Proc. of Int. Conf. on Current Trends in Eng., Science and Technology, ICCTEST

TCP Port Monitoring System

Manjunath B Kajjidoni [1] and Kishor Kumar R [2]
[1-2]Dept. of Telecommunication Engineering, DBIT, Bangalore, India
manju029@gmail.com kishor13kumar@gmail.com

***Abstract*—Network monitoring device monitors the traffic from multiple users and has a capability to block or grant access to individual users. The networking monitoring hardware is a device which monitors the internet activities. The port monitoring system is a feature of the network monitoring hardware. The port monitoring system monitors packets that belong to a set of specified TCP (Transmission Control Protocol) ports and provides a variety of services based on the packets monitored thus. Some of the services provided by the monitoring/policing hardware, specifically the TCP monitoring component are forbid certain transactions based on a few configured parameters, Identify the origin of suspected fraudulent transactions, to send a copy on line based on certain criteria, work at wire line speeds and thus providing a seamless experience to the connected users, recreate the whole TCP session like replaying a VOIP (Voice Over Internet Protocol) conversation, a set of Internet activities. Every application in the TCP/IP protocol is assigned with the particular port number for transaction, so by monitoring the TCP port number the application opened by the client can be known. In the project, port numbers which are to be monitored are stored in the configuration registers and upon receiving the packet, port number of the packet is compared with configured port number and the prescribed action is taken. The modules of the port monitoring system are coded using Verilog HDL language and implemented on FPGA Spartan 3.The simulation results are verified with the hardware implementation results. A control section field contains all the information regarding each packet which is very helpful for all the modules in the network monitoring device.**

***Index Terms*— TCP packet, Monitoring, Processing header.**

I. INTRODUCTION

Internet has made communication faster and easier to any corner of the world. Like every single innovation in science and technology, Internet has its downsides. These include risks like theft of personal information, spamming, virus threat, fraudulent transaction, inappropriate sites watched by the children at impressionable age etc. Therefore, it is crucial to monitor the network, in order to understand it and to react appropriately**.**

Internet Protocol (IP) contains address information for routing packets in Network Layer of TCP/IP model. IP, as an integral part of TCP/IP, is for addressing and routing packets. It provides the mechanism to transport datagram across a large network. The main purpose of IP is to handle all the functions related to routing and to provide a network interface to the upper-layer protocols, such as TCP from Transport Layer. Applications use this single protocol in the layer for anything that requires networking access. Network Layer is responsible for transmitting datagrams hop by hop, which sends from station to station until the messages reach their destination. Each computer should have a unique IP address assigned as an interface to identify itself from the network. When a message arrives from Transport Layer, IP looks for the message addresses,

Grenze ID: 02.ICCTEST.2017.1.86

performs encapsulation and add a header end to become a datagram, and passes to the Data Link Layer. As for the same at the receive side, IP performs decapsulation and remove network layer header, and then sends to the Transport Layer. The IP implements datagram fragmentation, so that delivery of packets to the destination increases. Once the large packet is fragmented into smaller Packets then they can travel across a packet switched network without tieing up a communications link. Multiple conversations between different parties can therefore share a single communications link. If any single packet is lost, it can be retransmitted instead of having to start the entire conversation all over again. When a device receives an IP packet it examines the destination address and determines the outgoing interface to use. This interface has an associated Maximum transmission unit that dictates the maximum data size for its payload. If the data size is bigger than the Maximum transmission unit then the device must fragment the data. Reassembly module assembles the fragmented IP packets into one full IP packet so that it can be delivered to the higher layer protocol. Here packet is not being delivered to the next higher layer till entire packet arrives.

The hardware solution is available which is based on the partially packet reassembly for intrusion detection. When arrived each packet of fragment packets, this packet is merged into previously arrived packet. This method is divided into three steps: packet store, merge and pattern matching. In this method there is assumption that packets will arrives only in the ordered form [1]. The software solution is the simplest available for monitoring the network. Apart from being inexpensive and being flexible, this solution suffers from serious limitations. It needs experts to install / setup the device**,** it needs maintenance, and it does not present a business model for an expert organization due to very limited IP protection. Therefore there is a need for hardware device which overcome the drawbacks of the software solution.

II. NETWORK MONITORING DEVICE

The Network monitoring Device is positioned between the switch and the router (which connects to the Internet) as shown in Fig 1. Monitoring device also has a capability to record the traffic which can be used for analysis.

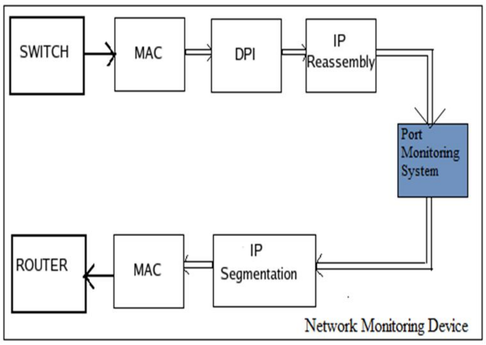


Figure.1 Monitoring Device

Deep packet inspection logic (DPI) receives the packets from different source, this module inspects all the headers (Data link, Internet, Transport) and all the inspected fields are updated in the one processing header. This processing header is attached to each packet and is different for each packet. Deep packet inspection module forwards the packet with only processing header without any Data link, Internet and Transport headers to reassembly module.

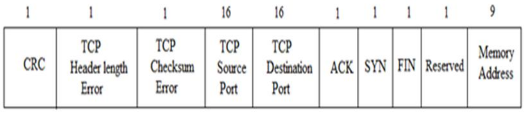
Reassembly module can receive the packets from different sources and it must have the capability to handle the data, Here it analysis the processing header to differentiate the packets from different sources. Here original message is constructed by assembling packets belonging to the respective source only. Upon constructing the original message this block removes the processing headers received .from the deep packet inspection block and attaches the only one processing header for original message which contains the useful information for the port monitoring device.

Port monitoring device analyses the processing header to decide whether the original message is allowed to pass or it should be blocked. Here entire message may be recorded for further analysis. If original message is blocked then it should inform the segmentation module so that it can erase the stored information related to blocked message. If the original message is allowed to pass then it should be forwarded to segmentation module.

Segmentation module upon receiving the original message divides the original message into fragments similar to fragments received at the reassembly side and before sending the fragments out it should attach all the headers like data link header, internet header, transport header to the respective fragment so that nobody knows the existence of this device.

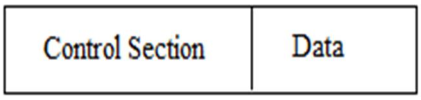
III. CONTROL SECTION (PROCESSING HEADER)

Reassembly module generates the packet as shown in figure 2. With the help of control section it will arrange all the packets in order and constructs the original packet. To this packet it will attach a new control section which is shown in figure 3. Here control section contains the information regarding source port address, destination port address, acknowledge flag, synchronizing flag, finish flag and memory address of headers. This control information is very useful for the port monitoring device because with the help of this it will decide whether all allow the packet to get passed or to block permanently.



1	1	1	16	16	1	1	1	1	9
CRC	TCP Header length Error	TCP Checksum Error	TCP Source Port	TCP Destination Port	ACK	SYN	FIN	Reserved	Memory Address

Figure. 2 Reassembly Module Generated Processing Header



Control Section	Data

Figure. 3 Packet Format

IV. TCP PORT MONITORING SYSTEM

The TCP packets are received from the previous block i.e filtering block which only forwards the TCP packets to the port monitoring block. Upon reception of the TCP packets from the Filter, the source port address of the TCP packet received is to be matched against the configured registers so as to perform the action on the packet of data. The block diagram of TCP port monitoring system is as shown in Figure 4.

The Control logic is attached from the reception of the packet so that the required information is available in the header and includes the advantage of being reconfigurable. The received TCP Packet which also includes a control logic reconfigured from the previous block which is the TCP reassembly block contains the source port address of the received packet of data. The first sixteen bits of the control logic is the source address of the TCP packet and is verified against Port addresses configured by the user in the configuration registers if the VALID BIT is SET. The port monitoring block is configured to monitor 16 TCP ports and overall 32 sessions. Depending on the port number of the received packet there are two possibilities that the port number matches with the configured registers or does not match. If the Port number matches, then the configured port addresses registers are checked for logic high in the three bits (F, R, B) of the configured registers that is either to forward, record or to block the packets. If the port numbers do not match then the packets are forwarded without any monitoring action. If the number of sessions exceeds 32 then the packets are simply forwarded to next block.

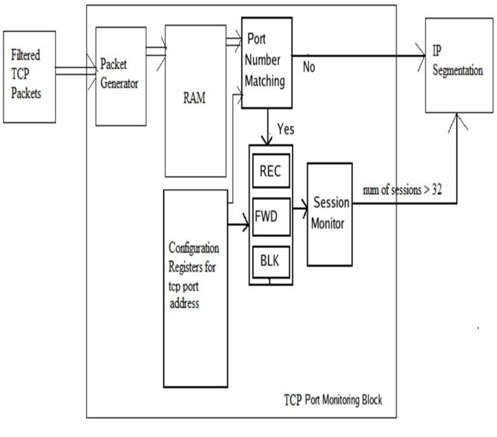


Figure. 4 TCP Port Monitoring System

V. SIMULATION RESULTS

This chapter shows the results achieved from simulation of the port monitoring system module for the different input packets that are configured for the specific action to be taken based on certain criteria. The different modules of the port monitoring system are coded using the Verilog HDL language. The simulation tools used are Xilinx 12.2 and Modelsim 6.3.Figure 5 to 7 shows the simulation result for the input packet with the size of 256 bits. The port monitoring system receives 8 bits of data per cycle. The packet generator module receives the incoming data bits when the valid_data signal is high and generates the packet of 256 bits when the valid_data is low or valid_out signal is high. The generated packet is stored in the memory when the rd_wr signal is low in the memory address specified by the addr. When the valid bit is high in configuration register the source port in the configuration register is compared with the source port in the processing header of the packet which is the bits from 28 to 43. If the source port matches then the required action is generated in the config_sig which is three bit signal which indicates Record, Block or Forward. When the config_sig is 001, the packet is just forwarded, when config_sig is 101 a copy of packet is stored and forwarded and when config_sig is 010 the packet is blocked that is data_out is cleared indicating that transaction is not allowed.

Figure. 5 shows the simulation results for the input packet001850000C00BB00C 9043 C01F50801091303DD 03E3038502770103E30 3850277. First 6 bytes (48 bits) are processing header which contains CRC error bit, TCP header length error, TCP checksum error and reserved bit which are first 4 bits, source port destination port and the 9 bit memory address and the next 26 bytes are data. The source port here is 389 which is configured for forwarding only. In this waveform the config_sig (RBF) is 001 which indicates that the packet has to be forwarded and the data in is copied to data out to transfer to next block.

Figure. 6 shows the simulation results for the input packet 000500000C 00BB00C9043 C01F508 010 91303 DD03E3 03850277010 3E3 03850277. Here the source port is 50 which is configured for record and forward The config_sig is 101; packet is forwarded and stored in the memory.

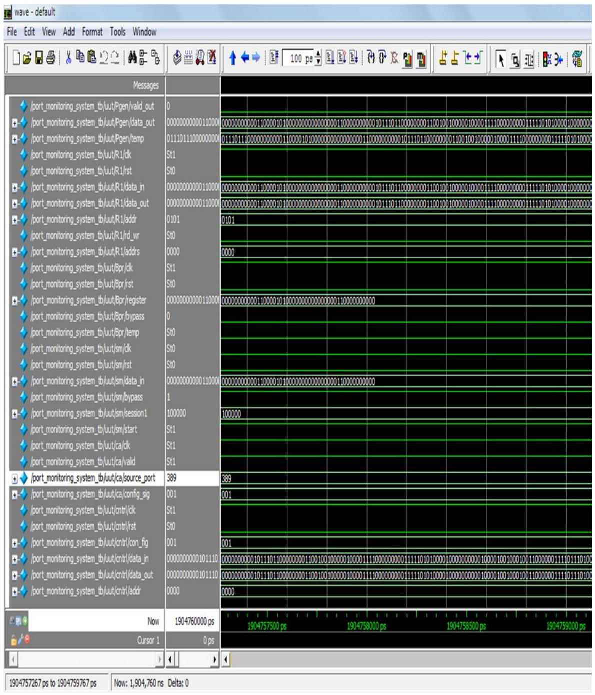

wave - default
File Edit View Add Format Tools Window
Messages
/port_monitoring_system_tb/uut/ca/source_port
389
/port_monitoring_system_tb/uut/ca/config_sig
001
Now
1904760000 ps
Cursor 1
0 ps

Figure. 5 Packet Forwarding

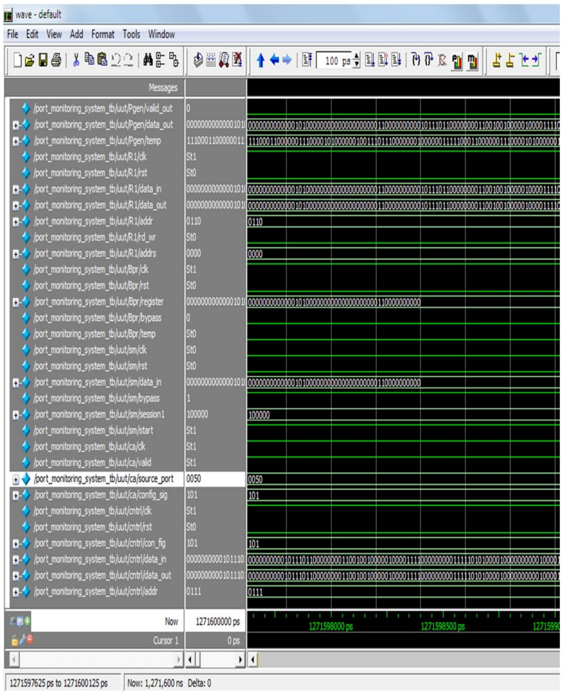
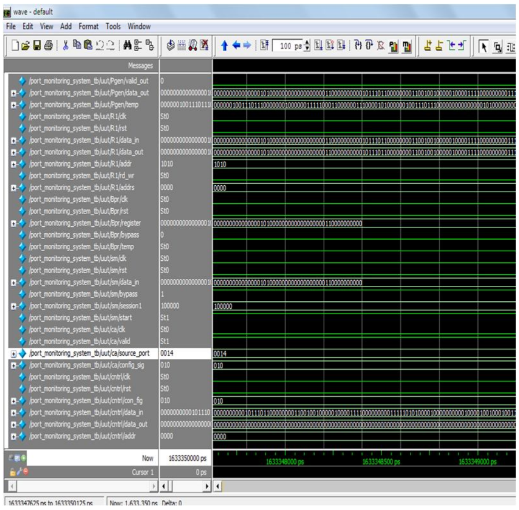

wave - default
File Edit View Add Format Tools Window
Messages
/port_monitoring_system_tb/uut/ca/source_port
0050
/port_monitoring_system_tb/uut/ca/config_sig
101
Now
1271600000 ps
Cursor 1
0 ps

Figure. 6 Packet Recording

Figure. 7 shows the simulation results for the input packet 000140000C00 BB00C 9043C01F50801091303DD03E3038502770103E303850277. Here the source port is 14 which is configured for blocking the transaction. In the result the config_signal is 010 which indicate the blocking of the packet. The data_out is cleared so that no packet is forwarded.



```
/port_monitoring_system_tb/uut/Pgen/valid_out      0
/port_monitoring_system_tb/uut/Pgen/data_out       00000000000000001
/port_monitoring_system_tb/uut/Pgen/temp           00000010011101111
/port_monitoring_system_tb/uut/R1/clk              St0
/port_monitoring_system_tb/uut/R1/rst              St0
/port_monitoring_system_tb/uut/R1/data_in          00000000000000001
/port_monitoring_system_tb/uut/R1/data_out         00000000000000001
/port_monitoring_system_tb/uut/R1/addr             1010
/port_monitoring_system_tb/uut/R1/rd_wr            St0
/port_monitoring_system_tb/uut/R1/addrs            0000
/port_monitoring_system_tb/uut/Bpr/clk             St0
/port_monitoring_system_tb/uut/Bpr/rst             St0
/port_monitoring_system_tb/uut/Bpr/register        00000000000000001
/port_monitoring_system_tb/uut/Bpr/bypass          0
/port_monitoring_system_tb/uut/Bpr/temp            St0
/port_monitoring_system_tb/uut/sm/clk              St0
/port_monitoring_system_tb/uut/sm/rst              St0
/port_monitoring_system_tb/uut/sm/data_in          00000000000000001
/port_monitoring_system_tb/uut/sm/bypass           1
/port_monitoring_system_tb/uut/sm/session1         100000
/port_monitoring_system_tb/uut/sm/start            St1
/port_monitoring_system_tb/uut/ca/clk              St0
/port_monitoring_system_tb/uut/ca/valid            St1
/port_monitoring_system_tb/uut/ca/source_port      0014
/port_monitoring_system_tb/uut/ca/config_sig       010
/port_monitoring_system_tb/uut/cntrl/clk           St0
/port_monitoring_system_tb/uut/cntrl/rst           St0
/port_monitoring_system_tb/uut/cntrl/con_fig       010
/port_monitoring_system_tb/uut/cntrl/data_in       0000000000101110
/port_monitoring_system_tb/uut/cntrl/data_out      00000000000000000
/port_monitoring_system_tb/uut/cntrl/addr          0000
Now        1633350000 ps
Cursor 1   0 ps
1633347625 ps to 1633350125 ps    Now: 1,633,350 ps  Delta: 0
```

Figure. 7 Packet Blocking

VI. CONCLUSION

The port monitoring system which is the feature of the networking monitoring device monitors the TCP ports and takes certain actions such as Blocking, Recording based certain criteria is designed and verified. The modules of the port monitoring system are implemented on FPGA and verified with the simulation results. If necessary to monitor more TCP ports, the devices can be connected in parallel for monitoring purpose

In this project the monitoring was done based on TCP port numbers, the monitoring can also be done based on the MAC address and IP address. If we need to monitor based on these criteria the configuration registers contain the MAC address or the IP address. First have to check for the matching of port numbers, if there is no match in the port numbers then we have to check the IP address and take the prescribed action. If there is no match in the IP address then we need to check MAC address and take the prescribed action. If there is no match in any of the port number and IP or MAC address then we have to forward the packet.

REFERENCES

[1] Ming-Han, Wan, Mong-Fong Horng, *"An Intelligent Monitoring System for Local-Area Network Traffic*", Eighth International Conference on Intelligent Systems Design and Applications, 2008 pp.657-660

[2] Bo Yang, Yi Li, Yuehui Chen, Runzhang Yuan "*A Flow-based Network Monitoring System Used for CSCW in Design"* The 9th International Conference on Computer Supported Cooperative Work in Design Proceedings

[3] J. Case, M. Fedor, M. Schoffstall, and J. Davin," *A Simple Network Management Protocol (SNMP),*" IETF RFC1157, 1990.

[4] A. Forouzan and Sophia Chung Fegan, *"Data Communication and Networking,"* Tata McGraw Hill Publications, 4th Edition, 2007.

[5] Trung Nguyeny, Mihai Cristeay, Willem de Bruijn and Herbert Bos, "*Scalable network monitors for high-speed links: a bottom-up approach,"* Proceedings of IEEE International Workshop on IP Operations & Management (IPOM'04), Beijing, China, 11-13 October 2004.

[6] Postel J, *"Transmission Control Protocol"*, RFC 793, Information Sciences Institute, University of Southern California, September 1981.

[7] Reynolds J, Postel J, *"Assigned Port Numbers",* RFC 1700, Information Sciences Institute, University of Southern California, October 1994