On the use of Multipath Transmission using SCTP

Imtiaz Ali Halepoto†, Muhammad Sulleman Memon††, Nazar Hussain Phulpoto†††, Ubaidullah Rajput††††, Muhammad Yaqoob Junejo†††††

†,††,††††Department of Computer Systems Engineering, QUEST Nawabshah, Pakistan
†††,†††††Department of Information Technology, QUEST Nawabshah, Pakistan
††††††Department of Information Technology, SBBUSB Naushahro Feroze Campus, Pakistan

Summary
Point-to-point protocols such as TCP uses the single path for data transmission. If the path is temporary not available due to errors, it stops the communication between two endpoints. The availability of Internet is increasing and such temporary path failures may be resolved by using the devices that establish multiple connections between two endpoints. Due to the popularity of such devices that provide the feature of multiple simultaneous connections, the protocol designs are also changed. Stream Control Transmission Protocol (SCTP) with Concurrent Multipath Transmission (CMT-SCTP) provides the feature of simultaneous multipath transmission. With such feature of CMT-SCTP, the mobile application developers are at keen interest to use it. This study evaluates the single and multipath transmission using SCTP and CMT-SCTP. The investigations on a simple scenario proved that CMT-SCTP distributes the data among the available paths. CMT-SCTP is also good for load sharing when some or all of paths encounter data loss due to link failures. A realistic scenario is proposed for the experimentation of SCTP and CMT-SCTP. The scenario also includes the background traffic by using TCP. In the simulations of lossless scenario, the CMT-SCTP improves throughput of SCTP by 23%. However, on the scenario of lossy paths the CMT-SCTP improves throughput by 79%.

Key words:
SCTP, multipath transmission, throughput, TCP and CMT-SCTP

1. Introduction

At present the mobile application development for the emergency services such as online video streaming and collaboration has gathered the attraction of many researchers. For emergency service type of mobile applications it is very important to have the Internet access available all of the time. The solution to Internet access is the multiple connections of a mobile phone through multiple service providers by using more than one interface cards. However, the question arises that the traditional TCP protocol only allows a device to have one point-to-point connection with another device or a network. TCP is a connection-oriented protocol that creates a single connection between a source and a destination. However, in reality the number of connections between two devices can be increased. The main advantage of multiple connections is the higher transmission rate due to the concurrent data transmission over connections between two devices. TCP is the backbone protocol for the Internet communication and most of the applications such as WWW (world wide web), file transfer and remote desktop are dependent on it. Another protocol, which provides all of the services of TCP, is the stream control transmission protocol (SCTP) [1, 2]. One of the best services of SCTP is the creation of multiple connections between two devices simultaneously. The new version of SCTP is CMT-SCTP [3]. The SCTP utilizes one connection for the data transmission and rest of the connections for backup and emergency service. On the other hand, CMT-SCTP utilizes all of the multiple paths between a source and a destination for the data transmission as well as for the emergency services. It is obvious to say that, the transmission rate of CMT-SCTP is more that the SCTP and TCP. Moreover, CMT-SCTP distributes the data among the number of paths. In order to distribute the data evenly on the multiple paths CMT-SCTP uses the round robin algorithm. An example of the SCTP protocol where multiple connections by using more than one IP addresses is shown in Fig. 1.

Considering the requirements of mobile application developers, CMT-SCTP is a fit. However, the deployment of CMT-SCTP to the Internet applications is still in process. There exist many research questions in the design of CMT-SCTP in term of flow control, error control and reliability. Many researchers proposed extensions of CMT-SCTP to tackle the issues [5-8,13-16]. The work in [9-12] compares CMT-SCTP with the basic SCTP while particularly analyzing the receiver buffer. This paper provides the details of single and multipath transmission using the SCTP and CMT-SCTP. A network topology is proposed to simulate the single and multipath transmission. To simulate the existing Internet traffic several number of background traffic agents are used in the experimentation. The agents use TCP protocol for the data transmission. The overall proposed network used the TCP flow as well the SCTP flow. This work also provides the details of background traffic on the performance of normal data flow of SCTP.

Manuscript received April 5, 2018
Manuscript revised April 20, 2018

In rest of the paper, the details of the experimentation along with the configuration parameters are presented in SECTION 2. The discussions on the obtained results are presented in SECTION 3. In last the conclusion is available in SECTION 4.

2. Details of Simulation and Topology

For the simulation NS-2 [17] is used. It is installed on Linux Ubuntu 14.04.5 LTS 64-bit PC (AMD64). The version of the simulator installed is NS-2.35. The hardware used is a standalone system that contains Dual-core AMD Fusion™ processor, with 2 gigabytes of RAM and 80 gigabytes of secondary memory. The implementation code for the required protocols SCTP and CMT-SCTP, in available in the simulator. The scenario contains 27 nodes that are connected as show in Fig. 2. Two nodes (sender and receiver) are set to as multihomed nodes. Node 0 is the sender and Node 24 is the receiver. Each node contains two interfaces (primary and secondary). The sender's primary interface is connected via duplex link to an intermediate node (Node 3) and then receiver's primary interface is also connected via duplex link to the same intermediate node. This connection of interfaces makes it as the primary transmission path (0-1-3-25-24) of the network. On the other hand the secondary interface of sender node is connected to receiver's interface via duplex link forming the secondary path (0-2-26-24). To configure the scenario as real-life, 10 nodes as the background traffic generators were added to the primary path. All these 10 nodes are connected with Node 1. Similarly the 10 nodes are added as the background traffic receiver nodes. These nodes are connected with Node 3. There are two types of proposed scenarios. In the first scenario all the paths carry data and there is no packet loss added, however due to the overflow of queue the packets may be discarded. In second scenario the 10% of packet loss is added so that it should work and appear as scenario that represents a real-time network.

The duplex links between first 10 background traffic nodes and sender's primary interface node are configured to 1Mbps of bandwidth and 30ms of delay. Each node uses a DropTail queue with the maximum size of 50. The duplex of the secondary path is set to 2Mbps bandwidth and 30ms of delay with 10% of packet loss in transmission. The duplex links between other 10 background traffic nodes and receiver's primary interface are configured to 1Mbps bandwidth and 30ms of delay. The implementation of protocols is applied on each node, on sender and receiver nodes the SCTP and CMT-SCTP agents are implemented on two scenario scripts, and are set to perform the multihoming feature.

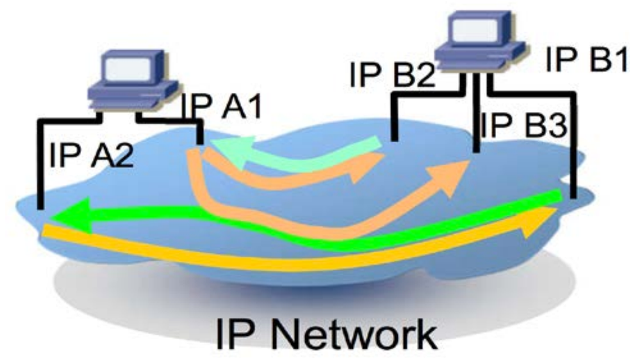

Fig. 1 Two endpoints in SCTP with multiple interfaces

In order to realize the existing Internet applications where TCP is a dominated protocol. The TCP is used for the nodes, which provide the background traffic in the given scenario. A TCP agent is attached to Node 4 (the 1st node of background traffic), and a connection is established to a TCP sink agent attached to Node 23 (the last node of background traffic). Similarly, all the nodes attached at Node 1 are configured as TCP agents and all the nodes configured at Node 3 are configured as TCP sinks.

As default, the maximum size of a packet that a TCP agent can generate is 1Kbyte but in proposed scenario the maximum size of a packet that a TCP agent can generate is set up to 1.5 Kbytes. A TCP sink agent generates and sends ACK packets to the sender (TCP agent) and then releases the received packets to the upper layers. For simple SCTP scenario a SCTP multihomed agent is attached to Node 0 for sender, and another SCTP multihomed agent is attached to Node 24 for receiver. For experimentation on CMT-SCTP, a "SCTP/CMT" multihomed agent is attached to Node 0 for sender, and another "SCTP/CMT" multihomed agent is attached to the Node 24. At the sender side, Node 1 and 2 are configured as NIC cards for sender. On the receiver side, Node 25 and 26 are configured as the NIC cards for the receiver. The overall scenario looks like a two path scenario between two endpoints. Due to the background traffic on the primary path, it is overloaded when compared with the secondary path. The parameters such as congestion window, receiver and sender window, slow start threshold and other important parameters are left to default values. The simulation time is 300 seconds. NS2 generates a trace file and an animation file after the completion of simulation. The trace file contains all of events of data and control transmission. The data obtained from the NS2 is analyzed by the AWK scripting language.

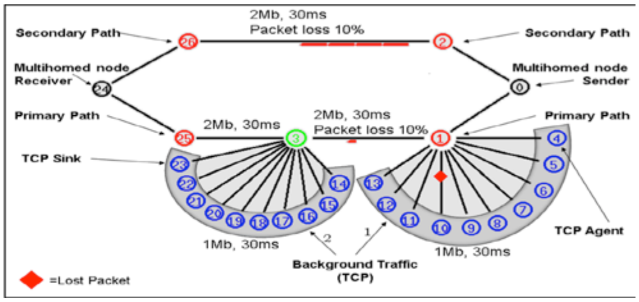

Fig. 2 Simulation Model

3. Results and Discussions

3.1 Throughput

The throughput is a network performance measure that refers to the amount of data travels from a sender to a receiver in one second. Network throughput is usually represented as an average and measured in bits per second (bps), or in some cases as data packets per second.

On the network scenario with no packet-loss using SCTP protocol the throughput measured on its primary path is 1.9032Mbps while on the secondary path of network the throughput measured is 2.25191e^-05Mbps. Hence it totals to 1.90322Mbps for the SCTP protocol. The throughput of SCTP is always the throughput of its primary path because it utilizes only one path at a time for transmission. The secondary path is occupied with the control information. A secondary path may also allow the data transmission when there is some problem in the primary path. The network scenario with no packet-loss using CMT-SCTP protocol the throughput measured on its primary path is 1.16856Mbps. The throughput of secondary path reaches to 1.17209 Mbps. The total throughput obtained by CMT-SCTP is 2.34059Mbps is the given scenario. The total throughput of CMT-SCTP is greater than SCTP because CMT-SCTP uses both of the paths for the data transmission (see Fig 3a). The values of throughput of primary and secondary path of CMT-SCTP are very similar because of the load-sharing property for data distribution of the protocol.

In the network scenario with 10% of packet-loss using SCTP protocol, the throughput measured on its primary path is 0.147815Mbps. On secondary path of network where control data is sent, the throughput measured is 0.0148973Mbps. The total throughput of the network with SCTP is measured as 0.162693Mbps. To total throughput in the given scenario is obtained by adding the throughput of primary and secondary paths. The network scenario with 10% of packet-loss using CMT-SCTP protocol the throughput measured on its primary path is 0.283721Mbps while on the secondary path of network the throughput measured is 0.168305Mbps. The total throughput of the network with CMT-SCTP is 0.451918Mbps. In this experiment, the throughput of CMT-SCTP is greater than SCTP. The throughput of SCTP is affected by the packet loss added on the path (see Fig 3a). If a packet loss occurs in the primary path (or if there are some errors in the primary path) the SCTP switches to the secondary path that waste some time. However, in CMT-SCTP the path error on one path may not affect the transmission on the other path.

In both of the experiments CMT-SCTP outperforms the SCTP. CMT-SCTP improves throughput of SCTP by 36% in the no-loss scenario, however on the scenario of paths with added packet loss the CMT-SCTP improves throughput by 79%.

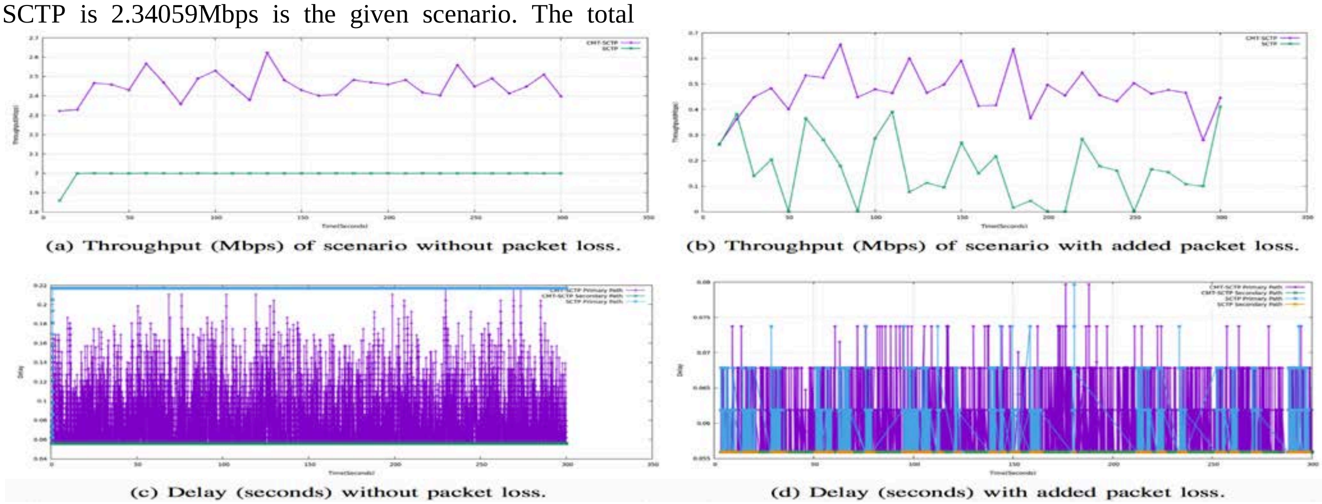

Fig. 3 Throughput and Delay

3.2 Delay

Delay is used as the second performance metric in the analysis. It is used for analysis in most of the telecommunications systems design. It specifies the fraction of time required by bits to traverse from source to destination. It is measured in seconds. Delay is calculated no matter which path the packets take for the transmission.

In lossless network scenarios SCTP does only send data packets via primary path while the secondary path is only used to send control information. It is the reason due to which the delay on the secondary path is not useful to measure. In primary path of SCTP the delay for each packet sent at any time varies and it is calculated. The delay of SCTP on primary path is 0.22seconds as shown in Fig 3c, whereas the delay of CMT-SCTP is 0.10seconds.

While analyzing the trace file obtained from the simulation, in the paths of 10% loss scenario with SCTP the protocol started transmission to destination via primary path at 180.704096seconds of simulation and the time duration that packet required to reach to the destination is 0.073744seconds. The same scenario with CMT-SCTP the packet transmission is started at 188.091152seconds of simulation and the time duration that packet required to reach to the destination is 0.079680seconds. In the proposed scenarios the delay of CMT-SCTP is shorter than the delay of the SCTP protocol. It is due the retransmission policies used by CMT-SCTP for the efficient data transfer. The delay of SCTP is also not worst when compared with the CMT-SCTP. It is because the proposed simulation scenario is very simple and contains only two paths. If primary path fails the sender moves to the secondary path. However, if a scenario is used when more than two paths, then the protocol requires some time to select the next available path.

4. Conclusion

Multipath transmission is beneficial in terms of data rate. More and more mobile applications require higher data rate and smoothness in transmission particularly in emergency such as online medical operation. SCTP and CMT-SCTP provide the ability to devices to connect with each other via multiple paths. In this work, SCTP is compared with the CMT-SCTP for multipath transmission. A realistic network scenario is proposed where the paths carry data both from SCTP and TCP protocols. Moreover, a random packet loss is also added in the network scenario. The experiment on the network proved that CMT-SCTP improvise the throughput and reduces the end-to-end delay when compared with SCTP. In future, the experimentation of SCTP over android applications would be a worthwhile research.

References

[1] Randall Stewart. Stream control transmission protocol. IETF, Standards Track RFC 4960, 2007. URL http://tools.ietf.org/html/rfc4960.

[2] Randall Stewart and P Amer. Why is sctp needed given tcp and udp are widely available. Internet Society Member Briefing, 17, 2004.

[3] Preethi Natarajan, Nasif Ekiz, Paul D Amer, Janardhan R Iyengar, and Randall Stewart. Concurrent multipath transfer using sctp multihoming: Introducing the potentially-failed destination state. In NETWORKING 2008 Ad Hoc and Sensor Networks, Wireless Networks, Next Generation Internet, pages 727–734. Springer, 2008.

[4] Halepoto, Imtiaz A., Francis CM Lau, and Zhixiong Niu. "Scheduling over dissimilar paths using CMT-SCTP." Ubiquitous and Future Networks (ICUFN), 2015 Seventh International Conference on. IEEE, 2015.

[5] Halepoto, Imtiaz Ali. "Scheduling and flow control in CMT-SCTP." HKU Theses Online (HKUTO) (2014).

[6] Halepoto, Imtiaz A., Francis CM Lau, and Zhixiong Niu. "Management of buffer space for the concurrent multipath transfer over dissimilar paths." Digital Information, Networking, and Wireless Communications (DINWC), 2015 Third International Conference on. IEEE, 2015.

[7] Bhangwar, Noor H., Halepoto Imtiaz A., Sadhayo Intesab H., Khokhar H., Laghari Asif A., "On Routing Protocols for High Performance." Studies in Informatics and Control 26.4 (2017): 441-448.

[8] Halepoto, Imtiaz A., Francis CM Lau, and Zhixiong Niu. "Concurrent multipath transfer under delay-based dissimilarity using SCTP." Computing Technology and Information Management (ICCTIM), 2015 Second International Conference on. IEEE, 2015.

[9] Hakim Adhari, Thomas Dreibholz, Martin Becke, Erwin P Rathgeb, and Michael Tuxen. Evaluation of concurrent multipath transfer over dis- similar paths. In Advanced Information Networking and Applications (WAINA), 2011 IEEE Workshops of International Conference on, pages 708–714. IEEE, 2011.

[10] Thomas Dreibholz, Martin Becke, Erwin P Rathgeb, and M Tuxen. On the use of concurrent multipath transfer over asymmetric paths. In Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE, pages 1–6. IEEE, 2010.

[11] Thomas Daniel Wallace. Concurrent Multipath Transfer: Scheduling, Modelling, and Congestion Window Management. PhD thesis, The University of Western Ontario, 2012.

[12] Thomas Dreibholz, Erwin P Rathgeb, Irene Rüngeler, Robin Seggelmann, Michael Tüxen, and Randall R Stewart. Stream control transmission protocol: Past, current, and future standardization activities. Communications Magazine, IEEE, 49(4):82–88, 2011.

[13] T Daniel Wallace and Abdallah Shami. A review of multihoming issues using the stream control transmission protocol. Communications Surveys & Tutorials, IEEE, 14(2):565–578, 2012.

[14] Michael Scharf and Sebastian Kiesel. Nxg03-5: Head-of-line blocking in tcp and sctp: analysis and measurements. In Global Telecommunications Conference, 2006. GLOBECOM’06. IEEE, pages 1–5. IEEE, 2006.

[15] Michael Tuexen, Peter Lei, and Randall R Stewart. Stream control transmission protocol (sctp) stream reconfiguration. 2012. URL http: //www.ietf.org/rfc/rfc6525.txt.

[16] Jonathan Rosenberg, Henning Schulzrinne, and Gonzalo Camarillo. The stream control transmission protocol (sctp) as a transport for the session initiation protocol (sip). Internet Engineering Task Force, Tech. Rep. RFC, 4168, 2005.

[17] Baldo, Nicola, et al. "ns2-MIRACLE: a modular framework for multi-technology and cross-layer support in network simulator 2." Proceedings of the 2nd international conference on Performance evaluation methodologies and tools. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2007.

Imtiaz Ali Halepoto received Bachelor of Engineering degree in Computer Systems Engineering from QUEST Nawabshah, Pakistan, and both M.Sc and PhD from the Department of Computer Science, the University of Hong Kong in 2010 and 2015. Currently, he is working as Assistant Professor at the Department of Computer Systems Engineering QUEST Nawabshah. His research interests are in communication, network protocols and the heterogeneous networks.

M. Sulleman Memon received the B.E in Computer Engineering and M.E. in Software Engineering from MUET, Jamshoro Pakistan in 1990 and 2004, respectively. He is now a PhD scholar at QUEST Nawabshah. He has submitted thesis. He is working as Assistant Professor in the Department of Computer Engineering. QUEST, Nawabshah. He is author of many International and national papers. He has presented his work at many countries of the word in International Conferences. His field of study is Wireless communications. He is Senior Member of IACSIT and member of Pakistan Engineering Council, ACM, and IEEE.

Nazar Hussain Phulpoto is working as Associate Professor at the Department of Information Technology, QUEST Nawabshah. He has PhD degree in Management Sciences and 15 years of research and teaching experience. His research expertise includes Management Sciences and Information Technology.

Ubaidullah Rajput received his Bachelor's Degree in Computer System Engineering from Quaid-e-Awam University of Engineering, Science and Technology (QUEST), Pakistan in 2005. He received his Master's in Computer System Engineering from NUST Islamabad, Pakistan in 2011. He successfully completed his PhD in Computer Engineering from Hanyang University, Korea in 2017. His research interests are security and privacy issues in crypto-currency, security and privacy issues in VANETS, Internet of Things (IoT), mobile social networks and cloud computing. He has more than 11 years of teaching and research experience and currently working as Assistant Prof. in Quest Pakistan. He has served as a reviewer in many conferences and journals. He is an author of many International and national papers.

Muhammad Yaqoob Junejo is working as Teaching Assistant at the Department of Information Technology, Shaheed Benazir Bhutto University Shaheed Benazirabad, Naushahro Feroze campus. He has B.E degree in Computer Systems Engineering from QUEST Pakistan. His research expertise includes computer networks and protocols for parallel data transmission.