Global Information Assurance Certification Paper

Cost Efficient Open Sourced NIDS
Keat Lim
November 21, 2000

Introduction

A Network Intrusion Detection System ("NIDS") makes it possible to monitor networks for any unauthorized activity. The idea of NIDS is not to prevent threats and attacks to the network but to provide an early warning system. There are several commercial IDS packages available that do take the advantage of possible signatures and automatically instruct a firewall to deter the attack (this method must be planned with care to not run into a possible Denial of Service attack on yourself). Unfortunately some of the commercial solutions are quite costly and may not run on all platforms. Using freely available tools on the Internet, it is possible to turn those old 486 computers into efficient NIDS boxes. In this project we will be evaluating a couple of software, mainly running a Linux based OS, with some freely available tools such as Snort. These machines can be distributed all over the network and the logs will be transmitted back to a central logging server where it will be parsed and audited.

The OS and tools

Linux is the operating system of choice just because it is freely available and will run on most old 486 boxes. Old 486 and classic Pentium machines are often discarded and can be easily salvaged. If possible look for identical batches of machines (i.e. the same model number) because if they have similar hardware, these machines can be cloned easily by using tools like Symantec Ghost or dd in Unix. The IDS tools used can also be compiled under some of the more widely used Unix flavors, therefore you can also use any old Sun boxes. Running it under any Microsoft Windows based OS will be trickier but not impossible

Snort is a lightweight NIDS tool that is freely available and is capable of analyzing real-time traffic. Rules are created to detect various attacks such as Distributed Denial of Service, backdoor attempts and buffer overflows. These rule databases can be created from scratch by the user or is downloadable from various Internet sites such as arachNIDS. arachNIDS provides an updated list of commonly known network attack signatures and catalogs each attack with a unique ID that provides additional information about the attack. Snort is available in a number of choices; in source code form for those who do not mind compiling or in binary form for several Solaris, Linux and Windows flavors.

The Setup

This will not be a step by step guide therefore will not go into too much details on the actual setup itself.

The Linux machine should be configured with the bare minimum for the OS to run. There is no need for any additional services like ftp or smtp and should be turned off. If you're not sure what services may still be running, use commands like netstat or lsof to assist you in removing the unwanted services. To enable the logging of the data into a central logging machine, configure the syslog.conf file appropriately. For example, the following line would send any logs with priority info or higher to the syslogd daemon on the central logging machine at 192.168.0.1

```
*.info            @192.168.0.1
```

Download the latest Snort source or binary and install it. Configure Snort to your preference and if you are logging this to another machine, use the –s function to log to syslog. The latest rules may be obtained from snort.org or whitehats.com. Here is an example rule:

```
alert tcp 192.168.0.1 555 -> any any
(msg:"BACKDOOR SIGNATURE - Possible PhaseZero Server Active on Network";
content:"phAse";flags:PA;)
```

The rule consists of the rule header followed by the rule options in the section enclosed in parenthesis. The rule option is optional and used for more accurate fingerprinting. In this rule header the action is to generate an alert if there is a TCP packet with the source IP of 192.168.0.1 and source port of 555 to any other IP or port. The rule option contains a descriptive message, a content search of the packet for a string "phase" and any TCP flags of PA (PSH and ACK). The following is an actual log captured by Snort on a cablemodem network:

```
11/07-00:53:00.152565 [**] fP-Login [**]
  4.48.xxx.xx:1488 -> 24.3.xxx.xxx:21
11/07-00:53:01.512073 [**] FTP-Password [**]
  4.48.xxx.xx:1488 -> 24.3.xxx.xxx:21
```

```
11/08-05:17:43.583351 [**] IDS279 - BACKDOOR ATTEMPT-Subseven v2.1 [**]
  64.111.xx.xx:2136 -> 24.3.xxx.xxx:27374
11/08-05:17:44.254496 [**] IDS279 - BACKDOOR ATTEMPT-Subseven v2.1 [**]
  64.111.xx.xx:2136 -> 24.3.xxx.xxx:27374
11/08-10:19:18.115472 [**] IDS279 - BACKDOOR ATTEMPT-Subseven v2.1 [**]
  38.26.xxx.xx:3179 -> 24.3.xxx.xxx:27374
11/08-10:19:18.920771 [**] IDS279 - BACKDOOR ATTEMPT-Subseven v2.1 [**]
  38.26.xxx.xx:3179 -> 24.3.xxx.xxx:27374
11/08-10:38:05.251556 [**] MISC-DNS-version-query [**]
  24.141.xxx.xxx:1286 -> 24.3.xxx.xxx:53
```

In the course of two days, there were a few ftp login attempts, a few Subseven backdoor attempts and queries for a DNS service. The only services available on the machine was ftp and dns and could be a false positive (alerts that may trigger an event but may not be an attack) but since there was no services open on port 27374, this was a definite probe from a Subseven tool to search for any Subseven backdoors. This backdoor attempt may be a regular occurrence because there are plenty of tools that 'script-kiddies' use to probe massive subnets at a time.

In addition to Snort, another tool that has been proven useful is an IP logger. Portsentry, protolog and iplog are the few loggers out there that will log any TCP,UDP or ICMP attempts. These tools logs any attempts made on the server therefore will capture additional information that a Snort rule may not. The caveats are that these logs may grow substantially and must be filtered or rotated regularly if space is a constraint.

To set up a central logging server using syslogd, make sure that there are plenty of storage spaces because the logs from the NIDS machines do tend to generate quite a bit of information, especially if the use of an IP logger is used. The initial attempt of a NIDS will generally generate a lot of false positives. After verifying the various network services that are generating these false positives, configure your NIDS to filter out these devices. Some of the common ones are MS WINS servers probing the network, or other printer tools attempting to interface to the printers using SNMP and may be legitimate. Once properly tuned, the central logging server should start to receive the logs you need and use some shell or perl script to automatically generate an alert to your email or your preference so you can be warned while it is happening.

Verdict

The NIDS is a powerful tool and when available with minimal costs, may be a valuable tool. Snort, while not as powerful as some of the commercially available tools, can be deployed easily and with the use of Linux, there are no additional costs. The data generated will assist you in keeping a more secure environment. Use of these NIDS in high traffic areas like in the DMZ or on the same segment as your web servers may increase your awareness of the type of traffic generated, especially if it is open to the Internet.

References

ArchNIDS Center – Advanced Reference Archive of Current Heuristics for Network Intrusion Detection Systems URL: http://www.whitehats.com/ids/index.html

Freshmeat.net - One of the largest index of Linux software and tools (Use it to search for tools mentioned in this paper) URL: http://www.freshmeat.net

Iplog – A TCP/IP traffic logger. URL: http://ojnk.sourceforge.net/

Redhat – One of several Linux distributions URL: http://www.redhat.com

Roesch, Martin. 'Snort – Lightweight Intrusion Detections for Networks' URL: http://www.snort.org/lisapaper.txt

Rpmfind.net – Archive and index of RPMs (Resource Package Managers) that are compiled binaries used by some Linux distributions. Useful for those who do not wish to compile the tools URL: http://rpmfind.net

Snort – Lightweight NIDS URL: http://snort.org

Syslog-ng – Replacement for the commonly used syslogd. Provides additional features such as filtering message based logs versus the priority/facility pairs in syslogd only. URL: http://www.balabit.hu/products/syslog-ng/