Study of snort-based IDS

Conference Paper · February 2010

DOI: 10.1145/1741906.1741914 · Source: DBLP

Mohuya Chakraborty
Institute of Engineering & Management

Indraneel Mukhopadhyay
Institute of Engineering & Management

Study of Snort-Based IDS

S Chakrabarti
Institute of Engineering
& Management
Y-12 Salt Lake Electronics Complex
Kolkata - 700091
+91-33-2357-2059
director@iemcal.com

M Chakraborty
Institute of Engineering
& Management
Y-12 Salt Lake Electronics Complex
Kolkata - 700091
+91-98746-28587
mohuyacb@yahoo.com

I Mukhopadhyay
Institute of Engineering
& Management
Y-12 Salt Lake Electronics Complex
Kolkata - 700091
+91-94335-52806
imukhopadhyay@gmail.com

ABSTRACT

General trend in industry is a shift from Intrusion Detection Systems (IDS) to Intrusion Prevention Systems (IPS). In this paper, we have investigated the motivations behind this trend. In addition, we have surveyed some of the available IDS/IPS tools. Real time analysis of several Internet attacks was done using SNORT, "the de facto standard for intrusion detection/prevention", and Nmap in order to study malicious behavior of our network. Simulation results of Scanning attack as well as DoS attack performed on test computer have been provided. A comparative analysis of the results obtained with Snort and EagleX showed the higher efficiency of Snort.

Category & Subject Descriptor:

C.2 Computer-Communication Networks C.2.0 General - Security and Protection

General Terms

Security

Keywords

Intrusion Detection System (IDS), Intrusion Prevention System (IPS), Host-based IDS, Network-based IDS, SNORT, Nmap.

1. INTRODUCTION

The common sentiment in industry is that Intrusion Detection Systems (IDS) will soon become old-fashioned. In order to replace IDS a system, which is much more capable and robust must be developed and implemented. Researchers all over the world are thinking of Intrusion Prevention System (IPS). IPS is not a system of new technology. Rather it is simply an evolved version of IDS incorporated by firewalls and network layer filtering to provide a more active filtering of malicious content. Our goal is to understand both IDS and IPS Snort and Nmap.

An Intrusion Detection System, IDS, is designed to detect unwanted attacks on systems. The primary responsibility of IDS is to detect unwanted and malicious network traffic. The two main types of ID systems are (i) ID systems which operate on a host and are called host-based ID systems (HIDS), and (ii) ID systems which operate on network data flow and are called network-based ID systems (NIDS). The focus of our research is primarily on NIDS [1]-[6].

1.1. Network-Based IDS

An NIDS, such as Snort, generally consists of one or more network based sensors, which monitors and filters all network traffic. The sensors help in filtering network traffic and generating alerts when suspicious or malicious traffic gets detected. There are two main types of detection techniques: *signature matching*, and *statistical anomaly detection*. Signature filters (used by Snort) is the most widespread type of filter. A signature detection filter simply looks at the signature of the data flow which the filter monitors. If the signature matches any of some predefined set of signatures, the filter will generate an alert. This technique is very simple to implement. However, its main drawback is that one must already know the signatures of malicious packets in order to detect them. This requires an additional amount of overhead to continually update the system with the most current malicious signatures. In addition, the attacker can slightly modify the attack in order to avoid matching the pre-existing signature rules for the original attack.

Statistical anomaly filters are meant to monitor a system and detect if something abnormal occurs. The normal behavior of a system is determined under normal, safe operating conditions, and then "exceptional" events are identified. When the system behavior deviates far from the "normal" behavior, alerts are generated. The major problem with this is that system behaviors change with time. As a result, the system behavior will deviate more and more from the "normal" reference model initially determined. As a result, there is a need for the reference model to evolve. However, if the reference model is updated too often, an attacker could spread out the attack over a longer period of time and possibly go unnoticed in the system [2].

ICWET'10, February 26–27, 2010, Mumbai, Maharashtra, India.

1.2. Host-Based IDS

Though we have not focused on HIDS, we briefly give a general description as to how it differs from NIDS. Unlike NIDS, which inspects all network traffic, HIDS inspects and detects abnormal behavior within a host. The central principle behind the effectiveness of HIDS is that attackers generally leave a trace of their activities. For instance, they might install software on a computer they have taken over. In general, an HIDS will maintain the the database of system objects that it should monitor. The database contains important information about these system objects including attributes, modification time, size, etc. In addition, the database may contain a checksum or hash of the system objects. These objects may later be used for comparison to the current system objects. If at some point of time an object becomes inconsistent, the system may generate an alert. However, the attacker may somehow gain access to the database. As a result, simply matching a hash does not guarantee that an intruder has not tampered the file in question. For files which are very dynamic, this checksum technique will not be effective.

IDSs are products designed to detect unwanted accesses or manipulations of a system. However, they typically do not prevent or protect from attacks. IDS systems are also the oldest systems available. On the contrary, the purpose of an Intrusion Prevention System (IPS) is to not only detect that an attack is occurring, but also to prevent it [7], [8]. In order to do so, it can be considered to be an advanced combination of a firewall and IDS. To illustrate this, we first consider what a firewall does. The idea behind firewalls comes from their use in building construction to prevent the spread of fires [6]. In a networking sense, they perform a similar function. Instead of preventing fire from spreading from one building to another, they prevent certain network traffic from traveling from one network section to another. It typically does this by examining portions of the IP packets and decides whether or not to let the packet pass [4], [8]. Firewalls can work in both directions, preventing traffic both into and out of a network. A firewall can be considered a very simple type of IPS in that it operates using IP addresses and ports and other header information, but it does not look into the contents of the packet [9]. An IPS is more advanced in that it can use application-layer information to attempt to determine intent of the packet. As mentioned, it combines the power of filtering that a firewall has, with the power of detection like an IDS has, along with the ability to prevent attacks. For these reasons IPSs are considered to be among the promising of network technologies. Similar to IDSs, IPS can be divided into several types. Also like IDSs, these include Network Based IPS (NIPS) and Host Based IPS (HIPS). The distinction is very similar to that of IDSs and so will not be discussed further.

Recent trends in industry show that more and more companies are choosing IPS based solutions over IDS based solutions, primarily due to the need to actively block worm and hacker attacks, instead of passively monitoring them as an IDS system would do. Although legitimate traffic is often blocked just as malicious traffic is, most system administrators believe that the benefits certainly outweigh the downsides, especially when considering the significant damage that a successful worm or hacker can have on an organization [10]. It should be noted that the blocking of legitimate traffic is not a unique problem in IPS systems. This problem occurs in any type of system that blocks traffic, including firewalls. The ability to determine the malicious nature of a packet with 100% accuracy is simply not possible. For this reason, any security implementation will suffer from false positives. However, a system administrator must weigh the benefits of blocking good traffic (false positive) versus allowing some bad traffic (false negatives) to determine how strictly to check data.

After an introduction in section I, we describe various types of attacks on IDS in section II. Computer vulnerabilities are discussed in section III. Simulation environment and real time analysis of attacks on IDS by using Snort and Nmap are discussed in section IV. Section V concludes the paper.

2. ATTACK ON IDS

There are many different types of attacks which can corrupt a system. Those attacks can be generally grouped into the following categories:

Confidentiality: allows the attacker to gain access to information without authorization.

Integrity: allows the unauthorized attacker to affect the state of the system. This could mean either affecting the system state or any data residing on or passing through the system.

Availability: principle of availability is violated if the attacker can prevent an authorized user from accessing a system resource.

Control: attack grants an unauthorized attacker a privilege in violation of the access control policy of the system, the attack is on the control principle. This attack can give means to further attacks on confidentiality, integrity, and/or availability.

2.1 Scanning Attacks

Scanning attacks can be used to assimilate information about the system being attacked. Using scanning techniques, the attacker can gain topology information, types of network traffic allowed through a firewall, active hosts on a network, OS and kernel of hosts on a network, server software running, version numbers of software, etc... Using this information, the attacker may launch attacks aimed at more specific exploits.

The above was gathered by launching a stealth SYN scan. This scan is called stealth because it never actually completes TCP connections. This technique is often referred to as half open scanning, because the attacker does not open a full TCP connection. The attacker sends a SYN packet, as though you he were opening up a real TCP connection. If the attacker receives a SYN/ACK, this indicates the port is listening. If no response is received, the attacker may assume that the port is not open. This is only one type of scan technique, there are many more available.

2.2 Denial of Service Attacks

There are two main types of denial of service (DoS) attacks: flooding and flaw exploitations. Flooding attacks can often be very simply implemented. For example, one can launch a DoS attack by just using the ping command: ping -f victim. This will result in sending the victim an overwhelming number of ping packets. If the attacker has access to greater bandwidth than the victim, this will easily and quickly overwhelm the victim. As another example, a SYN flood attack sends a flood of TCP/SYN packets with a forged source address to a victim. This will cause the victim to open half-open TCP connections - the victim will send a TCP SYN/ACK packet and wait for an ACK in response. Since the ACK never comes, the victim eventually will exhaust available resources waiting for ACKs from a nonexistent host.

2.3 Penetration Attacks

Penetration attacks contain all attacks which give the unauthorized attacker the ability to gain access to system resources, privileges, or data. One common way for this to happen is by exploiting a software flaw. For instance, in July of 2002 an exploit was found in sshd challenge response handling code which allowed the attacker to execute arbitrary code as the user running sshd (often root). This attack would be considered a penetration attack. Being able to arbitrarily execute code as root easily gives an attacker to whatever system resource imaginable. In addition, this could allow the user to launch other types of attack on this system, or even attack other systems from the compromised system.

3. COMPUTER VUNRABILITIES

There are different types of errors that arise in a system by which an attacker can gain access to the system.

Input Validation Error: two most common forms are buffer overflows and boundary condition errors. These both occur when a program does not properly check the input it receives from the system. A buffer overflow attack exploits the boundaries of some buffer, resulting in some data overwriting adjacent memory locations. A boundary condition error occurs when input to a program causes the program to exceed some boundary. For instance, the input may cause the system to run out of memory.

Access Validation Error: occurs when access control policy of the system is flawed. As a result, the attacker can utilize this to gain control of the system.

Exceptional Condition Handling Error: system becomes vulnerable because some type of exception has arisen. This exception could either not be caught, or handled incorrectly, thus allowing the attacker to exploit the system.

Configuration Error: error results as a fault of the end administrator whom is responsible for configuring the system. This is not a fault of how the system was designed, but a fault of the user who incorrectly configured the system.

Race Condition: flaw in a system where the output exhibits unexpected dependence on the timing of events. Attackers can exploit race conditions with respect to denial of service attacks. Also, attackers can take advantage of programs which reach race conditions while stuck in a privileged state. While in this privileged state, the attacker could convince systems to perform illegal operations.

4. SIMULATION ENVIRONMENT AND REAL TIME ANALYSIS

The machine, which is used as test bed for Intrusion Detection software, was installed with a full set of server applications to as to mimic a real-live enterprise server environment as closely as possible. The Apache HTTP server version 2.2 was installed as the web server. Additional services included an SMTP server, telnet server, FTP and FTP servers etc. Some additional non-default programs were installed to keep the install as simple as possible (including games, office software, and programming environments).

4.1. IDS Tools and Associated Programs

The installed IDS tool was the latest stable-version of Snort [11]-[14], currently version 2.0.1. By default, Snort logs straight to text files. Depending on how Snort is configured, it can log an extremely large amount of data, making it exceptionally hard to parse through and find appropriate material to review and assess. Optionally, Snort can be configured to log directly to a database. To this end, we installed MySQL5.0 on the server and configured the server to allow Snort to log on to it. This involved creating the schema in the MySQL database and creating users and setting permissions correctly. The snort configuration file was then setup to allow logging to the recently created MySQL database. We then installed a program, EagleX IDS Center2.1 [15]. EagleX is a full working IDS with database backend and PHP data analyzer front-end along with Snort IDS – powerful intrusion detection system. EagleX configuration and management software for Snort can easily be done using the wizards, e-mail notification feature, etc. It uses online updates of Snort rule set. Preconfigured Eagle X configuration tool can adapt to any system environment very easily. It provides a web front-end to display, analyze, and query information from Snort. Figure 1 shows the topology of the network of the test bed machine.

4.2 Snort Rules Configuration

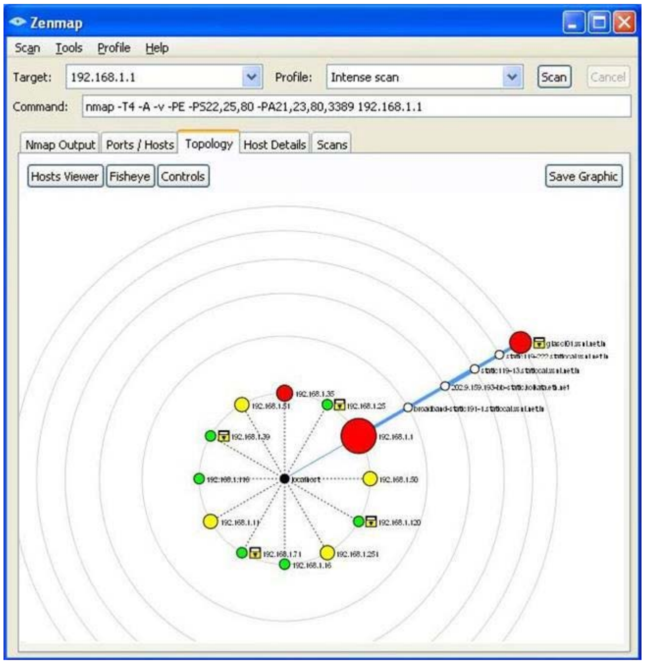
For testing out various attacks, Snort was configured with some rules that are inherent to Nmap – Zenmap 2.84. These rules include checks for Denial of Service (DoS) attacks, malware, viruses, and exploits, among many others [12].

4.3 Attacking the Server

We attacked our server using several different methods to see how it would reacts and logs the attack information.

4.3.1 Scanning Attack

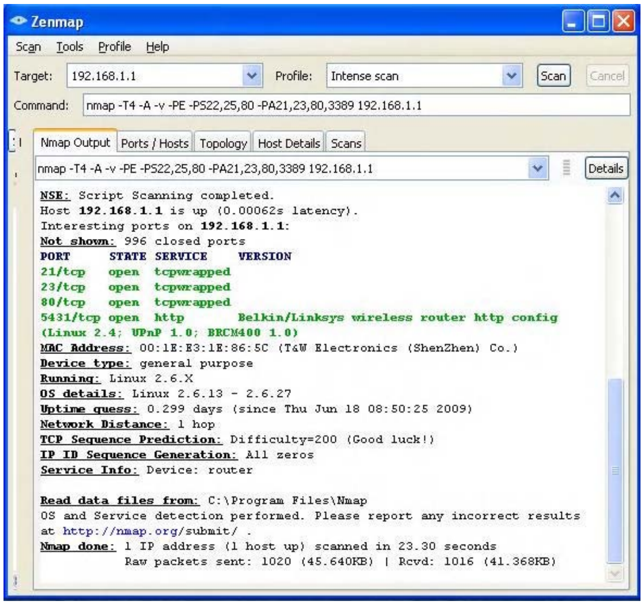
We launched several different types of scans (using Nmap – Zenmap 2.84) to try to discover as much information about our victim as possible. The first type of scan was a SYN stealth scan. Next, we launched a host enumeration and TCP scan to see if the system runs httpd, sshd, smtp, DNS, pop3d, imapd, or port 4564: Figure 2 shows scanning attack on test computer. An intense scan attack SYN is launched on the system to see different ports at the machine.



Target: 192.168.1.1 Profile: Intense scan

Command: nmap -T4 -A -v -PE -PS22,25,80 -PA21,23,80,3389 192.168.1.1

Figure 1. Topology of the test-bed machine network



Target: 192.168.1.1 Profile: Intense scan

Command: nmap -T4 -A -v -PE -PS22,25,80 -PA21,23,80,3389 192.168.1.1

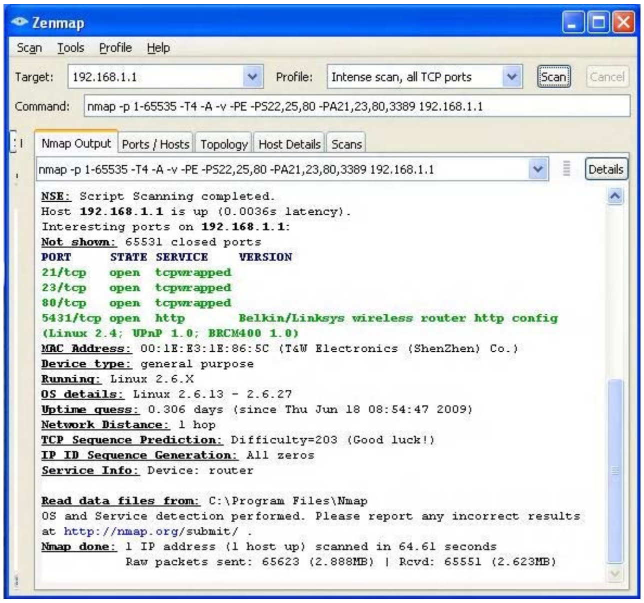
```
nmap -T4 -A -v -PE -PS22,25,80 -PA21,23,80,3389 192.168.1.1
NSE: Script Scanning completed.
Host 192.168.1.1 is up (0.00062s latency).
Interesting ports on 192.168.1.1:
Not shown: 996 closed ports
PORT     STATE SERVICE    VERSION
21/tcp   open  tcpwrapped
23/tcp   open  tcpwrapped
80/tcp   open  tcpwrapped
5431/tcp open  http       Belkin/Linksys wireless router http config
(Linux 2.4; UPnP 1.0; BRCM400 1.0)
MAC Address: 00:1E:E3:1E:86:5C (T&W Electronics (ShenZhen) Co.)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.13 - 2.6.27
Uptime guess: 0.299 days (since Thu Jun 18 08:50:25 2009)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=200 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: Device: router

Read data files from: C:\Program Files\Nmap
OS and Service detection performed. Please report any incorrect results
at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 23.30 seconds
           Raw packets sent: 1020 (45.640KB) | Rcvd: 1016 (41.368KB)
```

Figure 2. Scanning Attack on test computer

4.3.2 Denial of Service

We launched a very simple denial of service (DoS) attack by flooding the victim with ICMP echo requests. The purpose is to observe the ability of Snort to log the attacks rather than actually conducting a successful DoS attack because in reality, this would not have much of an effect since both computers are running on a LAN with identical bandwidth.

Figure 3 shows the output when we do scan of the machine by flooding it with packets. As number of attacks sent is more than the host computer can handle, there is DoS attack. Snort logs this kind of attack in the DoS category.



Target: 192.168.1.1 Profile: Intense scan, all TCP ports

Command: nmap -p 1-65535 -T4 -A -v -PE -PS22,25,80 -PA21,23,80,3389 192.168.1.1

```
nmap -p 1-65535 -T4 -A -v -PE -PS22,25,80 -PA21,23,80,3389 192.168.1.1
NSE: Script Scanning completed.
Host 192.168.1.1 is up (0.0036s latency).
Interesting ports on 192.168.1.1:
Not shown: 65531 closed ports
PORT     STATE SERVICE    VERSION
21/tcp   open  tcpwrapped
23/tcp   open  tcpwrapped
80/tcp   open  tcpwrapped
5431/tcp open  http       Belkin/Linksys wireless router http config
(Linux 2.4; UPnP 1.0; BRCM400 1.0)
MAC Address: 00:1E:E3:1E:86:5C (T&W Electronics (ShenZhen) Co.)
Device type: general purpose
Running: Linux 2.6.X
OS details: Linux 2.6.13 - 2.6.27
Uptime guess: 0.306 days (since Thu Jun 18 08:54:47 2009)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=203 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: Device: router

Read data files from: C:\Program Files\Nmap
OS and Service detection performed. Please report any incorrect results
at http://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 64.61 seconds
           Raw packets sent: 65623 (2.888MB) | Rcvd: 65551 (2.623MB)
```

Figure 3. DoS Attack Simulation

5. CONCLUSION

More and more intelligent and creative people are finding new ways to attack computer systems everyday. Until recently, system administrators were limited to choosing from a small variety of protection mechanisms, including the useful, yet limited, firewalls and Intrusion Detection System solutions. Our experiments with Snort show that it is in fact a useful tool, especially when considering the speed with which rules can be released to protect against newly discovered attacks. The industry is currently moving in a different direction this time by investing in Intrusion Prevention Systems. These systems combine the benefits of both firewalls and IDSs to block traffic and detect malicious behavior. Though EagleX has built-in IDS but still Snort is better equipped to handle attacks. It also adds the power of preventing attacks by looking at the application-layer information within the packet. This powerful combination will certainly serve the industry well over the next several years.

6. REFERENCES

[1] Bace, R., & Mell, P., “Intrusion Detection Systems”, NIST *Special PublicationonIntrusionDetectionSystem*. http://www.snort.org/docs/nist-ids.pdf.

[2] Cabrera, J., Lewis, L., Qin, L, Lee, W., & Mehra, R . , “Proactive Intrusion Detection and Distributed Denial of Service Attacks – A Case Study in Security Management”, *Journal of Network and Systems Management*, **Vol. 10**, **No. 2**, (pp 225-253), 2002, June.

[3] Comer, D. (2004), “Computer Networks and Internets”, 4th ed. Upper Saddle River, NJ: Pearson Prentice Hall.

[4] de Vivo, M., de Vivo, G., & Isern, G., “Internet Security Attacks at the Basic Levels”, *ACM SIGOPS Operating Systems Review*, **Vol. 32, No. 2**, SIGOPS, ACM, April 1998, (pp 4-15), 1998, April.

[5] Firewall (networking). Wikipedia. http://en.wikipedia.org/wiki/ Firewall_%28networking%29

[6] Intrusion-detection system. Wikipedia. http://en.wikipedia.org/wiki/ Intrusion_detection_system

[7] Intrusion-prevention system. Wikipedia. http://en.wikipedia.org/wiki/ Intrusion_prevention_system

[8] IPS gaining ground over IDS. (2005, February 14). Network World. http://www.networkworld.com/news/2005/021405ids.html

[9] NSS Group. (2004, January). Intrusion Prevention Systems (IPS). http://www.nss.co.uk/WhitePapers/intrusion_prevention_systems.htm

[10] Oppliger, R., “Internet Security: Firewalls and Beyond”, *Communications of the ACM*, May 1997/**Vol. 40, No. 5**, (pp 92-102) .

[11] Roesch, M., “Snort – Lightweight Intrusion Detection for Networks”, Proceedings *of LISA ’99: 13th Systems Administration Conference*, Seattle, WA, USA, November 7-12, 1999.

[12] Snort. http://snort.org/

[13] Whitman, M., “Enemy At The Gate: Threats to Information Security”, *Communications of the ACM*, **Vol. 46, No. 8**, August 2003, (pp 91-95).

[14] Zhang, X., Li, C., & Zheng, W., “Intrusion Prevention System Design”, *The Fourth International Conference on Computer and InformationTechnology (CIT'04)*, 2004

[15] Kistler, U. “Eagle-X Preconfigured Intrusion Detection System” http://www.engagesecurity.com/products/eaglex