Department of Computer Science and Engineering
Bangladesh University of Engineering and Technology

CSE406 - Computer Security Sessional

Project Report on SNORT3

Supervisor:

Abdur Rashid Tushar
Lecturer
Bangladesh University of Engineering and Technology

Submitted By:

Wasif Hamid
1905026
Level-4, Term-1

Md. Huzzatun Ali
1905027
Level-4, Term-1

March 11, 2024

Abstract

Snort 3, released as a successor to its highly regarded predecessor Snort 2.x, represents a significant advancement in the realm of network security. This latest iteration builds upon Snort's rich history, which dates back to its inception in 1998, and incorporates modernized features to tackle contemporary cyber-security challenges. With a modular architecture, Snort 3 offers enhanced flexibility and scalability, empowering users to deploy customized detection and prevention solutions tailored to their specific network environments. Moreover, Snort 3 introduces improvements in performance and efficiency, leveraging multi-threading and optimized processing algorithms to efficiently handle high-speed network traffic with minimal latency.

Beyond its technical enhancements, Snort 3 underscores usability and extensibility, streamlining rule creation and management processes while facilitating seamless integration with other security tools and platforms. Its support for advanced protocol analysis and protocol-specific inspection further enhances its ability to accurately identify and mitigate network threats. As organizations continue to grapple with increasingly sophisticated cyber threats, Snort 3 remains at the forefront of network intrusion detection and prevention, providing security professionals with a potent and versatile solution to safeguard their networks effectively.

Contents

List of Figures

1 Introduction

1.1 What Is Snort3?

Snort 3 is a powerful open-source intrusion detection and prevention system (IDPS) designed to detect and mitigate network threats effectively. It features modular architecture, improved performance, and emphasizes usability, making it adaptable for various network security needs.

1.2 How Does Snort3 Work?

Snort 3 functions by analyzing real-time network traffic, comparing it with known attack signatures, and applying customizable detection and prevention measures to identify and counteract threats. It employs modular architecture, enabling users to tailor its capabilities to their specific security needs. By inspecting packet content and behavior, Snort 3 can detect various network-based attacks, including malware and intrusion attempts, while also supporting anomaly detection to identify emerging threats. In essence, Snort 3 serves as a vigilant guardian, actively monitoring network activity to safeguard against cyber threats.

1.3 Snort Operation

- **Packet Sniffing:** Examining live network traffic in real-time.
- **Packet Logging:** Gathers and records network traffic in a log file.
- **Network Intrusion Detection:** Analyzing packets and comparing traffic with signatures.
- **Network Intrusion Prevention:** Takes targeted measures to obstruct identified threats.

1.4 Snort Rules

Snort employs 5 distinct types of rule sets:

- **Community Rules:** These rules are maintained and contributed by the Snort community, encompassing a broad spectrum of known threats and vulnerabilities.
- **Registered Rules:** These rules are designed for registered Snort users and provide additional coverage for specific threats or environments. Registered user rules offer more targeted protection.

- **Subscription Only Rules:** Subscription rules offer comprehensive coverage and are available to users through a subscription service. These rules often include advanced threat intelligence.
- **OpenAppID Detectors:** OpenAppID is a feature within Snort that enables the detection and control of application-layer protocols and applications. OpenAppID detectors allow Snort to identify and classify traffic based on specific applications or protocols.
- **Customized Rules:** In Snort, custom rule creation empowers users to finely tailor their intrusion detection and prevention capabilities to address specific threats and adapt to the nuances of their network environments. This feature enables users to define precise conditions and actions for detecting and mitigating potential security breaches, enhancing the effectiveness and flexibility of their defense mechanisms.

1.5 Snort Custom Rule

Creating custom rules involves defining the following components:

- **Header:** Specifies the action to take when the rule matches (e.g., alert, log, drop).
- **Protocol:** Identifies the network protocol to which the rule applies (e.g., TCP, UDP, ICMP).
- **Source and Destination IP Addresses:** Specifies the source and destination IP addresses or ranges involved in the network communication.
- **Source and Destination Ports:** Defines the source and destination ports associated with the network traffic.
- **Rule Options:** Includes additional criteria such as payload content, packet length, flags, and flow direction.
- **Rule Content:** Defines the specific content pattern or signature to match within the packet payload.
- **Rule Actions:** Determines the action to be taken when the rule matches (e.g., alert, log, drop).

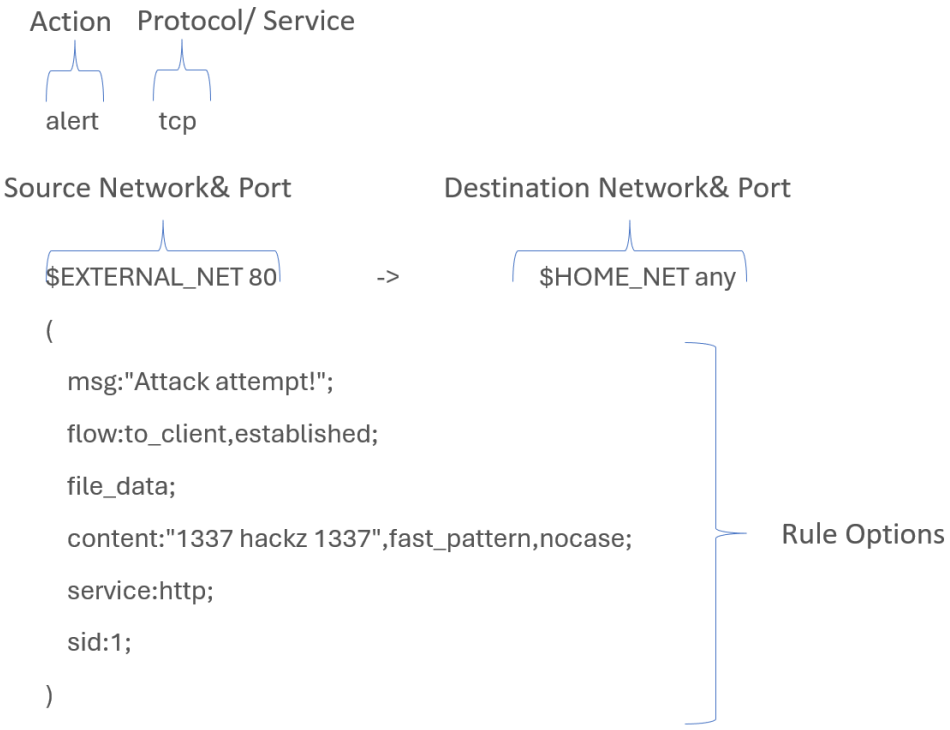

Figure 1: An example of a custom Snort3 rule

2 Prerequisites

2.1 Virtual Machine Setup

We used local VM to demonstrate the features of Snort3. Here is the list and the roles of all these virtual machines that we are going to use:

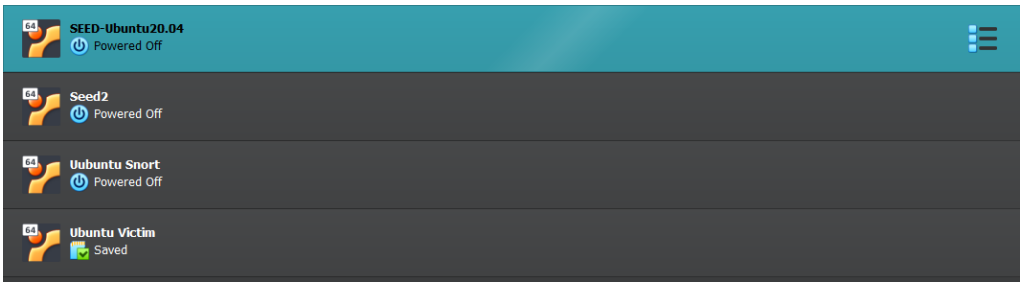

Figure 2: Used VM in Snort3 feature demonstration

- **SEED-Ubuntu20.04:** Attacker Virtual Machine.
- **Ubuntu Snort:** This is where we will install Snort.
- **Ubuntu Victim:** A vulnerable Victim Virtual Machine.

2.2 Snort Installation

To install snort in a Linux workstation, we have to follow the following steps (in the terminal). The documentation for proper installation is here: After

```
1. sudo apt install cmake flex g++ openssl pkg-config zlib1g bison autoconf libtool liblzma-dev
2. install luajit
        -- git clone https://luajit.org/git/luajit.git
        -- run make && sudo make install
3. install pcap
        -- download from https://www.tcpdump.org/
        -- run ./configure
        -- make && sudo make install
4 install pcre
        -- download from https://sourceforge.net/projects/pcre/
        -- run ./configure
        -- make && sudo make install
5. install libdaq
        -- download from https://github.com/snort3/libdaq/releases/tag/v3.0.14
        -- run ./bootstrap
        -- run ./configure
        -- make && sudo make install
6. install libnet
        -- sudo apt install check
        -- download from https://github.com/ofalk/libdnet/releases/tag/libdnet-1.17.0
        -- run ./configure
        -- make && sudo make install
7. install hwloc
        -- download from https://www.open-mpi.org/software/hwloc/v2.10/
        -- run ./configure
        -- make && sudo make install
8. install zlib
        -- download from https://zlib.net/
        -- run ./configure
        -- make && sudo make install
9. install snort3
        -- download from https://github.com/snort3/snort3/tree/master
        -- run  ./configure_cmake.sh --prefix=/usr
        -- add executable permission for
        -- "src/lua_wrap.sh"
        -- "src/managers/ffi_wrap.sh"
        -- cd build
        -- make -j $(nproc)
        -- sudo make install
```

Figure 3: Installation Documentation

following all these procedure, if we check whether Snort3 is properly installed, we will see the following window:

```
vboxuser@Uubuntu-Snort:~$ snort --v
--------------------------------------------------
o")~   Snort++ 3.1.81.0
--------------------------------------------------
--------------------------------------------------
Network Policy : policy id 0 :
--------------------------------------------------
Inspection Policy : policy id 0 :
--------------------------------------------------
pcap DAQ configured to passive.
--------------------------------------------------
host_cache
    memcap: 33554432 bytes

Snort successfully validated the configuration (with 0 warnings).
o")~   Snort exiting
vboxuser@Uubuntu-Snort:~$
```

Figure 4: Installation Check

3 Demonstration

3.1 Packet Sniffing

Packet sniffing in Snort encompasses the vital capability of the Snort Intrusion Detection System (IDS) to intercept and scrutinize network packets as they move across a network interface. This functionality lies at the core of Snort's operations, enabling it to conduct real-time analysis of network traffic in search of indicators of suspicious or malicious behavior.

Specifically, packet sniffing entails the process of capturing and examining network packets as they traverse a network interface. This initial phase serves as the foundation for the IDS to monitor the unprocessed network traffic, diligently scanning for any signs of potentially harmful or unauthorized activity. As such, we seamlessly transition from this essential packet sniffing function to the broader scope of Intrusion Detection without isolating it as a distinct feature demonstration.

3.2 Intrusion Detection

3.2.1 Workflow

In Snort 3, the intrusion detection system operates through a multi-step process to effectively monitor and analyze network traffic for potential threats. Here's an elaboration of how the intrusion detection system works in Snort 3:

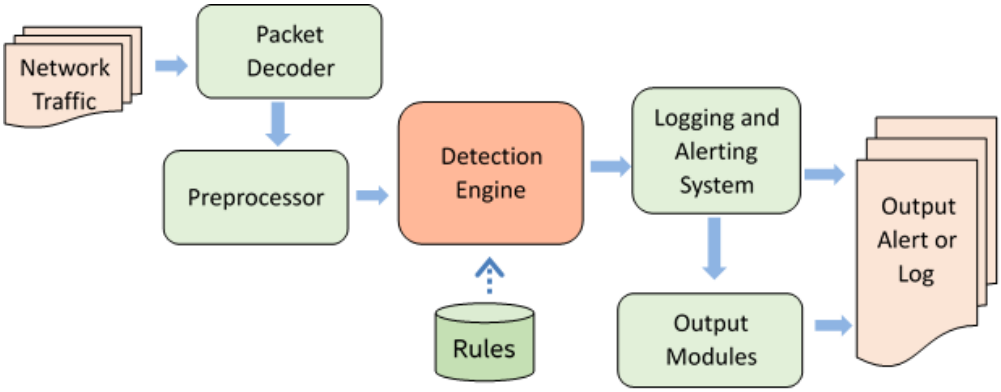

Figure 5: Intrusion Detection Workflow

3.2.2 Local File

For detecting intrusion we have to write some rules in local.rules file.

```
alert icmp any any -> $HOME_NET any (
    msg:"Ping attempt!";
    sid:1;
)
```

Figure 6: Local Rule for Intrusion Detection

3.2.3 Intrusion Attempt

Here is the intrusion attempt from the attacker

```
SEED-Ubuntu20.04:~$ ping 192.168.0.105
PING 192.168.0.105 (192.168.0.105) 56(84) bytes of data.
64 bytes from 192.168.0.105: icmp_seq=1 ttl=64 time=0.839 ms
64 bytes from 192.168.0.105: icmp_seq=2 ttl=64 time=0.746 ms
64 bytes from 192.168.0.105: icmp_seq=3 ttl=64 time=1.42 ms
64 bytes from 192.168.0.105: icmp_seq=4 ttl=64 time=1.22 ms
64 bytes from 192.168.0.105: icmp_seq=5 ttl=64 time=1.22 ms

--- 192.168.0.105 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4042ms
rtt min/avg/max/mdev = 0.746/0.992/1.418/0.256 ms
SEED-Ubuntu20.04:~$
```

Figure 7: Intrusion Attempt

3.2.4 Intrusion Detect

Here is the intrusion detection by using Snort3

```
09:33:10.950321 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 1, seq
 13, length 64
09:33:11.951630 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 1, seq
 14, length 64
09:33:12.951821 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 1, seq
 15, length 64
09:56:09.139626 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 1, length 64
09:56:10.142690 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 2, length 64
09:56:11.144296 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 3, length 64
09:56:12.146442 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 4, length 64
09:56:13.148888 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 5, length 64
09:56:14.151352 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 6, length 64
09:56:15.152558 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 7, length 64
09:56:16.154956 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 8, length 64
09:56:17.156566 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 9, length 64
09:56:18.159077 IP 192.168.56.101 > Ubuntu-Victim: ICMP echo request, id 2, seq
 10, length 64
^C
25 packets captured
```

Figure 8: Intrusion Detection

4 Conclusion

In summary, Snort3 stands out as a robust and adaptable intrusion detection and prevention system, playing a pivotal role in fortifying network security. Operating on predefined rules, it excels in real-time analysis of network traffic, offering heightened efficacy in identifying and thwarting diverse malicious activities. Whether employed for packet logging, packet sniffing, network intrusion detection, or intrusion prevention, Snort furnishes invaluable insights into network traffic, empowering organizations to guard against dynamic cyber threats. Its open-source framework and robust community support render it an indispensable asset within the realm of network security solutions, safeguarding the ongoing resilience of contemporary network infrastructures.

5 Resources

- Snort Official Website
- Youtube Playlist on Snort3 By Cisco Secure Firewall
- Youtube Video on Snort3 By GD Networking Newbie