ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 7, July 2017

Performance evaluation of transport layer protocols using ns-3

[1] Chandra Shekar J, [2]C.Y. Gopinath, [3]Dr Hemanth Kumar .A.R, [4]Dr M.N.Sreerangaraju
[1] PG Student [DEC], [2] Associate Professor, Dept. of ECE, [3][4]Professor, Dept. of ECE
[1][2][3][4]Bangalore Institute of Technology, Bengaluru, Karnataka, India

Abstract - **The transport layer is responsible for the end-toend transfer of messages from a session entity in the source machine to a session entity in the destination machine. The three major protocols available in the transport layer are : TCP, UDP and SCTP. SCTP provides some of the same service features of both UDP and TCP: it is message-oriented like UDP and ensures reliable, in-sequence transport of messages with congestion control like TCP; it differs from these in providing multistreaming, multi-homing and redundant paths to increase resilience and reliability. In this paper performance comparison of TCP, UDP and SCTP is presented both in wired and wireless environment. To compare the performance of SCTP with that of TCP and UDP under different scenarios and to understand the scope for improvement is the main purpose of this paper. The performance is analyzed using the metrics: Throughput, Jitter and Packet loss. Network simulator NS3 is the simulator used.**

Index Terms: SCTP; UDP; TCP; multi-homing; throughput; jitter; packet loss; NS3.

I.INTRODUCTION

The transport layer is responsible for process-to-process delivery of the entire message and delivery of data to the appropriate application process on the host computers. This involves statistical multiplexing of data from different application processes, i.e. forming data packets, and adding source and destination port numbers in the header of each transport layer data packet. Services provided by transport layer includes connection-oriented data transfer, reliable communication, flow control, error control and multiplexing. For connection-oriented communication TCP and SCTP are used & for connectionless transmission UDP is used. UDP treats each segment separately whereas TCP and SCTP creates a relationship between the segments using sequence numbers.

A computer network can make more than one transport layer protocol available to network applications. There are three major protocols – TCP, UDP and SCTP. Each of these protocols provides a different set of transport layer services to the invoking application. All transport layer protocols provide an application multiplexing/demultiplexing service. In addition to multiplexing/demultiplexing service, a transport protocol can possibly provide other services to invoking applications, including reliable data transfer, Connection-oriented communication, Flow control, Congestion avoidance(Congestion control), bandwidth guarantees, and delay guarantees.

TCP protocol could transmit data stream reliably [2]. TCP is used for email transfer and HTTP web browsing. Due to sequencing and because of Head of Line blocking, delay results. Telecommunication requires low delay and high reliability towards the data transmission. Although TCP can provide reliable communication service, it cannot fulfill the requirement of low latency and delay. TCP is prone to DOS attacks, such as SYN attacks. TCP follows strict ordered delivery and uses a single stream for transmission. It causes extra delay when message is lost or sequence of errors occurs within the network [1]

UDP does not provide reliable communication but is used in broadcasting and multicasting applications. For multimedia communication like IP-telephony and online games where packet loss is accepted, generally UDP is used [6]. UDP is suitable for purposes where error detection and correction are either not necessary or are performed in the application; UDP avoids the overhead of such processing at the level of the network interface. Time-sensitive applications often use UDP because dropping packets is preferable to waiting for delayed packets, which may not be an option in a real-time system.

Stream Control Transmission Protocol (SCTP) is a reliable, message-oriented, multi-homed transport protocol. Developed by the IETF SIGTRAN working group to transport SS7 over IP, it is now the third general-purpose transport developed by the IETF. SCTP is an unicast protocol that provide in-sequence packet delivery and rate-adaptive congestion control [4]. TCP socket complications arises while providing high availability, but SCTP can overcome this by use of multi-homed hosts. SCTP association can bind multiple IP addresses at each peer. This association of binding multiple IP address is not possible in UDP or TCP [5]. An SCTP sender chooses a single destination address as a primary one and transmits the data to this primary destination. Any failure to reach primary destination results in failure, such that sender dynamically chooses an alternate destination to provide failure and association survivability in the face of hardware failure [7]. SCTP's association contains a 4-way handshake with cookie mechanism which protects the server from DOS attack.

In the absence of native SCTP support in operating systems it is possible to tunnel SCTP over UDP, as well as mapping TCP API calls to SCTP ones.

The rest of this paper is organized as: Section II describes the background and a brief description of Transport Layer and the protocols TCP, UDP and SCTP. Section III describes the simulation model. In Section IV the results are discussed and finally the paper concludes in section V.

II. BACKGROUND

In computer networking, the transport layer turns the host to- host packet delivery service provided by the lower layers into a process-to-process communication channel. Residing between the application and network layers, the transport layer is in the core of the layered network architecture. It has the critical role of providing communication services directly to the application processes running on different hosts.

A transport layer protocol provides a logical communication between application processes running on different hosts. Logical communication implies that although the communicating application processes are not physically connected to each other (indeed, they may be on different sides of the planet, connected via numerous routers and a wide range of link types), from the applications' viewpoint, it is as if they were physically connected. Application processes use this logical communication provided by the transport layer to send messages to each other, without worrying about the details of the physical infrastructure used to carry these messages. Based on the needs of the application, an appropriate transport layer protocol between TCP, UDP and SCTP is chosen.

A. User Datagram Protocol (UDP)

The User Datagram Protocol (UDP) is an unreliable, connectionless transport layer protocol. With UDP there is no handshaking between sending and receiving transport layer entities before sending a segment. For this reason, UDP is said to be connectionless. It is a very simple protocol that provides only two additional services beyond IP: de multiplexing and error checking on data [6]. In fact, if the application developer chooses UDP , then the application is talking almost directly with IP. UDP takes messages from application process, attaches source and destination port number fields for the multiplexing/demultiplexing service, adds two other fields of minor importance, and passes the resulting segment to the network layer. The network layer encapsulates the segment into an IP datagram and then makes a best-effort attempt to deliver the segment to the receiving host. If the segment arrives at the receiving host, UDP uses the port numbers and the IP source and destination addresses to deliver the data in the segment to the correct application process [9]. IP knows how to deliver packets to a host, but does not know how to deliver them to the specific application in the host. UDP adds a mechanism that distinguishes among multiple applications in the host. UDP can optionally check the integrity of the entire UDP datagram [8].

UDP is also commonly used today with multimedia applications, such as Internet telephony, real-time video conferencing, and streaming of real-time audio and video [6]. Applications that also use UDP include Trivial File Transfer Protocol, DNS and SNMP.

B. Transmission Control Protocol (TCP)

The Transmission Control Protocol (TCP) is one of the main protocols of the Internet protocol suite. It originated in the initial network implementation in which it complemented the Internet

Protocol (IP). Therefore, the entire suite is commonly referred to as TCP/IP. TCP provides reliable, ordered, and error-checked delivery of a stream of octets between applications running on hosts communicating by an IP network. The Transmission Control Protocol (TCP) provides a full-duplex (two way) connection between two application layer processes across a datagram network. TCP provides these application processes with a connection-oriented, reliable, insequence, byte-stream service. TCP also provides flow control that allows receivers to control the rate at which the sender transmits information so that buffers do not over flow. TCP can also support multiple application processes in the same end system [6].

At the lower levels of the protocol stack, due to network congestion, traffic load balancing, or other unpredictable network behavior, IP packets may be lost, duplicated, or delivered out of order. TCP detects these problems, requests retransmission of lost data, rearranges out-of-order data and even helps minimize network congestion to reduce the occurrence of the other problems [8]. If the data still remains undelivered, the source is notified of this failure. Once the TCP receiver has reassembled the sequence of octets originally transmitted, it passes them to the receiving application. Thus, TCP abstracts the application's communication from the underlying networking details [1].

Before data transfer can begin, TCP establishes a connection between the two application processes by setting up variables that are used in the protocol. These variables are stored in a connection record that is called the transmission control block (TCB). Once the connection is established, TCP delivers data over each direction in the connection correctly and in sequence. TCP was designed to operate over the Internet Protocol (IP) and does not assume that the underlying network service is reliable [9]. To implement reliability, TCP uses a form of Selective Repeat ARQ [2]. TCP terminates each direction of the connection independently,allowing data to continue flowing in one direction after the other direction has been closed. TCP uses checksum for the purpose of error detection.

TCP does not preserve message boundaries and treats the data it gets from the application layer as a byte stream [2]. Thus when a source sends a 1000-byte message in a single chunk, the destination may receive the message in two chunks of 500 bytes each, in three chunks of 400 bytes, 300 bytes and 300 bytes, or in any other combination. In other words, TCP may split or combine the application information in the way it finds most appropriate for the underlying network [10].

TCP is optimized for accurate delivery rather than timely delivery and can incur relatively long delays (on the order of seconds) while waiting for out-of-order messages or retransmissions of lost messages. Therefore, it is not particularly suitable for real-time applications such as Voice over IP [6]. The major applications that use TCP are : World Wide Web(WWW), HTTP, SSH, FTP, telnet, SMTP, IMAP/POP [1].

C. Stream Control Transmission Protocol (SCTP)

SCTP can be characterized as message-oriented, meaning it transports a sequence of messages (each being a group of bytes), rather than transporting an unbroken stream of bytes as does TCP. As in UDP, in SCTP a sender sends a message in one operation, and that exact message is passed to the receiving application process in one operation [2].

SCTP applications submit their data to be transmitted in messages to the SCTP transport layer [5]. SCTP places messages and control information into separate chunks (data chunks and control chunks), each identified by a chunk header. The protocol can fragment a message into a number of data chunks, but each data chunk contains data from only one user message. SCTP bundles the chunks into SCTP packets. The SCTP packet, which is submitted to the Internet Protocol, consists of a packet header, SCTP control chunks (when necessary), followed by SCTP data chunks (when available) [3].

SCTP provides multi-streaming, which is the capability of SCTP to transmit several independent streams of chunks in parallel, for example transmitting web page images together with the web page text [3]. SCTP assigns a sequence number to each message sent in a stream. This allows independent ordering of messages in different streams. However, message ordering is optional in SCTP; a receiving application may choose to process messages in the order of receipt instead of in the order of sending [6].

The advantages of SCTP over TCP are: Multihoming support, enabling transparent fail-over between redundant network paths. Delivery of chunks

ISSN (Online) 2394-6849

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 7, July 2017

within independent streams eliminate unnecessary head-of-line blocking. Improved error detection suitable for Ethernet jumbo frames [7].

The next section describes the simulation model generated and the performance metrics that are measured to analyze the performance of the protocols.

III. SIMULATION MODEL

Simulation for TCP, UDP and SCTP is done separately in wired and wireless network environment using NS3 (version 3.26) simulator. For a wired network scenario, all the 3 cases of LAN, MAN and WAN are considered. The number of nodes, data rate and distance are varied and the corresponding results are tabulated.

A. Performance Metrics

The following are the different performance metrics considered.

PLR: PLR (Packet loss ratio) is the ratio between number of packets dropped or lost to the number of packets sent through the network.

PLR = (Number of packets dropped)/(Number of packets sent).

Jitter: Jitter is the variation in packet delays of consecutive packets.

Throughput: Throughput is the rate of successful transmission of packets delivered over a network. It is measured as number of bits transmitted per second.

IV. SIMULATION RESULTS

Simulation results are presented below:

A. Wired network scenario

The default settings that are used for the network scenario

created are given in the table I.

TABLE I
Default network settings

	LAN	WAN	MAN
Latency (ms)	1	40	75
Channel BW (Mbps)	100	100	100
PLR (%)	1	1	1

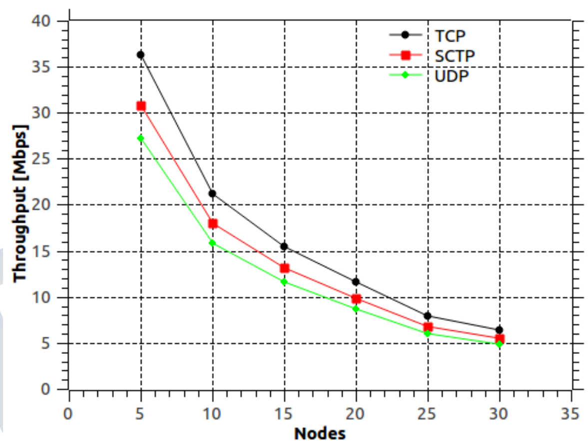

Fig. 1: Throughput vs nodes in LAN

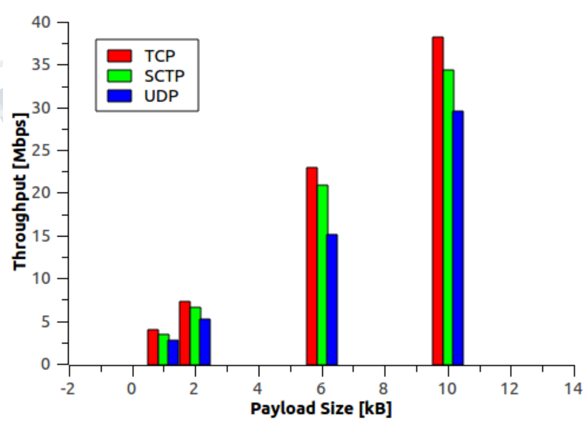

Fig. 2: Throughput vs payload size in LAN

International Journal of Engineering Research in Electronics and Communication Engineering (IJERECE)
Vol 4, Issue 7, July 2017

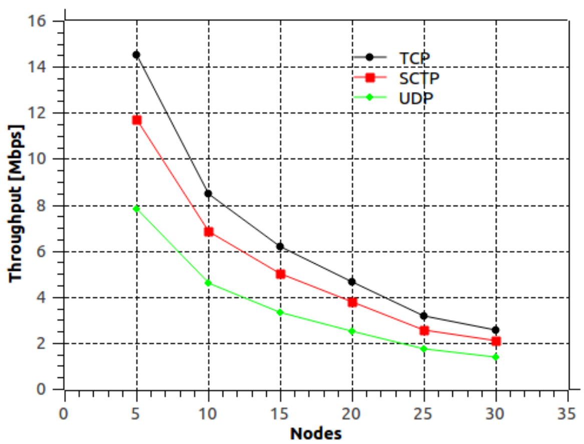

Fig. 3: Throughput vs nodes in MAN

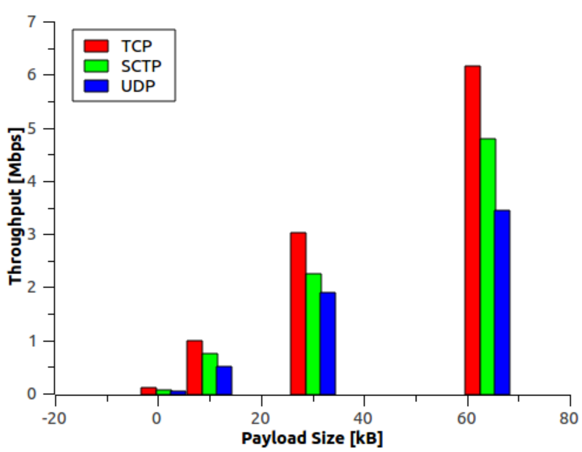

Fig. 4: Throughput vs Payload size in MAN

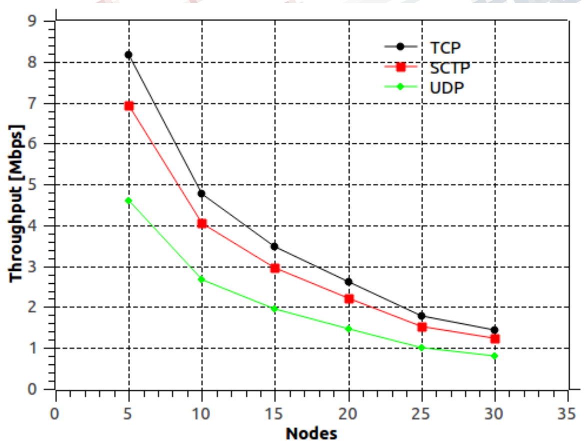

Fig. 5: Throughput vs nodes in WAN

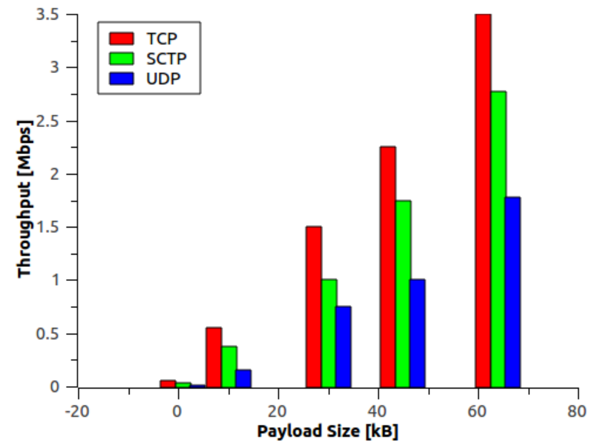

Fig. 6: Throughput vs payload size in WAN

As expected TCP has the highest throughput of the three protocols in all the 3 network scenarios LAN, MAN & WAN owing to its well developed error control and flow control mechanisms. UDP has the lowest performance for throughput, since it is a best-effort delivery protocol. SCTP performs better than UDP but its performance is significantly lower than that of the TCP which is evident from the figures 1 – 6. For all the 3 protocols the throughput reduces as the number of nodes increase as collisions increases. Also the throughput increases as the payload size increases since the packet will be utilized efficiently.

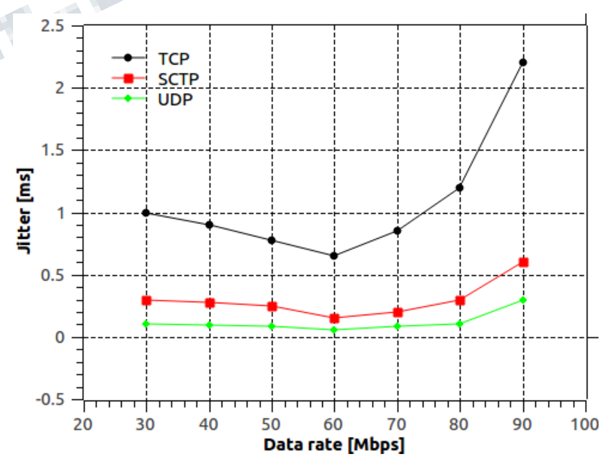

Fig. 7: Jitter vs data rate

Jitter is significant parameter only in a WAN scenario, hence the jitter performance is measured only in WAN network scenario & is presented in figures 7. In terms of jitter, UDP has the best performance. It has the lowest jitter value of the three protocols as there are no retransmissions in UDP. TCP has the highest jitter value owing to its retransmission policies and HOL blocking. Since SCTP also has retransmission policy, it has higher jitter but is lower than that of TCP because of multi-streaming and multihoming.

For all the 3 protocols the jitter there is a decrease in jitter as the data rate increases, but as the date rate goes beyond 60Mbps and starts approaching the channel bandwidth of 100Mbps, the jitter increases significantly due to collisions.

Variance of jitter with increase in the nodes is given in figure 8. As the nodes increase the jitter increases exponentially for all the 3 protocols because of increase in congestion due to more packets coming from nodes.

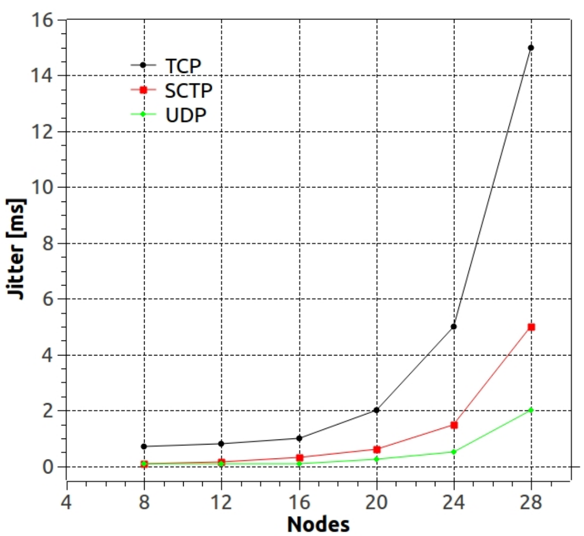

Fig. 8 Jitter vs nodes

B. Wireless network scenario

For the wireless scenario an LTE networking environment is simulated. Since packet loss is significant in the wireless networking environment, packet loss ratio (PLR) is also one of the metric being evaluated here along with Throughput and jitter as considered in a wired networking scenario. Default values for the wireless scenarios are: cell distance is 500m, eNB power is 30db and the UE power is 23db.

TCP has the best Throughput among the three protocols even in the wireless environment as given in figure 9. It is closely followed by SCTP. UDP has the worst throughput. Also the throughput reduces exponentially as the number of nodes increases.

PLR measured is presented in figure 10. TCP outperforms the other 2 protocols in-terms of PLR owing to its stable flow control mechanisms. It is closely followed by SCTP, it also has an inherent flow control mechanism that does well in restricting the packet loss. UDP has the highest packet loss ratio compared to the other two protocols. This can be expected as there is no flow control mechanisms implemented in UDP. The PLR increases for all the three protocols as the packet rate at each node is increased. As the packet rate increases, it can be observed that the behavior of SCTP approaches that of the TCP.

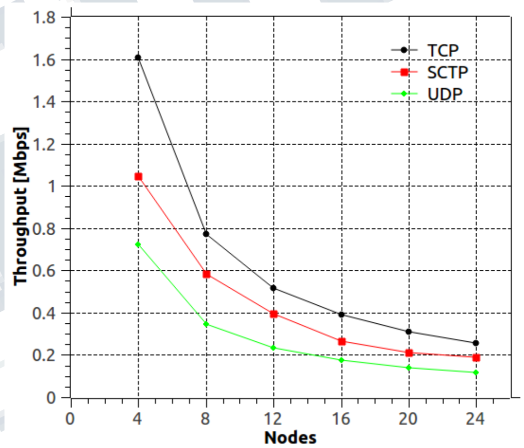

Fig. 9: Throughput vs nodes

Jitter performance is given in the figure 11. UDP has the lowest jitter. SCTP closely follows UDP in its jitter performance. TCP has the worst jitter because of its retransmission mechanisms and HOL blocking.

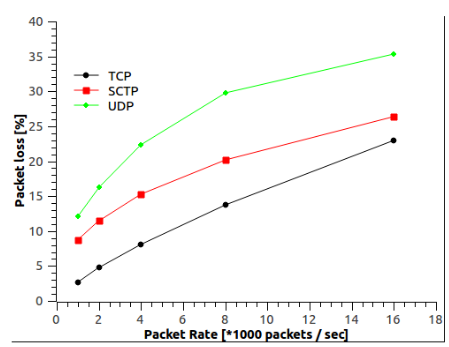

Fig. 10 PLR vs Packet rate

Also it is evident from figure 13 that the jitter increases exponentially as the number of nodes is increased in the network. This is due the increased loss of packets and network bottlenecks that arise due to increasing number of nodes.

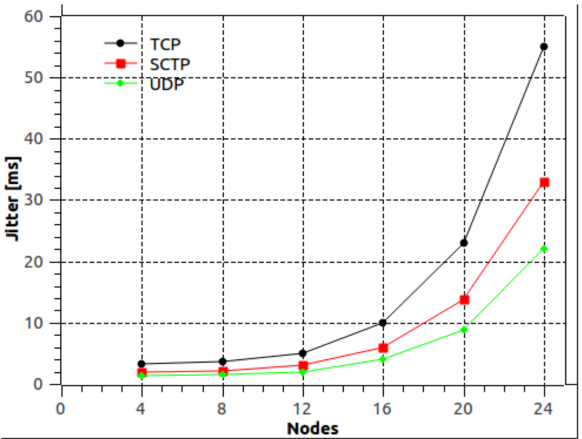

Fig. 11 Jitter vs nodes

V. CONCLUSION & FUTURE SCOPE

This paper presents a comparative analysis of three transport layer protocols: TCP, UDP and SCTP in both wired and wireless environment. The simulation results depict that the traffic agent TCP provides best throughput rate compared to the other protocols in both wired and wireless environment and packet drop rate or packet loss ratio is also negligible, but with highest jitter. However, UDP and SCTP exhibit less jitter but with low throughput and high packet loss as compared to TCP. TCP suffers disadvantage due to its high jitter. UDP has the better performance in terms of jitter but its throughput is the worst of the three protocols under consideration. SCTP performance is in between that of TCP and UDP. Its Throughput and packet loss performance is better than UDP but relatively lower than that of TCP. In terms of Jitter, SCTP performs better than TCP but it is significantly lower than that of UDP. Although SCTP does not outperform TCP and UDP in their use case scenarios, but it presents itself as a potential candidate for practical implementation at the transport layer both in wired and wireless network.

The multi-homing aspect of SCTP needs an in-depth analysis for the practical implementation and its possible use case for the improvement of throughput, jitter and delay has to be studied. Also the behavior of SCTP in a heterogeneous network, when used concurrently with the other available transport layer protocols have to be studied.

REFERENCES

[1] NRahat Saadia, Amir Qayum “Performance Analysis of Transport Layer Protocols for MPEG4 Video Traffic in IEEE802.11e Ad-hoc Wireless Networks”. 11th International Bhurban Conference on Applied Sciences & Technology (IBCAST).

[2] B.Suvarna, K V Krishna Kishore, G.P arimala, R.Prathap Kwnar “Performance Estimation of DSR, DSDV and AODV in TCP, UDP and SCTP”, 2014 International Conference on Reliability, Optimization and Information Technology

[3] Rung-Shiang Cheng , “Performance Evaluation of Stream Control Transport Protocol over IEEE 802.11ac Networks”,2015 IEEE Wireless Communications and Networking Conference (WCNC) - Workshop - Next Generation WiFi Technology

[4] M. Fazio, A. Biundo, M. Villari, A. Puliafito, “An in-depth analysis of the SCTP behavior in mobile ad hoc networks”, In 29th Annual IEEE International Conference on Local Computer Networks (LCN'04), pages 419-420, 2005.

[5] Tran Cong Hung, Tran Phu Khanh, “Analyze and Evaluate the performance of SCTP at transport layer”, ICACT 2010, ISBN 978-89- 5519-146-2.

[6] Sinda Boussen, Nabil Tabbane, Sami Tabbane. “Performance analysis of SCTP protocol in WiFi network”, IEEE 2009, 978-1-4244-5135-7

[7] Md. Nurul Islam, A. Kara, “Throughput Analysis of SCTP over a Multihomed Association”, Sixth IEEE International Conference on Computer and Information Technology (CIT'06), IEEE 2006.

[8] Andrea De Vendictis, Francesco Vacirca, Andrea Baiocchi, “Experimental Analysis of TCP and UDP

Traffic Performance over Infra-structured 802.llb WLANs", IEEE 2004.

[9] Hemant Kumar Rath , Abhay Karandikar, "Performance Analysis of TCP and UDP based Applications in a IEEE 802.16 deployed Network". IEEE 2008.

[10] Azeem Aftab, Azfar Ghani, Zain-ul-Abedin Abidi, Mohtashim Baqar, "Simulation Based Performance Evaluation of TCP Variants along with UDP Flow", International Conference on Open Source Systems and Technologies (ICOSST) 2016.

BIOGRAPHY

Chandra Shekar J received his B.E degree in E&C from Visvesvaraya Technological University, Belagavi, India. Currently he is pursuing M.Tech in Digital Electronics and Communication Engineering at BIT, Bangalore. His area of interest includes Wireless Communication, Embedded Systems and Computer Communication Networks.

C.Y. Gopinath received his B.E in E&C from U.V.C.E, Bangalore, in 1985. He specialized in Industrial Electronics from S.J.C.E, Mysore in 1989. He is presently working as Associate Professor in the Department of Electronics and Communication Engineering at BIT, Bangalore. He is currently research Scholar in Visvesvaraya Technological University. His area of interest includes Computer Architecture, Wireless Communication and Coding Techniques. He has published 5 international papers in various indexed journals, and national conferences.

Dr Hemanth Kumar A. R is presently working as Professor & Principal Investigator R&D,ECE, at BIT, Bangalore. He completed his B.E in E&C from Manipal Institute of Technology in the year 1999. He is specialized in Digital Communication from B.M.S College of Engineering in 2003 and received his PhD from Manipal University in 2011. His area of interest includes MANETS, Clustering the network, Path Planning and Routing algorithms. He has published 29 international papers in various indexed journals, and national conference in India, Singapore, Thailand.

Dr M.N.SREERANGARAJU received his BE in ECE. from Bangalore University in 1985, M.Tech in Ind.Electronics from Mysore University 1991 and Ph.D. from VMUniversity, Salem, TN,India. He is Professor in the Dept. of Electronics and Communication Engg., Bangalore Institute of Technology, Bengaluru,He has chaired and co- chaired in several key conferences in the area in the Communication and Networking areas and organized many Conferences and workshops.He has published more than 30 research papers in the leading National and International journals. His Research interests are in the area of : mobile and wireless communication and networks, cooperative communication networking, Sensor Networking,Cryptography and Network Security.