Academic Science

International Journal of Electronics, Electrical and Computational System
IJEECS
ISSN 2348-117X
Volume 4, Issue 10
October 2015

Intrusion Detection using Packet Sniffer

Shail Shah
Department of EXTC,
Dwarkadas J. Sanghvi COE
Mumbai,India

Akshit Shah
Department of EXTC,
Dwarkadas J. Sanghvi COE
Mumbai,India

Sahil Shah
Department of EXTC,
Dwarkadas J. Sanghvi COE
Mumbai,India

Shivani Bhattcharjee
Department of EXTC,
Dwarkadas J.Sanghvi COE
Mumbai,India

Abstract-

Packet sniffer is computer software which runs in a network attached device which intercepts and monitors traffic passing over a communication networks. Sniffing is the method in which the software receives all the frames in the data link layer passing through the device network adapter. In order to capture these packets the NIC device is set in the promiscuous mode and the sniffer decodes them. This paper focuses on the basics of packet sniffer, its implementation for Intrusion Detection and its development on various platforms. The development of packet sniffer allows the users to identify the threats and discard it along with the additional features which weren't available in the previous softwares.

Keywords: Intrusion, packet sniffing, promiscuous, NIC (Network interface card)

I.INTRODUCTION

Suppose a strange man is standing in front of your car. He looks around studying the surrounding and walks to the door and tries to open it. The door is locked and efforts are in vain. It seems your car is secure. So why to install an alarm? Similarly in computers we have firewalls, spam filters and activated password for authenticity. One might feel that computer is secured. But computers are still not totally safe even with the most advance protection. As a result we have alarms in cars similarly we must develop and implement intrusion detection techniques and tools in order to discover the threats and react against computer attacks.

The term Intrusion detection gives the information about detection of attacks and threats to the network covering a whole organization, and also providing remedies to those attacks. Responses which are automated basically include alerting an administrator via an email, console, shutting down the network, disconnecting internet links. Packet sniffer tools are used for intrusion detection but at the same time it can be used for malicious activities. Packet sniffers which are used for intrusion detection includes an engine, which has a property to analyze certain types of attacks in the network, such as packets floods and IP spoofing. Administrator is notified of the security attacks to the network by packet sniffer which gives an indication via email, console, shutting down the services, or even disconnecting internet links.

II. WORKING

The network interface card works in two modes

I) Non promiscuous mode (normal mode)

II) Promiscuous mode

The acceptance of packets in the network is based on matching of the MAC address of the Network interface card with that of packets. The comparison is made between the MAC address and if the comparison is successful the packets is accepted or else it is discarded out. This is done because NIC only recognizes the matched MAC address of the packets with its own. NIC behaves in two modes according to the requirement out of which non promiscuous mode means that the NIC is in don't care condition and it only reads the data which is directed to it and does not interfere with the other networks. This mode is usually not used and hence promiscuous mode is used which means that NIC is not intended and the packets are circulated all around the network. Packet sniffers which are used for sniffing sets the NIC in promiscuous mode of its own system due to this the all packets are received irrespective of packets being directed to a particular network. Hence for effective capturing and analysis of packets NIC is set in promiscuous mode.

The NIC of the node which is used to receive packet is set in the Promiscuous mode which is the basic requirement for the packet transfer from one node to another in the network. Memory is made available known as driver memory, in which the packet which is detected by the NIC at that particular node is stored. This memory content is directly passed to buffer known as kernel buffer. This kernel buffer can be used for various applications according to the packets in it [5]. Sniffing Process can be classified into three steps according to the process as. Packet sniffer collects raw binary data from the wire. This is mostly done in promiscuous mode. The Captured binary data is converted into a readable form. The packet sniffer takes the captured

network data and then the verification is done of its protocol based on the information extracted, and accordingly its analysis is started based on that protocol's specific features [1].

III. SNIFFING METHODS

Three types of sniffing methods are used. These are:

3.1. IP Based Sniffing

IP based sniffing is the most commonly used method of packet sniffing. The network interface card is set into promiscuous mode in IP sniffing. When network card is set into promiscuous mode then host will be able to sniff all packets of information in the network. The key point is that an IP based filter is used in IP sniffing in which the packets matching this IP address filter is only captured. The IP address filter is set in promiscuous mode so it can capture all the packets. This method only works in non switched network [3].

3.2. MAC based Sniffing

This is another method of packet sniffing and similar to IP based sniffing. The concept used is same as of IP based sniffing in which IP based filter is replaced by MAC address filter. The requirement of setting network interface card into promiscuous mode exists. In these methods sniffing packets are matched to MAC addresses [3].

3.3. ARP based Sniffing

This method works a little different as it does not put the network interface card into promiscuous mode. This is an effective method for sniffing in switched environment because ARP packets will be sent to us so it is not necessary to keep NIC in promiscuous mode. Sniffing is possible due to stateless nature of Address Resolution Protocol [3].

There are different types of network sniffing tools depending on the network, application or protocols are available in markets. This paper considers the primary and most useful packet sniffer like wireshark, tcpdump, Snort, Kismet etc.

IV. WIRESHARK

Wireshark is a packet analyzer which is used significantly in network troubleshooting and also in its analysis. Due to trademark issues, originally called Ethereal was renamed to Wireshark. Wireshark is cross-platform tool and to capture packets it uses pcap. It is compatible on various operating systems like Solaris and Microsoft Windows. It sees all traffic visible on that interface, not just traffic addressed to one of the interface's configured addresses but also broadcast/multicast traffic [1]. Wireshark allows the user to put the network interface card which is set in promiscuous mode, in order to make it visible to all traffic on that interface, not only traffic addressed to one of the interface's configured addresses but also it broadcasts or multicast traffic. On the contrary, when capturing is done with in promiscuous mode on a port of a network switch, not each of the traffic traveling through the switch will be sent to the port on which the capture is being done. Thus, promiscuous mode may not be sufficient for capturing packets and to see all traffic on the network. Wireshark is a tool which understands the structure of different networking protocols and hence it displays the encapsulation of packets with the meanings of fields of different packets specified by different networking protocols. It allows the users to capture the packets over the entire network on an interface at a given particular interval. There is a tool called capture tool in which the "Capture" menu is provided for the users to perform Packet Capture. Along with this it also provides several options for suiting the situations and the conditions that the analysts have in the mind while performing the process of capturing the packets. Analysts could

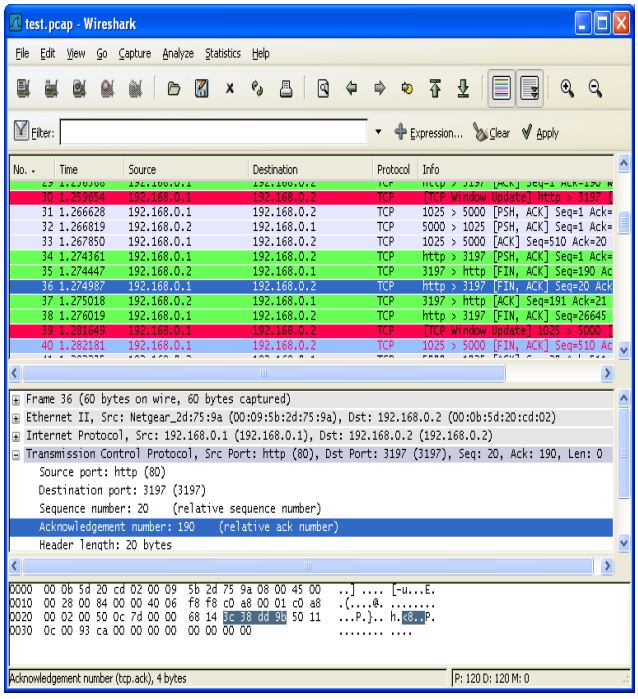

Figure 1:Wireshark

International Journal of Electronics, Electrical and Computational System
IJEECS
ISSN 2348-117X
Volume 4, Issue 10
October 2015

even set filters to avoid capturing unwanted traffic. The data is always converted into packets when it passes through your network interface while surfing on the Internet .It hunts for those packets in your TCP/ IP layer during the transmission and it will keep, and present this data on the screen [6].

V. SNORT

Snort is a packet sniffer that can echo network packets or parts of them, to the screen or to a log file you specify. Snort can be used to diagnose the network — say, to verify that packets are actually reaching a target computer... Snort supports some rules for interpreting network traffic as it is a network intrusion detection system (NIDS). You can create a rule set to have Snort alert you to suspicious activity such as packets that match the profile of known attacks [2].

You edit the Snort configuration file if you want snort to automatically detect intrusions. This file begins with a number of options to tell Snort what its home network is, what tools to use to *pre-process* packets, what tools to use to format output logs (to enable XML output, for instance), and so on. You can leave most of these options alone, but you may want to adjust the HOME_NET variable to reflect your local network's IP address. You can have the variable pick up of an address from a network interface [2]

var HOME_NET $eth0_ADDRESS

Alternatively, you can pass network address specifications in square brackets:

var HOME_NET [192.168.1.0/24,10.120.0.0/16]

Similar variables sets external network addresses and then enable you to point to your local SMTP (mail), Web, SQL, and DNS servers. The value that matches any address is the default value of the variable.. Some Snort rules use pointers to monitor traffic that's destined for or sent from the relevant machines. As a general rule, you should leave these values set at their default of $HOME_NET because this practice lets you see random probes for vulnerable servers, even on systems that are not running them. Sometimes attacks are also seen on the servers you don't realize are running. Snort follows number of rules to detect specific types of attack . In practice, many Snort installations load rules from files in the Snort configuration directory or a subdirectory of it, such as *rules*. These files have names that end in ***.rules***. If a packet matches the criteria specified by a rule, Snort logs it. If a packet doesn't match any rules, Snort will not do anything so do not bother by traffic Snort isn't configured to notice. If you want to expand Snort's capabilities you will need additional rules. You can search for a particular rule or browse the rule list. If a rule matches the type of activity you want to monitor, you can copy the "Signature" field from the rule's description into one of your *.rules* files, or you can create a new *.rules* file and add a reference to the new file to ***snort.conf*** file. Snort will create a log directory tree much like the one it creates when you use Snort as a packet sniffer. The difference is that the log files contain less data; only packet that matches a rule is logged. There is a file called *alert*, which contains a summary of all the suspicious activity. Snort has detected. You can also send alerts in real time to computers using the Server Message Block (SMB). You must also specify a configuration file that contains a list that contains machines to receive the popup message[2].

VI. TCPDUMP

Tcpdump works as a packet filter that runs on the command line interface. Tcpdump displays TCP/IP and other related packets which is being transmitted or received over a network to which the computer is attached. Tcpdump is compatible on various operating systems and performs various functions according to it. It views the entire data of an Ethernet frame or other link layer protocol .Tcpdump also allows the user to intercept the packers and to display TCP/IP transmitted or received over a network to which the computer is attached[4]. According to the operating systems, tcpdump uses the libpcap library to capture packets like it uses WinPcaplibpcap library in the Windows. Tcpdump works in a sequence like it analyzes network behavior, then its performance and followed by applications that generate or receive network traffic. Tcpdump also analyses the network infrastructure itself by determining all the necessary routing is occurring properly or not and allowing the user to further remove the Source of a problem . It can also be used for intercepting and displaying the communications of another computer. It has a limitation of only reporting what it finds in the packet. In case an IP address is duplicated inside the packet, tcpdump has no ability to report anything else. TcpDump has a size of 484 KB which makes it economical in terms of memory management. Tcpdump does not have a user friendly Graphical User Interface (GUI) which is

also considered as a limitation of it. Hence it is required for the user to study those commands and get used with the command prompt screen[1].

VII. KISMET

Kismet is one of the application is an open source wireless network analyzer that run on Linux, UNIX and Mac OS X. It does not run in windows OS. Kismet is passive sniffer used to detect any wireless 802.11a/b/g protocol complaint network, even when the network has a non broadcasting hidden service set identifier. Kismet detects, log the IP range of any detected wireless network and reports it noise and signal levels. It can sniff all data packet from detected network. Kismet can be used to optimize and troubleshoot signals strength for access points and clients, as well as detect network intrusions. Kismet becomes very easy to use as it runs on GUI mode. Kismet [4].Kismet has three separate parts. A *drone* can also be used to collect packets, and then pass them on to a *server* for interpretation. A server can either be used in conjunction with a drone, or on its own, interpreting packet data, and extrapolating wireless information.. The *client* communicates with the server and displays the information the server collects. Kismet has the ability to log all sniffed packets and save them in a tcpdump/Wireshark or Airsnort compatible file format. "Per-Packet Information" is captured by Kismet. headers. Kismet also features the ability to detect default or "not configured", probe requests, network and determine what level of wireless encryption is used on a given access point. Kismet supports channel hopping in order to find as many networks as possible. This means that it constantly changes from channel to channel in a non-sequential manner and in a user-defined sequence with a default value that leaves big holes between channels.

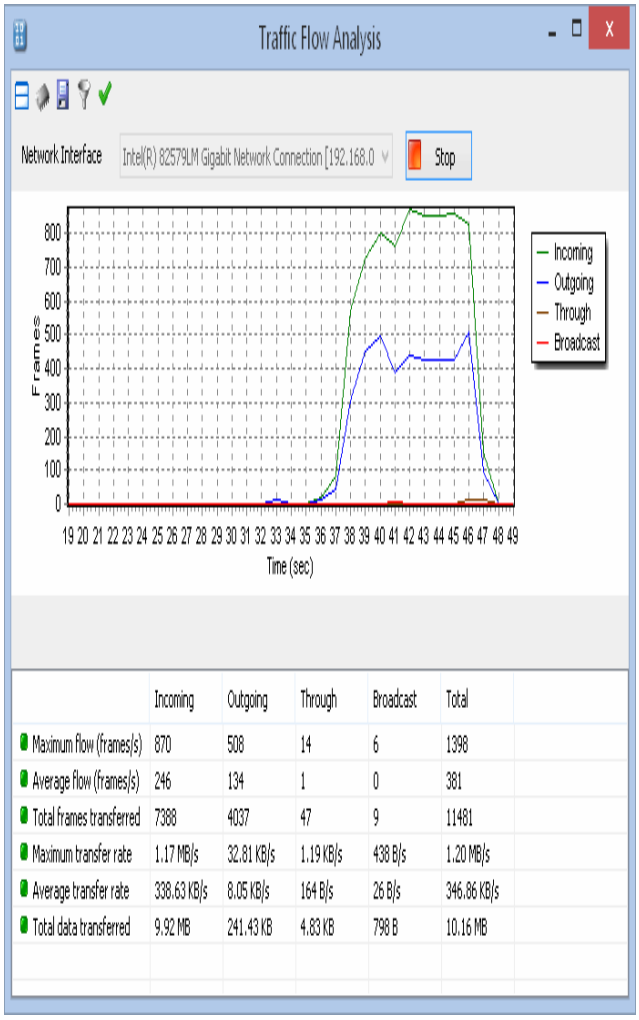

Figure 2:Traffic Floor Analysis

VII. SOFT-PERFECT NETWORK PROTOCOL ANALYZER

Soft-Perfect Network Protocol Analyzer is a professional tool for analyzing, debugging, maintaining and monitoring local networks and Internet connections. It captures the data passing through your dial-up connection and analyzes this data and then represents it in an easily readable form. Soft Perfect Network Protocol Analyzer is a useful tool for network administrators, security specialists, and network application developers. It is also useful for the people who want a comprehensive picture of the traffic passing through their network connection or segment of a local area network. This can be used to discard all network traffic except for the specific traffic patterns you wish to analyze because it contains fully-configurable filters . There is also a packet builder, which allows you to build your own custom network packets and send them into the network. This tool allows you to use the packet builder feature to check the protection of the network against any kind of attacks and intruders.

This tools works in promiscuous mode so that all packets are captured. Loopback communication is a type in which Soft-Perfect Network Protocol Analyzer can also capture network packets transmitted within the computer[1].

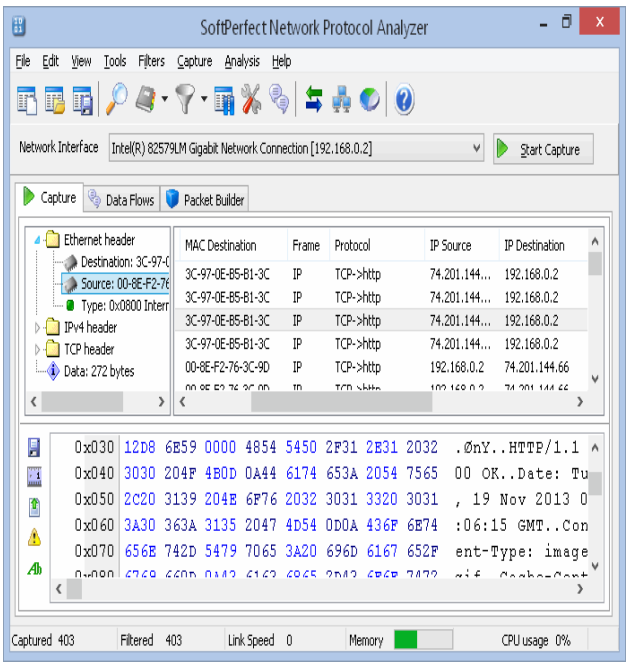

Figure3:Soft perfect network protocol analyzer

VIII. CONCLUSION

IDS tools are becoming the need for the day and for security not only in the corporate world but also for network users. Now days, security incidents are becoming more common and measures have to be taken to curb such incidents. Some of the firms are selling heuristic detection systems where artificial intelligence (AI) is used to detect intrusions. A highly effective hybrid class of IDS called Network Node IDS (NNIDS) where agents are placed on required host within the network which is to be protected. There are numerous available tools which are used to capture network traffic, but there are limitations in their working. Sometimes users have to use different tools for analyses of traffic since some of the tools only capture traffic network without performing any analyses. Some tools trace IP packets only and some tools capture tcp packets. Finally, cooperation with not only other IDS but also other network security components is mandatory to achieving a holistic network security posture for organizations of the future.

REFERENCES

[1] Pallavi Asrodia, Hemlata Patel, "Network traffic analysis using packet sniffer", International Journal of Engineering Research and Application (IJERA), Vol.2, pp. 854-857, Issue 3, May-June 2012.

[2] www.linux-mag.com/id/1358

[3] Ryan Splanger, "Packet sniffing detection with Anti sniff", University of Wisconsin-Whitewater, May 2003.

[4] Inderjit Kaur, Harkarandeep Kaur, Er. Gurjot Singh," Analysing Various Packet Sniffing Tools", International Journal of Electrical Electronics & Computer Science Engineering Volume 1, Issue 5 (October 2014), ISSN : 2348 2273

[5] Qadeer M.A., Zahid M., Iqbal A., Siddiqui M.R "Network Traffic Analysis and Intrusion Detection Using Packet Sniffer", ICCSN'10 Second International Conference, (2010), Page(s): 313-317(2010).

[6] Dulal C. Kar Felix Fuentes. Ethereal vs. tcpdump: A comparative study on packet sniffing tools for educational purpose. Journal of Computing Sciences in Colleges archive, Volume 20(4), pp 169-176(2005)