được gửi đến máy chủ được phân tích và chuyển tiếp đến máy chủ lưu lượng nếu nó không có khả năng độc hại. Trong khi NIDS thậm chí còn tập trung vào mạng lớn chứa các host này. HIDS đặc biệt hướng tới các nền tảng ứng dụng và hướng nhiều đến thị trường Windows trong thế giới máy tính, mặc dù có những sản phẩm hoạt động hiệu quả. Trên nền tảng ứng dụng UNIX và nhiều hệ điều hành khác.



Hình 8: Mô hình HIDS

Công việc của HIDS là theo dõi các thay đổi trong hệ thống bao gồm: tiến trình, đầu vào, mức sử dụng CPU, tình trạng RAM, sức khỏe hệ thống. Các dữ liệu này khi thay đổi bất thường hoặc có những khác biệt khả nghi sẽ gây ra báo động.

Ưu điểm của HIDS:

- Có thể xác định người dùng (user) liên quan tới sự kiện (event)
- NIDS không có khả năng phát hiện các cuộc tấn công diễn ra trên 1 máy còn HIDS thì có thể
- Có thể phân tích các dữ liệu mã hóa
- Cung cấp các thông tin về máy chủ (host) trong quá trình tấn công máy chủ này.

Nhược điểm của HIDS:

- Thông tin của HIDS không đáng tin cậy khi cuộc tấn công vào máy chủ (host) này thành công.
- Nếu hệ điều hành (OS) bị sập do một cuộc tấn công, đồng thời HIDS cũng sẽ thất bại
- HIDS phải được thiết lập cấu hình trên từng máy chủ (host) cần giám sát
- HIDS không thể phát hiện các cuộc dò quét mạng (Nmap, Netcat ...)
- HIDS cần sử dụng tài nguyên trên máy chủ (host) để hoạt động
- Khi bị tấn công từ chối dịch vụ DOS, HIDS có thể không hiệu quả
- Hầu hết HIDS được phát triển trên hệ điều hành Window. Tuy nhiên một số HIDS cũng chạy trên Linux hoặc Unix

Vì HIDS phải được cài đặt trên các máy chủ nên quản trị viên gặp khó khăn trong việc phiên bản, bảo trì và cấu hình phần mềm, việc này tốn nhiều thời gian và phức tạp. Thông thường hệ thống chỉ phân tích lưu lượng truy cập trên máy chủ, nhưng lưu lượng truy cập vào một nhóm máy chủ hoặc các hành động thăm dò như quét cổng không hoạt động. Nếu máy chủ bị xâm nhập, tin tặc có thể vô hiệu hóa HIDS trên máy chủ. Sau đó, HIDS bị vô hiệu hóa. Do đó, HIDS phải cung cấp chức năng cảnh báo đầy đủ. Điều này có thể trở thành một vấn đề trong môi trường hỗn

hợp khi HIDS cần phải tương thích với nhiều hệ điều hành, vì vậy việc lựa chọn HIDS cũng là một vấn đề quan trọng.

1.3.3. Các kỹ thuật phát hiện xâm nhập

a. Statistical anomaly detection:

Liên quan đến việc thu thập dữ liệu liên quan đến hành vi của người dùng hợp pháp trong một khoảng thời gian. Sau đó, các bài kiểm tra thống kê được áp dụng cho hành vi được quan sát để xác định với mức độ tin cậy cao xem hành vi đó có phải là hành vi của người dùng hợp pháp hay không.

- Phát hiện ngưỡng: Cách tiếp cận này liên quan đến việc xác định các ngưỡng, tùy thuộc vào người dùng, cho tần suất xuất hiện của các sự kiện khác nhau.
- Dựa trên hồ sơ: Một hồ sơ của hoạt động của từng người dùng được phát triển và sử dụng để phát hiện những thay đổi trong hành vi của các tài khoản cá nhân.

b. Rule-based detection:

Liên quan đến nỗ lực xác định một tập hợp các quy tắc có thể được sử dụng để quyết định rằng một hành vi nhất định là của một kẻ xâm nhập.

- Phát hiện bất thường: Các quy tắc được phát triển để phát hiện sự sai lệch so với các mẫu sử dụng trước đó.
- Thâm nhập nhận dạng: Một phương pháp tiếp cận hệ thống chuyên gia tìm kiếm hành vi đáng ngờ.

Tóm lại, các phương pháp thống kê cố gắng xác định hành vi bình thường hoặc được mong đợi, trong khi các phương pháp tiếp cận dựa trên quy tắc cố gắng xác định hành vi phù hợp.

Về mặt các loại kẻ tấn công được liệt kê trước đó, phát hiện bất thường thống kê có hiệu quả chống lại những kẻ giả mạo, những kẻ không có khả năng bắt chước các mô hình hành vi của các tài khoản mà họ thích hợp. Mặt khác, những kỹ thuật như vậy có thể không đối phó được với những kẻ thất bại. Đối với các cuộc tấn công như vậy, các phương pháp tiếp cận dựa trên quy tắc có thể nhận ra các sự kiện

và trình tự, trong ngữ cảnh, cho thấy sự thâm nhập. Trong thực tế, một hệ thống có thể thể hiện sự kết hợp của cả hai cách tiếp cận để có hiệu quả chống lại một loạt các cuộc tấn công.

Kết luận chương

Như vậy, chương 1 đã trình bày tổng quan về tấn công, xâm nhập, các dạng tấn công xâm nhập thường gặp; đồng thời chương 1 cũng đã khái quát về phát hiện xâm nhập cũng như các kỹ thuật phát hiện xâm nhập qua đó ta có cái nhìn rõ hơn về tấn công, xâm nhập.

CHƯƠNG 2. CÁC HỆ THỐNG PHÁT HIỆN XÂM NHẬP VÀ QUẢN LÝ LOG

2.1. CÁC HỆ THỐNG PHÁT HIỆN XÂM NHẬP

Như đã phân tích trong phần 1.3.2 về hệ thống phát hiện tấn công, xâm nhập thì IDS là hệ thống phát hiện xâm nhập nhằm phát hiện các cuộc tấn công vào máy tính hoặc các máy tính trong mạng.

Hệ thống luôn quan sát thông tin trên đường truyền nhằm phát hiện gói tin dựa vào các dấu hiệu trong nội dung gói tin, hoặc sự bất thường trong traffic của mạng. Khi phát hiện bất thường hệ thống sẽ cảnh báo cho người quản trị hoặc cho hệ thống xử lý

Snort, Suricata, Zeek và IBM Qradar là các hệ thống trong những hệ thống phát hiện xâm nhập phổ biến hiện nay sở hữu nhiều tính năng nổi bật cũng như được sử dụng rộng rãi bởi các cơ quan, tổ chức.

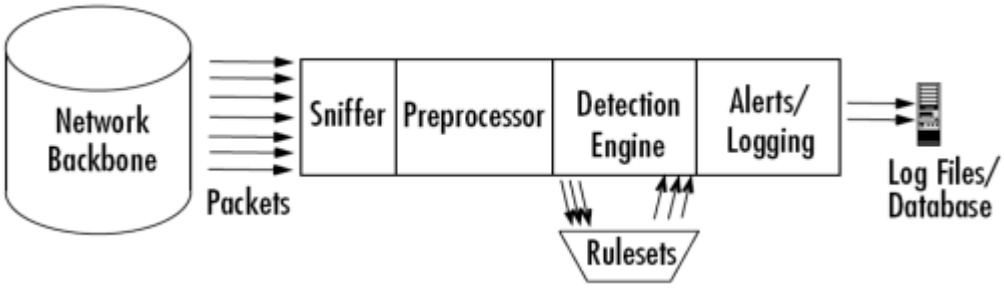
2.1.1. Snort

Snort là một hệ thống phòng chống xâm nhập mạng mã nguồn mở, có khả năng thực hiện phân tích lưu lượng theo thời gian thực và ghi nhật ký gói tin trên mạng IP [12]. Nó có thể thực hiện phân tích giao thức, tìm kiếm/đối sánh nội dung và có thể được sử dụng để phát hiện nhiều loại tấn công và thăm dò, chẳng hạn như tràn bộ đệm, quét cổng ẩn, tấn công CGI, thăm dò SMB, nỗ lực lấy dấu vân tay hệ điều hành và hơn thế nữa.

Snort hoạt động ở 3 chế độ chính là:

- Chế độ Sniffer, cho phép Snort đọc các gói tin trên mạng và hiển thị chúng lên màn hình.
- Chế độ Packet Logger, sẽ thực hiện ghi logs các gói tin thu thập được vào ổ đĩa dưới dạng tệp tin văn bản hoặc tệp tin nhị phân.
- Với chế độ Network Intrusion Detection System (NIDS), Snort sẽ thực hiện giám sát lưu lượng mạng, phân tích và đối chiếu chúng với một tập luật được định nghĩa bởi người dùng để phát hiện các xâm nhập. Đây cũng là chế độ phức tạp và nhiều tùy chỉnh nhất của Snort.

Về kiến trúc của Snort, bao gồm 5 phần chính như sau



Hình 9: Mô hình kiến trúc hệ thống Snort

Snort bao gồm 5 thành phần, với mỗi phần có một chức năng riêng biệt. Các phần chính đó là:

- ***Packet decoder***: Snort có thể nhận dữ liệu bằng cách phân tích một tệp tin PCAP hoặc lấy trực tiếp từ cảm biến đang giám sát mạng. Các dữ liệu này sau đó được đưa vào packet decoder để giải mã. Đầu tiên, nó giải mã các giao thức ở Data Link Layer, sau đó là giao thức IP, và TCP hoặc UDP. Packet decoder có thể đưa ra cảnh báo nếu nó phát hiện các header không đúng định dạng, các tùy chọn TCP bất thường hoặc các vấn đề tương tự.
- ***Preprocessor***: Preprocessor sẽ nhận dữ liệu từ packet decoder và thực hiện sắp xếp hoặc định dạng dữ liệu để Detection Engine có thể xử lý nó dễ dàng hơn. Nó có thể đưa ra cảnh báo, phân loại hoặc loại bỏ một gói tin trước khi nó được chuyển tới Detection Engine.
- ***Detection Engine***: Đây là thành phần quan trọng nhất trong kiến trúc của Snort. Detection Engine chịu trách nhiệm phát hiện các hành vi xâm nhập trong các gói tin. Để làm được việc này, nó thực hiện đối chiếu tất cả gói tin được nhận với các tập luật được định nghĩa bởi người dùng. Nếu một gói tin trùng khớp với bất kỳ luật nào, Snort sẽ thực hiện các hành động cụ thể được mô tả trong luật, ngược lại gói tin đó sẽ bị loại bỏ.
- ***Logging and Alerting System***: Bộ phận này sẽ sinh ra logs và các thông điệp cảnh báo dựa trên những gì mà Detection Engine tìm thấy trong gói tin.
- ***Output Module***: Output Module xử lý các cảnh báo và logs để đưa ra kết quả đầu ra dưới các định dạng khác nhau như syslog, tcpdump.

Tập luật là thành phần rất quan trọng của một hệ thống phát hiện xâm nhập. Đây là tập sẽ định ra dấu hiệu (mẫu) để đối chiếu, so sánh với dữ liệu ở đầu vào. Tập luật thường bao gồm rất nhiều luật, mỗi luật sẽ gồm hai thành phần cơ bản: **Rule Header** và **Rule Options.**

Rule header bao gồm các thông tin sau:

- Rule Action: Cho biết các hoạt động sẽ được thực thi khi "khớp" luật (dynamic, log, alert, pass, active, drop…).
- Protocol: Cho biết giao thức sẽ kiểm tra (TCP, UDP, ICMP, IP…).
- IP address: Cho biết thông tin về địa chỉ IP.
- Port number: Cho biết thông tin về cổng.
- Direction: Cho biết hướng của dữ liệu mà được so khớp.

Rule options chia làm 4 danh mục:

- General: cung cấp thông tin chung về luật (msg, reference, rev, classtype…).
- Payload: Tìm kiếm nội dung payload của gói tin (content, offset, depth, distance, within…).
- Non-payload: Tìm kiếm nội dung non-payload của gói tin (tos, id, ttl, ack, dsize…).
- Post-detection: Đưa ra các phương pháp thực thi tiếp theo (session, logto, tag…).

Ví dụ về luật của Snort:

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"Phat hien Ping "; sid:1;)

Phần header bao gồm các hành động được thực hiện khi phát hiện có gói tin trùng khớp với luật, giao thức của gói tin, địa chỉ, số cổng nguồn và đích.

alert icmp $EXTERNAL_NET any -> $HOME_NET any

Phần header của luật Snort bắt đầu với hành động được thực hiện khi đối chiếu luật trùng khớp như alert: đưa ra cảnh báo, drop: loại bỏ gói tin,… Tiếp theo là giao thức của gói tin có thể là ICMP, TCP, UDP hoặc IP. Kế tiếp là địa chỉ nguồn, cổng nguồn và chiều mũi tên chỉ hướng của lưu lượng mà luật áp dụng, cũng như là địa

chỉ đích và cổng đích. Với ví dụ trên, Snort sẽ thực hiện kiểm tra các gói tin ICMP từ địa chỉ IP bên ngoài với cổng nguồn bất kỳ tới địa chỉ IP đang được giám sát với cổng đích bất kỳ.

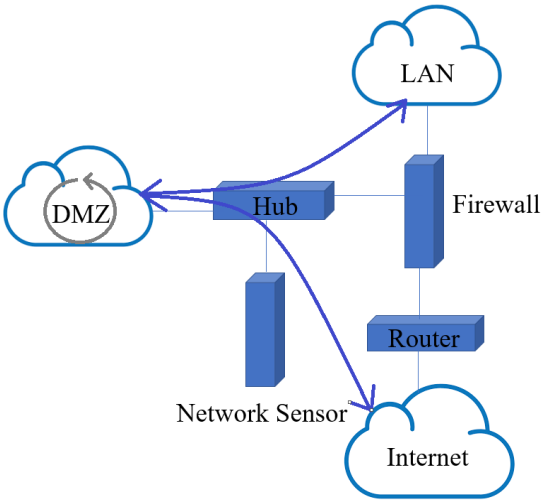
(msg:"Phat hien Ping"; sid:1;)

Phần Options mô tả những phần liên quan đến các gói tin được kiểm tra. Nó cho Snort chính xác những phần cần tìm kiếm và đối chiếu trong các gói tin để đưa ra hành động thích hợp. Tất cả các options của luật Snort đều được phân cách bởi dấu chấm phẩy (;). Các từ khóa option được phân cách với đối số của nó bởi dấu hai chấm (:). Ở ví dụ trên, Snort sẽ đưa ra thông điệp cảnh báo "Phat hien Ping" nếu có gói tin icmp được gửi từ địa chỉ IP khác với địa chỉ IP đang được giám sát.

Snort hỗ trợ nhiều hệ điều hành khác nhau cho phép nó dễ dàng triển khai trên bất cứ môi trường mạng nào. Các luật của Snort cũng được cập nhật nhanh chóng khi xuất hiện các kiểu tấn công, xâm nhập mạng mới. Snort sở hữu một cộng đồng người dùng lớn, do đó những vấn đề trong việc cài đặt, cấu hình hay tạo luật luôn được hỗ trợ và giải quyết nhanh chóng.

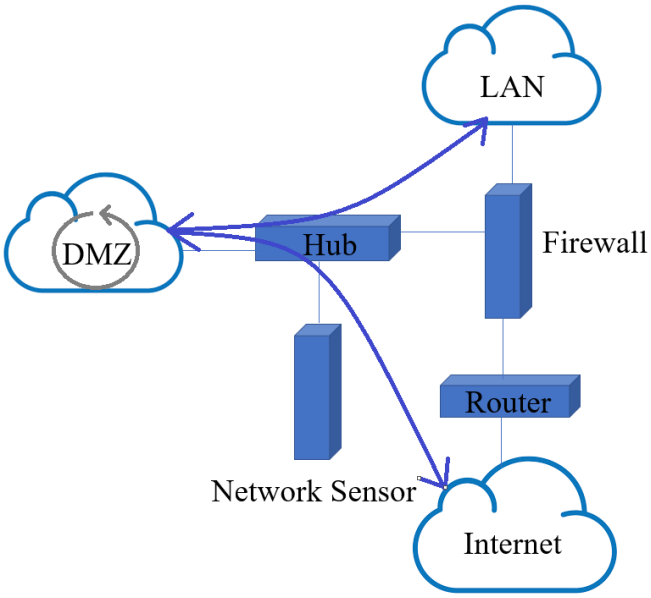
Vị trí của Snort trong hệ thống mạng

Vị trí giữa Router và Firewall



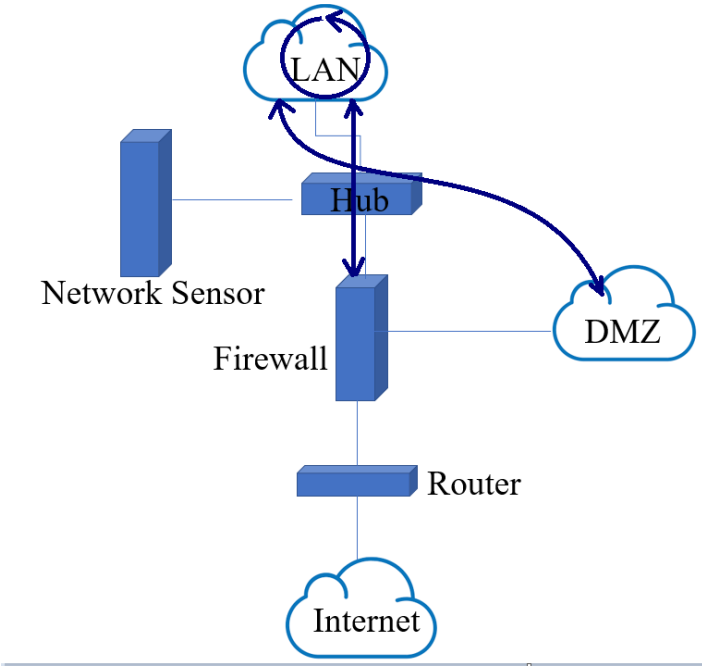
Hình 10: Snort-sensor đặt giữa Router và Firewall

Vị trí trong vùng DMZ



Hình 11: Snort-sensor đặt trong vùng DMZ

Vị trí sau Firewall

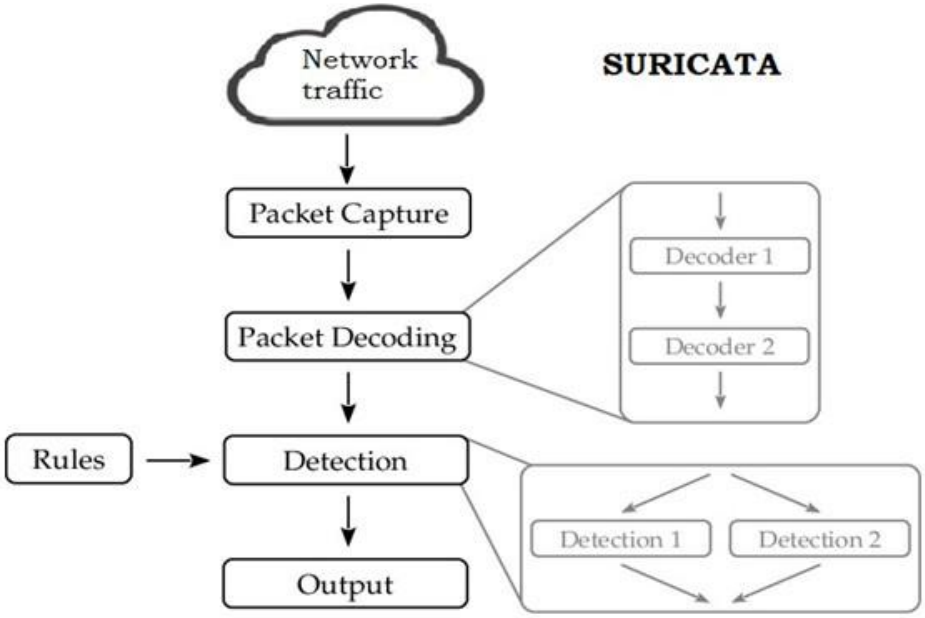


Hình 12: Snort-sensor đặt sau Firewall

2.1.2. Suricata

Suricata là một công cụ Giám sát an ninh mạng NSM (Network Security Mornitoring), phát hiện và ngăn chặn IDS/IPS (Intrusion Detection System/Intrusion Prevention System) mạng hiệu suất cao [4]. Suricata là mã nguồn mở được phát triển bởi OISF (Open Information Security Foundation) và do cộng

đồng điều hành. Nó hoạt động rất tốt với khả năng kiểm tra gói sâu và đối sánh mẫu, điều này khiến Suricata trở nên vô cùng hữu ích trong việc phát hiện các mối đe dọa và tấn công.



Hình 13: Mô tả sơ đồ Suricata

Kiến trúc của Suricata gồm 4 thành phần cơ bản: Packet Acquisition; Decode and Stream app, layer; Detect Modul; Outputs Modul. Bước đầu tiên trong quá trình xử lý là thu thập các gói tin với module Packet Acquisition. Mudule này có chức năng thu thập gói tin từ cổng mạng và chuyển tiếp chúng đến để giải mã gói tin (Decoder), nơi chịu trách nhiệm cho việc xác định các loại liên kết và chuẩn hóa dữ liệu cho các tiến trình khác.

Tiếp theo, dữ liệu sẽ được chuyển tới Stream module. Stream làm nhiệm vụ nhóm các dạng dữ liệu và reassembly các gói dữ liệu. Kế tiếp dữ liệu được đưa vào Detect Modul, nơi phân tích gói tin để phát hiện các tấn công mạng dựa trên các dấu hiệu.

Cuối cùng, cảnh báo được đưa ra khi có các dấu hiệu được phát hiện và được gửi tới Outputs Modul, dữ liệu đầu ra có thể được xác định ở nhiều dạng khác nhau.

Tương tự như Snort, các luật của Suricata cũng bao gồm hai phần chính là rule header và rule options. Ví dụ của một luật trong Suricata: alert icmp

$EXTERNAL_NET any -> $HOME_NET any (msg:”Phat hien Ping”; sid:2;)
Suricata sử dụng các luật tương tự với Snort và hoàn toàn tương thích với luật của Snort. Do đó, Suricata có thể sử dụng các luật của Snort để phát hiện tấn công, xâm nhập mạng. Cấu trúc luật của Suricata và Snort là hoàn toàn giống nhau. Tuy nhiên, luật của Suricata vẫn có một vài điểm khác với Snort. Suricata cho phép đặc tả trong luật một số giao thức của tầng ứng dụng như http, ftp, ssh, smb, dns,… Các giao thức tầng ứng dụng này có thể được Suricata phát hiện mà không cần chỉ rõ cổng. Trong Snort, để http_inspects và các bộ tiền xử lý khác được áp dụng cho lưu lượng mạng, nó phải qua một cổng đã được chỉ định trước đó. Suricata hỗ trợ một số từ khóa HTTP mà Snort không có như http_user_agent, http_host, http_content_type.

Một ưu điểm nữa của luật Suricata so với Snort là hỗ trợ Lua Script. Người dùng có thể sử dụng từ khóa lua (hoặc luajit) trong một luật để tham chiếu tới một Lua script cho phép linh hoạt trong việc truy cập gói tin, payload, HTTP buffers. Suricata hỗ trợ Lua script cho phép người dùng sử dụng các hàm có sẵn để phân tích gói tin nâng cao nhằm phát hiện các xâm nhập phức tạp, không thể mô tả bằng cú pháp luật thông thường.

Suricata là một công cụ tuyệt vời, chi phí thấp, giúp cung cấp thông tin chi tiết hơn về mạng. Mặc dù vậy, nó cần được xem như một lớp duy nhất trong một kế hoạch bảo mật toàn diện, thay vì một giải pháp hoàn chỉnh cho các vấn đề bảo mật. Một điểm khác biệt chính giữa Suricata và Snort là Suricata có kiến trúc đa luồng, có nghĩa là nó có thể sử dụng nhiều lõi cùng một lúc. Việc sử dụng nhiều CPU cho phép Suricata xử lý nhiều sự kiện cùng lúc mà không phải làm gián đoạn các yêu cầu khác. Đa luồng cũng cho phép Suricata cân bằng tải trên các CPU, cũng như cải thiện hiệu suất tổng thể trong phân tích lưu lượng mạng. Điều này là thuận lợi vì nó có nghĩa là Suricata có thể xử lý số lượng lớn lưu lượng truy cập mà không cần cắt giảm các quy tắc.

Suricata có các phiên bản chạy trên Windows, Mac, Linux, Unix và FreeBSD tương thích với một loạt các công cụ của bên thứ ba. Nó sử dụng các định dạng đầu

vào và đầu ra tiêu chuẩn như YAML và JSON, cho phép tích hợp dễ dàng với các công cụ như Splunk, Kibana và Elasticsearch. Một lợi thế quan trọng khác của Suricata là nó có một cộng đồng hỗ trợ và phát triển tuyệt vời. Họ đã cùng nhau xây dựng các nguồn tài nguyên phong phú bao gồm hướng dẫn cài đặt, hướng dẫn sử dụng, các câu hỏi thường gặp và cách khắc phục.

Suricata có thể được sử dụng trong ba vai trò chính. Đơn giản nhất là thiết lập nó như một IDS dựa trên máy chủ lưu trữ, giám sát lưu lượng của một máy tính cá nhân. Điều này không đặc biệt hiệu quả hoặc thực tế, nhưng nó có thể là một cách tuyệt vời để ai đó làm quen với Suricata. Là một IDS thụ động, Suricata có thể giám sát tất cả lưu lượng truy cập qua mạng và thông báo cho quản trị viên khi gặp bất kỳ điều gì độc hại. Khi Suricata được thiết lập như một IDS nội tuyến hoạt động và IPS, nó có thể giám sát lưu lượng đến và đi. Nó có thể ngăn chặn lưu lượng độc hại trước khi xâm nhập vào mạng, cũng như cảnh báo cho quản trị viên.

Đặt Suricata làm IPS có vẻ là lựa chọn rõ ràng nhất - tại sao bạn chỉ muốn theo dõi lưu lượng độc hại thay vì chặn nó? Thật không may, việc triển khai IPS không đơn giản như vậy - chúng cũng thường chặn lưu lượng truy cập hợp pháp nếu chúng không được định cấu hình đúng cách. Có thể khá khó khăn khi điều chỉnh các quy tắc để đảm bảo rằng chỉ có lưu lượng truy cập độc hại mới bị chặn. Nếu bạn đang cân nhắc sử dụng Suricata làm IPS, tốt nhất nên sử dụng nó làm IDS trước để đảm bảo rằng nó không chặn bất kỳ lưu lượng quan trọng nào.

Suricata là một công cụ mạng IDS hiệu suất cao (Hệ thống phát hiện xâm nhập), IPS và bảo mật mạng, được phát triển bởi OISF, đây là một ứng dụng mã nguồn mở đa nền tảng và là tài sản của một nền tảng phi lợi nhuận của cộng đồng Open Information Security Foundation (OISF). Nó được phát triển không nhằm thay thế hay cạnh tranh các công tụ hiện có, nhưng nó sẽ mang lại những công nghệ và ý tưởng mới trong lĩnh vực an ninh mạng.

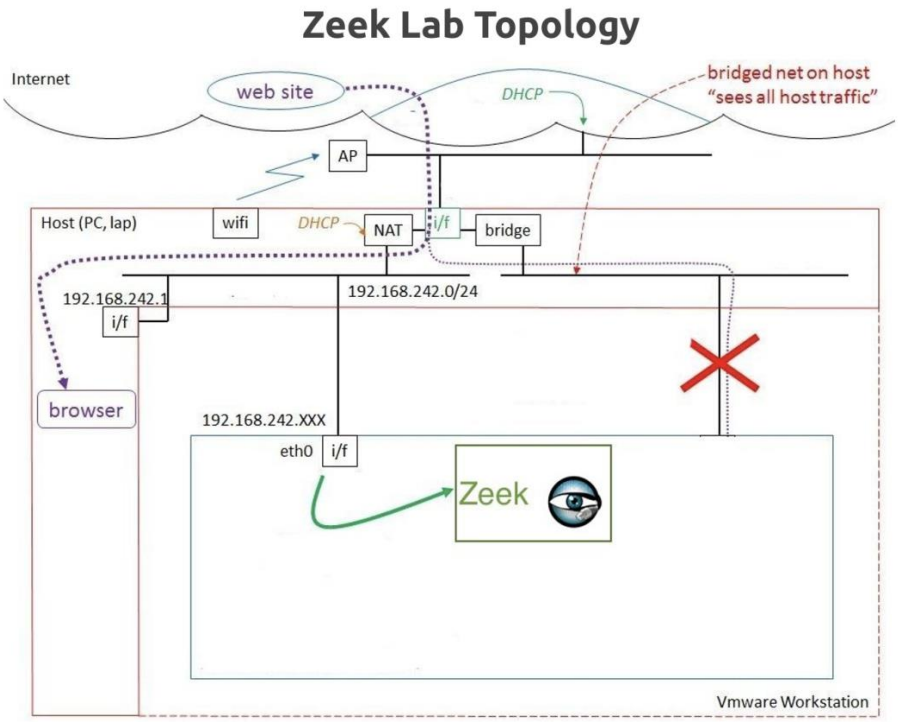
Nó dựa trên một bộ quy tắc phát triển bên ngoài để giám sát lưu lượng mạng và cung cấp cảnh báo cho người quản trị hệ thống khi xảy ra các sự kiện đáng ngờ. Được thiết kế để tương thích với các thành phần bảo mật mạng hiện có, cung

cấp chức năng đầu ra hợp nhất và các tùy chọn thư viện có thể cắm để chấp nhận cuộc gọi từ các ứng dụng khác. Là một công cụ đa luồng, nó cung cấp tốc độ và hiệu quả cao hơn trong việc phân tích lưu lượng mạng. Ngoài việc tăng hiệu quả của phần cứng (phần cứng và card mạng bị giới hạn), nó còn được xây dựng để tận dụng khả năng xử lý cao được cung cấp bởi chip CPU đa lõi đời mới.

2.1.3. Zeek

Zeek là một công cụ phân tích lưu lượng mạng mã nguồn mở thụ động. Nhiều nhà khai thác sử dụng Zeek như một trình giám sát an ninh mạng (NSM) để hỗ trợ các cuộc điều tra về hoạt động đáng ngờ hoặc độc hại. Zeek cũng hỗ trợ một loạt các nhiệm vụ phân tích lưu lượng ngoài phạm vi bảo mật, bao gồm đo lường hiệu suất và khắc phục sự cố.

Sơ đồ về Zeek



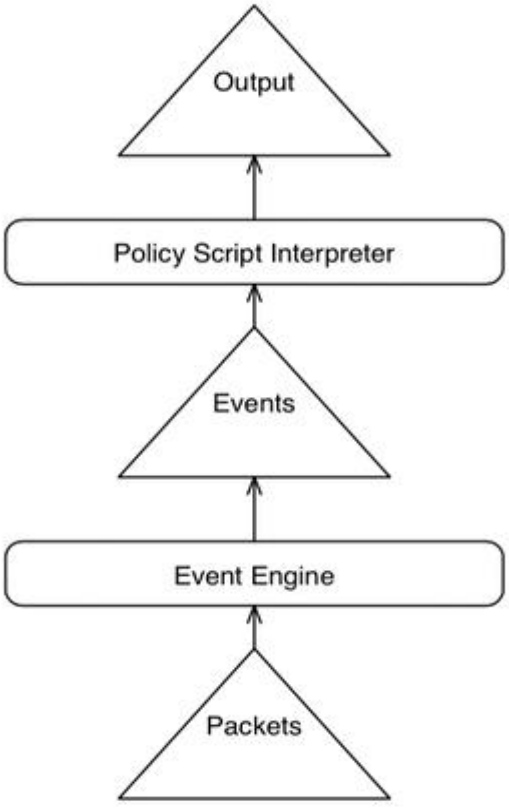
Hình 14: Sơ đồ về Zeek

Zeek không phải là một hệ thống phát hiện xâm nhập dựa trên chữ ký cổ điển (IDS); trong khi nó cũng hỗ trợ chức năng tiêu chuẩn như vậy, ngôn ngữ kịch bản của Zeek tạo điều kiện cho một phạm vi rộng hơn nhiều các phương pháp tiếp cận

rất khác nhau để tìm ra hoạt động độc hại. Chúng bao gồm phát hiện lạm dụng ngữ nghĩa, phát hiện bất thường và phân tích hành vi.

Zeek được tối ưu hóa để diễn giải lưu lượng mạng và tạo nhật ký dựa trên lưu lượng đó. Nó không được tối ưu hóa để đối sánh byte và người dùng tìm kiếm các phương pháp phát hiện chữ ký sẽ được phục vụ tốt hơn bằng cách thử các hệ thống phát hiện xâm nhập như Suricata. Zeek cũng không phải là một công cụ phân tích giao thức theo nghĩa của Wireshark, tìm cách mô tả mọi yếu tố của lưu lượng mạng ở cấp khung hoặc một hệ thống lưu trữ lưu lượng ở dạng bắt gói (PCAP). Thay vào đó, Zeek nằm ở "phương tiện hài lòng" thể hiện nhật ký mạng nhỏ gọn nhưng có độ trung thực cao, giúp hiểu rõ hơn về lưu lượng mạng và việc sử dụng.

Kiến trúc của Zeek



Hình 15: Kiến trúc của Zeek

Ở cấp độ rất cao, Zeek được phân loại về mặt kiến trúc thành hai thành phần chính. Event Engine (phần lõi) có nhiệm vụ chuyển đổi các lưu lượng mạng mà Zeek bắt được thành một chuỗi các sự kiện. Những sự kiện này phản ánh hoạt động

mạng theo các thuật ngữ trung lập về chính sách (policy neutral), nghĩa là chỉ mô tả những hoạt động xảy ra trên thực tế mà không xem xét hành động đó có chấp thuận hay vi phạm chính sách được đặt ra. Ví dụ, mỗi yêu cầu HTTP sẽ được chuyển đổi thành sự kiện http_request tương ứng với các mô tả về địa chỉ IP, cổng, URL được yêu cầu và phiên bản HTTP được sử dụng. Sự kiện này sẽ không đưa ra bất kỳ diễn giải gì thêm như là URL được yêu cầu có phải trang web độc hại hay không.

Các sự kiện trên sẽ được chuyển tới thành phần chính thứ hai của Zeek là Policy Script Interpreter có nhiệm vụ thực hiện một tập các trình xử lý sự kiện được viết bằng ngôn ngữ kịch bản của riêng Zeek. Các kịch bản này mô tả các chính sách bảo mật để xác định các sự kiện vi phạm và đưa ra cảnh báo cho quản trị viên. Ngôn ngữ của Zeek hỗ trợ nhiều chức năng khác nhau, nhưng nổi bật nhất là cho phép các kịch bản duy trì trạng thái theo thời gian. Chức năng này cho phép kịch bản theo dõi và tương quan sự thay đổi của sự kiện giữa các kết nối và máy chủ khác nhau. Các kịch bản của Zeek cũng có thể tạo cảnh báo theo thời gian thực và thực thi các chương trình bên ngoài tùy ý như là kích hoạt phần mềm phản ứng với một tấn công. Zeek cung cấp một tập các logs mở rộng ghi lại chi tiết các hoạt động của mạng. Các logs này không chỉ bao gồm một bản ghi đầy đủ về các kết nối trên mạng mà còn ở tầng ứng dụng như các phiên HTTP với URL được yêu cầu, các headers, và phản hồi của máy chủ; yêu cầu DNS và phản hồi, chứng chỉ SSL,... Thêm vào đó, Zeek cũng tích hợp một số chức năng cho việc phân tích và phát hiện xâm nhập như trích xuất các tập tin từ phiên HTTP, thông báo phiên bản lỗ hổng chứa nguy cơ, lỗ hổng trên hệ thống mạng, phát hiện tấn công vét cạn SSH. Ngoài ra, với việc sử dụng ngôn ngữ kịch bản của riêng mình, Zeek cung cấp cho người dùng nhiều phương pháp khác nhau để phát hiện hành vi độc hại, bao gồm phát hiện dựa trên chữ ký, phát hiện dựa trên bất thường và phân tích hành vi.

2.1.4. IBM Qradar

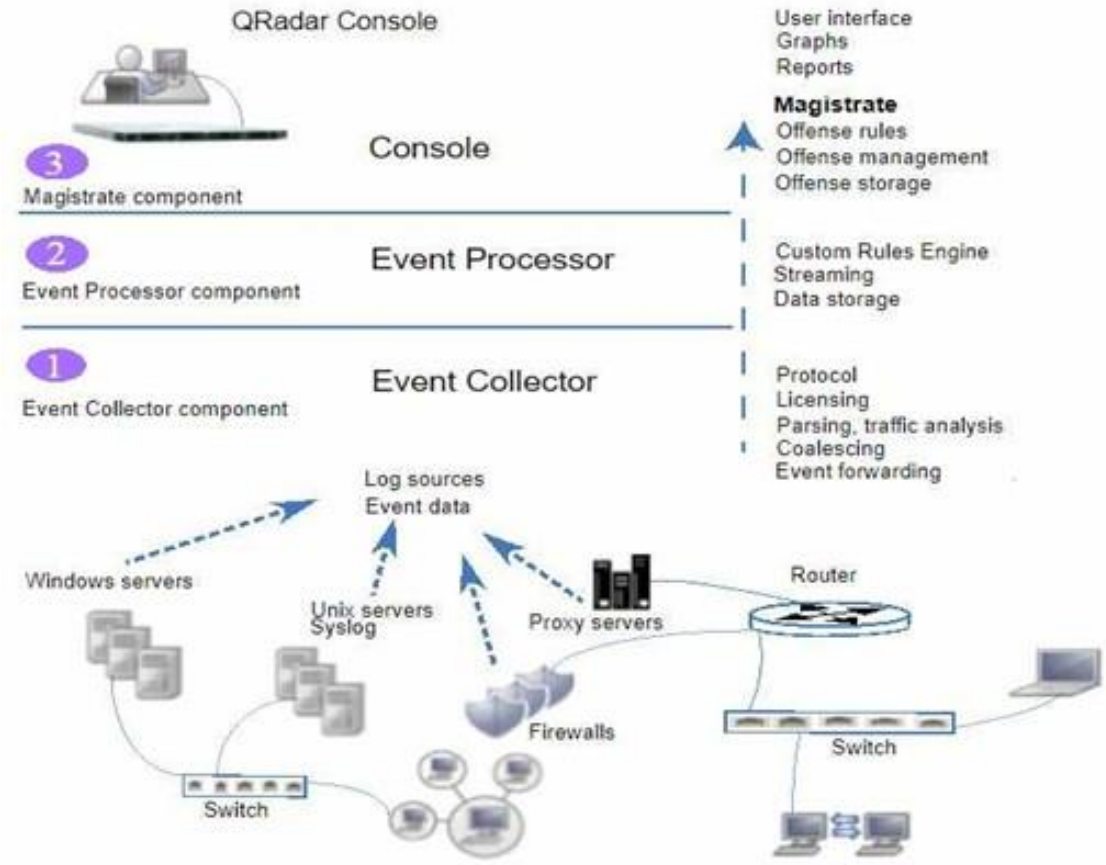
IBM Qradar là một hệ thống tích hợp các chức năng thu thập, xử lý, tổng hợp và lưu trữ dữ liệu mạng trong thời gian thực. Qradar sử dụng dữ liệu đó để quản lý

an ninh mạng bằng cách cung cấp thông tin và giám sát theo thời gian thực, cảnh báo và hành vi vi phạm cũng như phản ứng với các mối đe dọa mạng.

Thành phần và chức năng.

Hoạt động của nền tảng thông minh bảo mật Qradar bao gồm ba lớp và áp dụng cho bất kỳ cấu trúc triển khai Qradar nào, bất kể quy mô và độ phức tạp của nó. Sơ đồ sau đây cho thấy các lớp tạo nên kiến trúc Qradar.

- Event collector component
- Event processor component
- Magistrate



Hình 16: Minh hoạ sơ đồ IBM Qradar

a. Event Collector Component

Đối với Event

- Protocol: Nhận các dữ liệu đầu ra từ các giao thức mã nguồn Log mà các giao thức đó được Qradar hỗ trợ trong việc thu thập các sự kiện từ các thiết bị nguồn.
- Throttle: Giám sát số lượng sự kiện và lưu vào hệ thống để quản lý cấp phép đầu vào. Qradar sẽ cho phép lưu lượng các sự kiện đầu vào của mỗi thiết bị là bao nhiêu.
- Parsing: Lấy các sự kiện từ các thiết bị nguồn và phân tích.
- Log source traffic analysis & auto discovery: Áp dụng phân tích dữ liệu các sự kiện. Khi thu thập các trường sự kiện trên các thiết bị khác nhau, Qradar sẽ bóc tách các trường đó thành một chuẩn chung.
- Coalescing: Sự kiện được phân tích sau đó kết hợp lại dựa trên các loại sự kiện phổ biến. Khi 4 sự kiện được nhìn thấy với cùng một nguồn IP, IP đích, cổng đích và Tên truy nhập, các thông điệp tiếp theo cho đến 10 giây của cùng một khuôn mẫu được kết hợp lại với nhau. Điều này được thực hiện để giảm trùng lặp dữ liệu được lưu trữ.
- Event forwarding: Áp dụng quy tắc định tuyến cho hệ thống, chẳng hạn như gửi các tín hiệu đến các thiết bị ngoại vi, hệ thống SysLog bên ngoài, hệ thống JSON, hệ thống SIEM khác… Ở đây dữ liệu sẽ được chuyển đến Processor.

Đối với Flow

- Luồng trùng lặp: là quá trình loại bỏ các trùng lặp khi nhiều luồng QFlow giống nhau được thu thập cung cấp dữ liệu cho thiết bị xử lý luồng.
- Kết hợp bất đối xứng: Chịu trách nhiệm về kết hợp luồng bên trong và ngoài khi dữ liệu được cung cấp không đối xứng. Quá trình này có thể nhận ra các luồng từ mỗi bên và kết hợp chúng vào trong một bản ghi. Tuy nhiên, đôi khi không tồn tại dữ liệu ở hai bên.
- Throttle: Giám sát số sự kiện và lưu vào hệ thống để quản lý cấp phép đầu vào.

- Chuyển tiếp: Áp dụng quy tắc định tuyến cho hệ thống, chẳng hạn như gửi các tín hiệu đến các thiết bị ngoại vi, hệ thống SysLog bên ngoài, hệ thống JSON, hệ thống SIEM khác…

b. Event Processor Component

- Custom Rules Engine: Chịu trách nhiệm xử lý các sự kiện nhận được từ Qradar và so sánh chúng với các luật, duy trì theo dõi các hệ thống có liên quan đến sự kiện theo thời gian. Tạo ra các thông báo cho người dùng và các vi phạm.
- Host profile: Chịu trách nhiệm giải quyết các thông tin tài sản từ luồn dữ liệu thụ động. Luồng cung cấp thông tin về hoạt động mạng và cho phép Qradar xây dựng một cơ sở dữ liệu về tài sản.
- Streaming: Chịu trách nhiệm cho việc gửi dữ liệu sự kiện thời gian thực đến Console khi người dùng đang xem các sự kiện từ tab Log Activity với thời gian thực.
- Storage: Theo thời gian cơ sở dữ liệu các sự kiện và luồng được lưu trữ theo từng phút. Dữ liệu được lưu trữ nơi sự kiện được xử lý. Những sự kiện đi vào thiết bị, chúng được xử lý bởi ECS và được lưu trữ cục bộ trên thiết bị trong giai đoạn lưu trữ của ECS. Khi thu thập Log từ các Collector, các Log được cho là an toàn và không an toàn (có sự tấn công) sẽ được lưu trữ tại Ariel máy chủ Processor. Riêng Log không an toàn (có sự tấn công) sẽ được lưu trữ tại Ariel của Console.

c. Magistrate

Magistrate Processing Core (MPC) Chịu trách nhiệm tương quan giữa các hành vi vi phạm với các sự kiện được gửi từ nhiều Event Processor. Chỉ có Console mới có thành phần Magistrate

- Offenses rules: Giám sát và tác động đến các sự kiện vi phạm, chẳng hạn như tạo ra các thông báo emai khi có các sự kiện vi phạm các quy tắc được để ra.
- Offense management: Cập nhật, quản lý các hành vi vi phạm. Tiếp cận các hành vi vi phạm để đưa cho người sử dụng các thông tin về vi phạm qua Tab Offenses.
- Offense storage: Ghi dữ liệu hành vi vi phạm đến một cơ sở dữ liệu. Cơ sở dữ liệu trên Console thường lưu dữ các sự kiện vi phạm.

2.1.5. So sánh các hệ thống phát hiện xâm nhập

Qua những phân tích, đánh giá về khái niệm, kiến trúc cũng như cơ chế hoạt động của các hệ thống phát hiện xâm nhập phổ biến hiện nay là Snort, Suricata và Zeek, IBM Qradar ta có thể so sánh các hệ thống này theo bảng dưới đây:

Tiêu chí	**SNORT**	**SURICATA**	**ZEEK**	**IBM QRADAR**
Luồng xử lý	Đơn luồng	Đa luồng	Đơn luồng	Đa luồng
Sử dụng tài nguyên hệ thống	Trung bình	Nhiều	Ít	Nhiều
Tỷ lệ bỏ qua gói tin khi lưu lượng ít	Cao	Thấp	Trung bình	Thấp
Kỹ thuật phát hiện xâm nhập	Signature-based IDS	Signature-based IDS	Signature-based IDS và Anomaly-based IDS	Signature-based IDS
Tập luật	Sử dụng các luật từ VRT, Emerging Threat, cũng như là các tập luật được viết bởi cộng đồng	Sử dụng các luật từ VRT, Emerging Threat. Ngoài ra còn hỗ trợ các luật được viết bằng Lua script.	Sử dụng các luật được viết bằng chính Zeek script	Công cụ quy tắc tùy chỉnh (CRE)
Kết quả đầu ra	Có thể ghi kết quả đầu ra dưới dạng Syslog, tcpdump, csv hoặc unified2.	Cho phép ghi kết quả đầu ra dưới dạng Eve json và syslog. Ngoài ra còn hỗ trợ dùng Lua script để lấy kết quả đầu ra	Mặc định ghi kết quả đầu ra dưới dạng ASCII. Có thể tùy chỉnh để ghi dưới định dạng JSON hoặc ghi vào SQLite	Chuyển tiếp dữ liệu tới các mục tiêu ngoại vi, hệ thống Syslog bên ngoài, hệ thống JSON và các SIEM khác.
Hệ điều hành hỗ trợ	Linux, FreeBSD, OpenBSD, MacOS, Windows	Linux, FreeBSD, MacOS, Windows	Linux, FreeBSD, MacOS	Linux, Windows

Bảng 1: So sánh các hệ thống phát hiện xâm nhập

2.2. HỆ THỐNG QUẢN LÝ LOG ELK

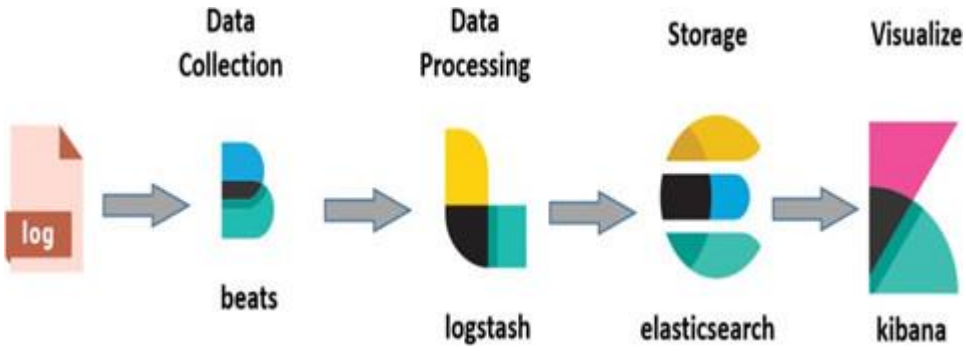
Hệ thống quản lý log ELK hay còn gọi là ELK Stack là một bộ sưu tập của ba sản phẩm mã nguồn mở - Elasticsearch, Logstash, và Kibana [15]. ELK Stack cung cấp ghi nhật ký tập trung để xác định các sự cố với máy chủ hoặc ứng dụng. Nó cho phép bạn tìm kiếm tất cả các bản ghi ở một nơi duy nhất. Nó cũng giúp tìm ra sự cố trong nhiều máy chủ bằng cách kết nối các bản ghi trong một khung thời gian cụ thể.

- E là viết tắt của ElasticSearch: dùng để lưu trữ nhật ký.

- L là viết tắt của LogStash: được sử dụng cho cả vận chuyển cũng như xử lý và lưu trữ nhật ký.

- K là viết tắt của Kibana: là một công cụ trực quan hóa (một giao diện web) được lưu trữ thông qua Nginx hoặc Apache

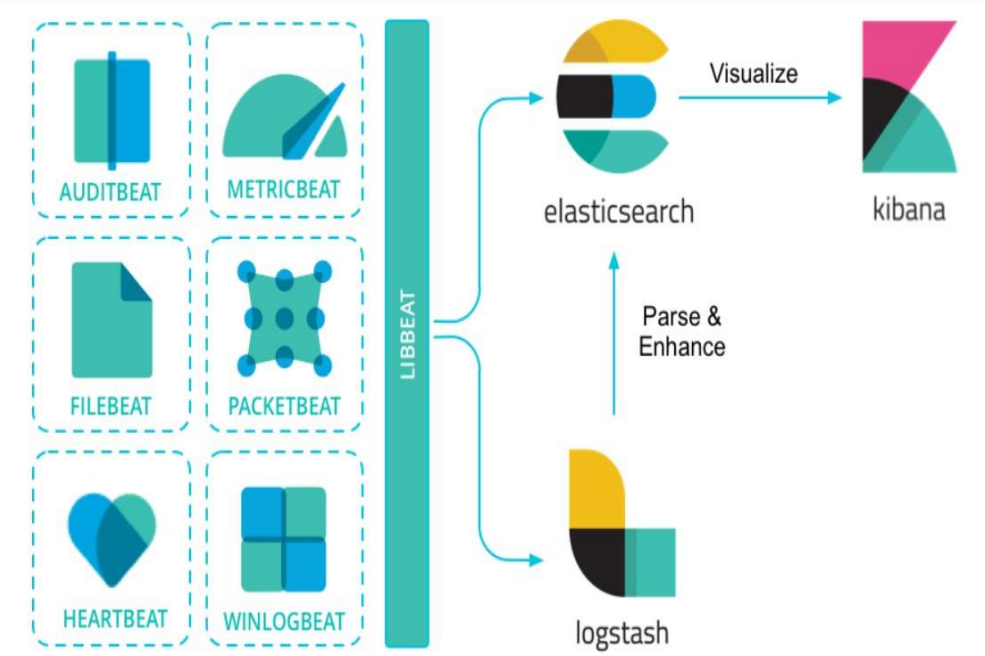
Trong các phiên bản mới hơn, ELK Stack bổ sung thêm một thành phần mới là Beats. Nó có nhiệm vụ gửi dữ liệu thu thập từ logs của máy chủ đến Logstash hoặc Kibana. Với việc bổ sung thêm Beats, ELK Stack cũng đổi tên thành Elastic Stack.



Hình 17: Kiến trúc Elastic Stack

Cơ chế hoạt động của Elastic Stack được mô tả qua hình 2.5. Đầu tiên, các logs được thu thập bởi các Beats được cài đặt ngay trên các máy chủ. Beats cung cấp nhiều ứng dụng để thu thập các loại logs khác nhau. Ví dụ, Filebeat được sử dụng để thu thập dữ liệu từ các tập tin logs, Winlogbeat dùng để thu thập các logs sự kiện trên Windows. Các logs này sau đó có thể được chuyển tới Logstash hoặc chuyển thẳng tới Elasticsearch. Logstash sẽ thực hiện tổng hợp các logs từ các ứng

dụng Beats hoặc từ các nguồn khác, xử lý, chuẩn hóa log và ghi vào cơ sở dữ liệu là Elasticsearch. Khi muốn xem logs, người dùng truy cập vào giao diện web của Kibana. Kibana sẽ đọc dữ liệu logs trong Elasticsearch, hiển thị lên giao diện cho người dùng tìm kiếm, truy vấn và phân tích.



Hình 18: Cơ chế hoạt động của Elastic Stack

2.2.1. Elasticsearch

Elasticsearch là một cơ sở dữ liệu NoSQL. Nó dựa trên công cụ tìm kiếm Lucene và nó được xây dựng với RESTful APIS. Nó cung cấp triển khai đơn giản, độ tin cậy tối đa và quản lý dễ dàng. Nó cũng cung cấp các truy vấn nâng cao để thực hiện phân tích chi tiết và lưu trữ tất cả dữ liệu một cách tập trung. Nó rất hữu ích để thực hiện tìm kiếm nhanh các tài liệu.

Elasticsearch cũng cho phép bạn lưu trữ, tìm kiếm và phân tích khối lượng lớn dữ liệu. Nó chủ yếu được sử dụng làm công cụ cơ bản để cung cấp năng lượng cho các ứng dụng đã hoàn thành các yêu cầu tìm kiếm. Nó đã được áp dụng trong các nền tảng công cụ tìm kiếm cho các ứng dụng web và di động hiện đại. Ngoài tính năng tìm kiếm nhanh, công cụ này còn cung cấp các phân tích phức tạp và nhiều tính năng nâng cao.

2.2.2. Logstash

Logstash là công cụ đường ống thu thập dữ liệu. Nó thu thập dữ liệu đầu vào và cấp dữ liệu vào Elasticsearch. Nó thu thập tất cả các loại dữ liệu từ các nguồn khác nhau và làm cho nó có sẵn để sử dụng thêm.

Các tính năng của Logstash bao gồm: các sự kiện được chuyển qua từng giai đoạn bằng cách sử dụng hàng đợi nội bộ, cho phép các đầu vào khác nhau cho nhật ký, lọc/phân tích cú pháp các nhật ký.

Lợi thế của Logstash: cung cấp tập trung dữ liệu, phân tích nhiều loại dữ liệu và sự kiện có cấu trúc/phi cấu trúc, cung cấp các plugin để kết nối với nhiều loại nguồn và nền tảng đầu vào khác nhau.

2.2.3. Kibana

Kibana là một công cụ trực quan hóa dữ liệu hoàn thành ngăn xếp ELK. Công cụ này được sử dụng để trực quan hóa các tài liệu Elasticsearch và giúp các nhà phát triển có cái nhìn sâu sắc về nó. Bảng điều khiển Kibana cung cấp các sơ đồ tương tác, dữ liệu không gian địa lý và đồ thị khác nhau để hình dung các quy tắc phức tạp. Nó có thể được sử dụng để tìm kiếm, xem và tương tác với dữ liệu được lưu trữ trong các thư mục Elasticsearch. Kibana giúp bạn thực hiện phân tích dữ liệu nâng cao và trực quan hóa dữ liệu của bạn trong nhiều bảng, biểu đồ và bản đồ.

Các tính năng của Kibana: cho phép tìm kiếm thông tin được lập chỉ mục trong thời gian thực, có thể tìm kiếm, xem và tương tác với dữ liệu được lưu trữ trong Elasticsearch, thực hiện hiện các truy vấn trên dữ liệu và trực quan hóa kết quả trong biểu đồ, bảng và bản đồ, v.v…

Lợi thế của Kibana: dễ hình dung, tích hợp hoàn toàn với Elasticsearch; công cụ trực quan hóa; cung cấp khả năng phân tích, lập biểu đồ, tóm tắt và gỡ lỗi trong thời gian thực; cung cấp giao diện thân thiện với người dùng.

ElasticSearch, LogStash và Kibana đều được phát triển, quản lý và duy trì bởi công ty có tên Elastic. Elastic Stack được thiết kế để cho phép người dùng lấy dữ liệu từ bất kỳ nguồn nào, ở bất kỳ định dạng nào và tìm kiếm, phân tích và trực quan hóa dữ liệu đó trong thời gian thực.

Kết luận chương

Chương II đã trình bày về kiến trúc, hoạt động cũng như tính năng của các hệ thống phát hiện xâm nhập mạng Snort, Suricata, Zeek. Bên cạnh đó chương này còn trình bày về các hệ thống phát hiện xâm nhập tích hợp IBM Qradar cho phép tìm kiếm, phát hiện xâm nhập cả hệ thống mạng và máy chủ, đồng thời hiển thị các logs thu thập được cho người dùng phân tích. Từ đó đưa ra bảng so sánh các hệ thống này. Cuối cùng, chương II đưa ra những giới thiệu tổng quan kiến trúc, cơ chế hoạt động của bộ công cụ thu thập logs Elastic Stack.

CHƯƠNG 3
XÂY DỰNG VÀ THỬ NGHIỆM MÔ HÌNH GIẢI PHÁP PHÁT HIỆN XÂM NHẬP DỰA TRÊN SNORT VÀ ELK CHO HỆ THỐNG MẠNG HỌC VIỆN THANH THIẾU NIÊN VIỆT NAM

3.1. *KHẢO SÁT VÀ TRIỂN KHAI MÔ HÌNH*

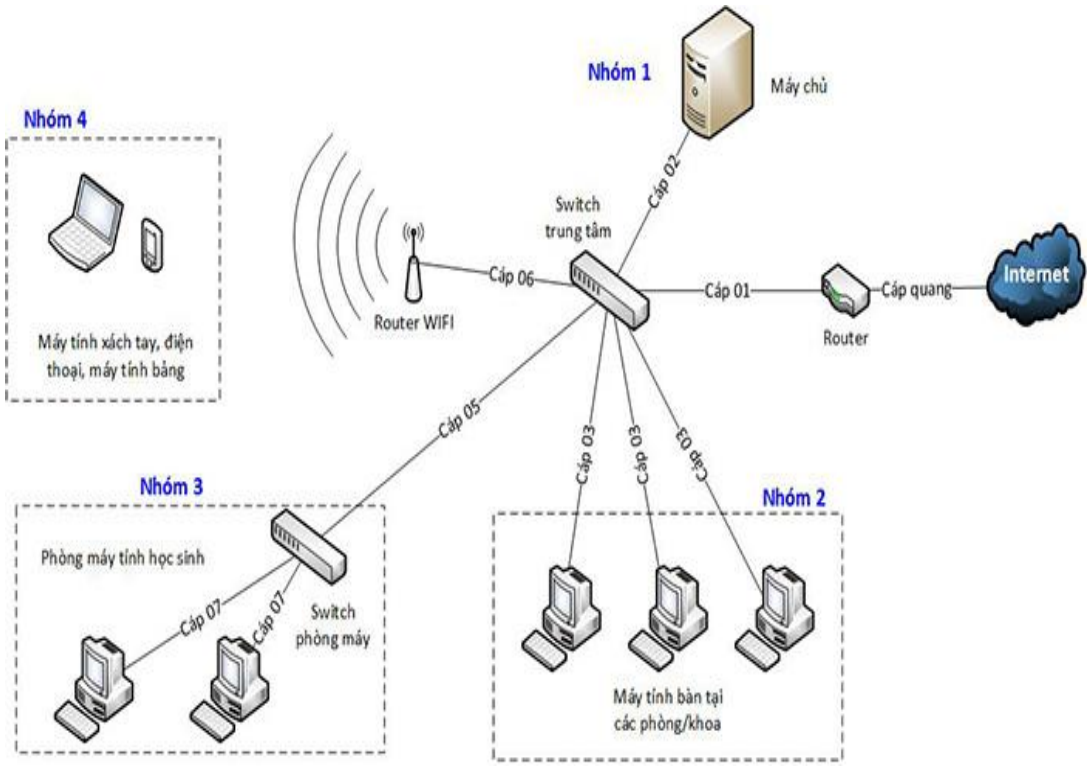
3.1.1. Khảo sát hệ thống mạng tại Học viện Thanh thiếu niên Việt Nam

Hình 19: Phối cảnh tổng thể học viện Thanh thiếu niên Việt Nam

Trường Học viện Thanh thiếu niên Việt Nam được xây dựng trên diện tích 13 ha bao gồm 163 phòng học và giảng đường đạt tiêu chuẩn, 17 phòng thực hành, thí nghiệm; ngoài ra còn có nhà tập Đa năng; Phòng làm việc của Giáo sư, phó Giáo sư, giảng viên của cơ sở đào tạo là 50, Trung tâm thư viện hiện đại với diện tích trên 640m2 đã đưa vào sử dụng; sân bãi, 1 tòa nhà hiệu bộ 11 tầng đang được xây dựng và nhiều trang thiết bị khác đáp ứng đủ nhu cầu dạy học.

Qua khảo sát hệ thống mạng, máy tính tại trường *Học viện Thanh thiếu niên Việt Nam* nhận thấy việc cần phải đưa hệ thống phát hiện xâm nhập vào sử dụng là hết sức cần thiết.

3.1.1.1. Sơ đồ mạng



Hình 20: Mô hình mạng máy tính trường Học viện Thanh thiếu niên Việt Nam

Cụ thể hệ thống mạng được chia làm các vùng sau:

- Nhóm 1:Hệ thống máy chủ
- Nhóm 2:Máy tính tại các phòng ban, khoa
- Nhóm 3:Hệ thống máy tính phòn thực hành
- Nhóm 4:Máy tính xách tay, điện thoại, máy tính bảng

3.1.1.2. Hiện trạng hạ tầng máy chủ

Theo thống kê đầy đủ, hệ thống máy chủ cả vật lý và ảo hóa cung cấp cho các dịch vụ công nghệ thông tin của của Học viện Thanh thiếu niên Việt Nam và các đơn vị thành viên để phục vụ công việc

Bảng : Số lượng máy chủ cung cấp cho các đơn vị thuộc Học viện thanh thiếu niên Việt Nam

Đơn vị	Số lượng máy chủ (Vật lý + ảo hóa)
Học viện Thanh thiếu niên Việt Nam	3 máy chủ

Trung tâm bồi dưỡng cán bộ	2 máy chủ
Tạp chí Thanh niên	2 máy chủ
Viện Nghiên cứu Thanh niên	2 máy chủ
Trung tâm phát triển kỹ năng xã hội	1 máy chủ
Trung tâm tin học ngoại ngữ	1 máy chủ

Bên cạnh các máy chủ còn có các thiết bị mạng, thiết bị lưu trữ đặc thù cũng cần phải được giám sát hoạt động thường xuyên.

3.1.1.3. Hiện trạng nền tảng hệ điều hành và phần mềm

Do dịch vụ cung cấp cho Học viện thanh thiếu niên Việt Nam và các đơn vị thành viên là rất nhiều và đa dạng nên nền tảng hệ điều hành và phần mềm sử dụng trong các dịch vụ công nghệ thông tin cũng rất phong phú về chủng loại và phiên bản.

a. Hiện trạng nền tảng hệ điều hành

Window	Linux	Oracle Solaris	HP-UX	Ubuntu
- Window Server 2012 - Window Server 2016	- Linux RedHat 6 - Linux RedHat 7 - Oracle Linux 7	- Solaris 10 - Solaris 11	- *HP-UX 10* - *HP-UX 11*	Ubuntu 20.04

Bảng 2: Hiện trạng nền tảng hệ điều hành được sử dụng tại Học viện Thanh thiếu niên Việt Nam

b. Hiện trạng nền tảng phần mềm

Cơ sở dữ liệu	Phần mềm lớp giữa	Web Server	Http Server	Cân bằng tải
Oracle: 12c, 11g, 10g	Oracle Weblogic: 10g, 11g, 12c	Microsoft IIS	Apache Http Server	HA-Proxy
SQL Server: 2005, 2008	IBM WebSphere Application Server	Apache Tomcat	Oracle WebCache	Ngnix
MySQL, MariaDB	JBoss		Oracle Http Server	KeepAlive
PosgreSQL	Oracle Service Bus 12c		IBM Http Server	

Bảng 3: Hiện trạng nền tảng phần mềm được sử dụng tại Học viện Thanh thiếu niên Việt Nam và các đơn vị thành viên

3.1.1.4. Hiện trạng quản lý, giám sát hệ thống

Hiện tại Học viện Thanh thiếu niên Việt Nam đang giám sát hệ thống một cách khá thủ công. Hàng ngày nhóm trực hệ thống sẽ thực hiện truy xuất định kỳ (từ 1 đến 3 lần tùy theo hệ thống) vào nơi lưu dữ liệu log trên các máy chủ và thực hiện đọc, tìm các mã lỗi một cách thủ công theo hướng dẫn của cán bộ quản trị trên các tệp dữ liệu log đó. Nếu phát hiện được mã lỗi, cán bộ trực hệ thống sẽ gửi thư điện tử thông báo tới cán bộ quản trị được biết để thực hiện kiểm tra và khắc phục lỗi.

Nhận thấy với cách thức kiểm tra dữ liệu log như hiện nay tồn tại các hạn chế sau:

- Cán bộ phải truy xuất dữ liệu thủ công và phân tán trên các máy chủ để đọc và tìm mã lỗi.
- Việc tìm kiếm thủ công sẽ rất chậm và có thể gây ra nhầm lẫn hoặc thiếu sót trong quá trình kiểm tra dữ liệu log, gây rủi ro sẽ xảy ra sự cố cho hệ thống mà người quản trị không được biết kịp thời.
- Dữ liệu log không được tận dụng để xây dựng các báo cáo và phân tích

để tìm ra những thông tin hữu ích.

- Mất nhiều nguồn lực con người để thực hiện các công việc đọc và tìm kiếm lỗi thủ công trong khi có thể tự động hóa được công việc này, giải phóng được nhân lực.

3.1.1.5. Những tồn tại của hệ thống mạng tại trường.

Hiện nay, trường chưa được cài hệ thống tường lửa để bảo vệ toàn thể máy tình các phòng ban, các khu vực mạng tránh khỏi các hiểm họa, nguy cơ về an toàn thông tin, các rủi ro có thể ảnh hưởng tới hệ thống mạng:

- Các cuộc tấn công vào khu vực DMZ khi mà các vùng đó chạy dịch vụ: DNS, web, mail.
- Các tấn công tới người dùng (Social engineering) trong mạng.
- Tấn công vào mạng LAN của trường
- Tấn công nghe trộm trên đường truyền
- Tấn công từ chối dịch vụ vào các hệ thống mạng văn phòng của phòng, ban.
- Tấn công bằng mã độc vào hệ thống mạng thông qua các phương tiện khác hoặc người dung. Mạng chưa có hệ thống cân bằng tải, hệ thống dự phòng, vv.. do vậy tính sẵn sàng của hệ thống còn nhiều điểm hạn chế.
- Xâm nhập vật lý tới các thiết bị cụ thể của hệ thống

Do đó việc triển khai một hệ thống phần mềm phát hiện xâm nhập cho Học viện Thanh thiếu niên Việt Nam là cấp bách và cần thiết, dựa trên đánh giá về những mặt tồn tại của hệ thống mạng và cơ sở vật chất, nguồn lực của nhà trường luận văn này đưa ra mô hình triển khai thử nghiệm cài đặt phần mềm Snort để áp dụng.

3.1.2. Mô hình triển khai

Hệ thống thử nghiệm được triển khai cài đặt gồm 2 máy như sau:

- Máy Kali Linux sử dụng các công cụ để thực hiện các kịch bản tấn công mạng đến máy mục tiêu là máy Ubuntu chạy SNORT và ELK
- Máy Ubuntu cài đặt và cấu hình hệ thống phát hiện xâm nhập SNORT và ELK để đưa ra cảnh báo.

- **Môi trường giả lập:** VMware Workstation Pro 12
- **Thiết lập cấu hình:**

✓ Client: Windows 10 – **Vmnet 11: 192.168.11.12**

✓ Attacker: Kali Linux 2016.2 – **Vmnet 11: 192.168.11.10**

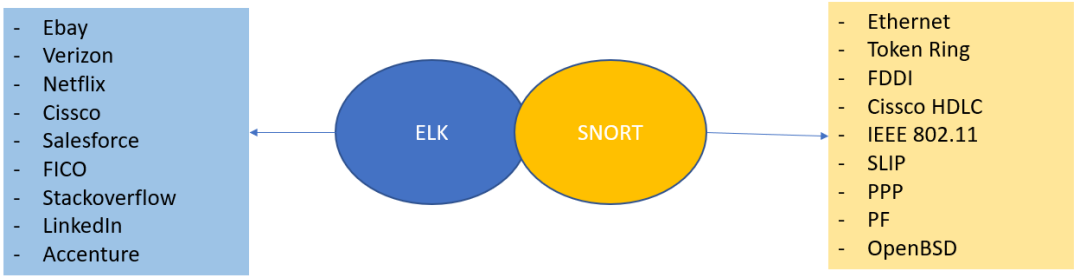
✓ Snort IDS/IPS: CentOS 6.8 – **Vmnet 11: 192.168.11.11/24**
Vmnet 12: 192.168.10.12/24

✓ WEB Server: CentOS 6.8 – **Vmnet 12: 192.168.10.13**

3.2. LỰA CHỌN CÔNG NGHỆ SỬ DỤNG

Bài toán quản lý dữ liệu tập trung, phát hiện và cảnh báo xâm nhập tự động có thể được thực hiện bằng nhiều giải pháp công nghệ khác nhau bao gồm cả các giải pháp phần mềm thương mại và phần mềm mã nguồn mở trong số đó thì Snort và ELK là nổi bật hơn cả. Các tính năng ưu việt của hai giải pháp này phải kể đến như: khả năng tìm kiếm mạnh mẽ, xây dựng được màn hình giám sát thời gian thực, báo cáo, cảnh bảo ngưỡng, phân tích dữ liệu lịch sử, truy tìm vết…

Một số nền tảng công nghệ được sử dụng trên những giải pháp này là:



Hình 21: Một số nền tảng công nghệ được sử dụng để quản lý dữ liệu

Mỗi giải pháp đều có những ưu nhược điểm và mức độ phù hợp riêng. Dưới đây là bảng so sánh về một số tính năng của ba giải pháp trên:

Tính năng	ELK	SNORT
Bản quyền	- Mã nguồn mở - Thương mại	- Mã nguồn mở - Miễn phí
Ngôn ngữ	Java, JRuby, NodeJS	TCP/IP, Novell'IPX

Định dạng dữ liệu	JSON	ASCII
Độ phức tạp cài đặt	Cần cài đặt 3 thành phần: - ElasticSearch - Logstash - Kibana Độ phức tạp trung bình.	Cần cài đặt một số thư viện trước khi cài Snort Sau đó sửa một số file cấu hình và update thư viện dùng chung Độ phức tạp trung bình
Nền tảng hỗ trợ	Unix, Window, Linux, Ubuntu, Solaris	Windows, Linux, OpenBSD, FreeBSD, NetBSD, Solaris, HP-UX, AIX, IRIX, MacOS.
Định dạng tệp log hỗ trợ	Các loại tệp dữ liệu log phổ biến như ngnix, http, database, tomcat, ...	Bất kỳ định dạng tệp nào như text, CSV, tệp log, .
Công cụ vận chuyển dữ liệu	- Apache Kafka - RabbitMQ - Redis	- MySQL - Oracle
Tổng hợp dữ liệu	Tổng hợp theo lô và Real-time	Tổng hợp theo lô và Real-time
Khả năng tìm kiếm	Khả năng tìm kiếm và phân tích mạnh mẽ với ElasticSearch	Có khả năng phát hiện một số lượng lớn các kiểu thăm dò, xâm nhập khác nhau như: Buffer overflow, CGI-attack, dò tìm hệ điều hành, ICMP, virus,…
Khả năng xây dựng báo cáo, màn hình giám sát, trừu tượng hóa dữ liệu	Với phần mềm Kibana trong bộ giải pháp, ELK cho khả năng dựng báo cáo, xây dựng màn hình giám sát mạnh mẽ, trực quan.	Tính năng báo cáo, giám sát được xây dựng sẵn

Bảng 4: So sánh tính năng của E.L.K và Snort

Nếu không sử dụng các giải pháp phần mềm ta cũng hoàn toàn có thể tự phát triển code các thành phần của hệ thống nhưng như thế rất tốn kém về nguồn lực, thời gian và hiệu quả mạng lại chưa chắc đã cao. Dựa vào những phân tích ở trên thì việc đi sâu tìm hiểu 2 giải pháp mã nguồn mở ELK và Snort là phương án tối ưu phù hợp nhất với nhu cầu và hiện trạng tại Học viện Thanh thiếu niên Việt Nam.

3.3. CÀI ĐẶT VÀ CẤU HÌNH HỆ THỐNG PHÁT HIỆN XÂM NHẬP

Snort là giải pháp phần mềm Intrusion Detection and Prevention Systems (IDPS) với chức năng và chi phí phù hợp cho các doanh nghiệp từ nhỏ và vừa (SMEs) tới các doanh nghiệp lớn. Phần mềm Snort được đánh giá cao bởi cả người dùng lẫn chuyên gia trong lĩnh vực Network Security Software.Snort là phần mềm được phát triển bởi Martin Roesh dưới dạng mã nguồn mở vào năm 1998. Snort ban đầu được xây dựng trên nền Unix nhưng sau đó phát triển sang các nền tảng khác. Hiện tại Snort đang được điều hành bởi Sorucefie và nằm trong những nền tảng phát triển nhất của hãng firewall Checkpoint. Snort được đánh giá rất cao về khả năng phát hiện xâm nhập. Tuy Snort miễn phí nhưng nó lại có rất nhiều tính năng tuyệt vời. Với kiến trúc kiểu module, người dùng có thể tự tăng cường tính năng cho hệ thống Snort của mình. Snort có thể chạy trên nhiều hệ thống như Windows, Linux, OpenBSD, FreeBSD, Solaris,…

Hiện tại phiên bản cập nhật mới nhất của Snort là Snort v3.0 bắt đầu được sử dụng từ năm 2021.

Trong phiên bản mới của Snort 3 đã được chuyển sang hệ thống thiết lập mới, cung cấp cú pháp đơn giản hóa và cho phép sử dụng các tập lệnh để tạo cấu hình động. LuaJIT được sử dụng để xử lý các tệp cấu hình và các plugin dựa trên LuaJIT có các tùy chọn bổ sung cho các quy tắc và hệ thống đăng ký.

Một số điểm nổi bật của Snort 3 :

- Đã thêm hỗ trợ tệp để ghi đè nhanh cài đặt liên quan đến cài đặt mặc định.
- Việc sử dụng snort_config.lua và SNORT_LUA_PATH đã bị ngừng để đơn giản hóa cấu hình.
- Đã thêm hỗ trợ để tải lại cài đặt một cách nhanh chóng.

- Hệ thống nhật ký sự kiện mới sử dụng định dạng JSON và dễ dàng tích hợp với các nền tảng bên ngoài như Elastic Stack.
- Tự động phát hiện các dịch vụ đang chạy, loại bỏ sự cần thiết phải chỉ định các cổng mạng đang hoạt động theo cách thủ công.
- Mã cung cấp khả năng sử dụng các cấu trúc C ++ được xác định trong tiêu chuẩn C ++ 14 (hợp ngữ yêu cầu trình biên dịch hỗ trợ C ++ 14).
- Một bộ điều khiển VXLAN mới đã được thêm vào.
- Cải thiện tìm kiếm các loại nội dung theo nội dung bằng cách sử dụng các triển khai thay thế cập nhật của thuật toán Boyer-Moore và Hyperscan.
- Khởi chạy tăng tốc bằng cách sử dụng nhiều luồng để biên dịch các nhóm quy tắc;
- Đã thêm một cơ chế đăng ký mới.
- Hệ thống kiểm tra RNA (Nhận thức mạng thời gian thực) đã được thêm vào, hệ thống này thu thập thông tin về tài nguyên, máy chủ, ứng dụng và dịch vụ có sẵn trên mạng.

3.3.1. Cài đặt snort IDS

✓ **Cài đặt Package yêu cầu:**

yum install -y gcc flex bison zlib* libxml2 libpcap* pcre* tcpdump git libtool curl daq

yum groupinstall - y "Development Tools"

✓ **Tải và cài đặt các file cài đặt riêng sau:**

libdnet-1.12.tgz

libdnet-1.12-6.el6.x86_64.rpm

libdnet-devel-1.12-6.el6.x86_64.rpm

✓ **Tải file snort mới nhất tại https://snort.org**

daq-2.0.6.tar

snort-2.9.8.3.tar

Các tùy chọn

Phần Rule Option nằm ngay sau phần Rule Header và được bao bọc trong dấu ngoặc đơn. Nếu có nhiều option thì các option sẽ được phân cách với nhau bằng dấu chấm phẩy ”;”. Nếu nhiều option được sử dụng thì các option này phải đồng thời được thoã mãn tức là theo logic các option này liên kết với nhau bằng AND.

Mọi option được định nghĩa bằng các từ khoá. Một số các option còn chứa các tham số. Nói chung một option gồm 2 phần: một từ khoá và một tham số, hai phần này phân cách nhau bằng dấu hai chấm. Ví dụ đã dùng:

msg: "Detected confidented";

msg là từ khoá còn “Detected confidented” là tham số. Sau đây là chi tiết một số các option của luật Snort.

Từ khoá ack

Trong header TCP có chứa trưởng Acknowledgement Number với độ dài 32 bit. Trường này có ý nghĩa là chỉ ra số thứ tự tiếp theo gói tin TCP của bên gửi đang được chờ để nhận. Trưởng này chỉ có ý nghĩa khi mà cờ ACK được thiết lập.

Các công cụ như Nmap sử dụng đặc điểm này ping một máy. Ví dụ, nó có thể gửi một gói tin TCP tới cổng 80 với cờ ACK được bật và số thứ tự là 0. Bởi vậy, bên nhận sẽ thấy gói tin không hợp lệ và sẽ gửi trở lại gói tin RST. Khi mà Nmap nhận được gói tin RST thì tức là địa chỉ đích đang “sống”. Phương pháp này vẫn làm việc tốt đối với các máy không trả lời gói tin thuộc dạng ping

ICMP ECHO REQUEST.

Vậy để kiểm tra loại ping TCP này thì ta có thể dùng luật như sau:

alert tep any any > 192.168.1.0/24 any (flags: A; ack: 0; msg: "TCP ping detected")

Từ khoá classtype

Các luật có thể được phân loại và gán cho một số chỉ độ ưu tiên nào đó để nhóm và phân biệt chúng với nhau. Để hiểu rõ hơn về từ khoá này ta đầu tiên phải hiểu được file classification.config (được bao gồm trong file snort.conf sử dụng từ khoá include). Mỗi dòng trong file classification.config có cú pháp như sau:

config classification: name, description, priority

Trong đó:

- name: là tên dùng để phân loại, tên này sẽ được dùng với tử khoá classtype trong các luật Snort.
- description: mô tả về loại lớp này
- priority: là một số chỉ độ ưu tiên mặc định của lớp này. Độ ưu tiên này có thể được điều chỉnh trong từ khoá priority của phần option trong luật của Snort.

Ví dụ:

config classification: DOS, Denial of Service Attack, 2

và trong luật:

alert udp any any > 192.168.1.0/24 6838 (msg:"DoS"; content: "server"; classtype: DoS:)

alert udp any any > 192.168.1.0/24 6838 (msg:"DoS"; content: "server"; classtype: DoS; priority: 1;)

Trong câu lệnh thứ 2 thì ta đã ghi đè lên giá trị priority mặc định của lớp đã định nghĩa.

Từ khoá content

Một đặc tính quan trọng của Snort là nó có khả năng tìm một mẫu dữ liệu bên trong một gói tin. Mẫu này có thể dưới dạng chuỗi ASCII hoặc là một chuỗi nhị phân dưới dạng các kí tự hệ 16. Giống như virus, các tấn công cũng có các dấu hiệu nhận dạng và từ khoá content này dùng để tìm các dấu hiệu đó bên trong gói tin. Ví dụ:

alert tcp 192.168.1.0/24 any-> [192.168.1.0/24] any (content: "GET""; msg: "GET match";)

Luật trên tìm mẫu “GET” trong phần dữ liệu của tất cả các gói tin TCP có nguồn đi từ mạng 192.168.1.0/24 và đi đến các địa chỉ không thuộc mạng đó. Từ “GET” này rất hay được dùng trong các tấn công HTTP.

Một luật khác cũng thực hiện đúng nhiệm vụ giống như lệnh trên nhưng mẫu dữ liệu lại dưới dạng hệ 16 là:

alert tcp 192.168.1.0/24 any -> ![192.168.1.0/24] any (content: "147 45 541"; msg: "GET match";)

Để ý rằng số 47 ở hệ 16 chính là bằng kí tự ASCII : G và tương tự 45 là E và 54 là T. Ta có thể dùng cả hai dạng trên trong cùng một luật nhưng nhớ là phải để dạng thập lục phân giữa cặp kí tự II.

Tuy nhiên khi sử dụng từ khoá content ta cần nhớ rằng:

Đối sánh nội dung sẽ phải xử lý tính toán rất lớn và ta phải hết sức cân nhắc khi sử dụng nhiều luật có đối sánh nội dung.

Ta có thể sử dụng nhiều từ khoá content trong cùng một luật để tìm nhiều dấu hiệu trong cùng một gói tin.

Đối sánh nội dung là công việc rất nhạy cảm.

Có 3 tử khoá khác hay được dùng cùng với tử khoá content dùng để bổ sung thêm các điều kiện để tìm kiếm là:

- Offset: dùng để xác định vị trí bắt đầu tìm kiếm (chuỗi chứa trong tử khoá content) là offset tính tử đầu phần dữ liệu của gói tin. Ví dụ sau sẽ tìm chuỗi "HTTP" bắt đầu tử vị trí cách đầu đoạn dữ liệu của gói tin là 4 byte:

alert tcp 192.168.1.0/24 any -> any any (content: "HTTP"; offset: 4; msg: "HTTP matched";)

- Dept : dùng để xác định vị trí mà từ đó Snort sẽ dùng việc tìm kiếm.Từ khoá này cũng thường được dùng chung với từ khoá offset vừa nêu trên. Ví dụ:

alert tep 192.168.1.0/24 any -> any any (content: "HTTP"; offset: 4; dept: 40; msg: "HTTP matched";).

Từ khoá này sẽ giúp cho việc tiêu tốn thời gian tìm kiếm khi mà đoạn dữ liệu trong gói tin là khá lớn.

- Content-list: được sử dụng cùng với một file. Tên file (được chỉ ra trong phần tham số của từ khoá này) là một file text chứa danh sách các chuỗi cần tìm trong phần dữ liệu của gói tin. Mỗi chuỗi nằm trên một dòng riêng biệt. Ví dụ như file test có dạng như sau:

"test"

"Snort" "NIDS"

Và ta có luật sau:

alert tep 192.168.1.0/24 any-> any any (content-list: "test";msg: "This is my Test";).

Ta cũng có thể dùng kí tự phủ định ! trước tên file để cảnh báo đối với các gói tin không tìm thấy một chuỗi nào trong file đó.

Từ khoá dsize

Dùng để đối sánh theo chiều dài của phần dữ liệu. Rất nhiều tấn công sử dụng lỗi tràn bộ đệm bằng cách gửi các gói tin có kích thước rất lớn. Sử dụng tử khoá này, ta có thể so sánh độ lớn của phần dữ liệu của gói tin với một số nào đó.

alert ip any any -> 192.168.1.0/24 any (dsize: > 6000; msg: "Goi tin co kich thuoc lon":)

Từ khoá flags

Từ khoá này được dùng để phát hiện xem những bit cở flag nào được bật (thiết lập) trong phần TCP header của gói tin. Mỗi cờ có thể được sử dụng như một tham số trong từ khoá flags. Sau đây là một số các cở sử dụng trong từ khoá flags:

Flag	Kí tự tham số dùng trong luật của Snort
FIN (Finish Flag)	F
SYN – Sync Flag	S
RST – Reset Flag	R
PSH – Push Flag	P
ACK – Acknowledge Flag	A
URG – Urgent Flag	U
Reserved Bit 1	1
Reserved Bit 2	2
No Flag set	0

Bảng 5: Các cử sử dụng với từ khoá flags

Ta có thể sử dụng các dấu +, * và ! để thực hiện các phép toán logic AND, OR và NOT trên các bit cờ muốn kiểm tra. Ví dụ luật sau đây sẽ phát hiện một hành động quét dùng gói tin TCP SYN-FIN: alert tcp any any -> 192.168.1.0/24 any (flags: SF; msg: "SYNC-FIN packet detected";)

Từ khoá fragbits

Phần IP header của gói tin chứa 3 bit dùng để chống phân mảnh và tổng hợp các gói tin IP. Các bit đó là:

- Reserved Bit (RB) dùng để dành cho tương lai.
- Don’t Fragment Bit (DF): nếu bit này được thiết lập thì tức là gói tin đó không bị phân mảnh.

More Fragments Bit (MF): nếu được thiết lập thì tức là các phần khác (gói tin bị phân mảnh) của gói tin vẫn đang còn trên đường đi mà chưa tới đích. Nếu bit này không được thiết lập thì có nghĩa là đây là phần cuối cùng của gói tin (hoặc là gói duy nhất). Điều này xuất phát tử nguyên nhân: Nơi gửi đi phải chia gói tin IP thành nhiều đoạn nhỏ do phụ thuộc vào Đơn vị truyền dữ liệu lớn nhất cho phép (Maximum Transfer Units - MTU) trên đường truyền. Kích thước của gói tin không được phép vượt quá kích thước lớn nhất này. Do vậy, bit MF này giúp bên đích có thể tổng hợp lại các phần khác nhau thành một gói tin hoàn chỉnh.

Đôi khi các bit này bị các hacker sử dụng để tấn công và khai thác thông tin trên mạng của ta. Ví dụ, bit DF có thể được dùng để tìm MTU lớn nhất và nhỏ nhất trên đường đi từ nguồn xuất phát đến đích đến.

Sử dụng fragbits, ta có thể kiểm tra xem các bit trên có được thiết lập hay không. Ví dụ luật sau sẽ phát hiện xem bit DF trong gói tin ICMP có được bật hay không: alert icmp any any -> 192.168.1.0/24 any (fragbits: D; msg: "Dont Fragment bit set":)

Trong luật này , D dùng cho bit DF, R cho bit dự trữ và M cho bit MF. Ta cũng có thể dùng dấu phủ định ! trong luật này để kiểm tra khi bit không được bật: alert icmp any any -> 192.168.1.0/24 any (fragbits: !D; msg: "Dont Fragment bit not set":)

✓ **<u>Bắt đầu cài đặt snort</u>**

cd /usr/local/src

tar -zxvf /root/ips/daq-2.0.6.tar.gz

tar -zxvf /root/ips/snort-2.9.8.3.tar.gz

```
# cd daq-2.0.6
# ./configure
# make && make install

# cd /usr/local/src/snort-2.9.8.3
# ./configure
# make && make install

# cd /etc
# mkdir snort
# cd snort
# cp /usr/local/src/snort-2.9.8.3/etc/* .
# tar -zvxf /root/ips/snortrules-snapshot-2983.tar.gz
# touch /etc/snort/rules/white_list.rules /etc/snort/rules/black_list.rules
```

✓ **Tạo user, group, cấp quyền:**

```
# groupadd -g 40000 snort
# useradd snort -u 40000 -d /var/log/snort -s /sbin/nologin -c SNORT_IDS -g snort
# cd /etc/snort
# chown -R snort:snort *
# chown -R snort:snort /var/log/snort
```

✓ **Cấu hình snort:**

```
# vim /etc/snort/snort.conf
```

output alert_unified2: filename snort.alert, limit 128, nostamp

output log_unified2: filename snort.log, limit 128, nostamp

ipvar HOME_NET any > ipvar HOME_NET 192.168.10.0/24

ipvar EXTERNAL_NET any > ipvar EXTERNAL_NET !$HOME_NET

var RULE_PATH ../rules > var RULE_PATH /etc/snort/rules

var SO_RULE_PATH ../so_rules > var SO_RULE_PATH /etc/snort/so_rules

var PREPROC_RULE_PATH ../preproc_rules > var PREPROC_RULE_PATH /etc/snort/preproc_rules

var WHITE_LIST_PATH ../rules > var WHITE_LIST_PATH /etc/snort/rules

var BLACK_LIST_PATH ../rules > var BLACK_LIST_PATH /etc/snort/rules

"Esc : wq!" lưu cấu hình vào file.

```
# cd /usr/local/src
# chown -R snort:snort daq-2.0.6
# chown -R 777 daq-2.0.6
# chown -R snort:snort snort-2.9.8.3
# chown -R 777 snort-2.9.8.3
# chown -R snort:snort snort_dynamicsrc
# chown -R 777 snort_dynamicsrc
```

✓ **Khởi động snort**

```
# cd /usr/local/src/snort-2.9.8.3/rpm
# cp snortd /etc/init.d/snortd
# cp /usr/local/src/snort-2.9.8.3/rpm/snort.sysconfig /etc/sysconfig/snort
# chkconfig --add /etc/init.d/snortd
# chkconfig snortd on
# cd /usr/sbin
# ln -s /usr/local/bin/snort snort
```

Nếu chưa có directory /var/log/snort thì tạo:

mkdir -p /var/log/snort

Cấp quyền:

chmod 777 snort

chown -R snort:snort snort

cd /usr/local/lib

chown -R snort:snort snort*

chown -R snort:snort snort_dynamic*

chown -R snort:snort pkgconfig

chown -R 777 snort*

chown -R 777 pkgconfig

cd /usr/local/bin

chown -R snort:snort daq-modules-config

chown -R snort:snort u2*

chown -R 777 daq-modules-config

chown 777 u2*

cd /etc

chown -R snort:snort snort

chown -R 777 snort

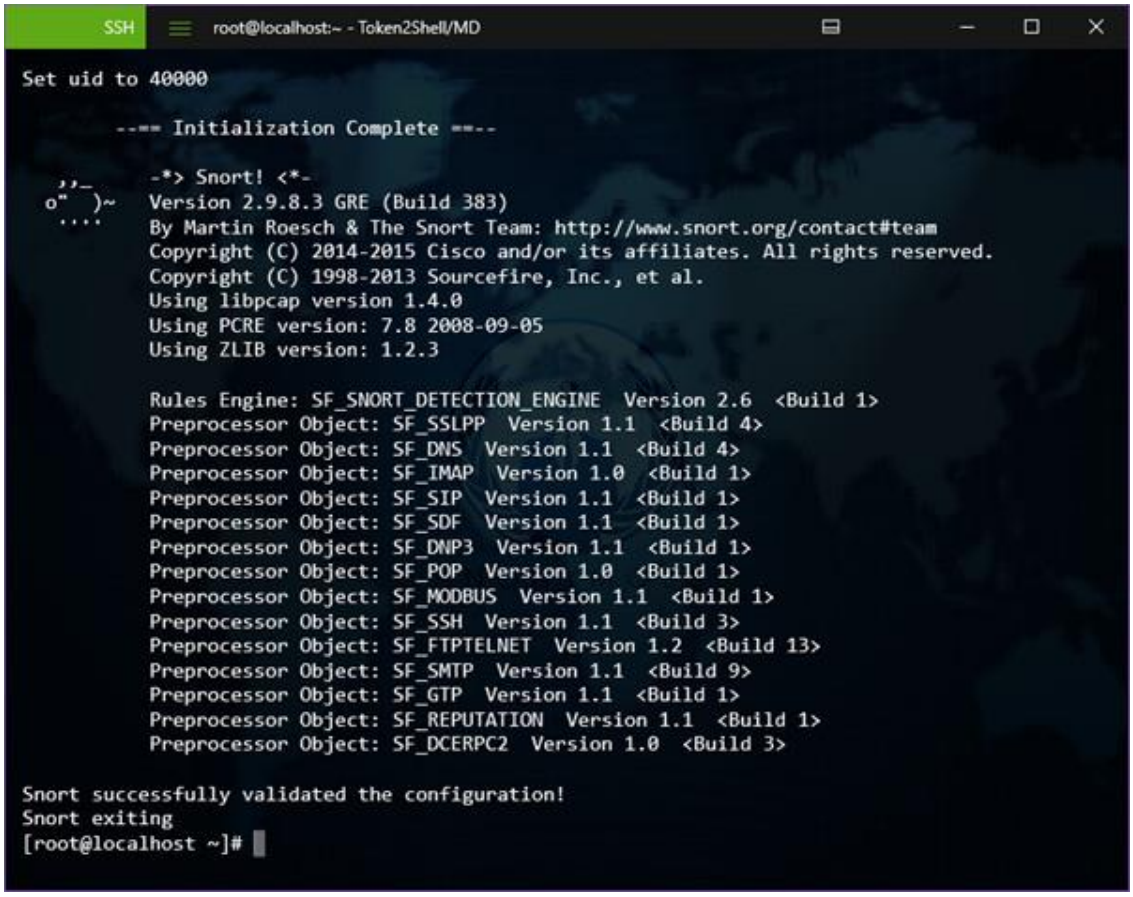
Tạo thư mực dynamicrules:

mkdir -p /usr/local/lib/snort_dynamicrules

chown -R snort:snort /usr/local/lib/snort_dynamicrules

chown -R 777 /usr/local/lib/snort_dynamicrules

✓ **Kiểm tra hoạt động của Snort IDS:**



Hình 22: Snort IDS đã hoạt động thành công

snort -T -i eth1 -u snort -g snort -c /etc/snort/snort.conf

Bật snort ids chạy nền:

snort -A fast -b -D -d -i eth1 -u snort -g snort -c /etc/snort/snort.conf -l /var/log/snort

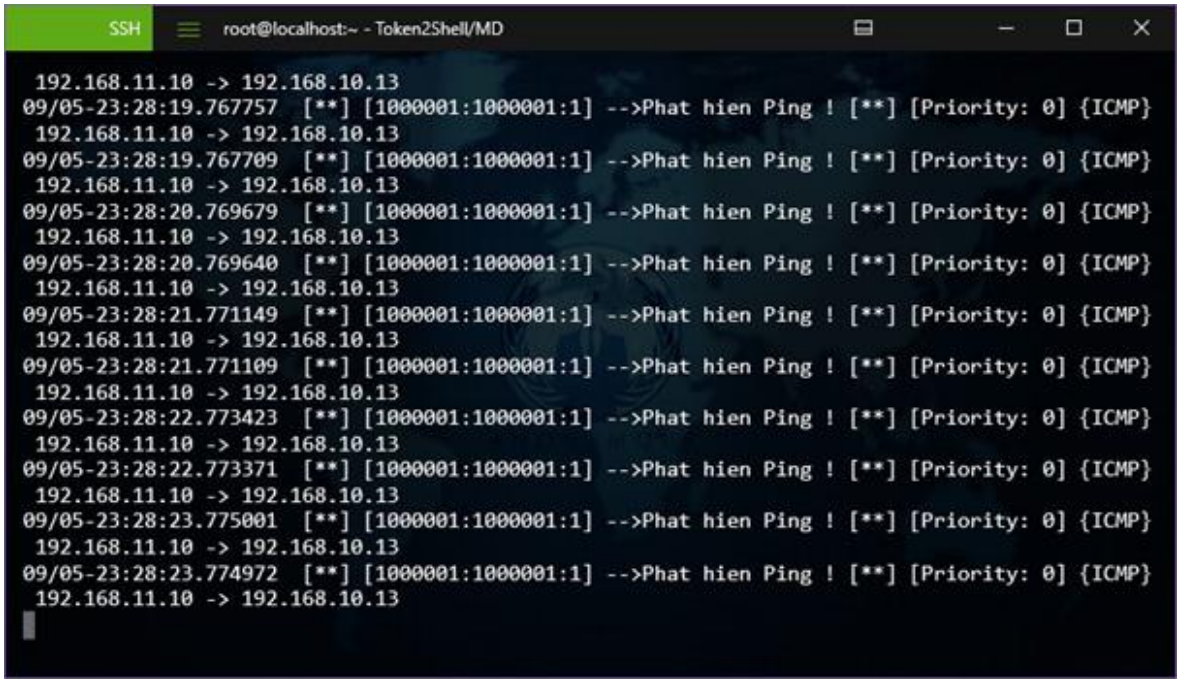
✓ **Thêm rule phát hiện ping để kiểm tra hoạt động của snort ids:**

vim /etc/snort/rules/local.rules

alert icmp any any -> $HOME_NET any (msg:"-->Phat hien Ping";gid:1000001 sid:1000001;rev:1;)

✓ **Xem cảnh báo phát hiện**

snort -c /etc/snort/snort.conf -i eth1 -A console



Hình 23: Snort IDS đã phát hiện Ping icmp từ địa chỉ nguồn 192.168.11.10 tới địa chỉ đích là 192.168.10.13

✓ **Một số Rules khác:**

alert icmp any any -> $HOME_NET 81 (msg:"Scanning Port 81"; sid:1000005;rev:1;)

alert tcp any any -> $HOME_NET 22 (msg:"Scanning Port 22"; sid:1000002;rev:1;)

alert icmp any any -> any any (msg:"UDP Tesing Rule"; sid:1000006;rev:1;)

alert tcp any any -> $HOME_NET 80 (msg:"HTTP Test!!!"; classtype:not-suspicious; sid:1000005; rev:1;)

✓ **Xem File Log cảnh báo snort:**

snort -A fast -b -D -d -i eth1 -u snort -g snort -c /etc/snort/snort.conf -l /var/log/snort

tail -f /var/log/snort/alert

3.3.2. Cài đặt snort IPS: snort inline mode

Mở cấu hình snort inline mode trong snort.conf:

vim /etc/snort/snort.conf

"## Under Step #2:" Thêm dòng sau:

config policy_mode:inline

Cấu hình giá trị biến DAQ để chạy **AFPacket** trong inline (IPS) mode:

"## Configure DAQ variables for AFPacket"

config daq: afpacket

config daq_mode: inline

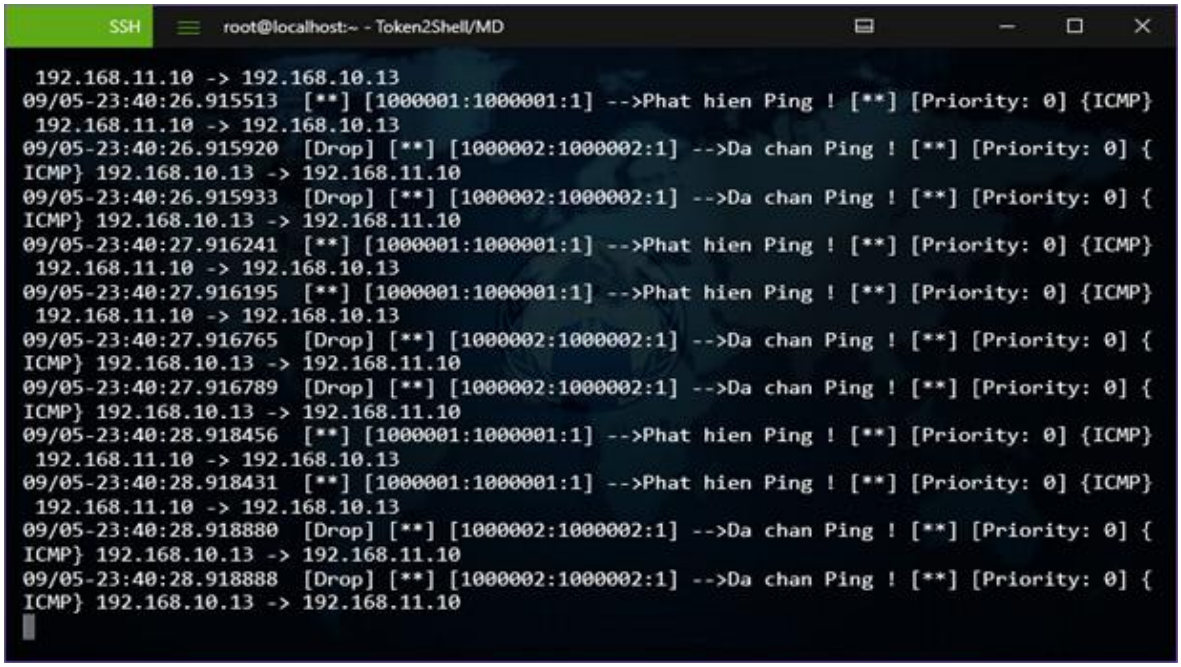
config daq_dir: /usr/local/lib/daq

config daq_var: buffer_size_mb=128

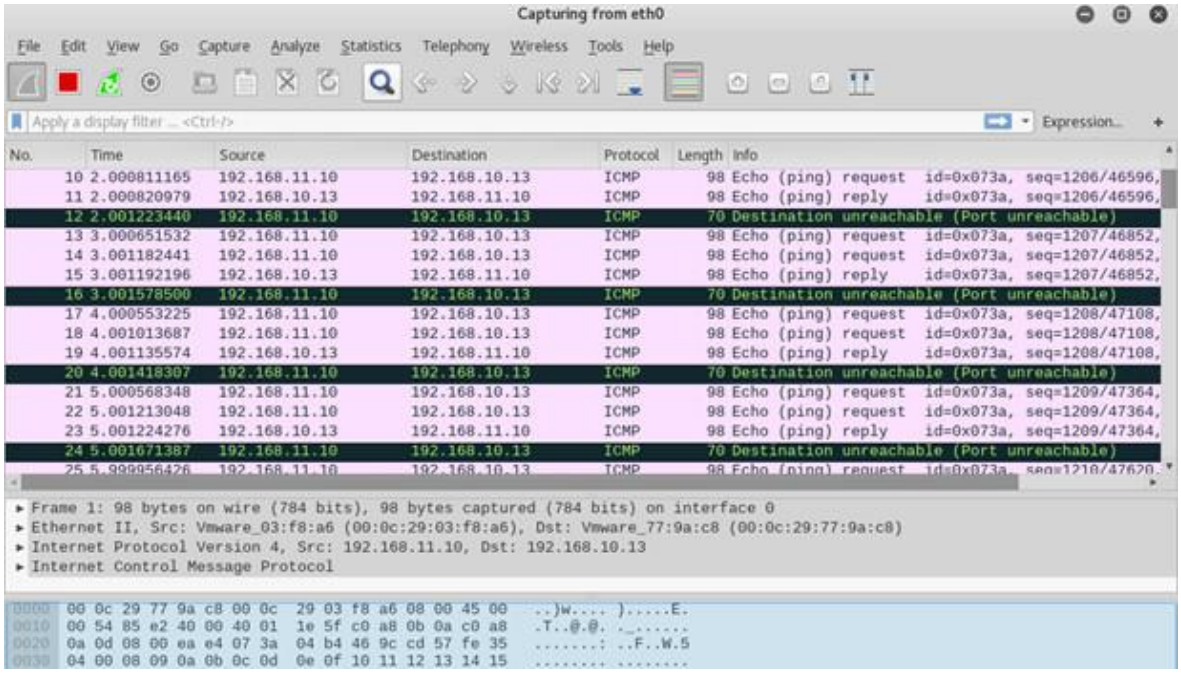
Lưu cấu hình. Thêm rule chặn ping và kiểm tra:

drop icmp any any -> any any (itype:0;msg:"-->Da chan Ping !";gid:1000002;sid:1000002;rev:1;)

snort -i eth1:eth2 -A console -c /etc/snort/snort.conf -l /var/log/snort/ -Q



Hình 24: Snort IPS đã phát hiện và chặn (DROP) các gói PING icmp từ Attacker



Hình 25: Các gói tin có giao thức ICMP đã bị chặn (drop) trên máy của Attacker

3.3.3. Cài đặt BASE quản lý phân tích Snort Log trên web

- BASE (Basic Analysis and Security Engine) cung cấp một trang web front-end để truy vấn và phân tích cảnh báo từ Snort. Các cảnh báo sẽ gửi đến một cơ sở dữ liệu MySQL, tính năng này được cung cấp bởi barnyard2 .
- Barnyard2 là một hệ thống đầu ra cho Snort, nó đọc các bản ghi nhị phân từ snort sử dụng định dạng unified2 và sau đó nó sẽ gửi lại các thông tin của bản ghi này tới cơ sở dữ liệu của user thiết lập trong mysql.

→Yêu cầu: đã cài sẵn PHP, Mysql, httpd

✓ **Cài đặt các gói yêu cầu cho BASE:**

pear channel-update pear.php.net

pear install Mail Mail_mime

pear install Numbers_Roman

pear install Image_Color-1.0.4

pear install Image_Canvas-0.3.5

pear install Image_Graph-0.8.0

✓ **Tạo cơ sở database mới cho snort:**

mysql -u root -p

mysql> create database snort;

mysql> grant select,insert,update,delete,create on snort. to snort@localhost;*

mysql> set password for snort@localhost=PASSWORD('123456');

mysql> flush privileges;

mysql> exit

✓ **Cấu hình định dạng đầu ra của snort log theo dạng unified2:**

vim /etc/snort/snort.conf

output unified2: filename snort.u2, limit 128

✓ **Cài đặt barnyard2:**

cd /root/ips

wget https://github.com/firnsy/barnyard2/archive/v2-1.13.tar.gz

tar -xzvf v2-1.13.tar.gz

cd barnyard2-2-1.13

autoreconf -fvi -I ./m4

./configure --with-mysql --with-mysql-libraries=/usr/lib64/mysql

make && make install

Tạo script cho barnyard2 chạy startup:

cp rpm/barnyard2 /etc/init.d/

chmod +x /etc/init.d/barnyard2

cp rpm/barnyard2.config /etc/sysconfig/barnyard2

Create links for Barnyard files and create archive directory:

ln -s /usr/local/etc/barnyard2.conf /etc/snort/barnyard.conf

ln -s /usr/local/bin/barnyard2 /usr/bin/

mkdir /var/log/barnyard2

chkconfig --add barnyard2

cp etc/barnyard2.conf /etc/snort/

mysql -u snort -p snort < schemas/create_mysql

chown snort:snort /var/log/barnyard2

touch /var/log/barnyard2/barnyard2.waldo

chown snort:snort /var/log/barnyard2/barnyard2.waldo

touch /etc/snort/rules/sid-msg.map

touch /etc/snort/rules/gen-msg.map

vim /etc/snort/barnyard2.conf

config reference_file: /etc/snort/reference.config

config classification_file: /etc/snort/classification.config

config gen_file: /etc/snort/rules/gen-msg.map

config sid_file: /etc/snort/rules/sid-msg.map

input unified2

config hostname: localhost

config interface: eth0

config alert_with_interface_name

output database: log, mysql, user=snort password=123456 dbname=snort host=localhost

 # vim /etc/init.d/barnyard2

...

 # chkconfig: 2345 70 60

...

BARNYARD_OPTS="-D -c $CONF -d $SNORTDIR/${INT} -w $WALDO_FILE -l $SNORTDIR/${INT} -a $ARCHIVEDIR -f $LOG_FILE -X $PIDFILE $EXTRA_ARGS"

...

Edit LOG_FILE variable in Barnyard sysconfig file:

vim /etc/sysconfig/barnyard2

LOG_FILE="snort.log"

service barnyard2 restart

Xem và kiểm tra:

/usr/local/bin/snort -q -u snort -g snort -c /etc/snort/snort.conf -i eth1:eth0 -D -A console

- ✓ **Khởi động lại snort:**

/etc/init.d/snortd restart

- ✓ **Cài đặt Adodb:**

cd /root/ips

wget http://jaist.dl.sourceforge.net/project/adodb/adodb-php5-only/adodb-520-for-php5/adodb-5.20.6.zip

unzip adodb-5.20.6.zip

mv adodb5 /var/www/adodb

- ✓ **Cài đặt BASE:**

cd /root/ips

wget http://nchc.dl.sourceforge.net/project/secureideas/BASE/base-1.4.5/base-1.4.5.tar.gz

mkdir /var/www/html/base

tar -xzvf /root/ips/base-1.4.5.tar.gz

cp -r base-1.4.5/* /var/www/html/base

chown -R snort:snort /var/www/html/base

cd /var/www/html/base

cp base_conf.php.dist base_conf.php

chmod 755 /var/www/html/base/base_conf.php

vim /var/www/html/base/base_conf.php

$BASE_urlpath = '/base';

$DBlib_path = '/var/adodb';

$alert_dbname = 'snort';

$alert_host = 'localhost';

$alert_port = '3306';

$alert_user = 'snort';

$alert_password = '123456';

 # chmod 777 /var/www/html/base

✓ **Cấu hình Apache:**

vim /etc/httpd/conf/httpd.conf

Alias /base /var/www/html/base/

<Directory "/var/www/html/base/">

AllowOverride None

Order allow,deny

Allow from all

</directory>

Alias /adodb/ "/var/adodb/"

<Directory "/var/adodb">

AllowOverride None

Order allow,deny

Allow from all

</Directory>

vim /etc/httpd/conf.d/base.conf

Alias /base /var/www/html/base/

<directory "/var/www/base/">

AllowOverride None

Order allow,deny

Allow from all

AuthName "Snort IDS"

AuthType Basic

AuthUserFile /etc/snort/base.passwd

Require valid-user

</directory>

✓ **Tạo file khẩu để truy cập web cho database:**

htpasswd -c /etc/snort/base.passwd snortadmin

service httpd restart

chcon -R -t httpd_sys_content_t /var/www/html/base/

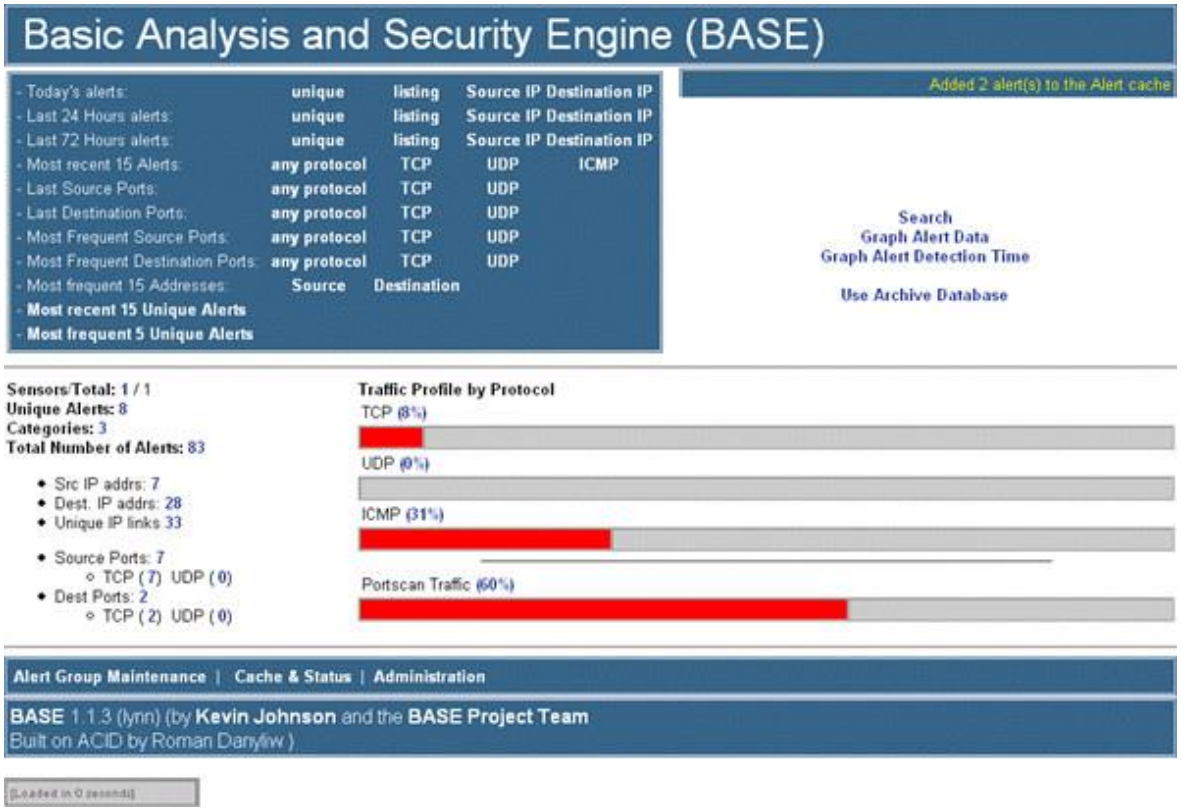
chcon -R -h -t httpd_sys_content_t /var/www/adodb

vim /var/www/html/base/base_main.php

date_default_timezone_set('Asia/Ho_Chi_Minh');

→Truy cập giao diện web để cài đặt base và quản lý:

http://192.168.0.100/base/base_db_setup.php



Hình 26: Giao diện chính của BASE trên web

3.3.4. Cài đặt ELK Stack

Server cài đặt ở đây thực hiện trên CentOS 8, trước tiên cần đảm bảo cài đặt Java (openjdk)

yum update -y

yum install java-1.8.0-openjdk-devel -y

Trước khi tiến hành cài đặt các thành phần của Elastic Stack, ta cần tải xuống và cài đặt khóa công khai của Elastic

sudo rpm --import https://artifacts.elastic.co/GPG-KEY-elasticsearch

Thao tác này sẽ thêm khóa ký công khai Elasticsearch vào hệ thống của bạn. Khóa này sẽ xác thực phần mềm Elasticsearch khi bạn tải xuống.

Thêm kho lưu trữ Elasticsearch bằng cách tạo tệp cấu hình kho lưu trữ:

elasticsearch.repo trong thư mục /etc/yum.rpos.d/

cd /etc/yum.repos.d/

sudo vim

elasticsearch.repo #

nhập nội dung sau

Hình 27: Cấu hình kho lưu trữ RPM Elasticsearch

Sau đó cập nhật các gói trong kho lưu trữ bằng lệnh

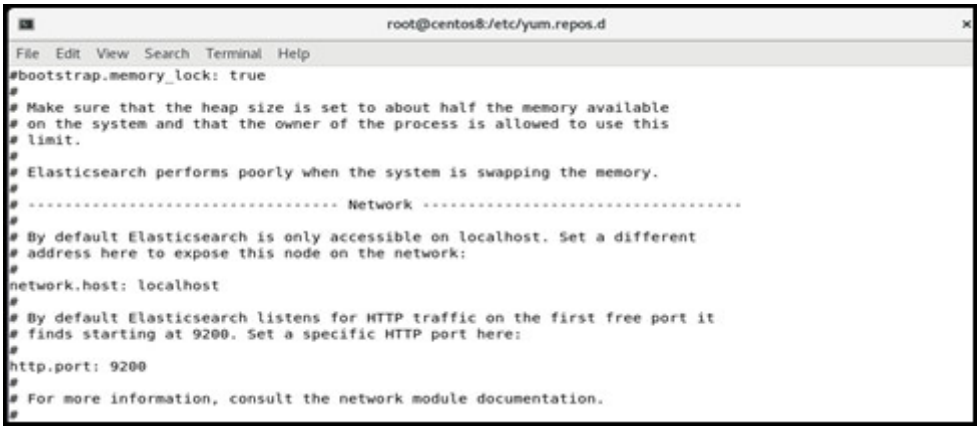
dnf update

Tiếp theo cài đặt và thiết lập Elasticsearch

sudo dnf install elasticsearch

Sau khi quá trình cài đặt kết thúc, mở và chỉnh sửa địa chỉ ip máy chủ trong file **/etc/elasticsearch/elasticsearch.yml**

sudo vim /etc/elasticsearch/elasticsearch.yml



Hình 28: Cấu hình địa chỉ Ip máy chủ cho Elasticsearch

Khởi động lại hệ thống, sau đó khởi động dịch vụ elasticsearch đồng thời kích hoạt tính năng dịch vụ elasticsearch khi hệ thống khởi động

sudo systemctl start

elasticsearch sudo systemctl

enable elasticsearch

Tiếp theo, ta cài đặt và thiết lập Kibana

sudo dnf install kibana

Thiết lập cấu hình cho Kibana trong file kibana.yml tại thư mục

/etc/kibana/

Khởi động và kích hoạt Kibana

sudo systemctl start kibana

sudo systemctl enable kibana

3.3.5. Cài đặt Filebeat

Cài đặt modul Filebeat

yum install filebeat

filebeat modules enable

system filebeat setup

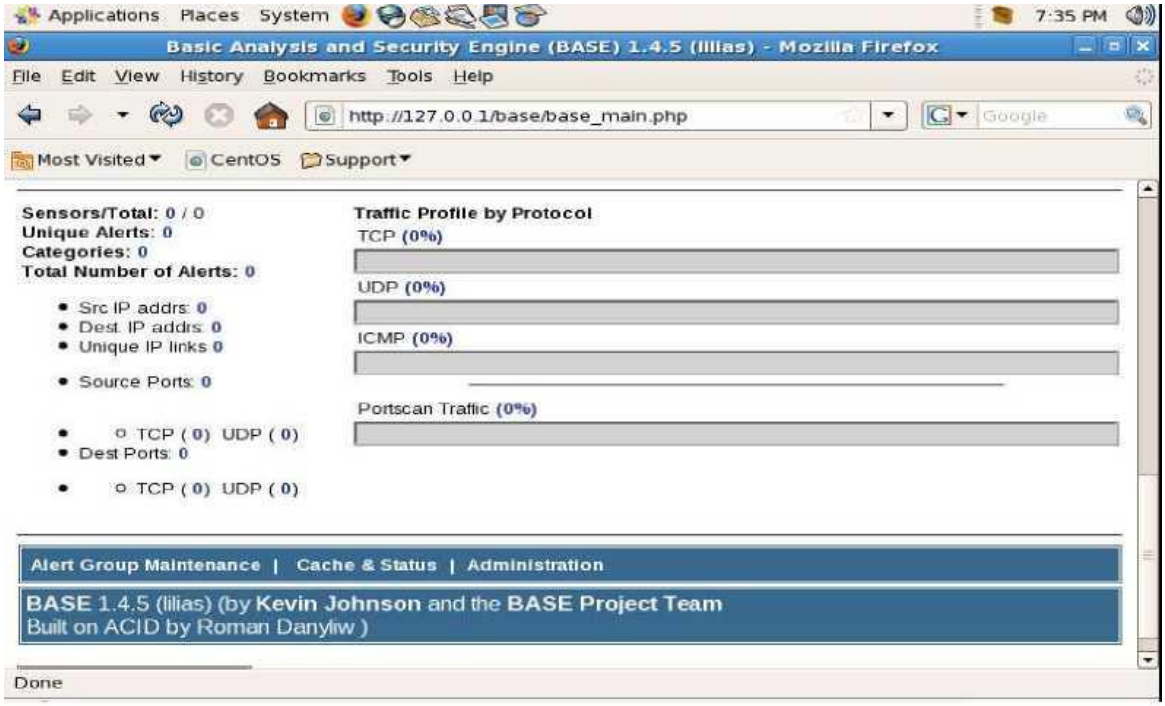
service filebeat start

Theo mặc định, Filebeat ghi trực tiếp file vào Elasticsearch. Để tùy chỉnh Filebeat, hãy chỉnh sửa file cấu hình /etc/filebeat/filebeat.yml.

Sau khi cài đặt xong Filebeat khởi chạy modul Suricata filebeat modules enable suricata

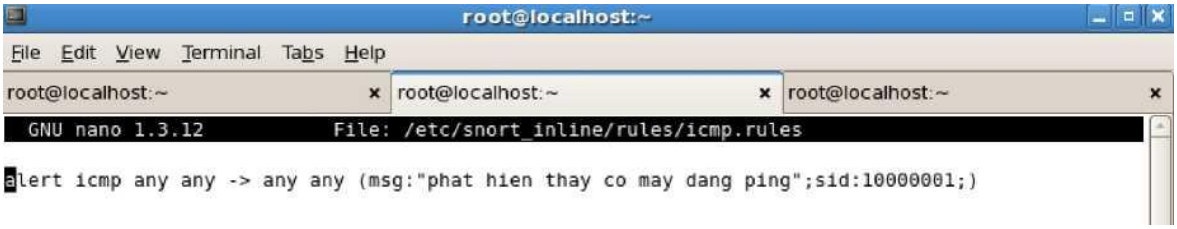
3.4. THỬ NGHIỆM KHẢ NĂNG PHẢN ỨNG CỦA SNORT IDS/IPS

Truy cập vào base http://127.0.0.1/base



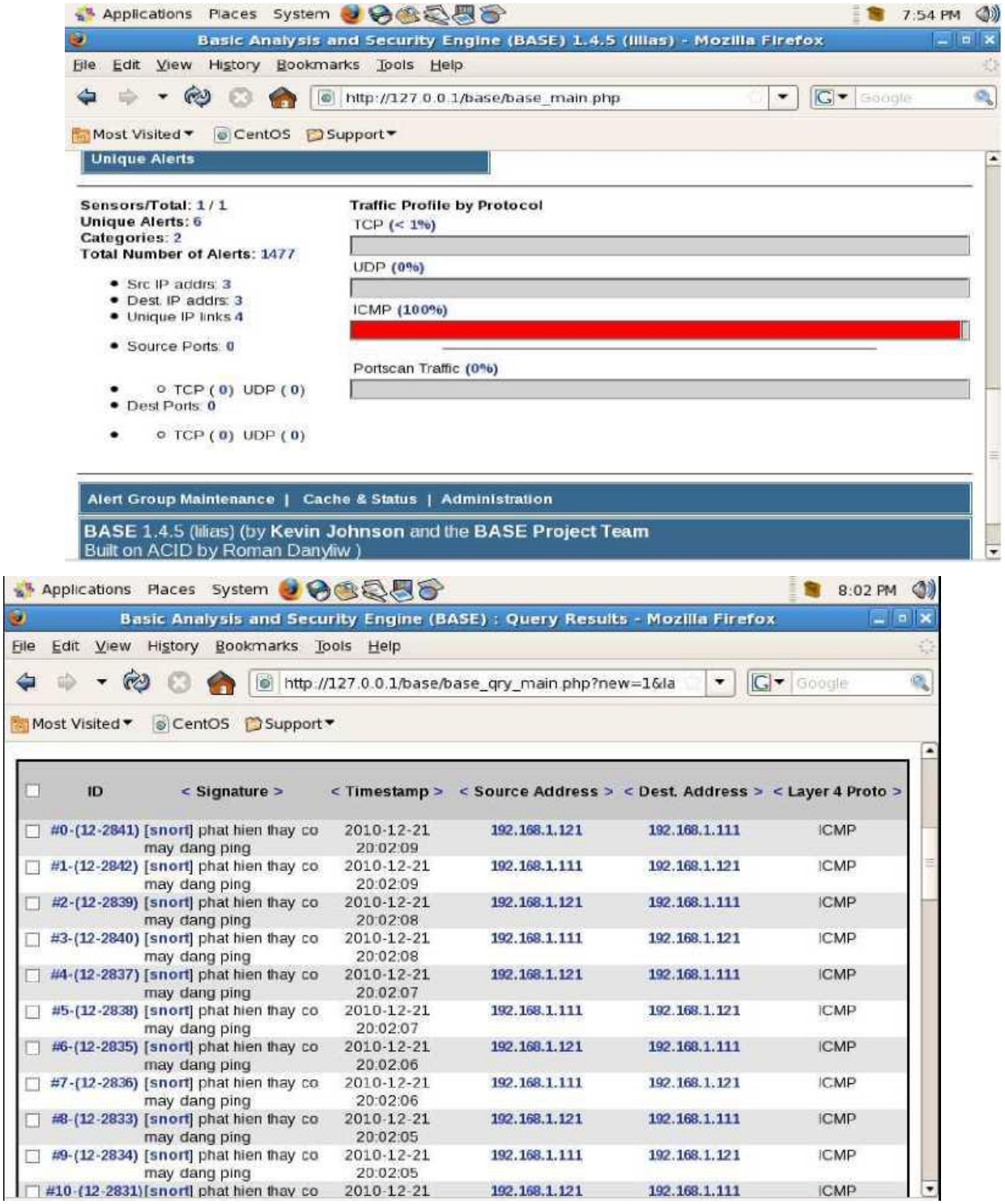
Hình 29: Truy cập vào base

Lúc này chưa có cảnh báo nào vì ta chưa khởi chạy snort. giả sử ta tạo một rules với dấu hiệu như sau:



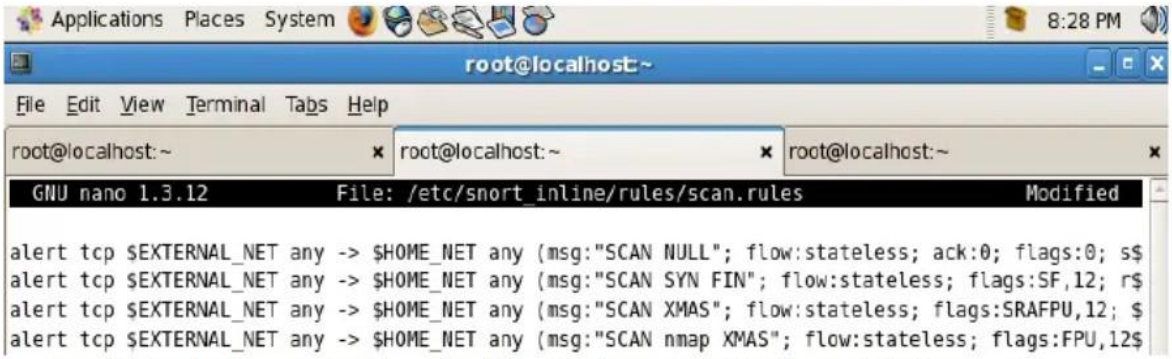
Hình 30: Tạo rules với dấu hiệu

Sau đó include nó vào file /etc/snort_inline/snort_inline.conf và khởi chạy snort: # snort_inline -c /etc/snort_inline/snort_inline.conf -Q rồi từ một máy khác ping đen với địa chỉ của máy ping là 192.168.1.121 và địa chỉ của máy IDS là 192.168.1.111 ta có kết quả sau.



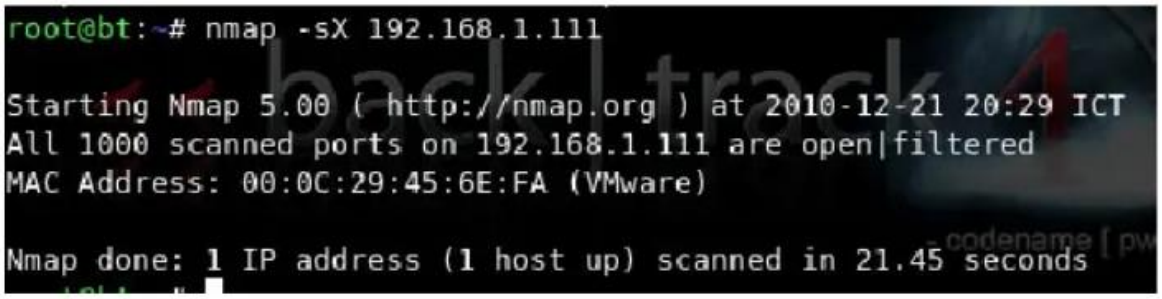
Hình 31: Kết quả khởi chạy Snort

Như vậy snort IDS đã hoạt động tốt, ta thử rules sau cho trường hợp phát hiện nmap scan cổng.

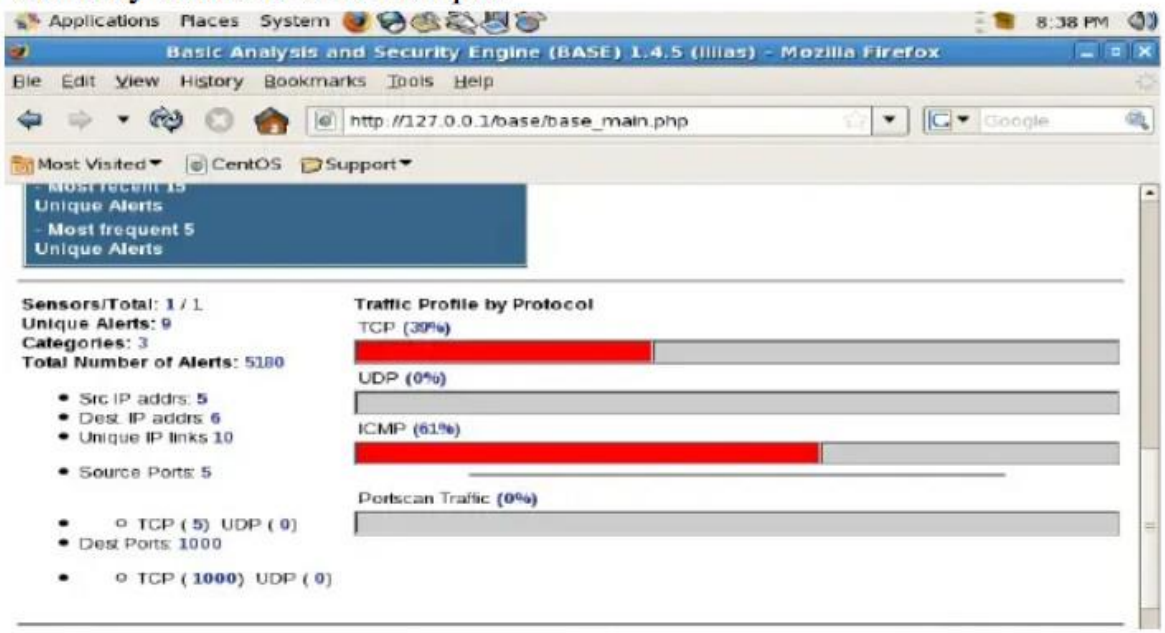


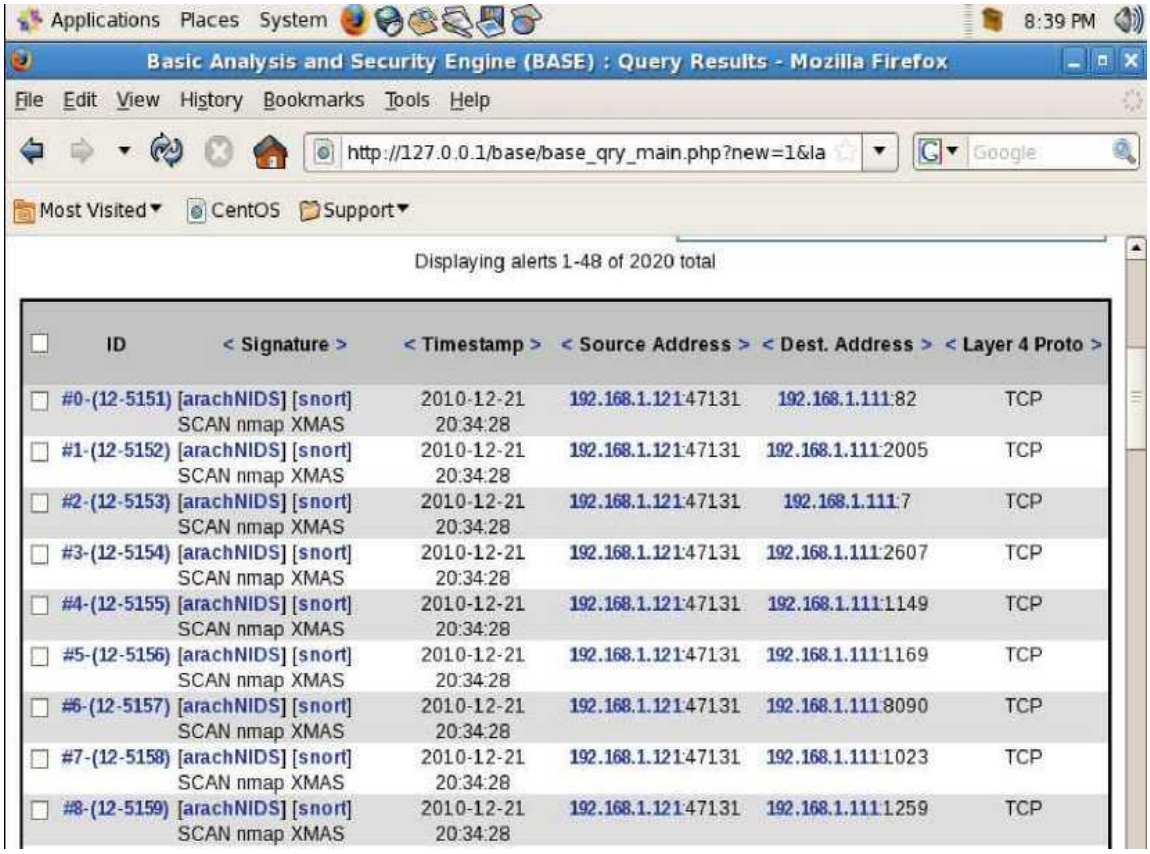
sau đó include scan.rules vào file /etc/snort_inline/snort_inline.conf

Khởi động lại snort_inline. từ máy tấn công bật nmap và scan cổng ta nhận được kết quả. như vậy snort đã thể hiên là một IPS



vào mày snort và xem kết quả.





3.5. MỘT SỐ PHƯƠNG THỨC TẤN CÔNG VÀ CÁCH PHÒNG CHỐNG

3.5.1. ARP Spoofing Attack

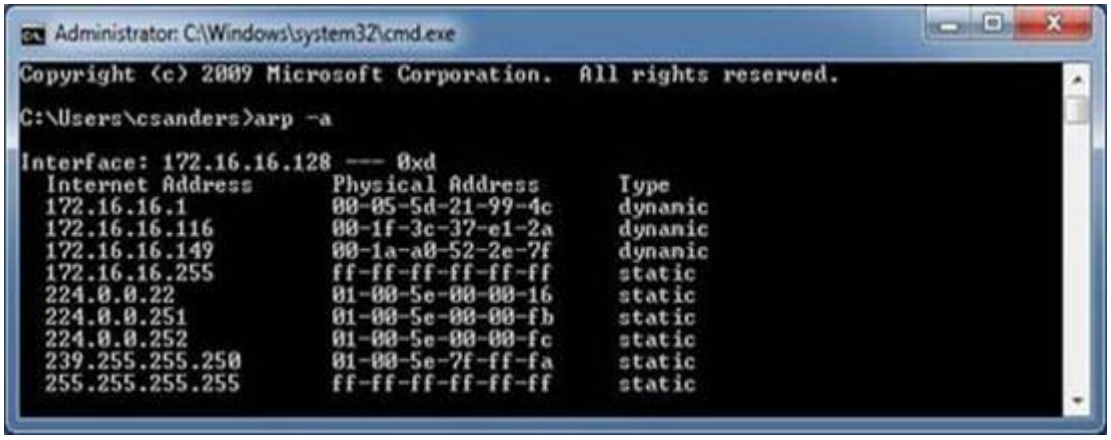
Đây là một hình thức tấn công Man in the middle (MITM) hiện đại có xuất sứ lâu đời nhất (đôi khi còn được biết đến với cái tên ARP Poison Routing), tấn công này cho phép kẻ tấn công nằm trên cùng một subnet với các nạn nhân của nó có thể nghe trộm tất cả các lưu lượng mạng giữa các máy tính nạn nhân. Đây là loại tấn công đơn giản nhất nhưng lại là một hình thức hiệu quả nhất khi được thực hiện bởi kẻ tấn công.

Khi bị tấn công các tín hiệu cảnh báo sẽ được ghi trong file log của Snort; sau đó Filebeat sẽ chuyển log sang ELK server để phân tích qua đó người quản trị có thể nhận biết dễ dàng

Cách phòng chống: Mã hóa ARP cache

Một cách có thể bảo vệ chống lại vấn đề không an toàn vốn có trong các ARP request và ARP reply là thực hiện một quá trình kém động hơn. Đây là một tùy

chọn vì các máy tính Windows cho phép bạn có thể bổ sung các entry tĩnh vào ARP cache. Bạn có thể xem ARP cache của máy tính Windows bằng cách mở Commad Prompt và gõ lệnh arp –a.



Hình 32: Xem ARP Cache

Có thể thêm các entry vào danh sách này bằng cách sử dụng lệnh:

arp –s <IP ADDRESS> <MAC ADDRESS>.

Trong các trường hợp, nơi cấu hình mạng của bạn không mấy khi thay đổi, bạn hoàn toàn có thể tạo một danh sách các entry ARP tĩnh và sử dụng chúng cho các client thông qua một kịch bản tự động. Điều này sẽ bảo đảm được các thiết bị sẽ luôn dựa vào ARP cache nội bộ của chúng thay vì các ARP request và ARP reply.

3.5.2. SYN Flood Attack

Syn flood là 1 dạng tấn công từ chối dịch vụ, kẻ tấn công gửi các gói tin kết nối SYN đến hệ thống. Đây là 1 loại tấn công rất phổ biến. Loại tấn công này sẽ nguy hiểm nếu hệ thống cấp phát tài nguyên ngay sau khi nhận gói tin SYN từ kẻ tấn công và trước khi nhận gói ACK.

Khi bị tấn công các tín hiệu cảnh báo sẽ được ghi trong file log của Snort; sau đó Filebeat sẽ chuyển log sang ELK server để phân tích qua đó người quản trị có thể nhận biết dễ dàng

Cách phòng chống: Sử dụng Iptables hoặc Snort IPS

Sử dụng Iptables:

 # iptables -A INPUT –p tcp --syn –m limit --limit 1/s --limit -burst 3 -j RETURN

Tất cả các kết nối đến hệ thống chỉ được phép theo các thông số giới hạn sau:

- **--limit 1/s**: Tốc độ truyền gói tin trung bình tối đa 1/s (giây)
- **--limit-burst 3: Số lương gói tin khởi tạo tối đa được phép là 3**

Sử dụng Snort IPS thêm rule sau:

dropt tcp any any -> $HOME_NET any (msg:"-->Da chan SYN FIN Attack ! "; flags: S;gid: 2000001;sid:2000001;)

3.5.3. Zero Day Attack

Zero-day là thuật ngữ chỉ sự tấn công hay các mối đe dọa khai thác lỗ hổng của ứng dụng trong máy tính cái mà chưa được công bố và chưa được sửa chữa.

"Windows Vista/7:SMB2.0 NEGOTIATE PROTOCOL REQUEST Remote B.S.O.D." là nguyên văn tiêu đề mô tả mã tấn công viết bằng Python mà Gaffie đưa lên blog bảo mật Seclists.org. Cuộc tấn công nhằm vào lỗi xuất phát từ System Message Block phiên bản 2.0 (SMB2) vốn có trong Windows Vista, Windows 7 và Windows Server 2008. Đi sâu vào lỗi do Gaffie công bố, nguyên nhân chính xuất phát từ cách thức driver srv2.sys xử lý các yêu cầu từ máy khách trong khi phần tiêu đề (header) của ô "Process Id High" chứa đựng một ký tự "&"(mã hexa là 00 26). Cuộc tấn công không cần đến chứng thực nhận dạng, chỉ cần cổng 445 có thể truy xuất. Mối lo ngại ở đây là cổng 445 thường được mở mặc định trong phần cấu hình mạng nội bộ (LAN) của Windows.

Khi bị tấn công các tín hiệu cảnh báo sẽ được ghi trong file log của Snort; sau đó Filebeat sẽ chuyển log sang ELK server để phân tích qua đó người quản trị có thể nhận biết dễ dàng

Cách phòng chống:

- Cập nhật bản vá lỗi.
- Lọc dữ liệu từ cổng TCP 445 bằng tường lửa (iptables).
- Khóa cổng SMB trong registry.

3.5.4. DOS - Ping Of Death Attack

Khi tấn công bằng Ping of Death, một gói tin echo đực gửi có kích thước lớn hơn kích thước cho phép là 65,536 bytes. Gói tin sẽ bị chia nhỏ ra thành các segment nhỏ hơn, nhưng khi máy đích ráp lại, host đích nhận thấy rằng là gói tin quá lớn đối với buffer bên nhận. Kết quả là, hệ thống không thể quản lý nổi tình trạng bất thường này và sẽ reboot hoặc bị treo.

VD : *ping 192.168.10.13 –l 65000*

Khi bị tấn công các tín hiệu cảnh báo sẽ được ghi trong file log của Snort; sau đó Filebeat sẽ chuyển log sang ELK server để phân tích qua đó người quản trị có thể nhận biết dễ dàng

Cách phòng chống:

Sử dụng các tính năng cho phép đặt rate limit trên router/firewall để hạn chế số lượng packet vào hệ thống.

Dùng tính năng lọc dữ liệu của router/firewall để loại bỏ các packet không mong muốn, giảm lượng lưu thông trên mạng và tải của máy chủ.

Sử dụng Snort IPS và thêm rule:

drop icmp any any -> $HOME_NET any (msg:"-->Da chan Ping Of Dead !"; dsize:>20000; gid:1000002; sid:1000002;rev:1;)

Kết luận chương

Chương 3 đã đưa ra mô hình thử nghiệm giải pháp phát hiện xâm nhập, đồng thời cũng trình bày chi tiết các cài đặt, thiết lập hệ thống phát hiện xâm nhập và tấn công mạng dựa trên nền tảng mã nguồn mở Snort và ELK. Dựa trên các kết quả cho thấy hệ thống hoạt động ổn định, có khả năng giám sát và phát hiện các bất thường và nguy cơ an ninh an toàn thông tin; đồng thời hệ thống cũng hỗ trợ các tính năng quản trị và hiển thị dữ liệu đa dạng, tiện dụng cho người dùng cuối.

Ngoài việc ưu tiên bảo mật , phát hiện xâm nhập mạng bằng cách sử dụng các phần mềm Snort, E.L.K chúng ta có thể áp dụng thêm các phương án bảo mật thông tin song song như: đảm bảo chính sách bảo vệ và quyền riêng tư phù hợp trong hệ sinh thái số của mình, thiết lập và duy trì các chính sách vệ sinh an ninh

mạng, đảm bảo cài đặt các bản vá và cập nhật mới nhất cho thiết bị, mật khẩu là chưa đủ, nên lấy xác thực đa yếu tố (MFA) làm tiêu chuẩn mới, sử dụng Zero Trust (liên tục xác thực người dùng và thiết bị (thay vì chỉ một lần), mã hóa mọi nội dung, cấp quyền truy nhập tối thiểu cần thiết và giới hạn thời lượng truy nhập, đồng thời sử dụng phân đoạn để hạn chế thiệt hại do bất kỳ vi phạm nào), tăng cường bảo mật làm việc từ xa, sử dụng thêm những dịch vụ bảo mật công nghệ thông tin bên ngoài một cách hiệu quả.

KẾT LUẬN

Những đóng góp của luận văn:

Luận văn này cung cấp một số thông tin hữu ích cho các nhà quản trị hệ thống trong việc hoàn thiện, nâng cấp an toàn thông tin trong các hệ thống mạng tại Học viện Thanh thiếu niên Việt Nam bằng cách sử dụng phần mềm Snort và ELK.

Các nhà lãnh đạo có thể tham khảo mô hình các phần mềm hệ thống IDS hoặc IPS để điều chỉnh công tác an toàn thông tin trong các hệ thống mạng tại đơn vị, tổ chức của mình.

Nhìn chung, việc sử dụng các hệ thống phát hiện xâm nhập Snort và ELK tương đối có giá trị và đạt độ tin cậy, có thể sử dụng làm nền tảng để tham khảo cho các nghiên cứu khác liên quan.

Các kết quả đạt được:

Luận văn này tập trung nghiên cứu về thu thập, xử lý, phân tích log truy cập, phục vụ phát hiện các hành vi bất thường và nguy cơ mất an toàn thông tin trong các hệ thống mạng. Các nội dung đã thực hiện trong luận văn bao gồm:

- Trình tổng quan về tấn công, xâm nhập mạng, các dạng tấn công xâm nhập thường gặp; đồng thời cũng đã khái quát về phát hiện xâm nhập cũng như các kỹ thuật phát hiện xâm nhập qua đó ta có cái nhìn rõ hơn về tấn công, xâm nhập.
- Mô tả một số nền tảng về công cụ xử lý và phân tích log truy nhập, từ đó rút ra so sánh, đánh giá để tìm ra mô hình triển khai phù hợp.
- Trình bày kiến trúc, hoạt động tính năng của các hệ thống xâm nhập mạng phổ biến hiện nay là Snort, Suricata và Zeek. Từ đó, đưa ra so sánh về ưu, nhược điểm của các hệ thống này. Giới thiệu các hệ thống phát hiện xâm nhập tích hợp IBM QRadar và bộ công cụ quản lý log Elastic Stack.
- Cài đặt hệ thống phát hiện tấn công, xâm nhập mạng với nền tảng mã nguồn mở như Snort và ELK. Thử nghiệm các kịch bản tấn công cụ thể và đưa ra kết quả cảnh báo khi bị tấn công.

Hướng phát triển:

Luận văn có thể được phát triển theo các hướng sau:

- Triển khai thử nghiệm mô hình phát hiện tấn công, xâm nhập mạng dựa trên Snort kết hợp ELK cho phát hiện bất thường và các nguy cơ ATTT trên hệ thống mạng thực. Hoàn thiện tối ưu hóa hệ thống để có thể phân tích xử lý logs nhanh chóng và tăng hiệu năng của hệ thống.
- Xây dựng và bổ sung thêm các tập luật giám sát, phát hiện bất thường và các nguy cơ ATTT, đảm bảo khả năng phát hiện kịp thời các nguy cơ mất an ninh an toàn; đồng thời có thể tích hợp thêm hệ thống ngăn chặn và tấn công mạng, tích hợp tính năng cảnh báo qua mail, telegram…

TÀI LIỆU THAM KHẢO

[1] NortonLifeLock Inc, "10 cyber security facts and statistics for 2018," 2018. https://us.norton.com/internetsecurity-emerging-threats-10-facts-about- todays-cybersecurity-landscape-that-you-should-know.html (accessed Feb. 03, 2021).

[2] CSO, "Top cybersecurity facts, figures and statistics for 2020." https://www.csoonline.com/article/3153707/top-cybersecurity-facts- figures-and-statistics.html, truy nhập tháng 3.2021.

[3] H. X. Dậu and N. T. T. Thủy, "Bài giảng Cơ Sở An Toàn Thông Tin," 2016.

[4] https://www.fortinet.com/it/resources/cyberglossary/, truy nhập tháng 5.2021.

[5] https://resources.cystack.net/, truy nhập tháng 5.2021.

[6] https://www.kaspersky.com/resource-center/definitions/what-is-cyber- security, truy nhập tháng 5.2021.

[7] https://www.upguard.com/blog/cyber-attack, truy nhập tháng 5.2021.

[8] https://willandway.vn/cac-hinh-thuc-tan-cong-mang-pho-bien/, truy nhập tháng 5.2021.

[9] https://portswigger.net/web-security/, truy nhập tháng 5.2021.

[10] Snort Team, "'Snort User Manual', 08/04/2020." https://snort-org-site.s3.amazonaws.com/production/document_files/files/000/000/249/origi nal/snort_manual.pdf, truy nhập tháng 5.2021.

[11] https://docs.zeek.org/en/master/about.html, truy nhập tháng 5.2021.

[12] https://www.ibm.com/docs/en/qsip/7.4?topic=deployment-qradar- architecture-overview, truy nhập tháng 5.2021.

[13] https://www.guru99.com/elk-stack-tutorial.html, truy nhập tháng 5.2021.