Intrusion Detection Using Snort

R Siva Girish Prof. Prasad B Honnavalli Gaurav C.G
PES University Bengaluru
Department of Information Security And Cyber Resilience

Introduction

SNORT is an open-source network intrusion detection system. Network intrusion detection systems are used to monitor network traffic, analyse incoming packets and determine whether the incoming traffic is healthy traffic for the network or simply a malicious request by an attacker. Network intrusions can be of various forms such as a denial of service, man in the middle or network scans. Snort when equipped with a robust ruleset can handle these attacks and can also be programmed to alert or drop malicious packets as the case may be. Our rule set consists of rules to detect man in the middle attacks, usage of reconnaissance tools like Nmap and variations of denial of service attacks (DDoS).Our ruleset is available on
https://github.com/sivagirish81/Intrusion-Detection-Using-Snort .

Objectives

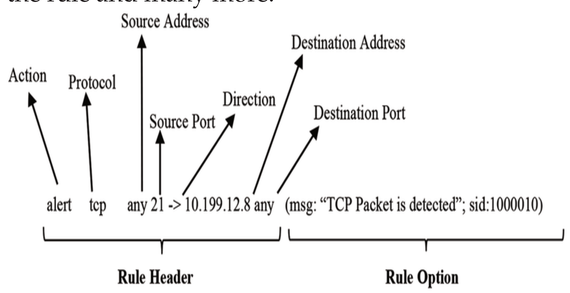
Network intrusion detection in the modern world is of paramount importance. As the number of people connected to the internet is increasing every day the potential threats such as cybercrimes are also constantly on the rise. Therefore it is important to know whether the network a person is connected to is secure or not. The cheapest and most feasible solution to this problem is to have a NIDS that can detect malicious packets, requests and drop any further request from the user that is sending them.

Challenges

One of the challenges faced was to understand the concept of Computer Networks, cause most of the intrusions that are performed are due to the vulnerabilities in the current TCP/IP model.
The other challenge we faced was to perform the attacks. In our work we have used a few tools. In order to use these tools one had to have sufficient knowledge of these tools work, their effectiveness and performance.
Understanding the inner workings of snort enabled us to understand how rules had to be written.

Rule Set

Snort is an open source software but the rulesets to detect various attacks are not necessarily open source.There are a lot of rules contributed by the open source community but a fair amount of the are unable to serve their purpose to detect the attack they are supposed to detect. Our goal was to develop our own open source rule set to detect a few selected attacks. In the process we have managed to detect man in the middle attacks , a few types of DOS and nmap scans. We have tested these rules by attacking a host using a VM running Kali Linux which is one of the most advanced open source penetration testing software.
Based on the gravity of the attack the user can decide to drop those incoming requests and proceed to service other meaningful requests from legitimate users.
Rules have been written for below range of attacks - :
1.Nmap Scans
2.Denial Of Service
3.Man in the middle
In Snort, the rules are divided into two parts.
Header section and option section.
The header section consists of type of output, the protocol, destination/source IP address along with their ports(ranging from 1 - 65535) followed by direction operator and dest/src IP address and port number.
The option section contains various option fields like msg to output the characters which are given as values to this field, content field to check for the given text in the message section of the packet, flag field to check if packet is a SYN/FIN/PSH, flow field which is attached to the pre-processor section of Snort component, sid field which is a unique value given to each rule of snort which indicates the process number of the rule and many more.



Description

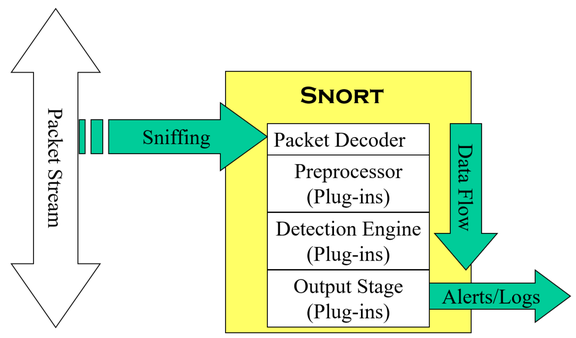


Figure 1: Snort - Architecture Source : Martin Roesch
Snort is comprised of a sniffing mechanism which sniffs all the packets along the input stream and sends it to the various plugins which perform specific operations and then unifies the output of all the plugins and alerts the corresponding violated rule

```
#Alert when nmap fin scan
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "Nmap FIN scan"; flags:F;sid
:10006; rev:1;)

#Alert when nmap ping scan
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "Nmap Ping Scan"; dsize:0;si
d:10007;rev:1;)

#Alert when Nmap tcp scan
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "Nmap TCP Scan"; sid:10008;
rev:2;)

#Alert when Nmap Xmas scan
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "Nmap XMAS Scan"; flags:FPU;
sid:10009;rev:1;)

#Alert when Nmap Null scan
alert tcp $EXTERNAL_NET any -> $HOME_NET any (msg: "Nmap Null Scan";flags:0;sid
:10010; rev:1;)

#Alert when Nmap udp scan
alert udp $EXTERNAL_NET any -> $HOME_NET any (msg: "Nmap UDP Scan"; sid: 10011;
 rev:1;)
```

Figure 2: A Few Rules Handpicked from our Ruleset based on nmap scans.
The above picture illustrates the rule header format.
Nmap is an advanced Reconnaisance tool to detect the nature of active devices connected to the same network

Man in the middle

Man in the middle attacks involves a third party that positions itself between one end host and another thereby illegally spying or modifying the packets. The goal of this attack is to either eavesdrop or impersonate another end host with intent to gain confidential information about the compromised users. This type of attack poses an immense danger to the end-users and hence must be prevented at all costs. The approach we used to tackle this problem was based on the ICMP protocol. The ICMP Protocol is designed in such a way that it has the capability to detect the shortest pathway in the network between two end hosts. Since in all cases of a man in the middle, the alternate host is introduced by the attacker and all the packets are being redirected to another host machine. This indicates that a longer route is being used despite the fact that a shorter route is available. Based on this concept we used Snort to detect ICMP packets of code 1 under type 5 which indicates that the datagrams have been redirected despite a shorter route being available.

Denial Of Service(DOS)

A DOS attack is one of the well-known threats, which focuses on stalling the server from servicing legitimate requests and sometimes cause complete shut down of the system. Few of the attacks for which these rules are developed are listed below.

SYN flood - Detected by checking the SYN requests sent from a particular source, if a certain threshold is crossed then we alert such packets.

FIN flood - Detected the same way as above but with the flag set to FIN.

Ping of Death - it causes failure at the server by sending fragmented ICMP packets to the server. By sending fragments which would total up to more than 65535bytes, the server would be overflowed when trying to combine these fragments which stops the functionality of the server.Detected by seeing the size of each fragment and the number of fragments received. If this count is higher we alert such packets.

Ongoing Research

The very existence of internet and it's exponential increase in traffic presents us with a unique challenge of curbing illegal attempts to access confidential information being transmitted across the web . We intend to write rules to detect Nmap scripted scans,complex man in the middle attacks as well us other more sophisticated variations of the distributed denial of service attacks.

References

[1] Jay and Caswell. Snort2.1 intrusion detection.

[2] Rafeeq Ur Rehman. Advanced ids techniques using snort, apache, mysql, php, and acid.