SCTP: Are you still there?

Zeynep Ince, Richard von Seck*
*Chair of Network Architectures and Services, Department of Informatics
Technical University of Munich, Germany
Email: zeynep.ince@tum.de, seck@net.in.tum.de

***Abstract*—Stream Control Transmission Protocol (SCTP) is a transport layer protocol initially designed for telephony signalling over IP networks. It is similar to Transmission Control Protocol (TCP) and User Datagram Protocol (UDP). Moreover, it has some crucial differences making the protocol outstanding such as multihoming and multistreaming. Through these features, multiple streams can be handled simultaneously, and the transmission goes on in case of a failure without interruption. So why is SCTP less used and known compared to TCP and UDP? This paper will try to answer this question in more detail by comparing SCTP with other transport protocols and detailing the current state-of-art of SCTP research.**

***Index Terms*—Stream Control Transmission Protocol, multihoming, multistreaming, Transmission Control Protocol, User Datagram Protocol**

1. Introduction

Reliable communication has been provided by Transmission Control Protocol (TCP) and unreliable communication has been provided by User Datagram Protocol (UDP) for many years [1]. However, because of the limitations they imposed, a third protocol named Stream Control Transmission Protocol was introduced. SCTP is a connection-oriented, message-based communication protocol in the transport layer that can handle multiple simultaneous streams. In 2000 it was standardized by the Internet Engineering Task Force (IETF) and the starting point of SCTP was to transport Public Switched Telephone Network (PSTN) signalling messages over IP networks [2]. TCP and UDP are the underlying concepts that made SCTP possible in the first place by combining their best features [3]. All the similarities aside, SCTP has some distinct differences and these features are multihoming and multistreaming. However, SCTP is still less known and used compared to TCP and UDP. In this paper, we will discuss the reason behind this situation by identifying the current state-of-the-art of SCTP research.

The remainder of this paper is organized as follows: First, we will take a brief look at the SCTP terminology in Section 2 and then compare SCTP with TCP and UDP in Section 3. Next, we will list some arguments about why it is less known and used in Section 4. In Section 5, we will provide an overview of the current use cases and actual specifications of SCTP and systems trying to optimize their performances using it. Finally, we will close with some conclusions and a brief overview in Section 6.

2. Brief Terminology of SCTP

Chunk: A unit of information that is sent within a packet.

Association: A protocol relationship between the two endpoints that can be uniquely identified by the transport addresses used by them; a broader concept than a connection [2].

Heartbeat: A type of chunk to check the availability of the idle destination addresses that are part of the built association. If the heartbeat acknowledgement is not returned, that particular IP address will be declared as "down" [4].

Stream: Unidirectional logical channel established from one endpoint towards another [3].

SACK: Selective Acknowledgment. When a message is received by one of the endpoints, the other endpoint should be notified back with a SACK [3]. A retransmission is only generated when SACKs report missing chunks [5].

Multihoming: Enables the SCTP host to establish an association with another host over multiple interfaces identified by different IP addresses [4].

Multistreaming: The capability to transmit several independent streams in parallel, meaning that each message sent to a data stream can have different final destinations [6].

2.1. Four-Way-Handshake

Since connections are initialized between unreliable hosts and over the unreliable internet communication system, a mechanism is needed to prevent errors [7]. The mechanism to establish an SCTP association is called four-way-handshake. The process takes place following these steps:

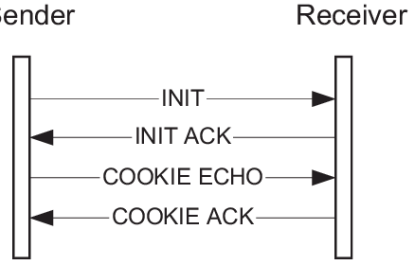

Figure 1: SCTP Four-Way-Handshake [8].

doi: 10.2313/NET-2022-07-1_09

1) The client sends an INIT signal to the server to start an association.
2) Once the server receives the INIT signal, it sends back an INIT-ACK response to the client, which contains a state cookie. This state cookie consists of a timestamp, referring to the life span of it and a Message Authentication Code (MAC), which is created by the server, including a secret key that only the server knows.
3) When the INIT-ACK signal is received, the client sends a COOKIE-ECHO response to echo the state cookie.
4) Subsequently, the authenticity of the state cookie is verified by using the secret key the MAC encapsulates. Before sending the COOKIE-ACK response, the server allocates the resources and the association's state is now ESTABLISHED.

The server does not keep any state information until the very end and the full handshake should be completed in order to have an actual state maintained on the server and no resources are allocated until the COOKIE-ECHO message is received by the receiver [8]. Moreover, when an endpoint decides to perform a shutdown, the association on each one of them will stop accepting new data and only deliver data in queue before closing [2].

3. Comparison of Features

Instead of listing the features of all protocols individually, we start by comparing them to have an overview of how they are positioned towards each other. See Table 1 for a brief overview.

Service/Features	TCP	UDP	SCTP
Transmission	Byte-oriented	Message-oriented	Message-oriented
Connection management	Connection-oriented	Connectionless	Connection-oriented
Reliability	Reliable	Unreliable	Reliable
Data delivery	Strictly ordered	Unordered	Partially ordered
Multistreaming	No	No	Yes
Multihoming	No	No	Yes

TABLE 1: Comparison of Protocols

Byte-oriented: TCP does not stream bytes over the Internet, opposing what it can be understood from this term. Enough bytes from the sending process are buffered by the TCP on the source host in order to fill a packet. This packet is then sent to its peer on the destination post. TCP there empties the contents into a receive buffer [7].

Message-oriented: Transported sequences of messages are in groups of bytes. In SCTP, these groups are called "chunks", as mentioned in Section 2.

Connection-oriented: Before the data is being exchanged, a connection should be established. In TCP this process is called the Three-Way-Handshake and in SCTP, as mentioned in Section 2.1, it is called the Four-Way-Handshake.

Connectionless: The transmission from the source to the destination starts right away, without verifying the state of the server.

3.1. TCP and UDP

TCP was standardized in 1981 and it has been the most widely chosen option for transmitting data ever since [9]. It is a connection-oriented, reliable protocol that guarantees none of the transmitted packets will get lost: TCP can retransmit them. In this sense, packets are sent strictly ordered and the receiver collects and reorders these segments conveniently. When a failover situation occurs where the data could not arrive in order, the TCP stack will wait for the retransmission and others are held. This is the situation that we call HOL blocking (Head of Line blocking) because of the strict order-of-transmission delivery of data [1].

On the other side, UDP is a connectionless protocol and packets do not necessarily arrive in order. The connection establishment is not checked like in SCTP and TCP. Packets can quickly go missing and the sender will not know whether the transmission is completed either, making UDP an unreliable protocol. The head-of-line blocking problem does not occur in UDP as well. Furthermore, TCP is byte-oriented while UDP is message-oriented.

3.2. UDP and SCTP

UDP is a connectionless, unreliable transport protocol contrary to SCTP. However, they are both message-oriented. In SCTP, an association is established after a four-way-handshake, but in UDP, no such process is needed. In UDP, the transmission starts right away without checking if it is received or not and there is no retransmission process either. It is helpful for cases where we need live real-time connections and retransmission is unnecessary. For example, retransmitting the position of an online game character from 5 seconds ago is not logical since it is not valid anymore. On the other side, SCTP can detect the loss of a packet rapidly in favor of SACK usage. Moreover, the UDP Header is much smaller when compared with SCTP and TCP; this makes UDP lighter and consequently attractive for fast and efficient handling of audio, image and video data traffic [10].

3.3. TCP and SCTP

TCP and SCTP are very similar protocols, and both of them are reliable and connection-oriented. SCTP was designed to push the edge of the envelope of TCP. The useful features of TCP were inherited and new features were added [11].

TCP uses a three-way-handshake where initial sequence numbers are being exchanged and SCTP uses a four-way-handshake including a signed cookie, as described in Section 2.1. However, the SCTP connection process is more complex [11] and the use of a cookie improves the vulnerability of TCP to SYN flooding [2]. Correspondingly, the security of the protocol is improved. [8].

SCTP is message-oriented, unlike TCP, which is byte-oriented. However, SCTP transmits its packets in chunks and TCP buffers enough bytes to fill a packet before sending, as described at the beginning of this section.

The two most important differences of SCTP compared to TCP are the multihoming and multistreaming features. In a traditional TCP connection, an IP address and a port is chosen from each end and with them, packets are sent and received. However, multihoming allows an SCTP association over multiple interfaces. Between the

different path options, a primary path is selected and the availability of the rest is constantly checked with heartbeats. If a failover situation occurs with the primary path, one of the other options is used and the transmission will go on without interruption [6]. Thus, the loss of a message does not affect the rest of the deliveries and the head-of-line blocking problem is avoided in SCTP. By means of this, the fault tolerance [12] and the robustness of the server is improved [8]. That is why it is a desired functionality [6].

Moreover, TCP has a strict order-of-transmission delivery mechanism [4] and SCTP has reliable ordered delivery and reliable unordered delivery services at the same time due to the multistreaming feature. This feature makes the transmission of several independent data streams in parallel possible. [8]. SCTP determines in which order to present the messages to the destination and this approach eliminates the head-of-line-blocking delay [4] that is caused by the strictly ordered delivery [1]. This combination of streams and unordered delivery simultaneously is helpful for Internet applications to have a better performance when a failover occurs, such as a network loss. By courtesy of this the overall latency is reduced and transmission efficiency is improved [3]. These features make SCTP valuable for real-time data transfers such as audio and video.

Taking into account all of these, we can say that SCTP is more robust and secure than TCP, under the assumption that the introduced features are used [13].

4. Why SCTP is less used

Even though all the advantages SCTP has over TCP, it is less known and used. The main reason behind this is that the client and server applications might need modifications such as upgrading IP stacks to use SCTP instead of TCP or UDP [9] and this would be too much work. In-network devices like NAT gateways, does not support SCTP well [14]. Moreover, TCP was first-to-market and for the most part, TCP is sufficient and works just fine. That is why TCP was the dominant transport protocol for a very long time [9].

In 2009, Google introduced a new transport protocol called QUIC: Quick UDP Internet Connections. It is a multiplexed low-latency transport protocol designed to improve the web performance [15]. As the Internet traffic increases rapidly, it is necessary to look for new technologies [16]. The most crucial advantage QUIC has over SCTP is that QUIC does not require changes to the operating system and this makes QUIC easily deployable with applications that are already in use. Briefly, QUIC can handle multiple request/response pairs concurrently on a single connection by using multiple streams and a packet loss does not block the rest of the connection [17], but it is still under active development and some specifications are still missing. Considering the apparent dominance of Google over the Internet, QUIC is widely used in Chrome clients [14]. This helped SCTP remain its obscurity too.

5. Current and Possible Use Cases

Still, SCTP has some essential use cases and it helps some systems optimize their performances. Multipath transport layer protocols such as SCTP are gaining increased attention every passing day [18]. SCTP is recently supported by a variety of operating systems, such as AIX, Solaris, Linux and Windows: Microsoft provides user space for SCTP implementations in the Windows family too [19].

5.1. Long-Term Evolution (LTE)

Diameter Protocol in LTE provides authentication, authorization, and accounting (AAA) services. LTE is closely related to 4^{th} generation mobile data transfer, which gives cause for higher data transmission. The multistreaming feature of SCTP is helpful at this point, making SCTP the up-front transport protocol being used to transport messages. Mobile consumers expect high-quality data experiences and invisible high-speed access.

5.2. Concurrent Multipath Transfer (CMT)

CMT is a process of using multiple networks to transfer data instead of selecting a single network interface for transmission. Several pieces of research have been done to test the multiple-path transmission of SCTP with CMT [19]. However, considering the dominance of TCP as a transport protocol, a proxy technique is needed to translate the TCP flows into SCTP streams without being obligated to make significant changes at end hosts or servers. Tachibana et al. [9] claim that the multihoming feature of SCTP would increase the aggregated throughput and the robustness of communication.

Liao et al. [5] introduced a modification of SCTP called cmpSCTP. With this solution, the transmission is updated based on real-time and all of the available paths are used simultaneously, unlike SCTP, where a chosen primary path is used. Moreover, as the states of paths change, the transmission strategy is also changed by cmpSCTP, and the flows between paths can be switched smoothly. Cloud computing is one of the examples where CMT and SCTP are used together by combining the multihoming feature of SCTP and multipath transfer technology of CMT [19].

5.3. Internet of Things (IoT) Sensors

The IoT is a giant network with connected devices and these devices share the data they collect over their sensors to help us understand and measure the planet around us. Sensors are embedded in most physical devices such as our smartphones and generate large amounts of real-time data. These are collected in sink nodes and transmitted over heterogeneous networks afterwards. It is essential that the packet loss rate is as low as possible and transmission quality is high. For this reason, the transport layer protocol should be chosen wisely. A switch of TCP and SCTP might be suitable for this case. Sun et al. [11] compared the performances of both protocols and proposed that SCTP is reliable, but TCP has higher transmission stability, which brought the idea of combining only the better sides of both protocols as a method out. This rendered the network's state prediction possible considering the packet loss rate. By courtesy of the multistreaming feature, multiple requests can be processed and

doi: 10.2313/NET-2022-07-1_09

with the multihoming feature the transmission efficiency was improved. Considering both of these features belong to SCTP, it can also be used in the field of IoT.

5.4. Session Initiation Protocol (SIP)

SIP is a protocol for managing communication sessions such as voice and video calls over internet telephony or mobile phone calls over LTE. Large amounts of data are exchanged between SIP entities and the protocol is independent of the underlying transport protocol. It can be used with TCP, UDP or SCTP. However, choosing SCTP as the transport protocol would provide some crucial advantages [12].

As we have mentioned before, SCTP uses Selective Acknowledgement (SACK) to generate the retransmission of a missing chunk and retransmissions take place only after SACKs report them. The loss of a SIP message is detected immediately. Moreover, this loss does not affect the rest of the transmissions, so if multiple transactions are happening at the same time SCTP will handle them with relative ease. SIP entities choose the server on the next-hop by checking if it supports SCTP to establish an association [12].

5.5. Satellites

Satellite links ensure some essential services we use every day, such as navigation services, television and telephony and without them, the Internet may not be the same. Satellite networks have a large transmission distance. Therefore, problems like corruption losses due to wireless links and long propagation delays come into sight. However, TCP was not designed for such networks [20].

SCTP is recommended for running over satellite networks because of many reasons. In a satellite environment, multiple segment losses are incidental and with SACKs, it is possible to react rapidly. The multihoming feature derives satellite networks to be fault-tolerant and reliable and the multistreaming feature eliminates the HOL blocking by reducing the receiver buffer size requirements [18].

5.6. Datagram Transport Layer Security (DTLS)

A datagram provides a connectionless communication service across packet-switched networks and DTLS is a communications protocol that maintains security to datagram-based applications. In this sense, using DTLS over SCTP means providing a secure channel to applications that are using SCTP as their transport protocol. By courtesy of this, eavesdropping is prevented, where information gets stolen while being transmitted—this way the confidentiality, authenticity and integrity of the network is ensured. Using DTLS over SCTP reinforces preservation of message boundaries, ordered and unordered delivery of SCTP user messages and a large number of unidirectional and bidirectional streams [21].

6. Conclusion

This paper has compared SCTP with other transport protocols and listed the current use cases. SCTP is a message-oriented, connection-oriented and reliable protocol and most importantly it can deal with multihomed hosts and manage multiple streams at the same time. However, TCP has been the dominant protocol for many years and SCTP was not as known and used compared to the other protocols. The main reason for that is TCP being first to the market. There are fewer client and server applications supporting SCTP, but many applications support TCP. Furthermore, TCP is sufficient in most cases. The introduction of QUIC by Google that is easily deployable without any changes and is similar to SCTP, let the protocol remain unknown.

SCTP is currently being used in LTE technology, Concurrent Multipath Transmission, IoT sensors, Session Initiation Protocols, satellites, Datagram Transport Layer Security and cloud computing. SCTP has a usage area in the telecommunications industry that it is sufficient for, but it might be the case that QUIC will be preferred for other industries in the future. It is similar to SCTP and will keep getting better since it is still under active development.

References

[1] R. Stewart, M. Tüxen, and P. Lei, "Sctp: What is it, and how to use it?" in *Proceedings of BSDCan: The Technical BSD Conference*. Citeseer, 2008.

[2] R. R. Stewart, "Stream Control Transmission Protocol," RFC 4960, Sep. 2007. [Online]. Available: https://rfc-editor.org/rfc/rfc4960.txt

[3] "Sctp and diameter parameters for high availability in lte roaming," Ph.D. dissertation, accessed: 17.12.2021. [Online]. Available: http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-163254

[4] R. Stewart and C. Metz, "Sctp: new transport protocol for tcp/ip," *IEEE Internet Computing*, vol. 5, no. 6, pp. 64–69, 2001.

[5] J. Liao, J. Wang, and X. Zhu, "cmpsctp: An extension of sctp to support concurrent multi-path transfer," in *2008 IEEE International Conference on Communications*. IEEE, 2008, pp. 5762–5766.

[6] S. Y. Shahdad, G. Amin, and P. Sarao, "Multihoming and multistream protocol in computer networks," 2014.

[7] J. Postel *et al.*, "Transmission control protocol," 1981.

[8] G. Camarillo, H. Schulzrinne, and R. Kantola, "Signalling transport protocols," 03 2002.

[9] A. Tachibana, Y. Yoshida, M. Shibuya, and T. Hasegawa, "Implementation of a proxy-based cmt-sctp scheme for android smartphones," in *2014 IEEE 10th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, 2014, pp. 660–665.

[10] H. Mohamad Tahir, M. A. Abu Seman, S. Shelen, M. S. Selan, S. H. Abdi *et al.*, "Performance comparison of sctp and udp over mobile ad hoc networks," *International Journal of Computer Science Issues*, vol. 9, no. 4, pp. 443–448, 2012.

[11] W. Sun, S. Yu, Y. Xing, and Z. Qin, "Parallel transmission of distributed sensor based on sctp and tcp for heterogeneous wireless networks in iot," *Sensors*, vol. 19, no. 9, p. 2005, 2019.

[12] J. Rosenberg, H. Schulzrinne, and G. Camarillo, "The stream control transmission protocol (sctp) as a transport for the session initiation protocol (sip)," *Internet Engineering Task Force, Tech. Rep. RFC*, vol. 4168, 2005.

[13] R. R. Stewart and Q. Xie, "Stream control transmission protocol (sctp): a reference guide," 2001.

[14] A. Joseph, T. Li, Z. He, Y. Cui, and L. Zhang, "A Comparison between SCTP and QUIC," Internet Engineering Task Force, Internet-Draft draft-joseph-quic-comparison-quic-sctp-00, Mar. 2018, work in Progress. [Online]. Available: https://datatracker.ietf.org/doc/html/draft-joseph-quic-comparison-quic-sctp-00

[15] A. Langley, A. Riddoch, A. Wilk, A. Vicente, C. Krasic, D. Zhang, F. Yang, F. Kouranov, I. Swett, J. Iyengar *et al.*, "The quic transport protocol: Design and internet-scale deployment," in *Proceedings of the conference of the ACM special interest group on data communication*, 2017, pp. 183–196.

[16] P. Megyesi, Z. Krämer, and S. Molnár, "How quick is quic?" in *2016 IEEE International Conference on Communications (ICC)*. IEEE, 2016, pp. 1–6.

[17] T. Viernickel, A. Froemmgen, A. Rizk, B. Koldehofe, and R. Steinmetz, "Multipath quic: A deployable multipath transport protocol," in *2018 IEEE International Conference on Communications (ICC)*. IEEE, 2018, pp. 1–7.

[18] J. Deutschmann, K.-S. Hielscher, T. Keil, and R. German, "Multipath communication over terrestrial and satellite links," in *2018 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN)*. IEEE, 2018, pp. 119–121.

[19] L. Zheng, X. Zhang, S. Zhang, and X. Chen, "Research on multipath network in cloud computing based on sctp," in *2021 8th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2021 7th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, 2021, pp. 30–35.

[20] S. Fu, M. Atiquzzaman, and W. Ivancic, "Sctp over satellite networks," in *2002 14th International Conference on Ion Implantation Technology Proceedings (IEEE Cat. No. 02EX505)*. IEEE, 2003, pp. 112–116.

[21] M. Tuexen, R. Seggelmann, and E. Rescorla, "Datagram transport layer security (dtls) for stream control transmission protocol (sctp)," *Request for Comments*, vol. 6083, 2011.