INTERNATIONAL JOURNAL OF ADVANCE RESEARCH, IDEAS AND INNOVATIONS IN TECHNOLOGY

ISSN: 2454-132X
Impact factor: 4.295
(Volume 3, Issue 6)
Available online at www.ijariit.com

Analysis of Network Traffic by using Packet Sniffing Tool: Wireshark

Praful Saxena
Senior Faculty
iNurture Education Pvt Ltd, Bengaluru
shyam.praful@gmail.com

Sandeep Kumar Sharma
e-Learning Coordinator
iNurture Education Pvt Ltd, Bengaluru
sandeep.s@inurture.co.in

Abstract: With recent technologies, the growth network is highly increased. The number of network user are rapidly increasing day by day which reflects the growth of network traffic also. So it's very important to monitor networks traffic as well as its user's activities to keep the network smooth and efficient. For large network it's very complicated task to monitor the network, because large amount of packets are available. For this purpose packet sniffing is used. Packet sniffing is important in network monitoring to watch network activities which help network administrators to find out weakness of network. This paper focuses on sniffing network traffic working in different environment. Working of Network sniffing tool Wireshark .By using this packet sniffer we can capture traffic as well as we analyzed capture traffic. We can generate reports on the basis of analyzed traffic. Many protocol like TCP, IP, UDP etc. are implemented and filtering on basis of protocol is also done. Alerts generated on the occurring of suspected activities.

Keywords: Network Sniffing, Wireshark, Packet Capture, Packet Filters.

I. INTRODUCTION

Network sniffing is the process of reading the data packets sent over a network .This can be achieved by the specialized software Tool or hardware equipment. Sniffing can be used to capture login credentials or we can eavesdrop on chatting messages or we can also capture the files during the transmission over the network. The communication between the computers is done by broadcasting messages on a network using IP addresses [4]. Once a message has been sent on a network, the recipient computer with the matching IP address responds with its MAC address [2]. The sniffing process is used by hackers either to get information directly or to map the technical details of the network in order to create a further attack. Hackers are always in favour of sniffing, because it can be done for a longer time without getting caught.

II. BASICS OF SNIFFING

When packets transmit from source to destination then it travels through many intermediate devices. A node whose NIC is set in the promiscuous mode receives all information travels in network. Each NIC have physical address which is different and unique from another network. When packet arrives at NIC then hardware address of frame matched with physical address that NIC have, but if we set it in promiscuous mode then all packets will arrives at that interface. When we use switch which already pass filtered data then we perform some method to capture all data of network. When NIC accept packets, packets are copied to driver memory then it passes to kernel and kernel passes it to user application. The process of Sniffing is also based on the devices used in the network. On this basis the Sniffing can be done in two modes Active sniffing and Passive Sniffing [3]. Passive sniffing is intercepting packages transmitted over a network that uses a hub**.** It is called passive sniffing because it is difficult to detect. It is also easy to perform as the hub sends broadcast messages to all the computers on the network. Active sniffing is intercepting packages transmitted over a network that uses a switch**.** There are two main methods used to sniff switch linked networks, ARP Poisoning, and MAC flooding [3].

III. WIRESHARK TOOL

Wireshark is a network packet analyzer used to analyses the network Traffic. Such type of tools try to capture network packets and tries to display that packet data as detailed as possible [1].We can think of a network packet analyzer as a measuring device used to examine what's going on inside a network cable, just like a voltmeter is used by an electrician to examine what's going on inside an electric cable .Wireshark is perhaps one of the best open source packet analyzers available today. This tool is available for Unix and Windows platforms [1].

Network Administrators uses this tool for troubleshooting networks problem. A network Security is also analyzed by using this tool. Wireshark is a GUI based network capture tool [1]. The GUI window gives a detailed breakdown of the network protocol stack for each packet, colorising packet details based on protocol, as well as having functionality to filter and search the traffic, and pick out TCP streams there is a command line based version of the packet capture utility, called T. Shark [1]. TShark provides many of the same features as its big brother, but is console-based. It can be a good alternative if only command line access is available, and also uses less resources as it has no GUI to generate. Wireshark can peer inside the network and examine the details of traffic at a variety of levels, ranging from connection-level information to the bits comprising a single packet. This flexibility and depth of inspection allows the valuable tool to analyze security events and troubleshoot network security device issues. Binary Versions can be downloaded for windows and also available through standard software distribution.

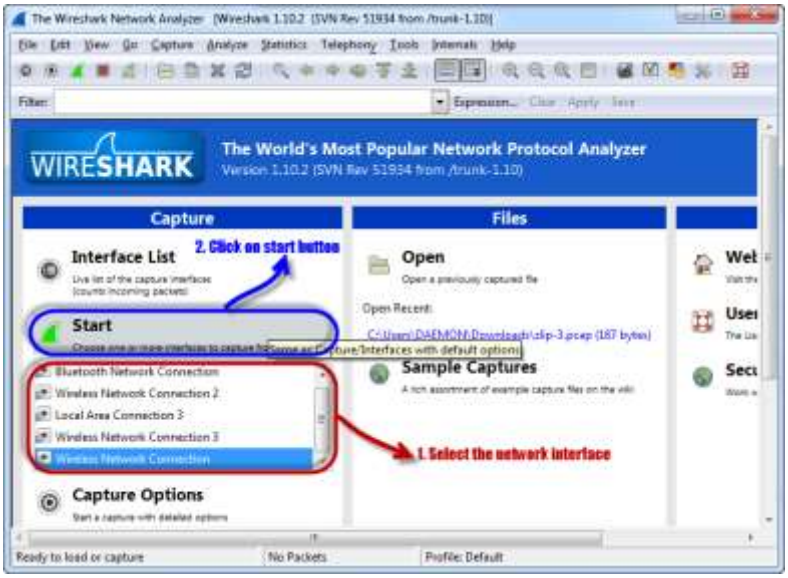

Fig.1 Wireshark Home Page

After installing the wireshark a homepage of this tool appears as shown in Figure 1. Wireshark is having lots of components to work with but some of the major components we will discuss here.

3.1 Wireshark Major Components

i) Command Menu: The command menus are standard pull down menus located at the top of the window. Of interest to us now are the File and Capture menus.

ii)Packet Listing Windows: The packet-listing window displays a one-line summary for each packet captured, including the packet number which is assigned by Wireshark; this is *not* a packet number contained in any protocol's header, the time at which the packet was captured, the packet's source and destination addresses, the protocol type, and protocol-specific information contained in the packet.

iii) Packet Header Detail Window: The packet-header details window provides details about the packet selected in the packet listing window. These details include information about the Ethernet frame and IP datagram that contains this packet. The amount of Ethernet and IP-layer detail displayed can be expanded or minimized by clicking on the right-pointing or down-pointing arrowhead to the left of the Ethernet frame or IP datagram line in the packet details window. The amount of Ethernet and IP-layer detail displayed can be expanded or minimized by clicking on the right-pointing or down-pointing arrowhead to the left of the Ethernet frame or IP datagram line in the packet details window. If the packet has been carried over TCP or UDP, TCP or UDP details will also be displayed.

iv) Packet Contents Window: The packet-contents window displays the entire contents of the captured frame, in both ASCII and hexadecimal format.

v).Packet Display Filter: The packet display filter field, into which a protocol name or other information can be entered in order to filter the information displayed in the packet-listing window which will be used further.

IV. CAPTURING A TRAFFIC

To capture the packets in network by using Wireshark here I am using an example of 2 PCS named PC1 and PC2. PC1 and PC2 are connected with the switch so that they can communicate each other. Before capturing a data the connection is verified by using ping command. The IP addresses of these two PCs are PC1 - 172.9.24.14 and PC2 -172.9.24.16. For capturing a packet using Wireshark following steps are too carried out:

i) For Capturing a packets first we have to select the available network interfaces. This is shown in a popup window from we have to select the interfaces.

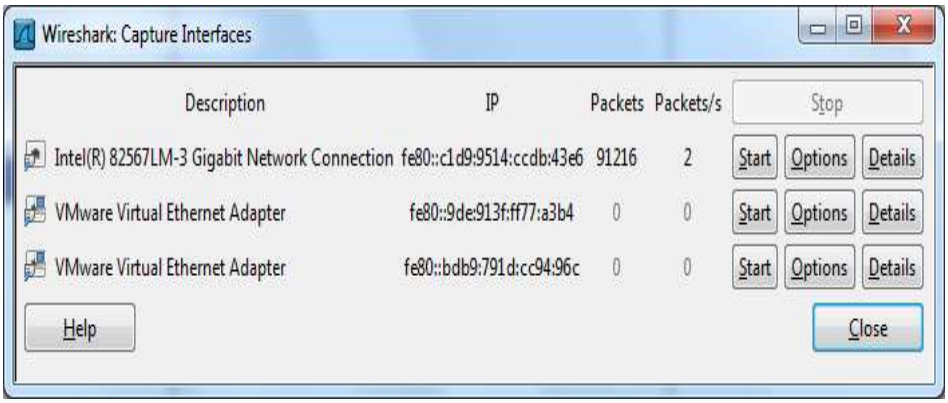

Fig 2. Interface Window

As shown in figure, we will select the Ethernet interface to begin the capturing of Packets.
ii). Now Generate the web traffic by using browser such Google chrome. And then start capturing the network.

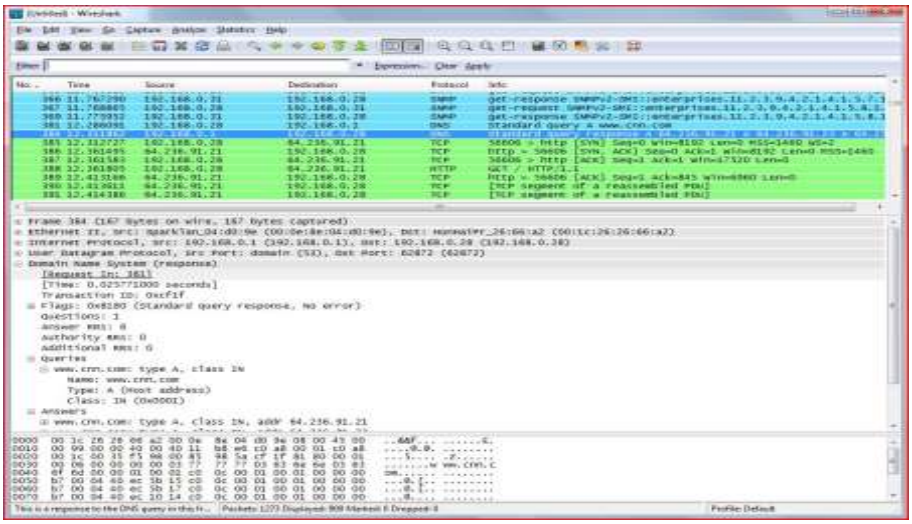

Fig 3. Wireshark Screen Capturing Packets

iii) In the screen shown above we see the flow of packets with detailed information such source IP address, Destination IP address and protocol used for the transmission.
iv) We can stop the capturing manually and it can easily done using STOP feature of Wireshark Tool.

V. ANALYZING THE RESULTS

After capturing the data using wireshark the window shown in figure 3 can be analyzed by intercepting it into 3 parts:

i) Packet List Panel: Each line in the top pane of the Wireshark window corresponds to a single packet seen on the network. The default display shows the time of the packet (relative to the initiation of the capture), the source and destination IP addresses, the protocol used and some information about the packet [5]. You can drill down and obtain more information by clicking on a row. This causes the bottom two window panes to fill with information.

ii) Packet Details Panel: The middle pane contains drill-down details on the packet selected in the top frame. The "+" icons reveal varying levels of detail about each layer of information contained within the packet [1]. In the example above, I've selected a DNS response packet. I've expanded the DNS response (application layer) section of the packet to show that the original was requesting a DNS resolution for www.cnn.com, and this response is informing us that the available IP addresses include 64.236.91.21.

iii). Packet Bytes Panel: The bottom window pane shows the contents of the packet in both hexadecimal and ASCII representations.

VI. WIRESHARK COLOR CODES

Wireshark is having a feature to differentiate the traffic according the protocols. Color codes of Wireshark is helpful when analyzing packets with Wireshark. In the example each row is color-coded. The darker blue rows correspond to DNS traffic, the lighter blue rows are UDP SNMP traffic, and the green rows signify HTTP traffic. Wireshark includes a complex color-coding scheme which you can customized.

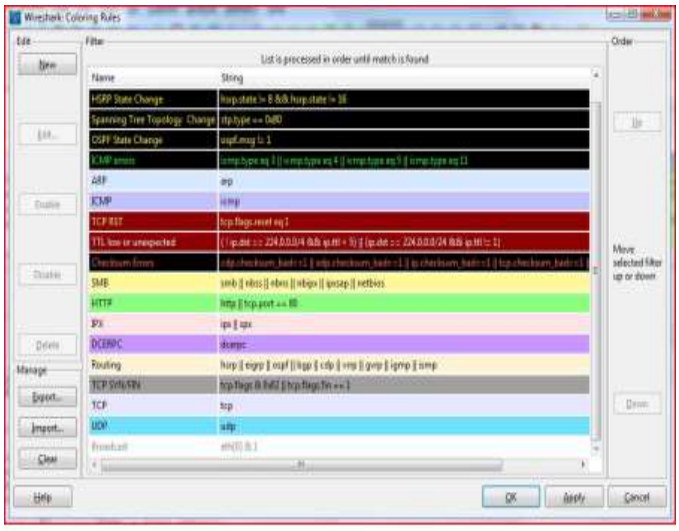

Fig 4. Wireshark Color Codes

VII. ANALYSING A TCP SESSION USING WIRESHARK

Start a capture, and generate some Web traffic by going to www.techipo.com, then stop the capture. Scroll back to the top of the capture trace. Find the first SYN packet, sent from your PC to the Web Server. This signifies the start of a TCP 3-way handshake. If you're having trouble finding the first SYN packet, select the Edit->Find Packet menu option. Select the Display Filter radio button and enter a filter of tcp.flags. (at this point you should get a list of the flags to choose from). Choose the correct flag, tcp.flags.syn and add == 1. Hit the Find button, and the first SYN packet in the trace should be highlighted [4]. A quick way to create a Wireshark Display Filter to isolate a TCP stream is to right click on a packet in the Packet List Panel and select Follow TCP Stream. This creates an automatic Display Filter which displays packets from that TCP session only. It also pops up a session display window, containing by default, an ASCII representation of the TCP session with the client packets in red and the server packets in blue.

The window should look something like Figure 7. This is very useful for viewing human readable protocol payloads, such as HTTP, SMTP, and FTP [4].

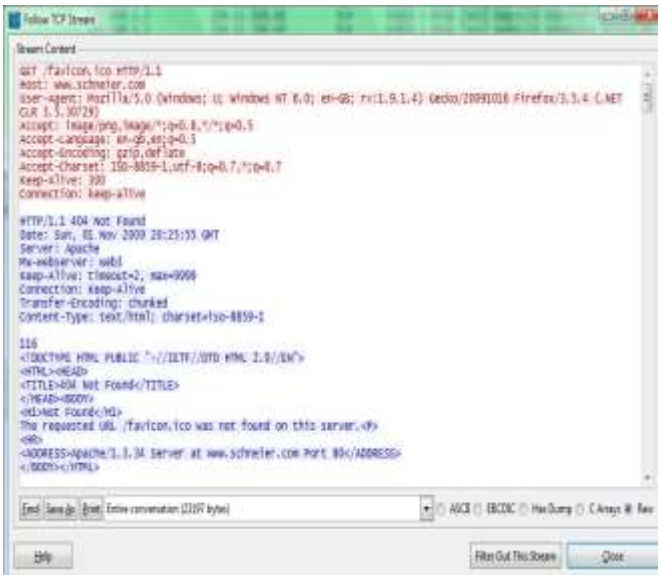

Fig 5 TCP Stream Capture

VIII. CONCLUSION

Packet sniffing is useful to Analyze the data during the Transmission in the network .Sniffing tools are useful to implement it. It can be used for network traffic monitoring, traffic analysis, troubleshooting and other useful purposes. Packet sniffers can capture things like passwords and usernames or other sensitive information. Networks Sniffing in non-switched network is easy but sniffing in switched network is difficult because we use switches in network which narrow the traffic and send to particular system, so for sniffing in this type of network we use some methods. There are many available tools. Packet sniffer can be enhanced in future by Incorporating features like making the packet sniffer program platform independent, and making tool by neural network Hence Sniffing should done in a manner to improve the performance of the network and to make it more secure .

REFRENCES

1. www.wireshark.org
2. Daniel Magers "Packet Sniffing: An Integral Part of Network Defense", May 09, 2002 SANS Institute 2000 – 2002.
3. Tom King, "Packet sniffing in a switched environment", SANS Institute, GESC practical V1.4, option 1, Aug 4th 2002, updated june/july 2006.
4. http://www.securityteam.com/unixfocus/Detecting sniffers on your network .html.
5. A. Dabir, A. Matrawy, "Bottleneck Analysis of Traffic Monitoring Using Wireshark", 4th International Conference on Innovations in Information Technology, 2007, IEEE Innovations '07, 18-20 Nov. (2007), Page(s): 158- 162(2007).
6. BoYu"Based on the network sniffer implement network monitoring Computer Application and System Modeling (ICCASM), 2010 *International Conference on Volume:* **7**, 2010, Page(s): V7-1-V7-3(2010).

