**BỘ THÔNG TIN VÀ TRUYỀN THÔNG**

**HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG**

------------------------------



**BÁO CÁO**

**AN TOÀN MẠNG**

***Đề tài*:**

**SNORT**

Giáo viên:

Lớp :D17CQAT01\_N

Sinh viên 1: Võ Minh Thuận N17DCAT070

Sinh viên 2: Vũ Tuấn Dũng N17DCAT010

Sinh viên 3: Đoàn Quang Huy N17DCAT033

**TP.HỒ CHÍ MINH – 2020**

**MỤC LỤC**

[MỤC LỤC i](#_Toc58576943)

[DANH MỤC VIẾT TẮT ii](#_Toc58576944)

[DANH MỤC BẢNG ii](#_Toc58576945)

[DANH MỤC HÌNH ii](#_Toc58576946)

[GIỚI THIỆU iii](#_Toc58576947)

[CHƯƠNG 1: GIỚI THIỆU 1](#_Toc58576948)

[1.1 TỔNG QUAN VỀ SNORT 1](#_Toc58576949)

[1.2 THÀNH PHẦN 1](#_Toc58576950)

[1.2.1 Packet Decoder 2](#_Toc58576951)

[1.2.2 Preprocessors 2](#_Toc58576952)

[1.2.3 Detection Engine 2](#_Toc58576953)

[1.2.4 Logging and Alerting System 2](#_Toc58576954)

[1.2.5 Output Modules 2](#_Toc58576955)

[1.3 HOẠT ĐỘNG 3](#_Toc58576956)

[1.3.1. Sniffer 3](#_Toc58576960)

[1.3.2. Packet Logger 4](#_Toc58576961)

[1.3.3. NIDS 4](#_Toc58576962)

[1.3.3.1. Cấu trúc Rules 4](#_Toc58576963)

[CHƯƠNG 2: TRIỂN KHAI 6](#_Toc58576964)

[**2.1** **MÔ HÌNH TRIỂN KHAI:** 6](#_Toc58576965)

[**2.2** **HỆ THỐNG THỰC HIỆN:** 6](#_Toc58576966)

[**2.3** **CÀI ĐẶT:** 6](#_Toc58576967)

[2.3.1 Cài đặt DVWA 6](#_Toc58576968)

[2.3.2 Cài đặt Snort 11](#_Toc58576969)

[**2.4** **CẤU HÌNH** 19](#_Toc58576970)

[2.4.1 Cấu hình máy Snort\_Router (Ubuntu) 19](#_Toc58576971)

[2.4.2 Cấu hình máy Victim (Ubuntu) 21](#_Toc58576972)

[2.4.3 Cấu hình máy Attacker ( Kali linux) 22](#_Toc58576973)

[**2.5** **TẤN CÔNG THỰC NGHIỆM** 23](#_Toc58576974)

[TÀI LIỆU THAM KHẢO 24](#_Toc58576975)

**DANH MỤC VIẾT TẮT**

IDS: Intrusion Detection System Hệ thống phát hiện xâm nhập

NIDS: Network Intrusion Detection System Hệ thống phát hiện xâm nhập mạng

IPS: Intrusion Prevention System Hệ thống ngăn chặn xâm nhập

SNMP: Simple Network Management Protocol Giao thức quản lý mạng đơn giản

SMB: Server Message Block Hệ thống ngăn chặn xâm nhập

IPS: Intrusion Prevention System Hệ thống ngăn chặn xâm nhập

**DANH MỤC BẢNG**

[Bảng 1. 1: Thông tin thiết lập ip 24](#_Toc58571131)

**DANH MỤC HÌNH**

[[Hình 1. 1: Lưu đồ hoạt động của snort 7](#_Toc54599862)](#_Toc56689183)

**GIỚI THIỆU**

Ngày nay, với sự phát triển của công nghệ 4.0 - toàn cầu hóa internet, thì sự đóng góp của hệ thống phát hiện, ngăn chặn xâm nhập (IDS/IPS) mạng là không thể thiếu.

Nhiều hãng đã cho ra các IDS/IPS bao gồm cả phần cứng và phần mềm, tính phí và miễn phí. Ví dụ như Bro, OSSEC, Snort, Suricata,v.v.

Trong phạm vi báo cáo, chúng tôi sẽ tìm hiểu về Snort. Snort là một phần mềm mã nguồn mở và miễn phí đối với người dùng, bao gồm nhiều tính năng vượt trội, phục vụ cho việc phát hiện xâm nhập và ngăn chặn từ những cuộc tấn công có mục đích.

**CHƯƠNG 1: GIỚI THIỆU**

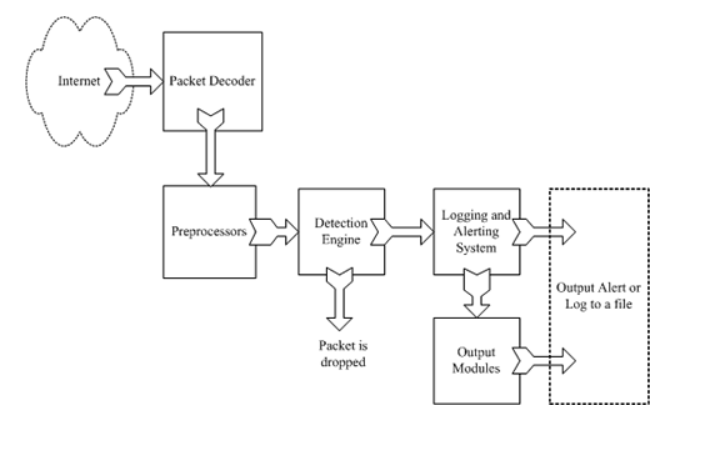
* 1. TỔNG QUAN VỀ SNORT

Snort là một phần mềm mã nguồn mở miễn phí được phát triến bởi Martin Roesch [1] vào năm 1998, với hoạt động chính như một hệ thống phát hiện xâm nhập mạng (NIDS), nó có khả năng phát hiện các cuộc tấn công mạng và ngăn chăn chúng dựa vào các quy tắc được đặt trước, bên cạnh nó cũng có thêm 2 chức năng như sniffer – đánh hơi bắt gói, và ghi log của các gói tin đi qua.

* 1. THÀNH PHẦN

Bao gồm 5 thành phần chính [2]:

* Packet Decoder
* Preprocessors
* Detection Engine
* Logging and Alerting System
* Output Modules



Hình 1. 1: Lưu đồ hoạt động của snort

* + 1. Packet Decoder

Bộ giải mã gói, các gói tin từ Internet đi vào mạng sẽ đi qua bộ giải mã gói, bộ giải mã gói sẽ bắt gói từ những loại giao tiếp mạng khác nhau, giải mã để chuẩn bị cho quá trình preprocessors. Các loại giao tiếp có thể là Ethernet, SLIP, PPP và v.v.

* + 1. Preprocessors

Thực hiện tìm kiếm và sắp xếp hoặc sửa đổi dữ liệu trong các gói tin để chuẩn bị cho giai đoạn tiếp theo. Nó cũng phát hiện những bất thường trong header của các gói tin và sinh ra cảnh báo.

Giai đoạn này cũng được sử dụng để chống việc phân mảnh các gói tin gây khó khăn cho quá trình nhận dạng tấn công, giải mã các URL HTTP mà được hacker chèn các kí tự tương đương với kí tự thực thi, v.v.

* + 1. Detection Engine

Chịu trách nhiệm phát hiện các hành động xâm nhập trong các gói tin, bằng việc so khớp các quy tắc được định nghĩa trước đó, nếu việc so khớp thành công, một hoạt động thích hợp từ phần mềm sẽ được thực hiện, như ghi lại các gói hoặc sinh ra cảnh báo, hoặc loại bỏ gói.

* + 1. Logging and Alerting System

Thành phần này sẽ phụ thuộc vào những gì mà Detection engine tìm thấy bên trong một gói tin, mới có thể đưa ra được nên ghi log hay là sinh ra cảnh báo.

Với định dạng log giống tcpdump và mặc định được lưu trong thư mục /var/log/snort.

* + 1. Output Modules

Output Modules sẽ hoạt động khác nhau theo nhiều hướng phụ thuộc vào cách chúng ta muốn lưu đầu ra như nào. Về cơ bản, nó có thể thực hiện những hoạt động như sau:

* Ghi lại log tại vị trí mặc định /var/log/snort/alerts hoặc vị trí khác nếu mình muốn.
* Gửi SNMP traps tới máy chủ quản lý, để cảnh báo
* Gửi thông điệp đến hệ thống quản lý log tập trung
* Ghi log đến cơ sở dữ liệu như MySQL và Oracle
* Tạo đầu ra XML
* Sửa đổi cấu hình trên các router và tường lửa
* Gửi thông điệp SMB đến máy chủ Windows
  1. HOẠT ĐỘNG

Snort có khả năng thực hiện phân tích lưu lượng mạng và ghi lại và lưu trữ các gói bắt được trên mạng. Và khả năng nổi trội của nó là phát hiện những hành động mạng bất thường và cảnh báo tới người dùng, đồng thời ngăn chặn lưu lượng độc hại từ những cuộc tấn công, ví dụ như DDoS, PortScan, Buffer Overflow, v.v.

Hoạt động 3 chế độ chính:

2. Sniffer

Đọc và hiện thị các gói tin bắt được trên mạng và hiển thị chúng lên màn hình người dùng. Với các thông tin cơ bản như sau:

* Ngày và giờ
* Địa chỉ IP nguồn
* Địa chỉ IP đích
* Port nguồn
* Port đích
* Giao thức vận chuyển (TCP/UDP)
* Thời gian sống của gói tin
* Loại dịch vụ
* Định danh của gói
* Độ dài IP header
* IP payload
* Bit phân đoạn hoặc không phân đoạn được đặt trong IP header
* Trạng thái cờ TCP A và P
* TCP sequence number
* Acknowledgement number trong TCP header
* TCP Window field
* TCP header length

1. Packet Logger

Ghi lại các gói tin bắt được vào đĩa để lưu trữ

1. NIDS

Giám sát, phân tích lưu lượng mạng dựa vào các quy tắc được định nghĩa từ người dùng, và sau đó thực hiện những công việc cụ thể đã được đặt trước như gửi cảnh báo hoặc chặn lưu lượng bất thường

* + - 1. Cấu trúc Rules



Cấu trúc rule snort gồm 2 phần: rule header và rule options.

* Rule Header

Gồm có các thông tin sau: actions, protocol, IP nguồn , port nguồn, direction operator, IP đích, port đích.

* actions: cảnh báo, ghi log, hoặc lọc bỏ gói
* Protocol: TCP, UDP, ICMP, IP
* IP, port nguồn: biến hoặc nêu ra cụ thể
* Direction operator: chỉ ra hướng của lưu lượng mà rule áp dụng, có thể là một chiều ví dụ -> hoặc <-, và <> là cả 2 chiều.
* Rule Options

Các tùy chọn quy tắc là trung tâm của công cụ phát hiện xâm nhập của Snort, kết hợp tính dễ sử dụng với sức mạnh và tính linh hoạt. Tất cả các tùy chọn được được phân tách bằng dấu chấm phẩy (;). Các từ khóa tùy chọn quy tắc được phân tách với các đối số của nó bằng dấu hai chấm (:)

*Lựa chọn chung:*

**msg**: là một thông điệp có ý nghĩa thường gồm những quy tắc đang được phát hiện. Tùy chọn quy tắc thư cho Snort biết những gì sẽ xuất khi quy tắc khớp.Nó là một chuỗi văn bản đơn giản

**sid/rev**: id snort là một định danh duy nhất cho mỗi rule. Thông tin này cho phát đầu ra plugin nhận định rule một cách dễ dàng và nên được sử dụng cùng với từ khóa **rev** (revision) để biết rule này được chỉnh sửa bao nhiêu lần.

*Lựa chọn phát hiện:*

**content**: đây là thuộc tính quan trọng cho phép người dùng đặt rule mà nó tìm kiếm nội dung được chỉ ra trong gói payload và kích hoạt các phản ứng dựa vào dữ liệu đó. Trong dữ liệu tùy chọn có thể chứa văn bản hỗn hợp và dữ liệu nhị phân.

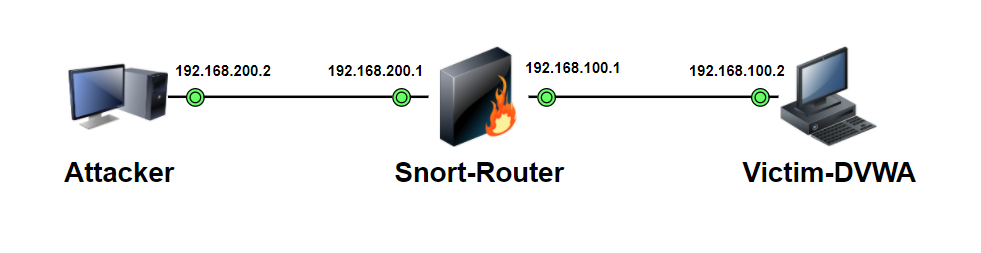
**pcre**: cho phép các quy tắc được viết bằng cách sử dụng biểu thức chính quy tương thích với perl, cho phép các nhận dạng kết quả có độ phức tạp hơn các kết quả nội dung đơn giản

xem chi tiết tại đây: https://paginas.fe.up.pt/~mgi98020/pgr/writing\_snort\_rules.htm

**CHƯƠNG 2: TRIỂN KHAI**

1. **MÔ HÌNH TRIỂN KHAI:**

* Mô hình nâng cao



|  |  |  |  |
| --- | --- | --- | --- |
| Máy | IP | Subnet mask | Nhiệm vụ |
| Snort - Router | Ens33:  192.168.100.1  Ens38:  192.168.200.1 | Ens33:  255.255.255.0  Ens38:  255.255.255.0 | - Forwarding lưu lượng đến máy Victim  - Phát hiện và ngăn chặn tấn công SQL victim |
| Victim | Ens33:  192.168.100.2 | Ens33  255.255.255.0 | Web server lưu trữ DVWA |
| Attacker | Eth0:  192.168.200.2 | Eth0:  255.255.255.0 | Tấn công Sql injection đến trang web DVWA |

Bảng 1. 1: Thông tin thiết lập ip

* Kế hoạch triển khai

Attacker tấn công SQL injection vào trang web DVWA của máy Victim, máy Snort-Router sẽ phát hiện và ngăn chặn dựa vào rules tự tạo, rules talos, emerging threat, và so sánh độ chính xác giữa các rules

1. **HỆ THỐNG THỰC HIỆN:**

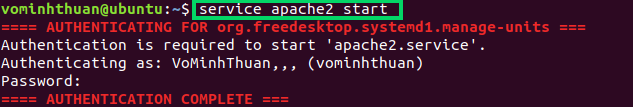
* CPU: Intel(R) Core(TM) i7-3630QM CPU @ 2.40GHz (8 CPUs), ~2.4GHz
* RAM: 8GB DDR4
* Hệ điều hành: Windows 10 pro 64-bit
* GPU: NVIDIA GEFORCE GT 650M

1. **CÀI ĐẶT**
   1. Cài đặt DVWA
      * 1. Giới thiệu

* DVWA là một trang web có tồn tại các lổ hổng bảo mật được viết bằng PHP, để phục vụ chổ việc test thử các lổ hổng
  + - 1. Cài đặt
* Cài đặt Apache2



* Khởi động dịch vụ apache2



* Cài đặt Mariadb



* Cài đặt thư viện liên quan



* Tải web DVWA



* Giải nén thư mục DVWA



* Duy chuyển thư mục vừa giải nén vào thư mục chứa Web



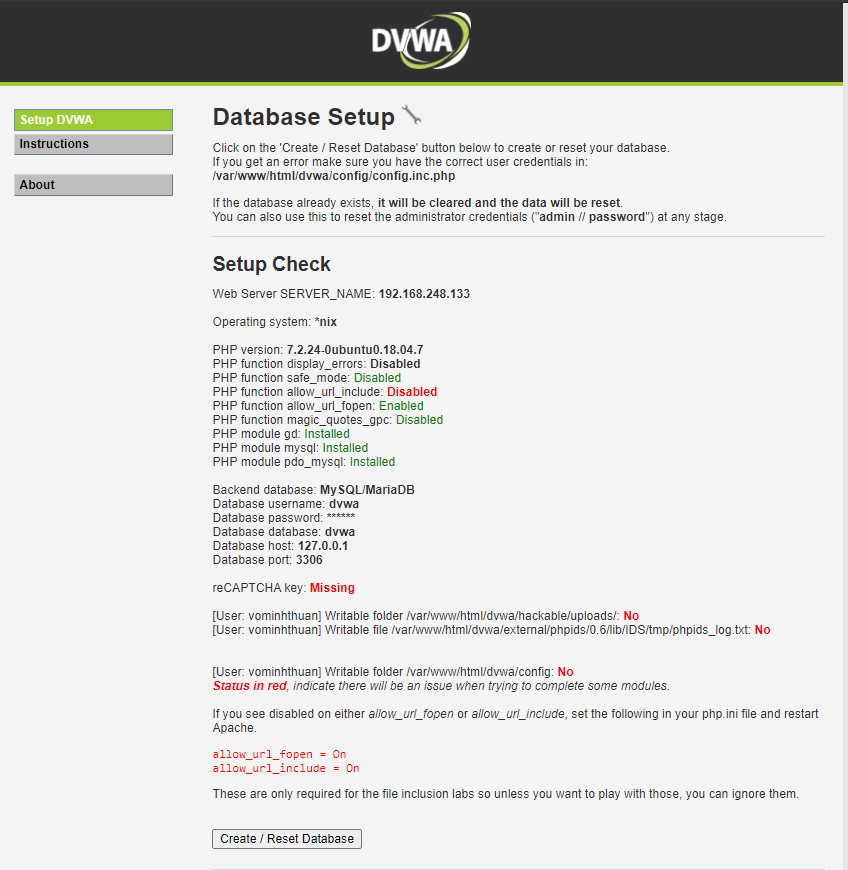
* Đổi tên thư mục để dể truy cập



* Copy file config.inc.php.dist -> config.inc.php để web hoạt động



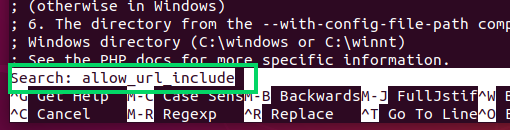
* Truy cập trang web DVWA với địa chỉ localhost/dvwa



* + - 1. Cấu hình
* Sửa đổi file php.ini



* Ctr + W rồi tìm từ khóa allow\_url\_include



* Đổi off -> on

 => 

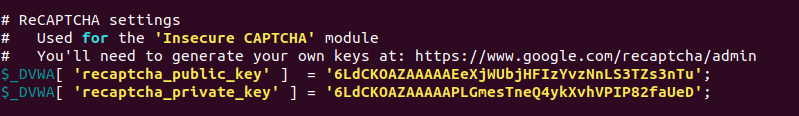
* Xong rồi restart lại apache2



* Vào file



* Cập nhật mục màu vàng với public key và private key lấy từ trang web được đề cập

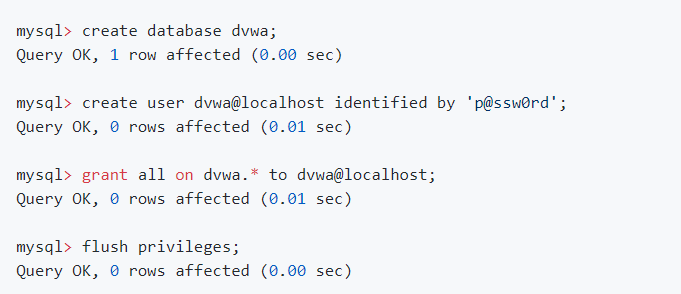






* Tạo bảng database dvwa





* Đăng nhập

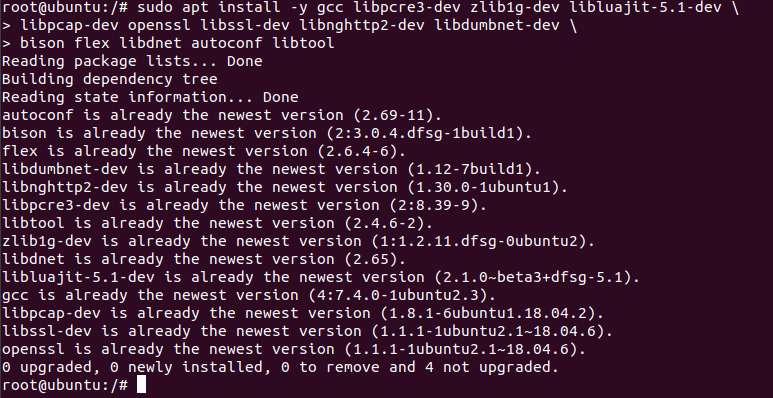


* User name :admin
* Password: password
* Thay đổi cấp độ bảo mật



* + 1. Cài đặt Snort
* Đầu tiên ta cài đặt các thư viện hỗ trợ cần thiết.

# sudo apt install -y gcc libpcre3-dev zlibig-dev libluajit-5.1-dev libpcap-dev openssl libssl-dev libnghttp2-dev libdumbnet-dev bison flex libdnet autoconf libtool



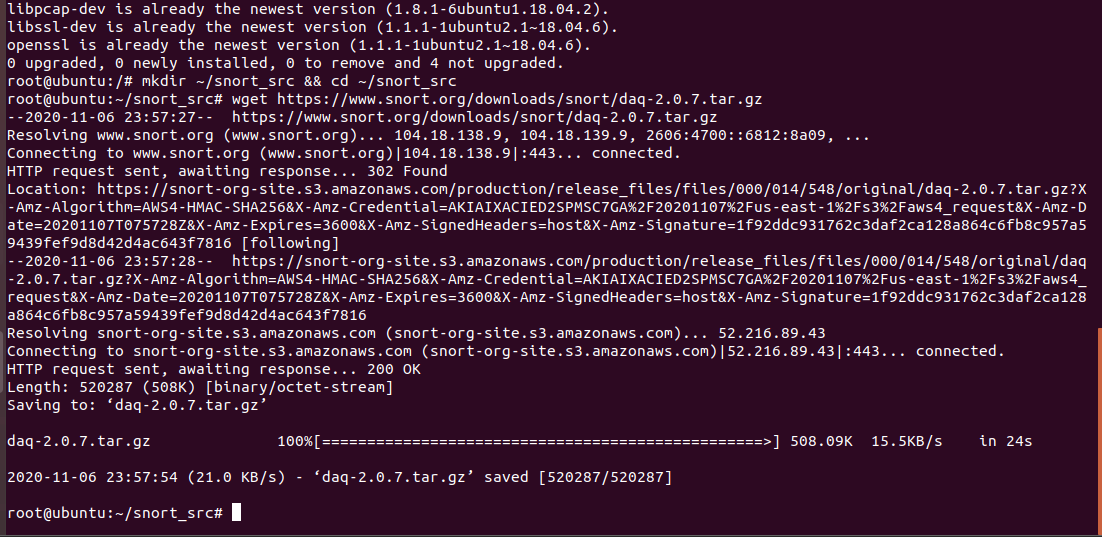
* Tạo thư mục để tải về tạm thời và đi tới thư mục đó.

# mkdir ~/snort\_src && cd ~/snort\_src



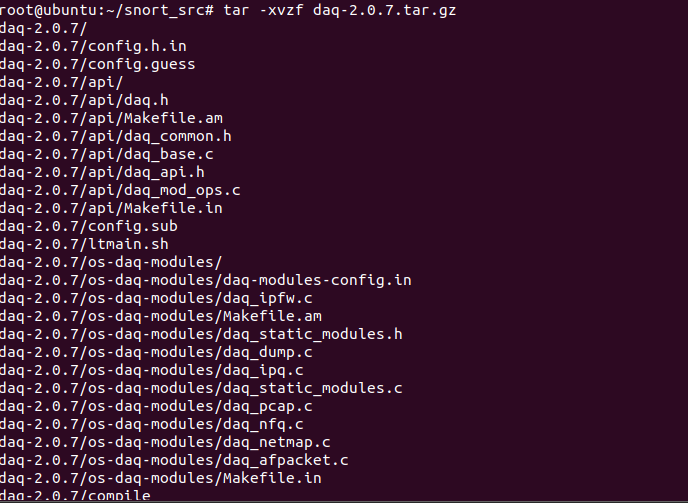
* Snort sử dụng thư viện thu thập dữ liệu (DAQ) để thực hiện các cuộc gọi trừu tượng đến thư viện bắt gói. Ta sẽ tải về từ trang web của Snort thông qua lệnh wget:

# wget <https://www.snort.org/downloads/snort/daq-2.0.7.tar.gz>



* Giải nén file:

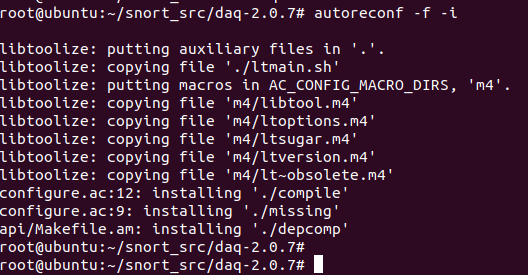
# tar -xvzf daq-2.0.7.tar.gz

e

* Sau đó ta di chuyển tới thư mục của thư viện và tiến hành cài đặt autoconf và libtool để tự động cấu hình lại DAQ

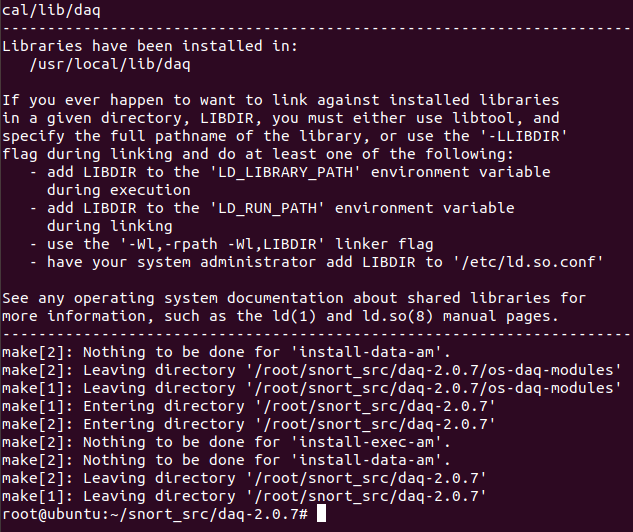


# autoreconf -f -i



* Sau đó ta tiến hành tạo và hoàn thành cài đặt DAQ qua câu lệnh:

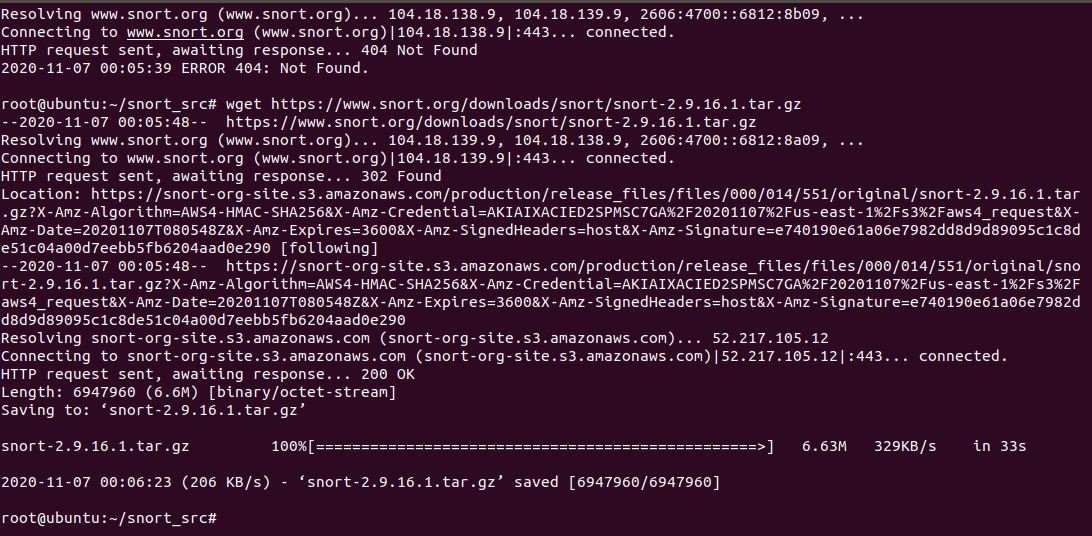
# ./configure && make && sudo make install



* Sau khi cài đặt xong, qua trở lại thư mục ban đầu và tiến hành tải và cài đặt snort phiên bản mới nhất:

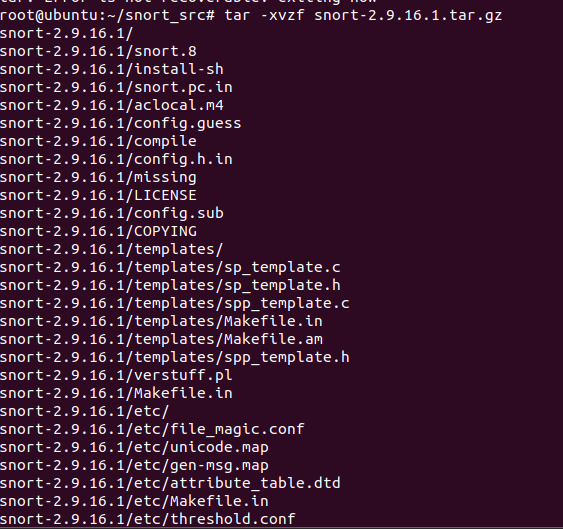


# wget https://www.snort.org/downloads/snort/snort-2.9.16.tar.gz



* Giải nén file

# tar -xvzf snort-2.9.16.1.tar.gz

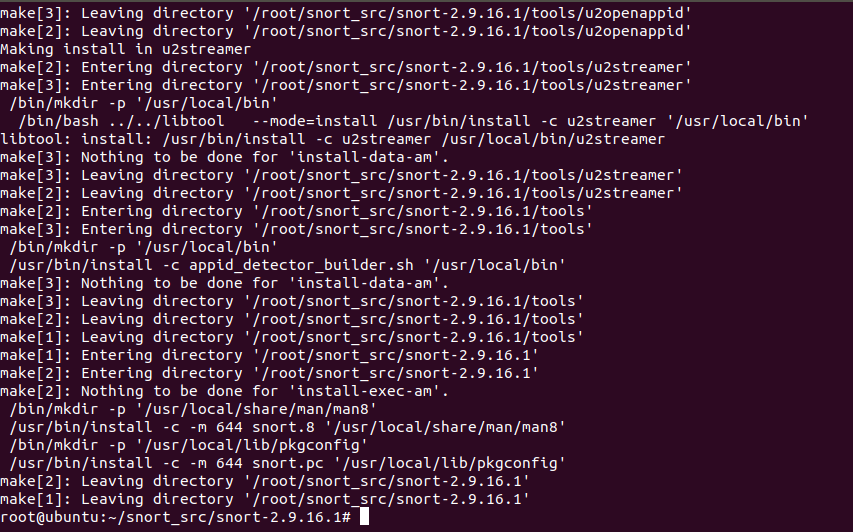


* Tương tự như DAQ, ta cấu hình phần cài đặt và hoàn tất qua câu lệnh:

# cd snort-2.9.16.1



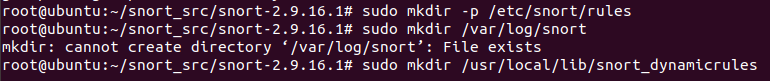
# ./configure --enable-sourcefire && make && sudo make install



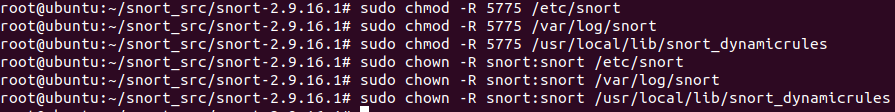
* Cấu hình Snort để chạy trên NIDS



* Thiết lập hệ thống thư mục để cung cấp địa chỉ cho cấu hình Snort



* Thiết lập quyền cho các địa chỉ mới



* Tạo tập tin cho danh sách đen(black\_list) và white\_list cũng như local.rules

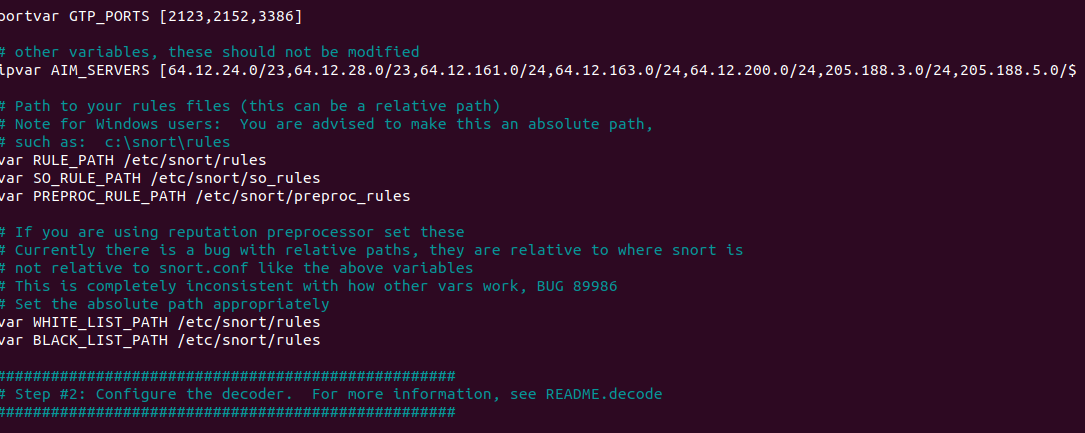


* Sau đó copy các file cấu hình đã tạo vào thư mục đã download Snort:



* Thực hiện cấu hình địa chỉ cho các thư mục trong file cấu hình

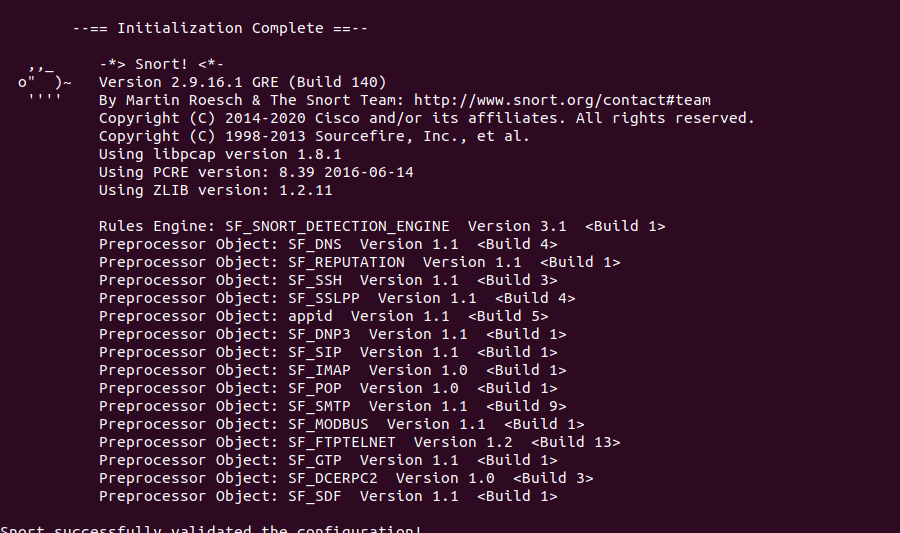




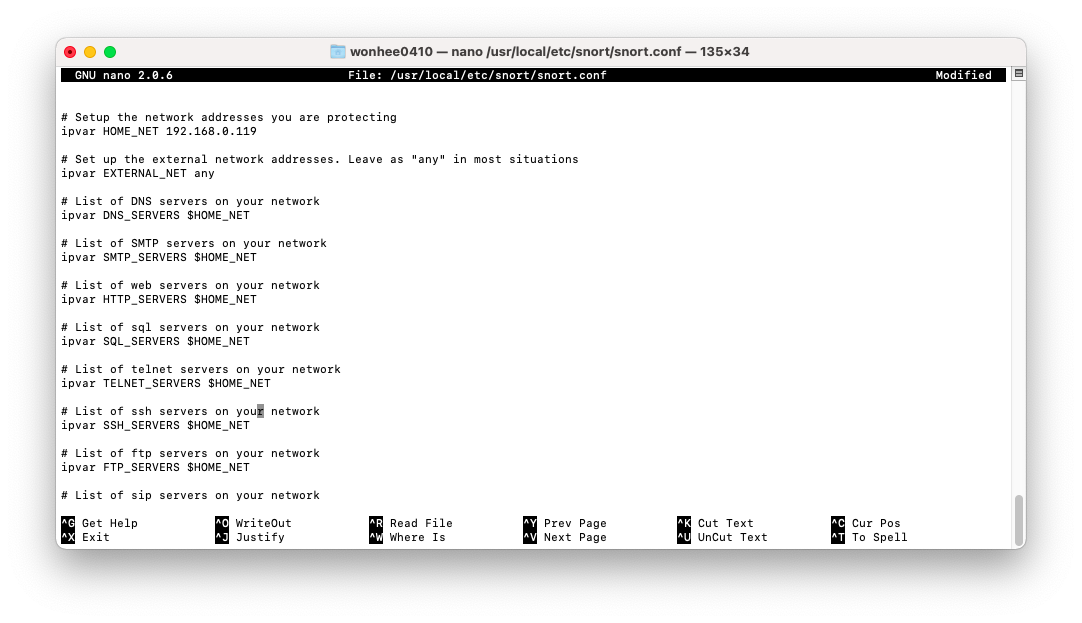


* Lưu lại file cấu hình và chạy câu lệnh **sudo snort -T -c /etc/snort/snort.conf**

Để kiểm tra xem quá trình cấu hình báo lỗi hay không



* Thiết lập file snort.conf và chỉnh sửa IP HOME\_NET

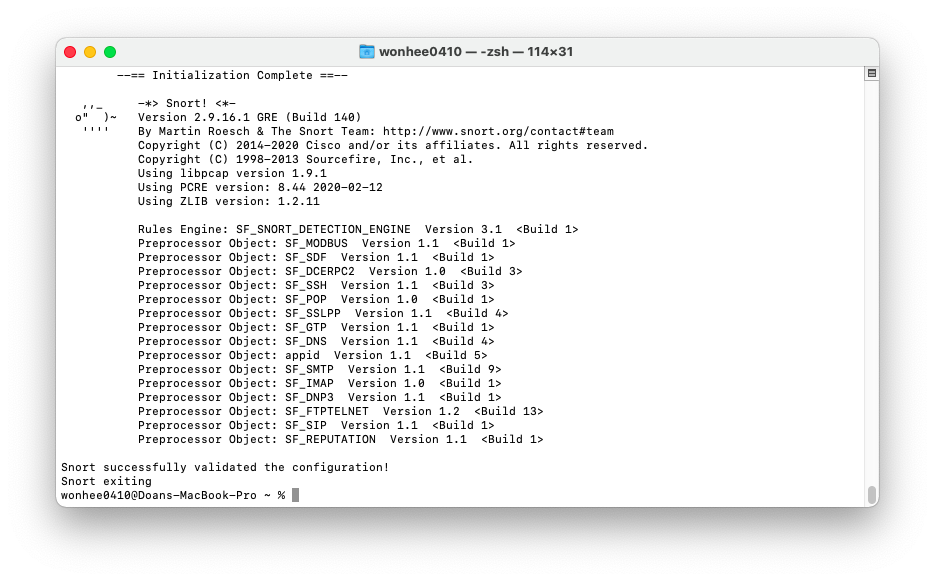


* Chỉnh sửa đường dẫn các rules





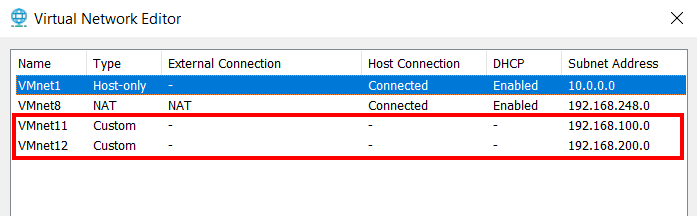
* Xác minh file config có đúng hay không



1. **CẤU HÌNH**

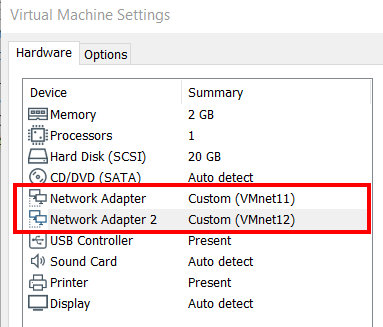
* Xây dựng mô hình bằng máy ảo Vmware

Bước 1: Thiết lập 2 lớp mạng ảo trên VMware



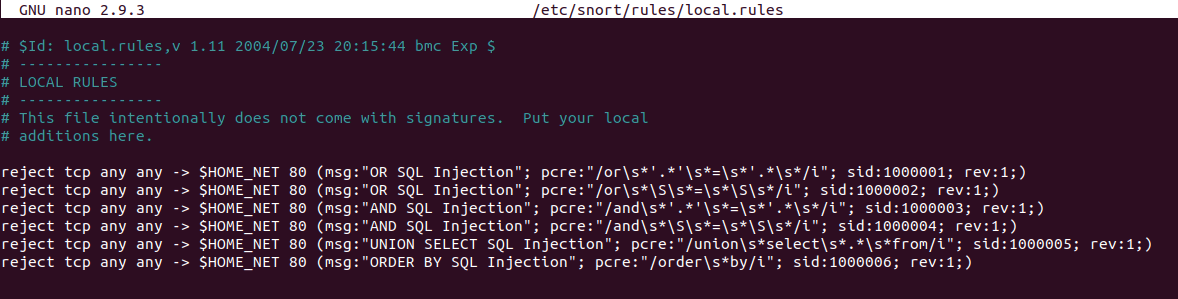
1. Cấu hình máy Snort\_Router (Ubuntu)

* Ens33
  + Static IP: 192.168.100.1
  + Netmask: 255.255.255.0
* Ens38
  + Static IP: 192.168.200.1
  + Netmask: 255.255.255.0
* Add 2 card mạng tương ứng với 2 lớp mạng ảo trên



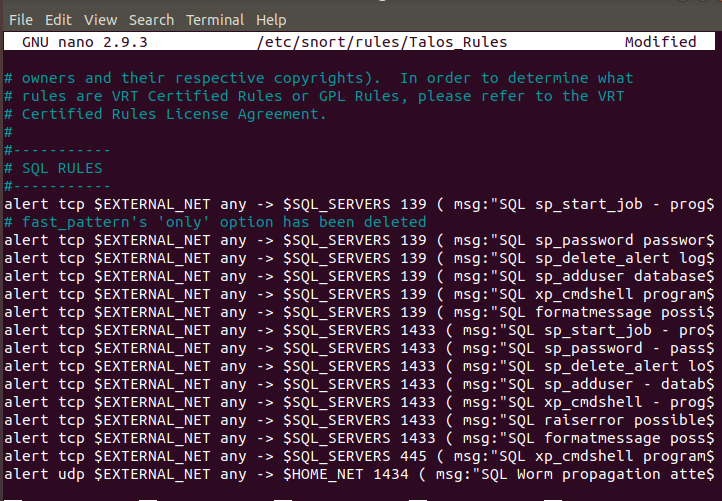
* Tải snort ( Xem chi tiết mục 2.3.2)
* Thiết lập rules
* Rule tự tạo

# sudo nano /etc/snort/rules/local.rules

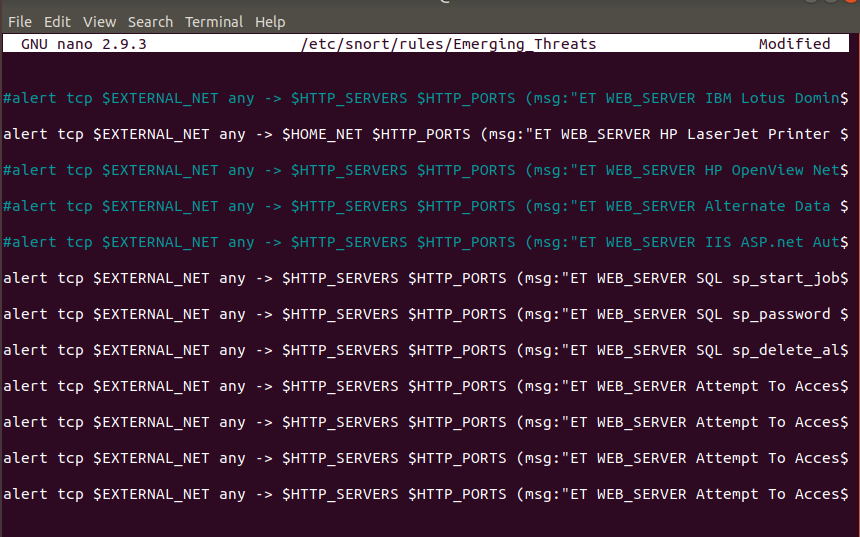


* Rule talos

Tải rules talos <https://www.snort.org/downloads/registered/Talos_LightSPD-2020-12-10-001.tgz>

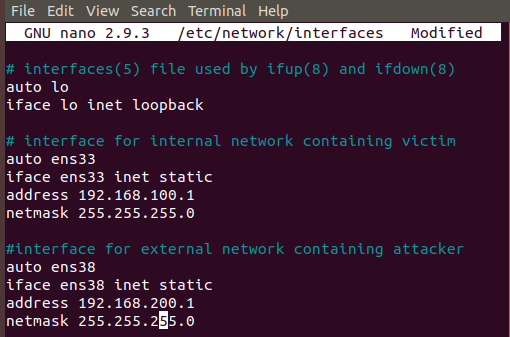


* Rule emerging threats



* Thiết lập ip static cho 2 card mạng

# sudo nano /etc/network/interfaces

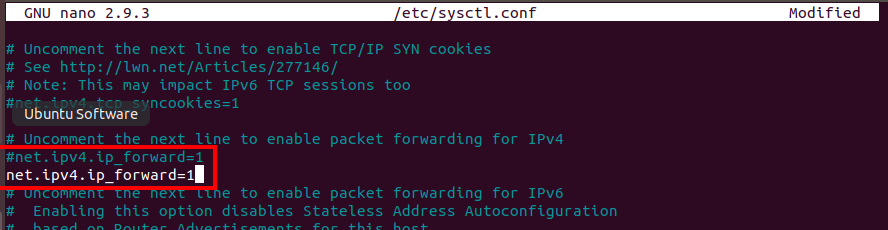


(lưu ý: tùy theo tên interface khác nhau mà ta sẽ thiết lập tên interface khác nhau)

* Cho phép packet forwarding Ipv4

\_ Với mục đích để các lớp mạng tương ứng các interface khác nhau có trong máy có thể giao tiếp được với nhau.

# sudo nano /etc/sysctl.conf



# sudo sysctl -p

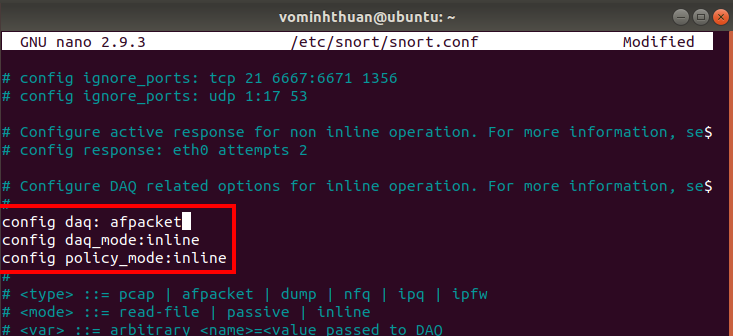
\_ Kiểm tra trạng thái của ip\_forward



* Chuyển mode inline

# sudo nano /etc/snort/snort.conf

Chỉnh sửa file snort.conf như sau:

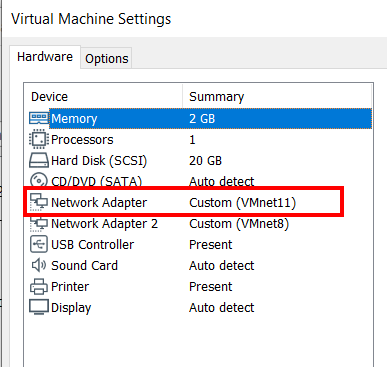


# sudo reboot

1. Cấu hình máy Victim (Ubuntu)

* Ens33
  + Static IP: 192.168.100.2
  + Netmask: 255.255.255.0
  + Gateway: 192.168.100.1 ( cũng là địa chỉ của máy Snort-Router)
* Thiết lập card mạng

\_ Với card 1 dùng kết nối mạng nội bộ Vmnet11

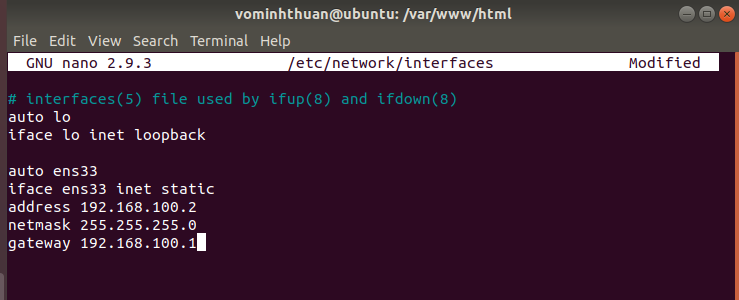


* Cài đặt dịch vụ web(dvwa) trên máy victim

*Xem chi tiết mục 2.3.1*

* Cấu hình ip static cho card mạng

# sudo nano /etc/network/interfaces

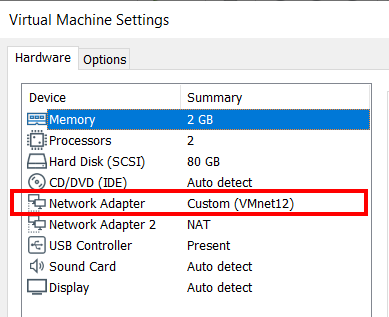


# sudo reboot

1. Cấu hình máy Attacker ( Kali linux)

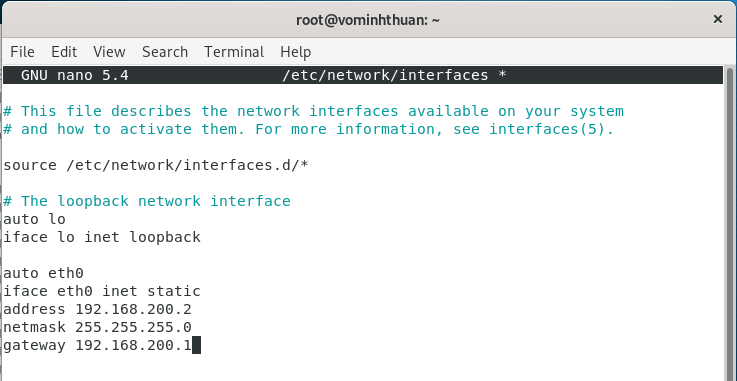
* Eth0
  + Static IP: 192.168.200.2
  + Netmask: 255.255.255.0
  + Gateway: 192.168.200.1
* Thiết lập card mạng

\_ Với card 1 dùng kết nối mạng nội bộ Vmnet12



* Cấu hình ip static cho card mạng

# nano /etc/network/interfaces



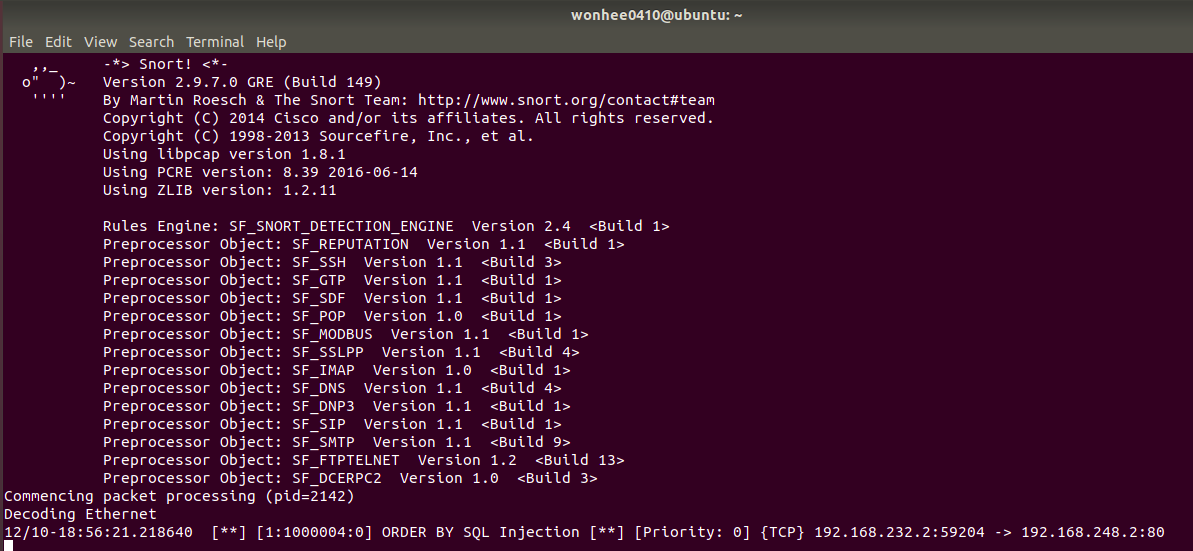
# service networking restart

1. **TẤN CÔNG THỰC NGHIỆM**

*( Ở phần thực nghiệm tấn công ip có thay đổi do vấn đề phân công làm nhóm , với lớp mạng 192.168.100.0 sẽ đổi thành 192.168.248.0 và 192.168.200.0 -> 192.168.232.0)*

* Bên máy Snort-Router
* Khởi động snort

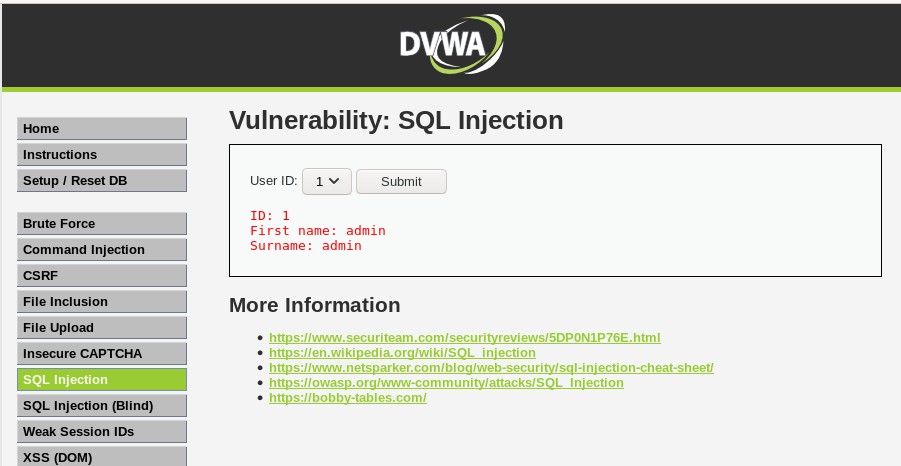
# sudo snort -A console -c /etc/snort/snort.conf -i ens33:ens38



* Bên máy Victim
* Khởi động DVWA

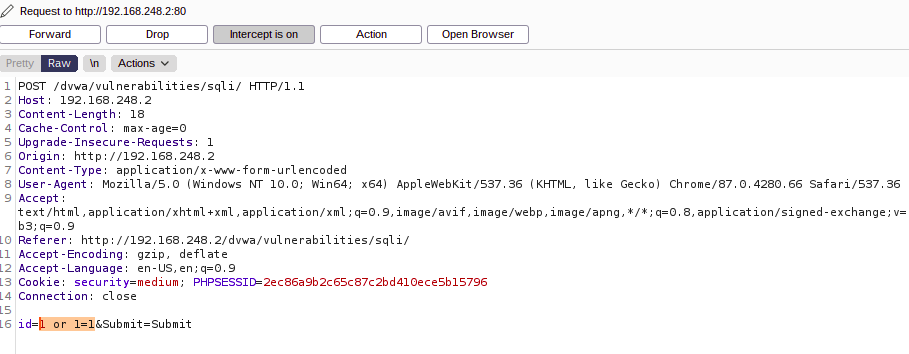
# sudo service apache2 restart

* Bên máy Attacker
* Truy cập web DVWA chứa trên máy Victim

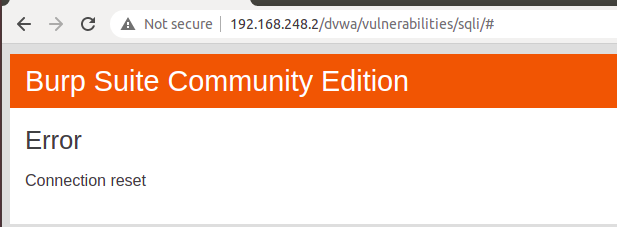


* Thực hiện tấn công SQL injection với level medium, dùng Burp Suite
* **Rules tự viết:**
* **Trường hợp: 1 or 1=1**

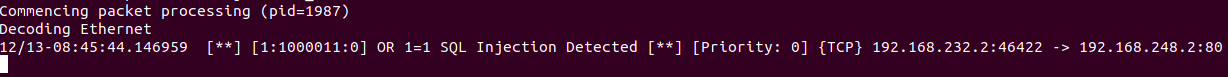
Chèn nội dung “1 or 1 =1” vào request gửi đến victim



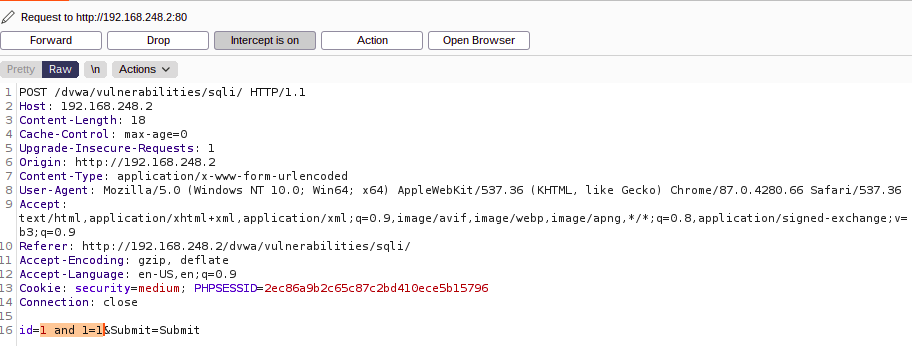
* Tấn công đã bị chặn

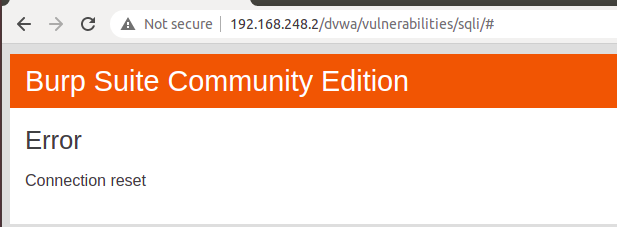


* Snort cảnh báo tấn công

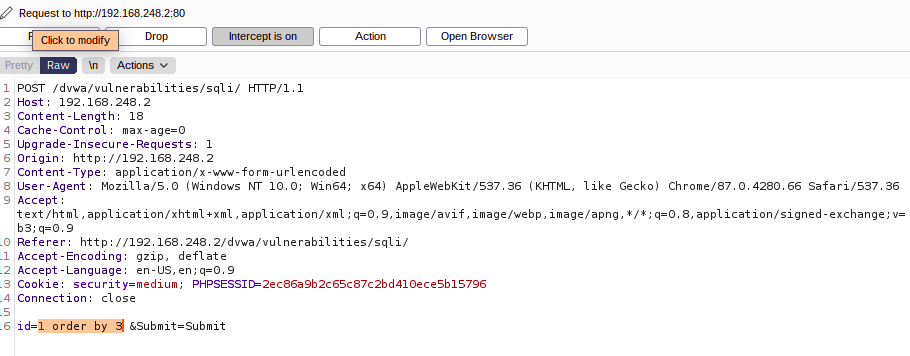


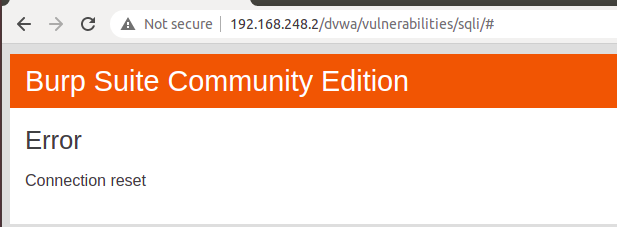
* **Trường hợp: 1 and 1=1**



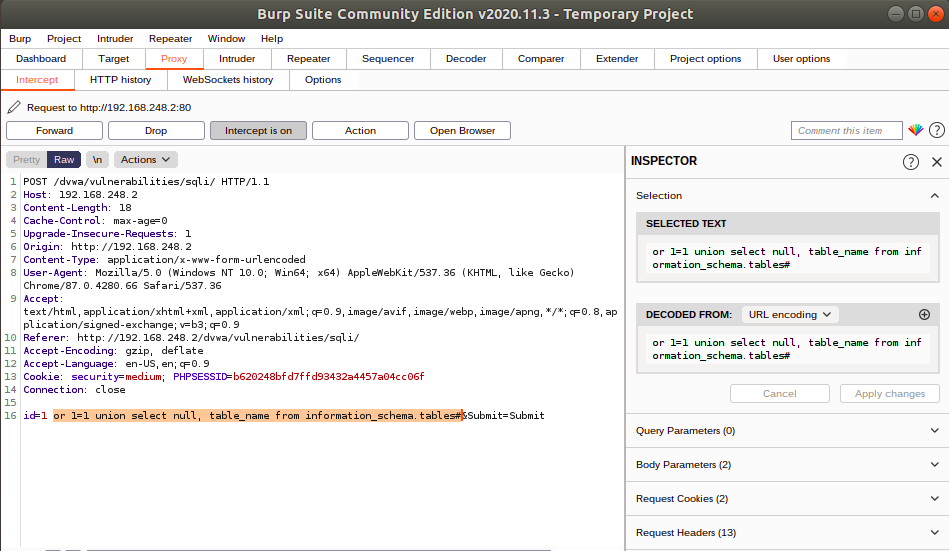


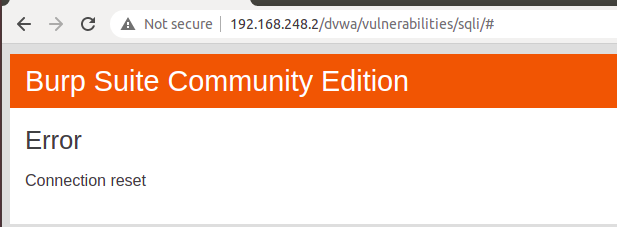
* **1 order by 3**

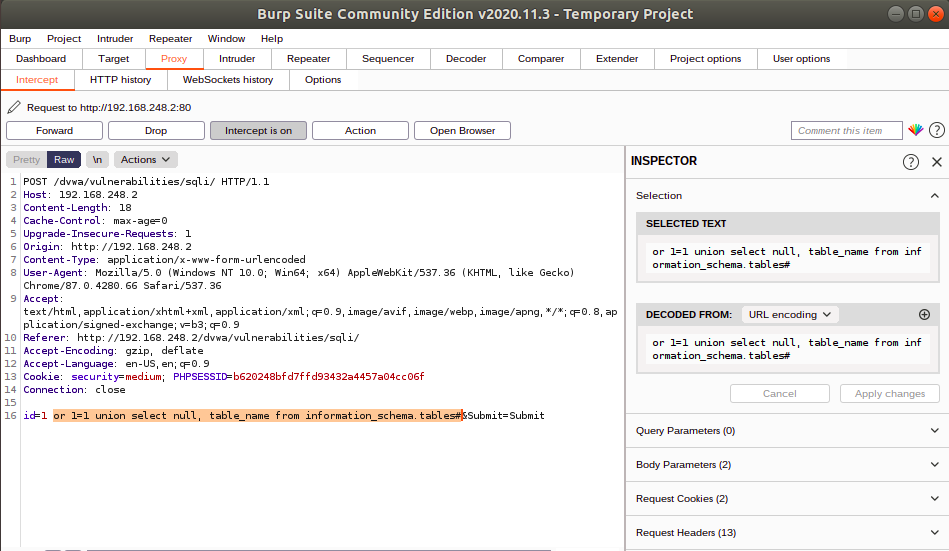




* **Trường hợp: 1 or 1=1 union select null, table\_name from information\_schema.tables#**

****

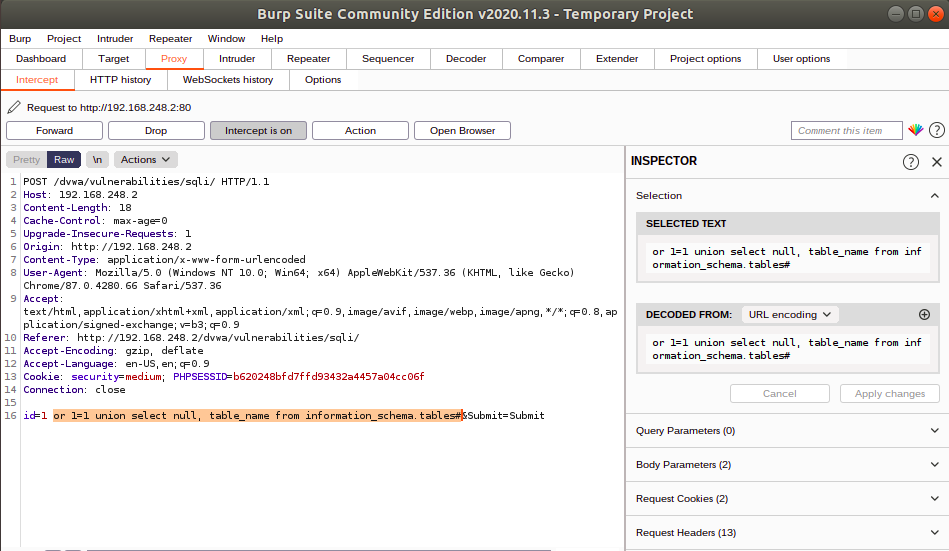




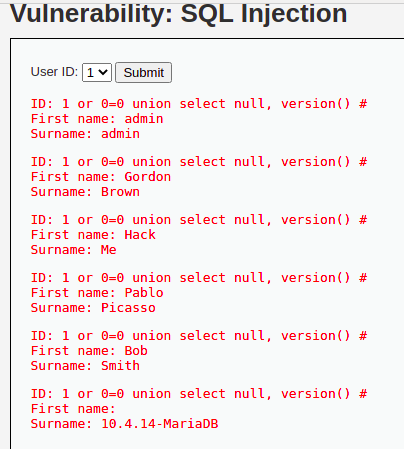


Tấn công đã bị phát hiện và ngăn chặn bởi Snort khi attacker tấn công máy victim 192.168.248.2 (máy chứa DVWA)

* **Emerging Threat rules:**
* **Trường hợp : 1 or 1=1 union select null, table\_name from information\_schema.tables#**

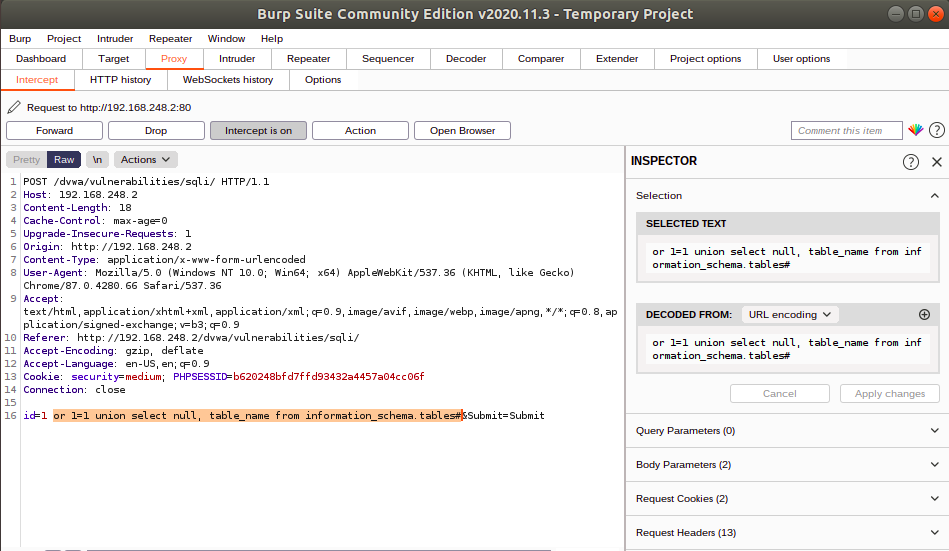


* Snort không cảnh báo
* Chúng ta vẫn lấy được thông tin từ server



* **Talos rules:**

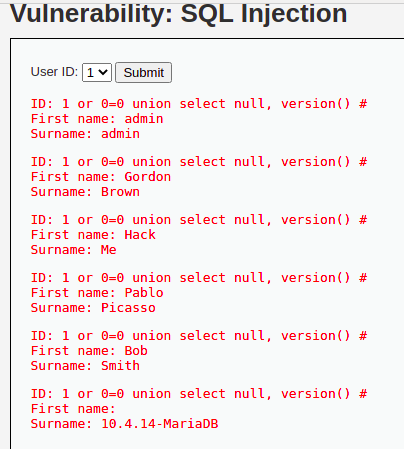
**Trường hợp : 1 or 1=1 union select null, table\_name from information\_schema.tables#**



* Có cảnh báo phát hiện tấn công



* Chúng ta vẫn lấy được thông tin từ server



1. **KẾT LUẬN**

Với 3 rule là rule tự tạo, rule talos và emerging threat thì kết quả cho thấy được rule tự tạo đã ngăn chặn được tấn công vì rule này được thiết lập là từ chối (Reject) trong khi rule talos và emerging thì không chặn được cuộc tấn công do chúng được thiết lập chỉ cảnh báo (Alert)

|  |  |  |  |
| --- | --- | --- | --- |
|  | Rules tự tạo | Emerging Threats | Talos rules |
| Thiết lập | Reject | Alert | Alert |
| Phát hiện tấn công dạng:  **1’ or ‘1’=’1** | Có | Không (chỉ phát hiện ở DVWA mức low) | Không |
| Phát hiện tấn công dạng:  **1 or 1=1** | Có | Không (chỉ phát hiện ở DVWA mức low) | Có |
| Phát hiện tấn công dạng:  **Order by** | Có | Không (chỉ phát hiện ở DVWA mức low) | Không |
| Phát hiện tấn công dạng:  **Union Select** | Có | Có | Có |
| Độ chính xác | Cần phải cân nhắc kỹ rules để tránh cảnh báo giả | Được cộng đồng hỗ trợ đóng góp | Được cộng đồng hỗ trợ đóng góp |

**TÀI LIỆU THAM KHẢO**

[1] <https://en.wikipedia.org/wiki/Snort_(software)>

[2] <https://learning.oreilly.com/library/view/intrusion-detection-systems/0131407333/0131407333_ch01lev1sec3.html>

**PHỤ LỤC**

**Phần 1:Trả lời câu hỏi**

Câu 1 Lý do để dùng snort thay vì suricata?

Trả lời:

Cơ bản vì đây là chủ đề do cô phân công, nhóm không tự chủ, nên về ý kiến khách quan thì mình sẽ chon suricata vì nó là bản cập nhật của Snort.

Câu 2 Nhưng điểm yếu của Snort là gì?

Trả lời:

Do sự sâu rộng của giao diện, việc kiểm thử và quản lý có thể khá vất vả. Có nhiều nơi khác nhau để thiết lập các mục riêng lẻ nên sẽ tốt hơn nếu đơn giản hoá giao diện một chút.

Việc nâng cấp có thể nguy hiểm phần nào đó khi nếu nâng cấp phiên bản ví dụ tư 5.x lên 6.x có thể làm hỏng hệ thống nếu không được thực hiện đúng cách vì có rất nhiều công việc bổ sung bên ngoài giao diện người dùng.

Nhiều lúc có hiện tượng không ổn định với các bug của Cisco.

Từ quan điểm quản trị thì việc các FTD Images(Firepower Threats Defense) đang được đẩy qua các ASA(Adaptive Security Appliance) sẽ kém hiệu quả hơn, không có giao diện Command Line..

Câu 3 Hướng phát triển của snort

Trả lời:

Phát hiện và ngăn chặn tấn công mạng , đặc biệt là 2 kiểu tấn công phổ biến cross-site scripting(xss) và SQL Injection được các hacker sử dụng phổ biến.

Mở rộng phạm vi quét của Snort bằng cách cập nhật thường xuyên làm đa dạng hoá các signature của các cuộc tấn công. Sử dụng AI cho phép chủ động trong việc nhận biết các thay đổi của các cuộc tấn công đến từ hacker so với các signature có sẵn giúp ngăn chặn các cuộc tấn công đột biến không đoán trước.

Tối ưu hoá tài nguyên sử dụng của Snort khi số lượng log trở nên quá tải hoặc dung lượng hoạt động tăng đột biến ảnh hưởng đến RAM và CPU của máy giúp cho Snort có thể hoạt động trên nhiều loại máy và hệ điều hành khác nhau.

Câu 4 Có các loại rule mặc định nào?

Trả lời:

* Có 5 loại rule mặc định có sẵn trong Snort:
* alert: tạo ra một cảnh báo sử dụng các phương pháp cảnh báo được lựa chọn, và sau đó ghi lại vào trong gói tin.
* log: ghi lại gói tin.
* pass: bỏ qua gói tin.
* activate: cảnh báo và kích hoạt một rule động(dynamic rule) khác.
* dynamic: không làm gì cho đến khi được kích hoạt với activate rule, và sau đó hoạt động như một log rule.

Câu 5 Khi viết rules drop và rules reject khác nhau ở điểm nào?

Trả lời:

Drop: block và ghi log lại các gói tin

Reject: block các gói tin, ghi log lại và gửi một TCP reset nếu giao thức là TCP hoặc một thông báo ICMP port unreachable nếu giao thức là UDP

Câu 6 SQLi là gì?

Trả lời:

SQL injection là một lỗ hổng bảo mật web cho phép kẻ tấn công can thiệp bằng các câu truy vấn mà ứng dụng thực hiện đối với cơ sở dữ liệu của nó. Nó thường cho phép kẻ tấn công xem dữ liệu mà chúng thường không thể trích xuất. Điều này có thể bao gồm dữ liệu thuộc về người dùng khác hoặc bất kỳ dữ liệu nào khác mà bản thân ứng dụng đó có thể truy cập. Trong nhiều trường hợp, kẻ tấn công có thể sửa đổi hoặc xóa dữ liệu này, gây ra những thay đổi liên tục đối với nội dung hoặc hành vi của ứng dụng. Trong một số tình huống, kẻ tấn công có thể leo thang một cuộc tấn công SQL injection để xâm phạm server hoặc cơ sở hạ tầng back-end khác hoặc thực hiện một cuộc tấn công từ chối dịch vụ.

Ví dụ: Có thể sử dụng một thủ thuật liên quan đến dấu nháy và đặt trường mật khẩu thành:

**password' OR 1=1**

Kết quả là máy chủ cơ sở dữ liệu chạy truy vấn SQL sau:

**SELECT id FROM users WHERE username='username' AND password='password' OR 1=1'**

Do câu lệnh OR 1 = 1, mệnh đề WHERE trả về id đầu tiên từ bảng users bất kể tên người dùng và mật khẩu là gì. Id người dùng đầu tiên trong cơ sở dữ liệu thường là quản trị viên. Bằng cách này, kẻ tấn công không chỉ bỏ qua xác thực mà còn có được đặc quyền của quản trị viên.

Cách phòng chống:

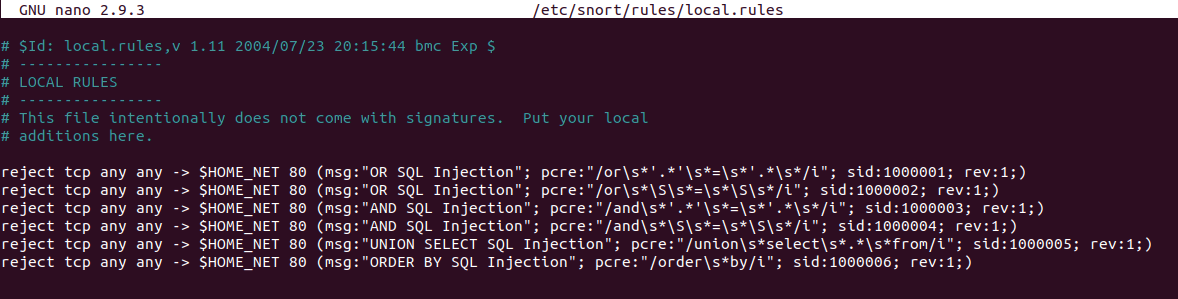
Cách chắc chắn duy nhất để ngăn chặn các cuộc tấn công SQL Injection là xác thực đầu vào và các truy vấn được tham số hóa bao gồm các câu lệnh đã chuẩn bị. Code của ứng dụng không bao giờ được sử dụng đầu vào trực tiếp. Developer phải khử tất cả đầu vào, không chỉ đầu vào biểu mẫu web, chẳng hạn như biểu mẫu đăng nhập. Họ phải loại bỏ các phần tử mã độc tiềm ẩn như dấu ngoặc kép. Bạn cũng nên tắt khả năng hiển thị lỗi cơ sở dữ liệu trên các trang của mình. Lỗi cơ sở dữ liệu có thể được sử dụng với SQL Injection để lấy thông tin về cơ sở dữ liệu của bạn.

Câu 7 Nêu luật để phát hiện bypass SQL injection bằng cách sử dụng đầu độc tham số HTTP (HTTP Parameter Pollution)

Trả lời:

HTTP parameter pollution (HPP) là một kỹ thuật tấn công web mà kẻ tấn công sẽ tạo ra các tham số trùng nhau trong HTTP request. Lợi dụng các phản ứng khác nhau của các công nghệ web khi xử lý các tham số trùng nhau này để inject những đoạn mã độc hại nhằm tấn công server và người sử dụng.

* Tuy nhiên, vẫn có thể dùng bộ rules chống SQL Injection đã thiết lập từ trước để chống bypass bằng đầu độc tham số, Snort có thể detect ở URL, ở cả GET và POST cho nên hoàn toàn có thể ngăn chặn được cuộc tấn công.



Câu 8 Ưu điểm của Snort so với Suricata

* Dễ dàng cài đặt từ package
* Cộng đồng hỗ trợ lớn, lâu năm hơn
* Nhiều tài liệu tốt từ official website và trên internet
* Công nghệ mô-đun phát hiện: Snort sensors có dạng mô-đun và có thể giám sát nhiều máy từ một vị trí vật lý và logic. Snort được đặt trước tường lửa, sau tường lửa, bên cạnh tường lửa và ở mọi nơi khác để giám sát toàn bộ mạng. Do đó, các tổ chức sử dụng Snort như một giải pháp bảo mật để tìm hiểu xem có những nỗ lực xâm nhập trái phép trong hệ thống mạng hoặc liệu một hacker có xâm nhập trái phép vào hệ thống mạng hay không.

Câu 9 Snort được cấu tạo từ những thành phần nào ?

Trả lời: Snort gồm có 5 thành phần:

* + Packet Decoder,
  + Preprocessors,
  + Detection Engine,
  + Logging and Alerting System,
  + Output Modules

Câu 10 Phát hiện tấn công dựa vào điểm nào của request http?

Trả lời:

Sẽ dựa vào content của request http được gửi tới server

Module Detection Engine sẽ so khớp rule, và có thể đưa ra cảnh báo

Câu 11 SNORT hoạt động ở tầng nào trong mô hình OSI ?

Trả lời:

Snort hoạt động tên từ Data Link đến Application

Câu 12 Snort có thể phân tích được các thông tin bị mã hóa không ?

Trả lời:

Snort không thể phân tích nếu như thông tin được mã hóa,

về cơ bản thì để giải mã được được thì chúng ta phải cần key giải mã

**Phần 2: Tự đánh giá**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **4** | **3** | **2** | **1** |
| Hình thức báo cáo (1 điểm) | **x** |  |  |  |
| Thuyết trình (1 điểm) | **x** |  |  |  |
| Phần cơ sở lý thuyết (2 điểm) | **x** |  |  |  |
| Phần thực hiện (5 điểm) | **x** |  |  |  |
| Hiểu được nội dung của các nhóm khác (cô đặt câu hỏi sau thuyết trình cho các nhóm) (1 điểm) |  |  |  |  |

**Phần 3: Phân công công việc**

|  |  |  |
| --- | --- | --- |
| STT | Họ và tên | Công việc |
| 1 | Võ Minh Thuận | Nội dung cơ sở lý thuyết  Biên soạn tài liệu |
| 2 | Vũ Tuấn Dũng | Soạn slide thuyết trình  Trả lời câu hỏi |
| 3 | Đoàn Quang Huy | Cấu hình, cài đặt, thực nghiệm triển khai mô hình |

Link video thực nghiệm:

<https://drive.google.com/drive/folders/1FVR5krgUaJ_EHkxZ9us6Q2hUs92wtekS?fbclid=IwAR2ioMkgpqewNvVWC5n_iWC3JZVfK-nIKLD_CAIAluWX66gQT-vfDFI2wg8>