International Journal of Computer Networks and Communications Security

VOL. 2, NO. 5, MAY 2014, 179–187
Available online at: www.ijcncs.org
ISSN 2308-9830

Packet Sniffer – A Comparative Study

Dr. Charu Gandhi[1], Gaurav Suri[2], Rishi P. Golyan[3], Pupul Saxena[4], Bhavya K. Saxena[5]

[1] Assistant professor, Department of computer science, JIIT, Noida-201307

[2, 3, 4, 5] Student, Department of computer science, JIIT, Noida-201307

E-mail: [1]charu.kumar@jiit.ac.in, [2] gaurav.jiit128@gmail.com, [3] rishi.golyan5@gmail.com, [4] pupul9910103466@gmail.com, [5] bhavya.jiit128@gmail.com

ABSTRACT

Packet Sniffer is a tool which captures all the packets on the network irrespective of the final destination of the packet. Packet Sniffer could be used to monitor the bottlenecks in the network, alarm the irregular behaviour in the network, capture passwords and VoIP from any system in that network. This paper gives a brief introduction of what is a packet sniffer, its structure and what is its working. Then key features of top packet sniffing tools (i.e. Wireshark, TCPdump and Colasoft Capsa) are discussed. Further, the above tools are compared on the basis of characteristic behaviour and quantitative parameters. Finally, one gets the best tool amongst these three in a particular situation.

Keywords: *Packet Sniffer, Wireshark, Colasoft Capsa, TCPdump, Packet capture, Network monitoring tools.*

1 INTRODUCTION

Packet sniffing is a technology which captures the packets passing through the network in which it is installed. Packet sniffer is a tool which monitors all the network data. Furthermore, it can intercept and log incoming and outgoing traffic across the network.

The information that travels across a network is transmitted in form of "packets." For example, in a network, the packet is sent from one computer to another, initially the packet is broken up into smaller segments with destination and source address attached, and other useful information. But, if packet sniffer is installed in any of the node of the networks (either source or destination), then one can analyze the performance of network or could find out bottleneck in the network.

Packet sniffers are mostly used by network administrators as they help to troubleshoot the network problems, network intrusion detection system to monitor attackers, finding bottlenecks in networks and converting binary network data in human readable form such as collecting clear usernames and passwords, VoIP communications, mapping network, etc. These are some illegal uses of packet sniffer, unless the administrators have the permission for that particular network in your organization. Packet sniffer can also be referred as network analyzer or protocol analyzer.

Packet sniffers are of two types: Active and Passive. Passive packet sniffers do not respond back, i.e. they only collect data and are impossible to detect them. Passive sniffers are useful in areas such as telecommunication,. Radar systems, medical equipments, etc. Colasoft Capsa, TCPDUMP and Wireshark are examples of passive packet sniffers. Active packet sniffers can send the data in the network and hence could be detected by other systems through different techniques. For example, active packet sniffer can fake replies to the broadcast or can forward it to a legitimate host. Scapy, smart RF and network ACTIV protocol packet sniffer are some of the active packet sniffers.

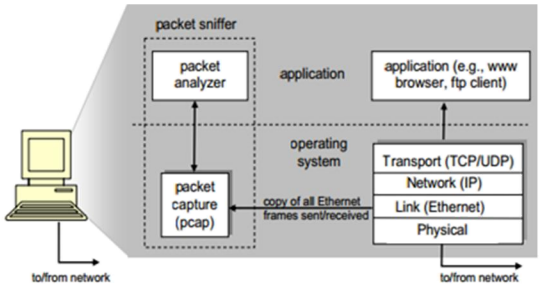


Fig. 1. structure of packet sniffer

The structure of packet sniffer consists of two parts:- packet analyser and packet capture(pcap). Packet analyser works on application layer whereas pcap captures packet from all other layers such as physical layer, link layer, IP and transport layer. Packet analyser communicates with the pcap which further captures packets from the applications running on the network. Figure 1 shows the basic structure of packet sniffer [1].

2 WORKING

Fig. 2. data encapsulation in a packet

Most of the packet sniffers work as a pcap application. The normal flow in a pcap application is to initialize network interface, then further set the filter, to filter the packets to be accepted and rejected. Packets are accepted and log is maintained continuously until the interface is closed, and further processes the packets captured.

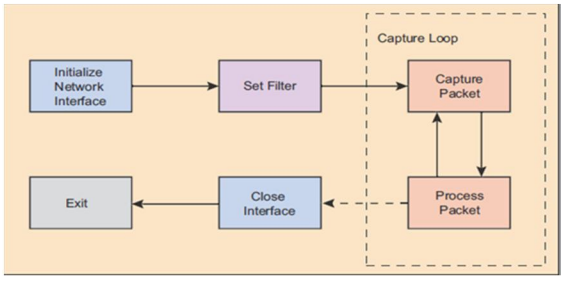


Fig. 3. normal program flow of a pcap application

To capture the information in these packets it does the following steps [2]:-

Step 1: Initially a socket is created. To deal with raw binary data, raw sockets are created. For each socket created it have a socket handle, socket type, local and remote address.

Step 2: Then the NIC (network interface card) is set to a promiscuous mode. Dictionary meaning of promiscuous mode is demonstrating an unselective approach. All packets moving in a network reaches the NIC of all the nodes and then further checks IP address of the destination node and IP address of the current node. Hence, when promiscuous mode is active it accepts all the packets arriving on its NIC irrespective of the destination address.

Step 3: Final step is protocol interpretation. Protocol interpretation means the data to be fetched for the protocols mentioned such as TCP, IP, UDP, ICMP, etc.

3 KEY FEATURES OF POPULAR SNIFFERS

Increasing computer networks increase the demand of network administrators which further increase the demands of packet sniffers. Top 5 passive packet sniffers used are: - Wireshark, TCPdump, Colasoft packet sniffer - Colasoft Capsa, etherdetect and ettercap[3]. In this section, key features of top 3 passive packet sniffers are discussed in detail i.e. Wireshark, TCPdump and Colasoft Capsa.

3.1 TCPdump

TCPdump[4] is a tool used for packet capturing, network monitoring and protocol debugging. It is the oldest and most commonly used command line tool, which works only on the Linux based systems (windump is a modified version for windows).TCPdump is a free and open source software. It can be used to read live capture or already captured log file. It can be run remotely by Telnet or SSH login. It gives the least overhead as it does not use any graphical interface and captures data in libpcap formats, which is used in most of the tools [5]. It uses a large range of packet filters. At the end of the communication or whenever TCPdump is stopped, it displays number of packets displayed and numbers of packets dropped [6]. It does not have any graphical display. Whereas, third party tools can be used such as xplot[7] or gnuplot[8] to display the graphs regarding the transaction. Major advantage of TCPdump over other packet sniffers is that it can be used remotely with giving the least overhead and hence, preferred by those administrators who like to work from a different network.

3.2 Wireshark

Similar to TCPdump, Wireshark[9] is an open source tool which has much more filtering and sorting options, including ,the GUI, which lacks in TCPdump. As seen in Figure 4, first part shows the packet capture, second one shows packet detail and the last one is the raw data of that particular packet. Wireshark could be used by command line version called Tshark. It works on more than 1100 protocols.

VoIP can be captured and if and only if properly decoded then it could even be played. It supports Linux based, windows and Mac operating systems. Wireshark supports geolocalisation of MaxMind, which means that cities and localities could be seen by given IPs giving the information of origin of packets [10]. Wireshark is not for layman as it involves a lot of network layer filtering options.

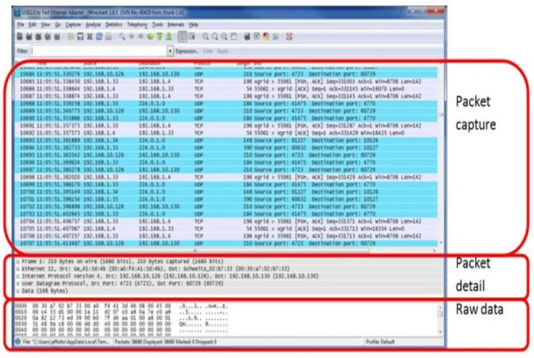


Fig. 4. Wireshark

3.3 Colasoft Capsa

Colasoft Capsa[11] supports most of the features of Wireshark with powerful TCP flow analysis and its easier interpretation. It has versatile network traffic, bandwidth and utilization analysis. It has in-depth packet decoding feature with multiple network behaviour monitoring. It has a matrix representation and eclipse visualisation of the network. Colasoft Capsa extends the network security analysis with notifying alerts only by email and audio.

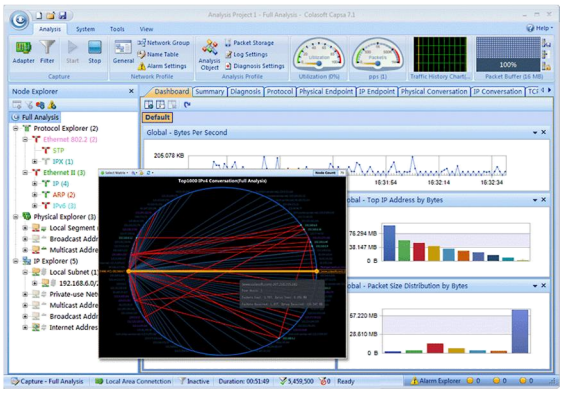


Fig. 5. Colasoft Capsa

It diagnoses the network problems by detecting and locating suspicious hosts, causing the problem and alerts computer against network anomalies. One of the demerits of Colasoft Capsa is that it is quite expensive. Whereas, a free version is available with limited features[12]. Another disadvantage of Colasoft Capsa is that, it works only on windows platform. Further, it covers only about 300 protocols which is very less when compared to Wireshark's 1100 protocols.

4 CHARACTERISTIC EVALUATION

To compare the above tools qualitatively we need to finalize some parameters. These can be OS support, disk usage, cost, number of protocols supported, etc.

4.1 TCPdump vs. Wireshark

4.1.1 Similarities

Both, TCPdump and Wireshark, have wide range of packet filters to filter the incoming traffic through NIC. Neither TCPdump nor Wireshark has intrusion detection function. They cannot generate alarms for attacks or hints when a passive attack or anything strange happens in the network. If someone is looking to manipulate data on network, then he should above both tools fail in the area of manipulation. None of them can send message in the network or do active things. Both tools capture file in libpcap format. Both can act as command line tool (Tshark in place of Wireshark).

4.1.2 Differences

Wireshark has a user friendly interface displaying the information inside packets in a meaningful manner. On the other hand, TCPdump does not have a graphical interface. Graphical interface helps in better understanding of the tools and it's working. Better the interface, more will be its users. It's harder to learn TCPdump and its filtering rules rather than Wireshark, because the rules of TCPdump may first appear completely cryptic. Colasoft Capsa has the best interface including the packet flow in the network shown graphically via matrix, whereas in Wireshark we can analyse the packet capture and make comparative graphs with the constraints of protocols, destination IP, etc. but in TCPdump, one cannot draw a single graph without the use of third party tool. The user gets information only in form of text words. Further, more the tool is graphical more will it be its system requirements. Hence, TCPdump has the least overhead when compared to others. Moreover, TCPdump is the only tool amongst the above discussed tools to be used remotely, because of its very least load on the system. TCPdump is less intrusive than Wireshark, as TCPdump displays only the data on the packet headers whereas Wireshark displays all the information inside the packets. TCPdump shows only TCP/IP based packets whereas Wireshark works for more than 1100 protocols. TCPdump has some problem with IPV6 commands. Hence, IPV6 users should go for Wireshark. TCPdump output is uncontrolled whereas in Wireshark, we can sort them up or do manipulations accordingly. TCPdump works on UNIX based operating systems whereas Wireshark works on UNIX based operating system as well as on windows. TCPdump works on most Unix-like operating systems such as Fedora, RHEL, Solaris, BSD, Mac OS, etc. TCPdump uses the libpcap library to capture incoming and outgoing packets. TCPdump can be used on Windows and is called WinDump; which uses WinPcap, the Windows port of libpcap.

4.2 Tcpdump VS. Wireshark VS. Colasoft Capsa

The following table, Table 1 summarizes how packet sniffers are differentiated on the properties such as their features for different operating systems, including cost, identification of abnormal protocols, disk usage, etc.

Table. 1. Characteristic comparison of TCPdump, Wireshark and Colasoft Capsa

s. no.	Property	TCPdump	Wireshark	Colasoft Capsa
1	Os supported	Unix based	Windows and unix based	Windows
2	Disk usage	448kb	81mb (windows) & 449mb (unix)	32mb
3	Cost	Free	Free	$999
4	Open source	Yes	Yes	No
5	No. of protocols supported	Tcp/ip	More than 1000	300
6	Libpcap based	Yes*	Yes	No
7	Multiple interfaces at a single instance	No	No	Yes
8	Alarms on traffic, protocols	No	No	Yes

9	Decode protocoL (Hex, ASCII EBCDIC)	Only hex and ASCII	Only hex and ASCII	Yes
10	Identify abnormal protocol	No	No(only creates a warning)	Yes
11	Identify packets with forged data	No	Yes	Yes
12	Display protocol in OSI 7 layer structure	No	Yes	Yes
13	Locate hosts running a specific service	No	Yes	Yes
14	Network communication in matrix map	No	No	Yes
15	Evaluate critical business traffic and non-business traffic	Yes(by filters)	Yes(by creating filters)	Yes(inbuilt)
16	Reconstruct TCP communication	No(by TCPflow)	Yes (but not formatted)	Yes
17	UDP traffic	No	Yes	Yes

5 EXPERIMENTAL SCENARIO AND PARAMETER SELECTION

From the above discussion, one can find out the best tool on the behaviour or the characteristics required. For example, if a person wants to see everything graphically, then he should prefer Wireshark or Colasoft Capsa. Whereas if someone wants to work remotely with least bandwidth usage, then he should prefer TCPdump over Wireshark and Caspa. Many more similar cases could be made by one or combination of properties mentioned above.

But, it can be seen that in many cases Wireshark and Colasoft Capsa have similar characteristics. Hence, there is a need to find out distinct parameters which may define the internal load and performance of the tool. To compare the tools, there is a need of common scenario in which admin could compare the working of the tools and their performances. One of the sample scenarios taken is explained:-

- Start one of the tool, clear its all history captures.
- As new capture file captures all the communication with the network, hence, stop all other communications with the network.
- Now open internet explorer and go to Gmail and sign in with your account
- Compose a new mail which includes a model attachment file (this file is common for all tools).
- Send the mail to yourself and sign out.
- Stop capture procedure in the tool.
- Continue the same procedure with same model attachment file for the rest of the tools

After one has the common scenario for the comparison, one needs the parameters on which the tools need to be compared. Following are the parameters on which Colasoft Capsa and Wireshark are compared (TCPdump could not be compared here because it does not have any graphical interface to show the bounded output)

5.1 Packet size distribution

Theoretically, there is no ideal packet for a communication in a network or for maximum/ optimal throughput. But there could be division in packet size as short packets, medium packets and long packets. Short packets consists of packets whose length is 88 bytes or less. Packets with size 1518 bytes or more belong to long packets. Whereas packets of size between short and long packets refer to medium sized packets. There is no right answer to the question which packet size gives best result [13].

Short packets increase load on device and hence, more short packets is a good way to stress device. On the other hand, long packets increase load on the network which means less the long packets, less will be the stress on network. Further, dealing with long packets means dealing with high ratio of packet payload and packet headers. Hence, packet size distribution should be one of the benchmarks to evaluate performance of networking tools.

5.2 Throughput

Throughput or bits per second (bps) is the amount of data which a system processes. The system throughput means the aggregate of all the terminals with sum of the data rates [14]. This is generally measured in bits/bytes per second. More is the throughput of a system, better is the performance of the tool involved in packet sniffing. Hence, Throughput is a key concept for professional performance testers to understand, and throughput is one of the top metrics used to observe how well a tool is performing.

5.3 Packets per second (PPS)

Packets per second (PPS), refers to the number of packets transferred in one second. When packets per second are multiplied with average packet size, we can get a Figure which could be similar to throughput. But, there is difference between bps and PPS. Bps is number of bits of data per second that can be processed without dropping data whereas PPS is the number of packets of data per second that can be processed before dropping data. PPS is used instead of bps only when we have to look inside the packet header. Hence, when compared to bps, PPS is equally important performance benchmark.

5.4 Dropped packets

In the given scenario, the results are analyzed after completing the transactions in all the tools. After the communication is over, it is seen that some packets are dropped. Dropping of packets could be due to many reasons including, firewalls. But here due to similar scenario for comparison of tools, the packet loss in one tool should also be there in the other. If it is not same, then one of the reasons for excessive packet loss could be that all packets coming to the NIC could not be saved to the capture file due to lack in buffer of the tool. This leads to the amount of dropped packets and hence gives you the performance benchmark.

5.5 Response time

In a network, there is a three way communication when one system asks for the packet from the other system. Then the system sends the packet to former system and finally, when the former system receives the packet, it further sends an acknowledgement signal to show that it has received the packet. Time taken in the above process is called response time. In short, Response time means the length of time taken to respond a given stimulus or event [15]. It is one of the benchmark for performance because less response time indicates less number of retransmissions occurred in the communication. Hence, lesser the response time, better the performance.

6 QUANTITATIVE COMPARISON

In the previous section, the benchmarks were chosen to compare the tools. Now, the tools performance is compared on decided parameters as follows:

6.1 Packet size distribution

Inferred from previous section, tool which supports maximum medium sized packets is a better tool on this benchmark. As it can be seen from Figure,

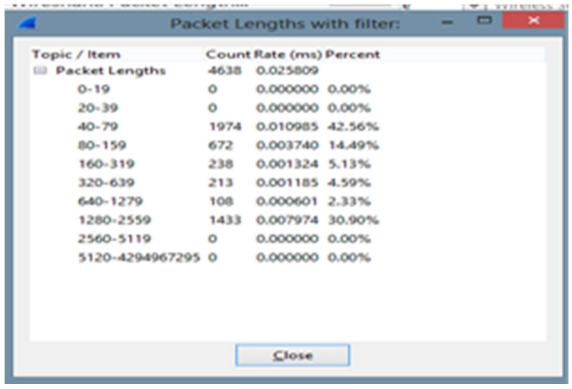


Fig. 6. Wireshark packet distribution

Wireshark does about 44% work with short packets, 20-25% work with long packets and 30-35% work with medium sized packets.

In Colasoft Capsa, the table infers that it works 25-30% with short packets, 15.13% with long packets and about 50-55% with medium sized packets. Hence, Colasoft Capsa neither stresses the network nor does it stress the system by sending too many small sized or medium sized packets.

Table. 2. packet size distribution in Colasoft Capsa

Item	Statistical value	Percentage
<=64	1,037	22.08
65-127	1,990	42.37
128-255	255	5.43
256-511	238	5.06
512-1517	447	9.51
>=1518	729	15.23

On comparison we get the following result:

1) Number of Short packets are more in Wireshark
2) In the case of average length packets, in Colasoft Capsa, around half of the work is done in average length packet size, whereas in Wireshark, it is less.
3) In Colasoft Capsa, 15% and in Wireshark, roughly 20% work is done in long length packets.

RESULT- To summarize, the average length packet size measured in Wireshark is 558.76 B and in Colasoft Capsa is 434B.

Colasoft Capsa gets a slight edge over Wireshark

6.2 Throughput

As throughput refers to the system's ability to handle the number of bits per second, the tool with higher throughput would give better performance.

It can be seen in the Figure 7, that Colasoft Capsa (brown) has large range of throughput and is changing swiftly. These random changes in the throughput are not good as it hinders the systems performance and is not good with respect to a network performance.

Whereas, in the same graph, Wireshark (black) is ranging in a pattern which is good for the network and shows a constant behaviour.

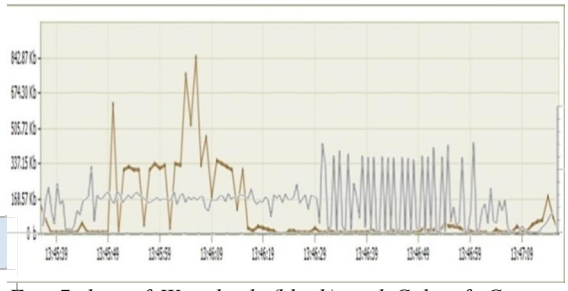


Fig. 7. bps of Wireshark (black) and Colasoft Capsa (brown)

RESULT: Average bps in Colasoft Capsa is 6.34 Kbps whereas in Wireshark it is 115.398 kbps

As we know more the bps, better would be the packet sniffer's performance. Hence, here Wireshark has a edge over Colasoft Capsa due to constant variation and not showing high cut- offs.

6.3 Packets per second (PPS)

Packets per second means the number of packets processed per second by the system. But the packets per second alone could lead to a deluded result. Hence, combine it with the average package length. As seen from Figure, the range of PPS is more in Colasoft Capsa, whereas in Wireshark there is a pattern followed which is good for the network to handle the traffic. Further, PPS in Colasoft Capsa has higher variations, on the other hand, Wireshark is stable throughout.

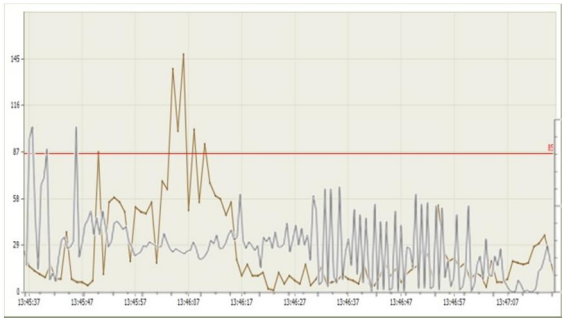


Fig. 8. PPS of Wireshark (black) and Colasoft Capsa (brown)

PPS is more in Colasoft Capsa but in Colasoft Capsa average length packet size is 417 B whereas in Wireshark average length packet size is 558.760 B

Bandwidth= average packet size * average PPS

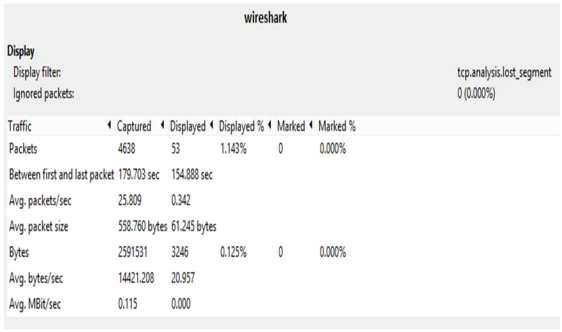
BWColasoft Capsa= 417 * 8 * 7 = 23,352bps

BWWireshark= 558.76 *8 * 25.809 =1,15,372bps
RESULT - Wireshark is ahead on comparing with

Colasoft Capsa, because the bandwidth calculated here is more than that of Colasoft Capsa and is similar to that in previous case.

6.4 Packets per second (PPS)

As it is seen, there could be packet loss due to many reasons. But if the scenario is similar for both the cases then excess packet loss shows that the tool was not able to read all the packets and before the packets could be read, the NIC drops the packets.



wireshark

Display

Display filter:	tcp.analysis.lost_segment
Ignored packets:	0 (0.000%)

Traffic	Captured	Displayed	Displayed %	Marked	Marked %
Packets	4638	53	1.143%	0	0.000%
Between first and last packet	179.703 sec	154.888 sec			
Avg. packets/sec	25.809	0.342			
Avg. packet size	558.760 bytes	61.245 bytes			
Bytes	2591531	3246	0.125%	0	0.000%
Avg. bytes/sec	14421.208	20.957			
Avg. MBit/sec	0.115	0.000			

Fig. 9. Wireshark packet loss

In Wireshark, one needs to set the filter to filter all the dropped packets from the captured file. If one has already have a conversation filter open, then one could just put that filter in brackets and add "tcp.analysis.lost_segment". This will show the lost packets when compared.

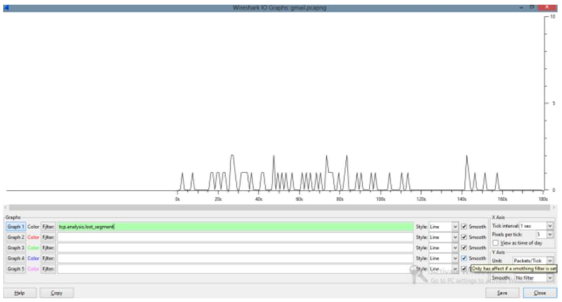


Fig. 10. graph of packets loss per second.

In Colasoft Capsa, if administrator wants to just see the ratio of packet loss then Colasoft ping tool could be used. But if administrator wants to monitor the packet loss then diagnosis tab could be used and then by going to the "TCP retransmission" tab one can monitor the lost packets and diagnose them for network management. Follow is the Figure obtained from Colasoft ping tool:

Fig. 11. Colasoft Capsa packet loss

RESULT: In Wireshark, the Figure shows the no. of dropped packets as the filter is TCP.analysis.lost_segment which gives us 53 out of 4638 packets I.e. 1.1%. And in Colasoft Capsa, Figure shows 24% packet loss. Clearly, Wireshark has lesser packet loss than Colasoft Capsa and hence is preferred over Colasoft Capsa for less packet retransmissions.

6.5 Response time

Response time means the time taken to respond a stimulus. The tool which gives lesser response time in similar conditions has performed better than the other one.

Here, we compared the response time as shown below:

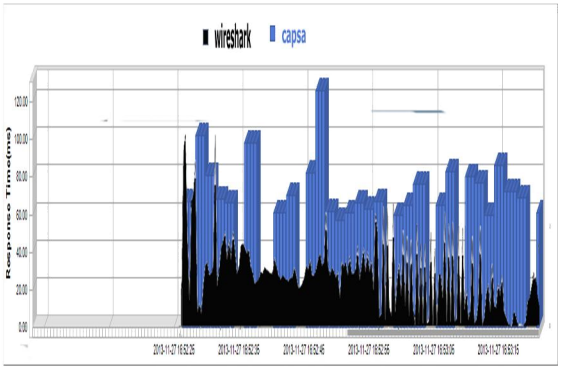


Fig. 12. compared response time of Wireshark (black) and Colasoft Capsa (blue).

As Figure shows, here again Colasoft Capsa is volatile and has higher range than Wireshark. Maximum response time in Colasoft Capsa is 130 milliseconds whereas in Wireshark maximum response time is 95 milliseconds. Clearly it can be seen that response time of Colasoft Capsa is much higher than Wireshark. Hence, in this benchmark too, Wireshark is better than Colasoft Capsa.

7 CONCLUSION AND FUTURE SCOPE

The proposed comparison of packet sniffers on qualitative and quantitative parameters shows none of the tool leads all the parameters. On the one hand, TCPdump has least overhead, but on the other Colasoft Capsa has maximum network security. The following table shows frequent cases required and the best tool in that scenario.

The present study has been made to suggest best packet sniffing tool, according to the user's requirements. The advantages and disadvantages would help to develop a new packet sniffer which could hide all the -disadvantages of the most used packet sniffers and could outperform them on quantitative and qualitative parameters.

Table 2: Conclusions

S. No.	Case	Best tool
1	Packets dropped	Wireshark
2	Network security	Colasoft Capsa
3	Response time	Wireshark
4	Network alarms	Colasoft Capsa
5	bps(throughput)	Wireshark
6	Packet size	Colasoft Capsa
7	PPS	Wireshark
8	User interface	Colasoft Capsa
9	Number of protocols	Wireshark
10	Network communication	Colasoft Capsa

8 REFERENCES

[1] "Tutorial on Wireshark". Internet: http://webhost.bridgew.edu/sattar/CS430/HW/LABS/wireshark.htm [Oct. 10,2013].

[2] L. Garcia,"programming with libpcap," in Hacking- Practical protection hard core IT magazine, Vol 3, 2008, pp. 38-46.

[3] K. zhou, "Top 5 Most Welcomed Packet Sniffers," [online], 2009, http://snifferclub.blogspot.in/2009/05/top-5-most-welcomed-packet-sniffers.html.

[4] All about TCPdump [Online] Available http://www.TCPdump.org/.

[5] H. styn, "TCPdump fu," Linux journal,. [online], 2011, http://www.linuxjournal.com/content/TCPdump-fu?page=0,1 .

[6] TCPdump Command, Command Reference, [online], http://publib.boulder.ibm.com/infocenter/pseries/v5r3/index.jsp?topic=/com.ibm.aix.cmds/doc/aixcmds5/TCPdump.htm

[7] All about Xplot [Online] Available http://www.xplot.org/

[8] All about Gnuplot [Online] Available http://www.gnuplot.info/

[9] All about Wireshark [Online] Available http://www.Wireshark.org/.

[10] How to Use GeoIP With Wireshark, "Wireshark", [online], http://wiki.Wireshark.org/HowToUseGeoIP

[11] All about Colasoft Capsa [Online] Available www.colasoft.com

[12] Colasoft Capsa- Compare Editions, "Colasoft-Maximize network value" [online], http://www.colasoft.com/Colasoft Capsa/editions.php

[13] A. Shah, D. Bhatt, P. Agarwal, and P. Agarwal, "Effect of Packet-Size over Network Performance", International Journal of Electronics and Computer Science Engineering, Vol. 1, pp. 762-766, 2012.

[14] J. Colantonio, "Performance testing basics", http://www.joecolantonio.com/2011/07/05/performance-testing-what-is-throughput , 2011.

[15] Stating Response Time Requirements, RPM solutions, [online], 2004, http://www.loadtest.com.au/Terminology/ResponseTime.htm