International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Packet Analysis with Network Intrusion Detection System

Rashmi Hebbar[1], Mohan K[2]

[1]M.Tech Student, Department of Computer Networks & Engineering,
Shrinivas Institute of Technology, Valachil, Post, NH 48, Farangipete, Karnataka

[2]Associate Professor, Department of Computer Science & Engineering,
Shrinivas Institute of Technology, Valachil, Post, NH 48, Farangipete, Karnataka

Abstract: ***Attacks on a computer network grow stronger each and every day. Network intrusion Detection System is one of the fundamental components to monitor and analyze the traffic to find out any possible attacks in the network. They are the safety measurements of any network. NIDS plays an important role in privacy security. But the problem is that at what level these NIDS will efficiently able to work? In this paper, the framework for the network intrusion using anomaly method by considering machine learning algorithm.***

Keywords: Wireshark; Snort; Network Traffic; Packet Analysis

1. Introduction

Data packets are the basic entities of all communication systems. Security of a network thus implies security of the data packets. The enormous attacks from the internet increase day by day. The quality of service is became more issue hence it should require powerful traffic analysis and distribution engine for network application. The complete packet inspection is required to examine the data part along with the header content of the packet. A packet analyzer is a computer software or hardware that can intercept and log traffic passing through a digital network or part of a network. As data streams flow across the network, the sniffer captures each packet and eventually decodes and analyzes its content according to the appropriate specification. DPI is widely used approach in industry as primary intrusion analysis method to improve the QoS. The complete packet analysis helps the network operator to obtain the profile of each application running inside the network and act as the primary way for intrusion detection system. It became the responsibility to make network security by using monitoring tool. This network monitoring tool helps the intrusion detection system easy by gathering the detailed information. Intrusion detection System (IDS) is a device or that monitors the network activities for malicious user and produces reports to a management station. IDS can be classified into two categories; Anomaly based detection and Signature based detection. Anomaly based detection uses traffic activities and creates threshold to identify the anomalies within the network [4].

In this work, we identify the intrusion by capturing the real time network traffic by using Snort and perform the detailed analysis on the captured packet using network monitoring tool called Wireshark. We capture the real traffic from the wired or wireless medium and perform the intrusion detection on snort. Snort uses the predefined rule that allows monitoring the status of a network. It can give compact information about the packet. If the packet matches the rule only then it is logged; otherwise packet dropped [1]. These logged file are imported to wireshark interface for the detailed analysis of each captured packet.

The rest of the paper is organized as follows: Section II will discuss the background of the proposed methods, section III says the Experimental setup and result and finally section IV ends with the conclusion.

2. Background

2.1 Snort

Snort [2] is a free and open source network intrusion detection system developed by Martin Roesch. It checks all the network traffic to identify any type of intrusion and alert the user. It uses the signature based detection and analyses of all the network traffic to identify the known attack signature. Snort is available under GNU [3], runs on multiple platform and freely configurable. All the components work together to find particular attack and take the appropriate action that is required for that particular attack.

Basically snort consists of following major components as shown in the figure.

- Packet Decoder
- Pre-processor
- Detection Engine
- Logging & Alert System
- Output Modules

You should use Times Roman of size 10 for all fonts in the paper. Format the page as two columns:

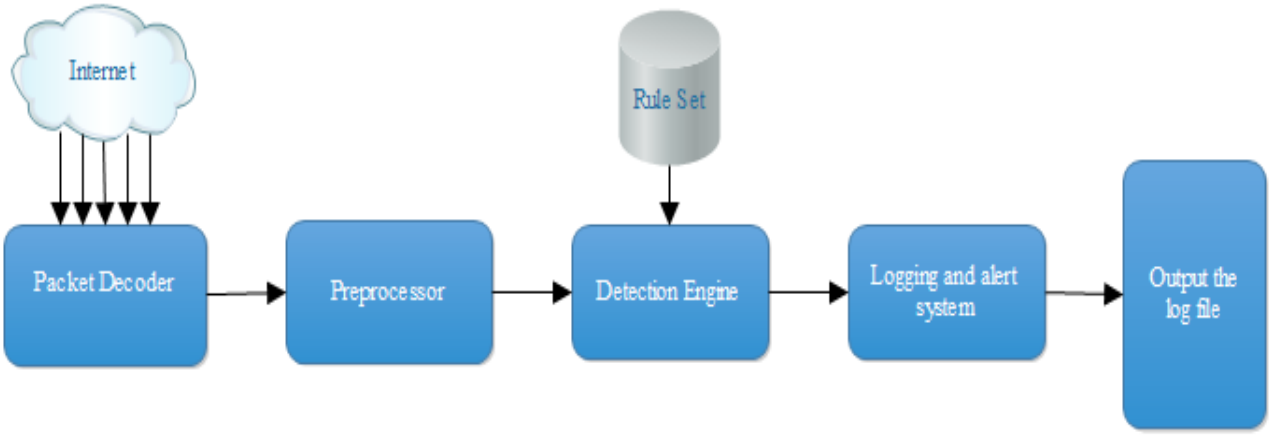

Figure 1: Components of Snort

- Packet Decoder
  The packet decoder captures the packet from different types of network interfaces and setup packets for preprocessing.
- Preprocessor
  Pre-processors are used to arrange and modify packets before analyzed by the detection engine. And they try to detect the some basic anomalies in the packet header. Preprocessors are very important for any IDS to prepare data packets to be analyzed against rules in the detection engine.
- Detection Engine
  Detection Engine is the heart of the snort. Its responsibility is to analyze all the packets passing through it for the sign of intrusion by making use of certain predefined rule. If the packet matches the rule appropriate action will be taken; otherwise the packet is dropped. It may also take different amount of time to respond for different packets irrespective of how many rules we define.
- Logging and Alert System
  Depending upon what detection engine finds out, the activity is logged or alert is generated. Logs are kept in simple text files or tcp-dump style files. The location of logs and alert can be modified using –l command in the command prompt.
- Output Modules
  It is used to control the type of output produced by the logging and alert system. Some of its function includes may be generating log reports, sending SNMP traps, logging into databases (like MySql), sending a message to sysylog server etc.

2.2 Wireshark

Wireshark is a network packet analyzer and it will try to capture the network traffic and display the packet data in detailed vie. Wireshark is very similar to tcpdump, but has a graphical front-end, plus some integrated sorting and filtering options. It is formally known as Ethereal [5] also captures the packet in human readable format. It is free and open source and used for network troubleshooting, analysis, software and communication protocol development. Live data can be read from different types of network including Ethernet, IEEE 802.11, PPP, and other interfaces. Captured files can be programmatically edited or converted via command line switches to the "editcap" program.

Wireshark will typically display information in three panels. The top panel lists frames individually with key data on a single line. Any single frame selected in the top pane is further explained in the tool's middle panel. In this section of the display, Wireshark shows packet details, illustrating how various aspects of the frame can be understood as belonging to the data link layer, network layer, transport layer or application layer. Finally, Wireshark's bottom pane displays the raw frame, with a hexadecimal rendition on the left and the corresponding ASCII values on the right [6], [7].

3. Experimental Setup and Result

The analysis of the signature based IDS can be done by installing the Snort, Winpcap and Wireshark on the computer. Snort is an open source network intrusion detection system, and can run on any platform. In this experiment the snort on windows platform is discussed. Snort has three main modes of operations.

Snort as Sniffer

Snort acts like tcpdump in sniffer mode. The Snort sniffer mode output is slightly different than the other command-line sniffers. It is actually very easy to read and you may find you prefer it for quick captures. One very nice feature of sniffer mode Snort is the network traffic summary at the end of the capture. Sometimes this can be a very useful troubleshooting tool for network administrators.

 # snort –v –d –e
-v Dump the packet header to standard output
-d Dump the packet payload(includes all TCP, UDP, ICMP packets)
-e Display the link layer data

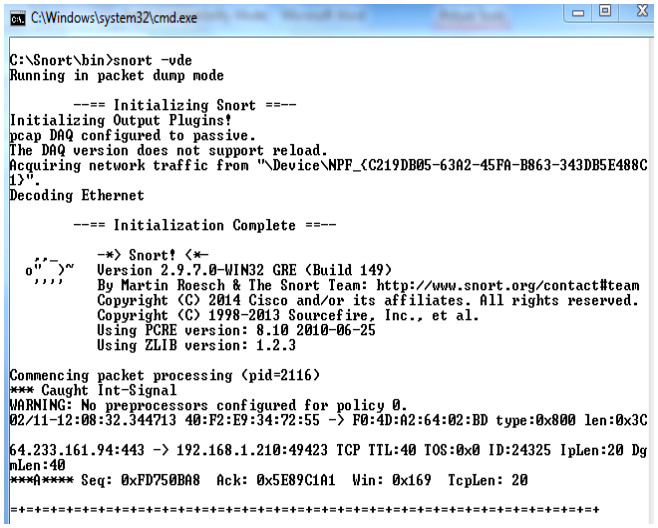

Figure 2: Snort sniffer mode

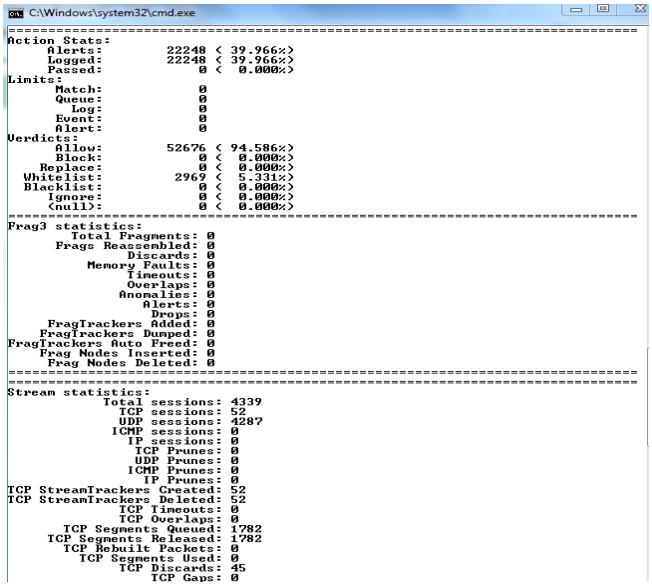

Figure 3: Snort NIDS mode

Snort as a Packet Logger

Once after the sniffing is done packet has to be logged. Logging is as simple as adding the -l option, followed by the directory in which you wish to store the logs. Snort's default logging directory is *\snort\log*. If you specify a directory that does not exist, Snort exits with an error message. We can use the -d, -a, and -e options to control the amount of information logged for each packet

 # snort –l {log-directory} will log the packet to specified folder.

If we want to log the data with home subnet then use following command.

 # snort –vde –l c:\snort\log –h 192.168.1.210/24

Snort as NIDS

When used as an NIDS, Snort provides near real-time intrusion detection capability. Snort does not log the captured file but it applies the rule, if any matches found then it will log or alert the system.

snort –c C:\Snort\etc\snort.conf will starts the snorts in NIDS mode. It will continuously watch the network until user hit Ctrl-C or using kill-USR. Bellow command display the alert in the log directory.

#snort –vde –l c:\snort\log –c c:\snort\etc\snort.conf

The generated alert file can be exported to the wireshark for the detailed analysis of captured packet. Wireshark has two filtering languages: one used when capturing packets, and one used when displaying packets. Display filters allow concentrating on the packets that the administrator is interested in, while hiding the currently uninteresting ones. The different protocol analysis is done here.

As soon as we open the file, it will display the all captured data in window. It shows the source and destination address of the packet, protocol, information of the packet. The bellow figure shows the network traffic captured by the snort and imported on wireshark interface. We are able to see the details of any packet by selecting any packet. The header part consist of source and destination IP address, protocol, time to live field, protocol version, header length and various types of services. The data of the header field is shown in the decimal form whereas data of payload is displayed in the hexadecimal form.

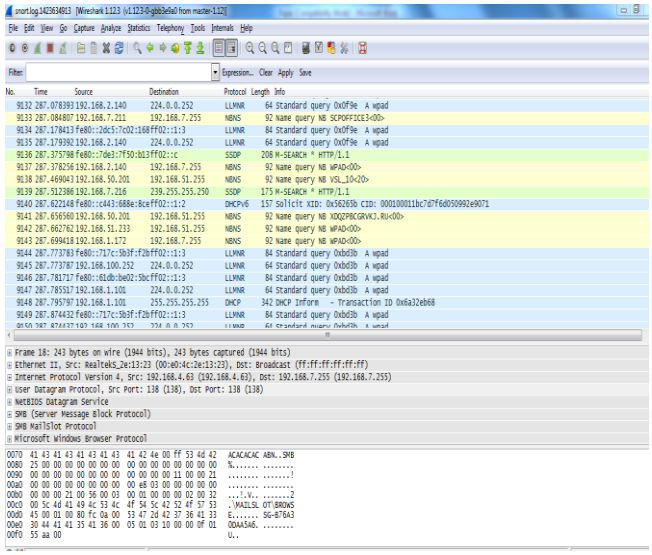

Figure 4: Packet Analysis using Wireshark

IO graphs are the most important facility provided by the Wireshark. At a time we can draw graph of five protocols of

different colors with different tick interval and pixels per tick on X-axis and units and scale on Y-axis. Styles can also be changed instead of lines shown you can select impulse, Fbar, dot from the drop down and you get a different look of the graph. The bellow figure shows the IO graph tcp, udp and http protocol in the captured network traffic.

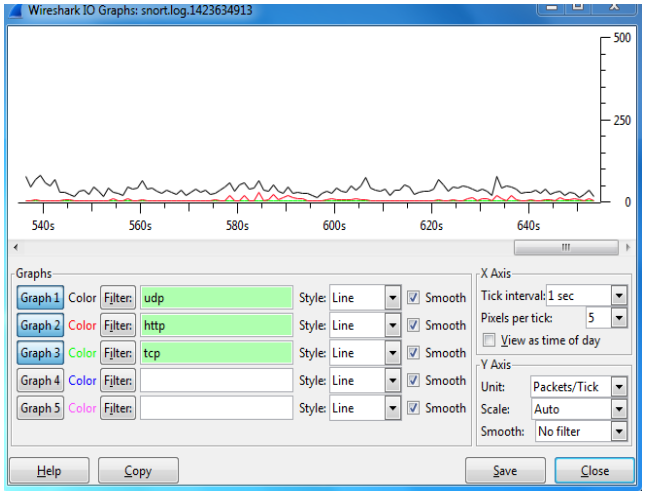

Figure 5: IO Graph of captured Traffic

4. Conclusion

In this paper the live network traffic is captured and perform the detailed analysis on captured packet using wireshark and Snort. Wire shark alone will not be able to generate an alarm or take a security action against the unauthorised access. If any intrusion occurs then the intrusion detection tool snort will prevent the system and alert will be generated for those abnormal activity. The graphs of captured files shows the details of network dynamics and insight into the problems.

References

[1] P. Agarwal, S.Satapathy, "Implementation of Signature-based Detection System using Snort in windows", International Journal of Innovation & Advancement in Computer Science (IJIACS), May 2014.
[2] R. Vanathi, S.Gunasekaran, "Comparison of Network Intrusion Detection Systems in Cloud Computing Environment", 2012 International Conference on Computer Communication and Informatics (*ICCCI* - 2012), Jan. 10 – 12, 2012, Coimbatore, India.
[3] T. Richard Stallman, "GNU General Public Licence", 1989,http://www.gnu.org/copyleft/gpl.txt
[4] J. Singh, Manisha J. Nene, "A Survey on Machine Learning Techniques for Intrusion Detection Systems", International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 11, November 2013.
[5] U. Banerjee, A. Vashishtha, M. Saxena, "Evaluation of the Capabilities of WireShark as a tool for Intrusion Detection", International Journal of Computer Applications (0975 – 8887) Volume 6– No.7, September 2010.
[6] file:///D:/Studies/M.Tech/Project/Papers/My%20Paper/1nd%20Paper/3.4%20Modes%20of%20Operation.htm
[7] file:///D:/Studies/M.Tech/Project/Papers/My%20Paper/1nd%20Paper/What%20is%20Wireshark%20%20%20Definition%20from%20WhatIs.com.htmr