



Global Information Assurance Certification Paper

Copyright SANS Institute
Author Retains Full Rights

This paper is taken from the GIAC directory of certified professionals. Reposting is not permitted without express written permission.

Interested in learning more?

Check out the list of upcoming events offering
"Security Essentials: Network, Endpoint, and Cloud (Security 401)"
at <http://www.giac.org/registration/gsec>

An Analysis of the Snort Network Intrusion Detection System

by

Mark D. Tollison

Submitted

December 10, 2000

Research Paper Index

- [Overview](#)
- [Introduction](#)
- [What is Snort?](#)
- [Snort Command Line Summary and Usage](#)
- [Snort as a Packet Sniffer](#)
- [Snort as an Intrusion Detection System](#)
- [Conclusion](#)
- [References](#)
- [Links of Interest](#)

Overview

In today's world, everyone is increasingly dependent on the ability to have instant access to information. The explosion of the internet, along with wireless and broadband technologies, allow companies and individuals, unprecedented "Real Time" access to vast amounts of information. In our daily lives we are inundated with email, voice mail, facsimile, pager and other types of information. In our personal lives, we use our computers to chat with friends, listen to digitized music, make travel reservations and buy products. As Internet access costs have plummeted, corporations are using the Internet as the media of choice for corporate data and, increasingly, voice communications. Any outage in any of these systems is not only a nuisance but a major event to productivity. The Internet has and will continue to revolutionize the way business is conducted.

Unfortunately, there is a dark side to the use of the Internet. The many advantages, such as cost, openness and flexibility of this vast computer network are heavily impacted by security risks. It is a daily occurrence to read about another malicious hacker who has defaced a web site, gained unauthorized access to a large corporation's information resources, or shut down an E-commerce site via a distributed denial of service attack. The recent attack on Microsoft via a Unicode Bug or Web Server Folder Traversal Vulnerability, seemed to cause minor monetary damage to the company but this intrusion definitely raised credibility issues with its information security program [1].

Virus or malicious code is distributed via email and forces system administrators to scan for, and if found, remove the code from these systems. A recent report stated that about 50,000 viruses had been created in the last 14 years [2]. Some of these attackers are initiated by novices who are just trying out one of the many new "Hacker Tools" available on the Internet. However, I believe a greater threat is from the dedicated individual who has a financial, theological, or personal reason to disrupt a company's vital information flow. Espionage is a valid threat to the intellectual property of a company. There is no more cost effective way to gain a competitive advantage than to have access to your rivals' information.

These types of threats are just a few of the reasons that information security is becoming increasingly important. Just as we lock our homes and offices to keep out intruders, we must protect our vital computing resources from unauthorized access. One part of an overall information security initiative is the area of Intrusion Detection.

The focus of this research paper is to educate the reader about the need for an intrusion detection capability. Specifically, I examine the role of network intrusion detection and how one open source tool, Snort, can be used to establish a defense against many attacks. I will present some background information on the features of SNORT and list some examples against a widely used vulnerability scanning tool, Nmap [3].

[Index](#)

Introduction

Before I examine the specifics of Snort, I believe that it is important to quickly review the concepts of intrusion detection. As stated in the GIAC Level One Security Essentials Course, Intrusion Detection Systems (IDS) are separated into two main categories, network and host based systems. Each type of intrusion detection system complements the other and both should be used in an effective information security program. Host based systems allow administrators to effectively monitor back doors into systems. Network Intrusion Detection Systems (NIDS) give administrators the ability to determine who is trying to gain unauthorized access to critical computing resources. Network intrusion detection systems are effective in learning about and monitoring the various exploits that intruders can use. One open source network intrusion detection system is Snort.

[Index](#)

What is Snort?

What is Snort? The following information from the Snort web site [4] describes the system.

"Snort is a libpcap-based [PCAP94] packet sniffer and logger that can be used as a lightweight network intrusion detection system (NIDS). It features rules based logging to perform content pattern matching and detect a variety of attacks and probes, such as buffer overflows [ALE96], stealth port scans, CGI attacks, SMB probes, and much more. Snort has real-time alerting capability, with alerts being sent to syslog, Server Message Block (SMB) "WinPopup" messages, or a separate "alert" file. Snort is configured using command line switches and optional Berkeley Packet Filter [BPF93] commands. The detection engine is programmed using a simple language that describes per packet tests and actions. Ease of use simplifies and expedites the development of new exploit detection rules. For example, when the IIS Showcode [IISBT99] web exploits were revealed on the Bugtraq mailing list [BTQ99], Snort rules to detect the probes were available within a few hours." [5]

In order to better understand what the description says, let's begin the analysis of Snort by examining some of the command line features.

[Index](#)

Snort Command Line Summary and Usage

Snort has an extensive set of features and command line options. These are documented fully in the accompanying "Readme" file which is included in the distribution. A sample of the basic command line options are summarized in the following table.

snort -[options] <filters>

COMMAND LINE OPTION	DESCRIPTION
-A <alert>	<alert> mode can be either Full, Fast or None. Type of alerting to the alert file Full mode, normal mode. Fast mode - Only Timestamp, Message, IPs, and P. None - No alerting.
-a	Display ARP packets
-b	Log packets in tcpdump format. FASTEST OPERATION
-c <cf>	Use configuration or rules file <cf>.
-C	Dump the ASCII characters in packet payloads only. NO HEXDUMP
-d	Dump the application layer data
-e	Display/ and log the layer 2 packet header data
-F <bpf>	Read BPF filters from file <bpf>. Can be used for complex filters
-i <if>	Use network interface <if>.
-l <ld>	Log packets to directory <ld>.
-N	Turn off logging. Alerts still function normally.
-p	Turn off promiscuous mode sniffing.
-r <tf>	Read the tcpdump-generated file <tf>.
-v	Verbose output to console. Limited use. Will cause slowdown and possible packet loss.
-V	Show the version number and exit.
-?	Show the usage summary and exit.

Also, two very valuable features are the support for filters and rules. Filters allow the program to be tailored for specific needs such as monitoring a single host computer or subnet. The use of rules allow Snort to be used as a fully functional intrusion detection system. In the next sections I will run some simple examples using a small computer network to show how Snort can be used in its two modes, as a packet sniffer and as an intrusion detection system.

[Index](#)

Snort as a Packet Sniffer

For this example, I am running the Windows version of Snort on a small two computer network. The first computer is running Windows 98 and is running the Snort system. The second computer is a RedHat Linux system. To test Snort as a sniffer, I will use the ICMP Ping program to generate Echo packets.

In an MSDOS window, the command line for Snort is the following.

```
D:\Snort\program\snort-1.6.3\Binary\snort.exe -l D:\Snort\TEMP -i 4 -a -e -d -A full
```

For this demonstration, I started Snort, sent Echo requests from the Linux computer, and terminated Snort using CTRL C. Snort displayed the following statistics in the MSDOS window.

```
=====
Snort received 26 packets and dropped 0 (0.000%) packets
```

```

TCP: 0 (0.000%)
UDP: 0 (0.000%)
ICMP: 24 (92.308%)
FRAGS: 0 (0.000%)
ARP: 2 (7.692%)
IPv6: 0 (0.000%)
IPX: 0 (0.000%)
OTHER: 0 (0.000%)

```

Folder 169.254.193.10

Folder 169.254.17.196

I am not going to discuss how to interpret all the packet information. This information can be obtained from various references including "Using TCP/IP"[6]. It is important to notice that Snort reads the packets from network driver correctly and stores the information in files in a very readable format. These files can be viewed manually or using one of many third party add-ons. Data mining or data filtering techniques could be used to find important events within large data files.

Index

For this next example, the small twocomputer network is again used. However, two configuration changes are made. The first is that Snort now uses a rules database. Being open source, the rules database is frequently updated by various authors. Also, various sites have downloadable rules files for Snort. The latest rules database can be obtained from the Snort web site.[] Once downloaded, the rules database must be configured with site specific information. It only took me a few minutes to add the required IP addresses information.

The second configuration change is that I am using the Nmap port scanning utility to generate suspect data packets. [7] This versatile and fully featured utility can be used to map and identify hosts with networks. I am going to limit my Nmap scan to use TCP fingerprinting techniques to try to guess the operating system on the computer running Snort. If all goes

well, I should be able to obtain a good comparison of Snort operation and logging to the actual attack vector.

In an MSDOS window, the command line for Snort is the following.

```
D:\Snort\program\snort-1.6.3\Binary\snort.exe -l D:\Snort\TEMP -c
D:\Snort\Rules\12062k.rules.txt -i 4 -a -e -d -A full
```

For this demonstration, I started Snort, ran Nmap with TCP/IP Fingerprinting option. Upon Nmap completion, I terminated Snort. The Snort system displayed the following statistics in the MSDOS window.

```
=====
Snort received 3103 packets and dropped 0 (0.000%) packets

Breakdown by protocol:
  TCP: 3097      (99.807%)
  UDP: 1         (0.032%)
  ICMP: 3        (0.097%)
  FRAGS: 0       (0.000%)
  ARP: 2         (0.064%)
  IPv6: 0        (0.000%)
  IPX: 0         (0.000%)
  OTHER: 0       (0.000%)
=====
```

Investigating the D:\Snort\TEMP directory revealed that Snort had generated an alert file. This ASCII alert file contained ten alert entries. One of the entries was the following.

```
[**] IDS05 - SCAN-Possible NMAP Fingerprint attempt [**]
12/10-10:05:07.959989 0:60:97:8A:F5:31-> 0:50:BA:A4:EA:97 type:0x800 len:0x4A
169.254.193.10:42856 -> 169.254.17.196:139 TCP TTL:51 TOS:0x0 ID:4196
**SF*P*U Seq: 0xA5757576 Ack: 0x0 Win: 0x1000
```

This alert report that an Nmap TCP/IP Fingerprinting attempt was made against the host computer. Snort effectively generated an alert showing that this attempt had occurred.

[Index](#)

Conclusion

The Snort lightweight Intrusion Detection System has all many features a system administrator needs to establish an efficient network based intrusion detection system. Through third party add-ons, the system can be enhanced to make logs reviews and administrator easier. This demonstration shows that Snort can be an effective part of intrusion detection architecture. Hopefully, this introduction will peak your interest in this system and I invite you to explore Snort in more detail.

[Index](#)

References

- [1] Delio, Michelle. "Hackers Crack Into MS System." 27 October 2000.
URL: <http://www.wired.com/news/culture/0,1284,39778,00.html> (10 December 2000)
- [2] "Virus Woes Lead to New Tactics." Investors Business Daily. 30 November 2000.
- [3] Fyodor. "General Information." URL: <http://www.insecure.org/nmap/index.html#intro>

(10 December 2000)

- [4] Roesch, Martin "Snortorg Latest News" URL: <http://www.snort.org/snortnews/news.asp>
(10 December 2000)
- [5] Roesch, Martin "What is Snort?" URL: http://www.snort.org/what_is_snort.htm
(10 December 2000)
- [6] Ray, John. "UsingTCP/IP." January 1999. QUE Corporation.

[Index](#)

Linksof Interest

Snort 1.6.3 Source Code - <http://www.snort.org/Files/snort-1.6.3.tar.gz>

Snort 1.6.3-patch2 - <http://www.snort.org/Files/snort-1.6.3-patch2.tar.gz>

Snort-1.6.3-win32-source.zip - <http://www.snort.org/Files/snort-1.6.3-win32-source.zip>

Snort-1.6.3-win32-static.zip <http://www.datanerds.net/~mike/binaries/snort-1.6.3-win32-static.zip>

Snort Solaris Package - <http://www.snort.org/Files/snort-1.6.3-sol-2.6-sparc-local>

SnortWIN32 Graphic Interface for Snort - www.xato.net/downloads<http://www.xato.net/downloads>

Nmap Downloads - <http://www.insecure.org/nmap/index.html#download>

[Index](#)