

SCTP: State of the art in Research, Products, and Technical Challenges

Shaojian Fu and Mohammed Atiquzzaman
Telecommunications and Networks Research Lab
School of Computer Science
University of Oklahoma,
Norman, OK 73019-6151, USA.
Email addresses: {sfu, atiq}@ou.edu

Abstract—The Stream Control Transmission Protocol (SCTP) is being standardized by IETF as a reliable transport protocol to address a number of limitations of TCP. Due to its attractive features such as multi-streaming and multi-homing, SCTP has received much attention from the research community and industry. This paper serves as a tutorial by introducing the main features of SCTP, and discussing the state of the art in SCTP research activities. We also provide a survey on the available products which implement SCTP. Finally, the challenges faced by SCTP research community are identified with a view to stimulating further research.

I. INTRODUCTION

The past five years have witnessed a strong trend of convergence in PSTN/ISDN and IP-based networks, most evident in the bloom of IP telephony or Voice over IP (VoIP) applications. To reduce communication costs, it is now a common practice to transport voice over wide-area IP networks. However, many of the important services provided by the PSTN network need support of the SS7 signaling network, which is a separate network for carrying setup & tear-down messages, billing information, routing queries, etc. SS7 is designed as an open-ended common-channel signaling standard which is currently deployed by virtually all telephone service providers and interexchange carriers. To achieve complete IP telephony, there is one more task required beyond just transporting raw voice streams between VoIP gateways: transport of SS7 signaling message over IP networks.

Until late 2000, the Transmission Control Protocol (TCP) and the User Datagram Protocol (UDP) have been the only available transport layer protocols in the TCP/IP protocol suite. Transport of SS7 signaling message has a stringent requirement on reliable and timely delivery, since the information carried is critical to the operation of the network. Errors or delay in the delivery of these messages may result in circuit establishment failure or billing errors. Since the UDP is not a reliable connection-oriented protocol, it cannot be used as the transport protocol for signaling message. The SIGTRAN (Signaling Transport) working group (founded in November 1998) evaluated the applicability of TCP to this task, and several deficiencies of TCP were identified [1]:

The research reported in this paper was partially funded by NASA Grants NAG3-2528 and NAG3-2922.

- TCP's strict by-order delivery gives rise to head-of-line (HOL) blocking in some applications.
- TCP is stream-oriented instead of being message-oriented.
- TCP can't support multi-homing, which is crucial in high-availability environments such as SS7 signaling transport.
- TCP is vulnerable to blind denial of service (DoS) attacks by SYN segments.

To overcome the above limitations of TCP, a new transport protocol, called Stream Control Transmission Protocol (SCTP), was proposed by IETF in October 2000 to accomplish signaling transport [1]. It was soon noticed that SCTP should be useful in a wider range of applications instead of just the signaling transport area. As a result, the standardization work of SCTP was moved from SIGTRAN to the Transport Area Working Group (TSVWG) of IETF in February 2001.

The design of SCTP absorbed many strengths that made TCP a success during the explosive growth of the Internet, such as the window based congestion control, error detection and retransmission, etc. Moreover, SCTP incorporated several new features that were not available in TCP. Two of the most prominent of these features are:

- **Multi-homing:** Multi-homing allows two endpoints to set up an association with multiple IP addresses for each endpoint (In SCTP, "**association**" is the name for the communication relationship between end points, and is similar to "**connection**" in TCP). This built-in support for multi-homed endpoints can utilize the redundancy in network, and allow high-availability applications to perform switchover to an alternate path without interrupting the data transfer during link failure situations.
- **Multi-streaming:** Multi-streaming is used to alleviate the head-of-line (HOL) blocking effect resulting from TCP's strict by-order delivery policy. Each stream is kind of a *sub-flow* within the overall data flow, and the delivery of a *sub-flow* is independent from other sub-flows.

We will discuss a number of SCTP features, including the above two, in detail in Sec. II.

Due to its new attractive features, SCTP has received much attention from the research community, and is rapidly becoming one of the hottest topics in networking technology. Randall Stewart, one of the main designers of SCTP, had provided an excellent tutorial on SCTP back in Year 2001 [2]. The tutorial

mainly provided a basic introduction to SCTP with emphasis on the SCTP association setup and data transfer process. The *objective* of this tutorial is to update the readers with the very recent research activities related to SCTP and the problems that remain open to be solved. The *contributions* of this article can be summarized as follows:

- Provide a comprehensive survey on the state-of-the-art in the area of SCTP's congestion control, multi-homing, multi-streaming, and its application in wireless/mobile environment;
- Summarize the different implementations of SCTP and several commercially available products;
- Identify some of the challenges and issues faced by the SCTP research community.

The rest of the paper is organized as follows. In Sec. II, the main features of SCTP are reviewed to familiarize readers with the general picture of SCTP. We discuss the state of the art in SCTP research activities in Sec. III, and provide a survey of available SCTP products in Sec. IV. To stimulate further research in the area of SCTP, we identify a number of issues and challenges to be solved in Sec.V

II. CORE FEATURES OF SCTP

Like TCP, SCTP fits in the transport layer of the Internet protocol stack. Fig. 1 shows an schematic view of a SCTP association with multi-homing and multi-streaming features.

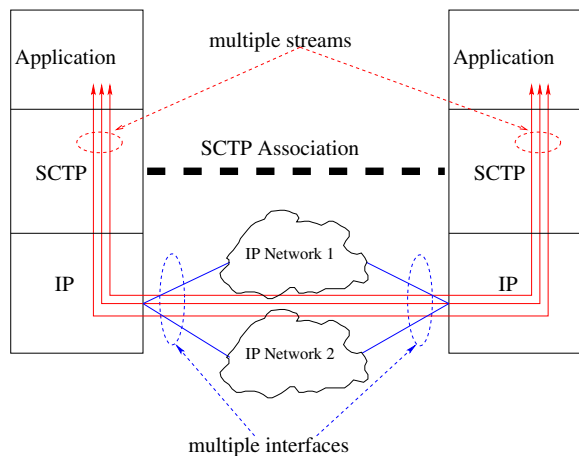


Fig. 1. Schematic view of an SCTP association.

A. Multi-homing

Multi-homing allows an association between two end points to cross multiple IP addresses or network interface cards. An example of SCTP multi-homing is shown in Fig. 2. In this figure, both endpoints A and B have two interfaces bound to the SCTP association. The two end points are connected through two kind of links: satellite at the top and ATM at the bottom. One of the addresses is designated as the primary while the other one can be used as backup in the case of failure of the primary address, or when the upper layer application explicitly requests the use of the backup. Retransmission of lost packets

can also be done over the secondary address. The built-in support for multi-homed endpoints by SCTP is especially useful in environments where applications require high-availability, such as SS7 signaling transport. The multi-homed SCTP association can speedup the recovery from link failure situations without interrupting the data transfer.

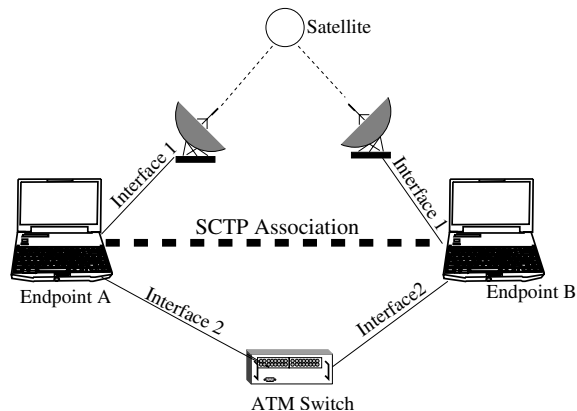


Fig. 2. An SCTP association with multi-homed endpoints.

B. Multi-streaming

Multi-streaming allows data from the upper layer application to be multiplexed onto one channel (called association in SCTP) as shown in Fig. 3. Sequencing of data is done within a stream; if a segment belonging to a certain stream is lost, segments (from that stream) following the lost one will be stored in the receiver's stream buffer until the lost segment is retransmitted from the source. However, data from other streams can still be passed to the upper layer application. This avoids the head of line blocking (HOL) found in TCP where only one stream carries data from all the different upper layer applications.

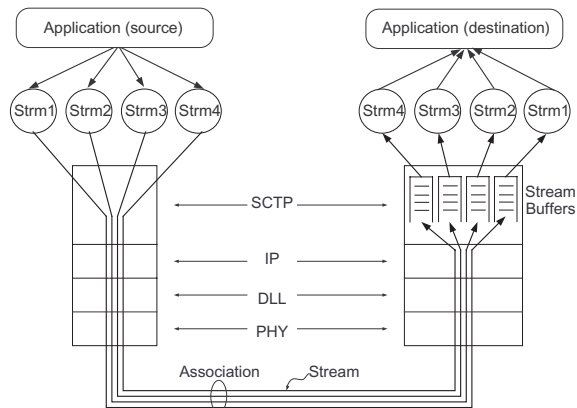


Fig. 3. An SCTP association consisting of four streams carrying data from four upper layer applications.

An example of using multi-streaming in Web browsing is shown in Fig. 4. Here, an HTML page can be split into five objects: a java applet, a ActiveX control, two images, and plain text. Instead of creating a separate connection for each object (as in TCP), SCTP can make use of its multi-streaming feature

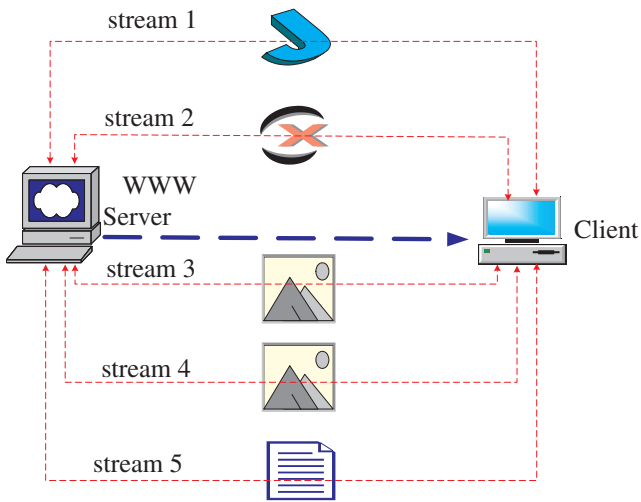


Fig. 4. Multi-streaming in Web browsing.

to speedup the HTML page transfer. By transmitting each object in a separate stream, the HOL blocking between different objects can be eliminated. If packets belonging to one object is lost during the transfer, the other objects can still be transmitted while the lost object is retransmitted. This will give a better response time to users, and eliminate the unfair network-sharing problem caused by opening multiple TCP connections simultaneously by a single user.

C. Congestion Control

SCTP congestion control is based on the well proven rate-adaptive window-based congestion control scheme of TCP. This ensures that SCTP will reduce its sending rate during congestion in the network, and prevent congestion collapse in a shared network. SCTP provides reliable transmission, and detects lost, reordered, duplicated or corrupt packets. It provides reliability by retransmitting lost or corrupt packets. However, there are several major differences between the congestion control mechanisms of TCP and SCTP as summarized below:

- SCTP incorporates a fast retransmit algorithm based on SACK gap reports similar to that described in [3]. This mechanism speeds up the loss detection and increases bandwidth utilization. One of the major differences between SCTP and TCP is that SCTP doesn't have an explicit fast-recovery phase, but achieves this automatically with the use of SACK [1].
- In contrast to TCP, the use of SACK is mandatory in SCTP. This allows more robust reaction in the case of multiple losses from a single window of data, thereby avoiding a time-consuming slow start stage after multiple segment losses, resulting in bandwidth savings and increased throughput.
- During the slow start or congestion avoidance of SCTP, the $cwnd$ is increased by the number of acknowledged bytes; while in TCP it is increased by the number of ACK segments received.
- During congestion avoidance of SCTP, $cwnd$ can only be

TABLE I
COMPARISON OF TCP AND SCTP

Protocol	TCP	SCTP
Setup messages	three-way handshake	four-way handshake
Shutdown messages	four-way handshake	three-way handshake
Half-open support	supported	not supported
Ordered delivery	strict ordered	ordered within stream
Unordered delivery	not supported	supported
Message boundary	no boundary stream-oriented	boundary preserved message-oriented
Multi-homing	not supported	supported
SACK support	optional	mandatory
Keep-alive heartbeat	optional	mandatory
Heartbeat interval	≥ 2 hours	30 secs by default

increased when the full $cwnd$ is utilized; this restriction does not exist in TCP.

- TCP begins fast retransmission after the receipt of three DupAcks while SCTP begins after four DupAcks. SCTP is able to clock out new data on receipt of the first three DupAcks and retransmit a lost segment by ignoring whether or not the flight size is less than $cwnd$. But TCP can only begin data retransmission on the receipt of the third DupAck.

D. Differences between TCP and SCTP

In Sec. II-C, we have described the differences between the congestion controls mechanisms of TCP and SCTP. In Table I, we describe other differences between the two protocols. The first three rows compare the messages exchanged during TCP connection/SCTP association setup & shutdown. *Half-open* in the third row refers to a situation where one endpoint has finished its data transfer while expecting to receive further data from its correspondent endpoint, i.e. the connection/association is open only for one direction. TCP supports the half-open connection through four-way handshake shutdown sequence [4], while SCTP uses a three-way handshake for shutdown, but and does not support half-open association.

The fourth and fifth rows of the table relate to the delivery of segments to the application at the receiver. TCP only supports strict ordered delivery, and can result in HOL blocking in some cases. SCTP can independently deliver to application layer the

received segments that belong to different streams, provided that the sequence within the stream is preserved; SCTP can also support unordered delivery optionally, which is not possible in TCP.

The next comparison considers message boundary after transmission by the transport layer protocols. TCP is a stream-oriented protocol, and the application data are treated as a continuous byte stream instead of discrete messages. Therefore, the developers must add their own markings between messages, and have to use TCP **PUSH** flag to ensure that the complete message is received within a reasonable time. By comparison, SCTP is message-oriented. As long as there is space in the receiver buffer, the whole message is delivered without ever getting mixed with another message.

The last two rows of Table I relate to the keep-alive messages. A *keep-alive* mechanism periodically probes the other end of a connection when the connection is otherwise idle, even when there is no data to be sent. In TCP, whether this mechanism should be implemented by the transport layer or by the application itself is highly controversial. The opponents of implementing keep-alive in TCP think that this mechanism will unnecessarily waste bandwidth. If a specific TCP implementation chooses to implement the keep-alive mechanism, the default value of heart-beat interval shouldn't be less than two hours [5]. SCTP designers believe that the ability to monitor the reachability of the peer's address is crucial in high-availability applications. For example, in the SS7 network, it is desirable to receive a link failure alarm as soon as possible to take care of the problem immediately. In this sense, conserving bandwidth is not a principal consideration. Therefore, the keep-alive heartbeat is provided in SCTP as a standard mechanism instead of relying on implementations, as in TCP. In the case of SCTP, the default heartbeat interval is also reduced to a small value of 30 seconds.

This comparison of the two transport layer protocols clearly reveals the improvements of SCTP over TCP. These improvements reflect the better understanding of the deficiencies of TCP by the research community during the past twenty years of practice.

III. STATE OF THE ART IN RESEARCH ACTIVITIES

In this section, we will provide a comprehensive survey of the current research activities in the area of SCTP, with a view to providing readers with a clear picture of the state of the art in SCTP research.

A. Congestion Control

SCTP's congestion control mechanisms are slightly different from that of TCP (see Sec. II-C). If this difference causes SCTP to perform more aggressively than TCP, then on the average, SCTP traffic will consume more network resources than TCP, resulting in unfairness when users of the two protocols are competing for network resources. When SCTP is used as a general purpose transport protocol, an important concern is whether it can co-exist fairly with TCP in a shared network such as the Internet. Jungmaier et. al. investigated the flow control and bandwidth-sharing behavior of SCTP when SCTP associations and TCP connections share common wide area network

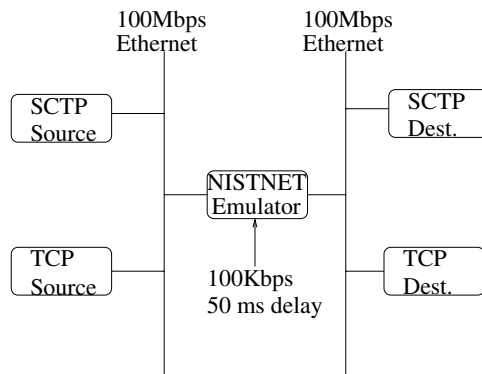


Fig. 5. TCP and SCTP sharing a common WAN link

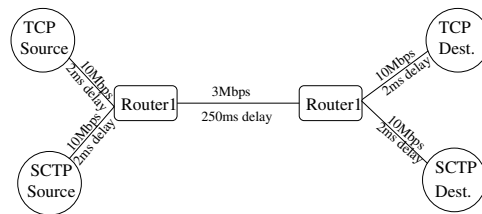


Fig. 6. TCP and SCTP sharing a common satellite link.

links [6]. The experimental configuration is shown in Fig. 5. By measuring the link layer load imposed by the different flows of the two protocols, it was shown that SCTP traffic has the same impact on the congestion control decision of TCP connections as normal TCP traffic, i.e. if we replace the transport protocol for some of the existing applications from TCP to SCTP, the performance of the remaining TCP applications is not affected. This ensures that the introduction of SCTP traffic into an existing TCP/IP network will not degrade the performance of TCP traffic, and the traffic of the two protocols can share the network resources fairly. It is a desirable property that helps gradual and seamless deployment of SCTP in the Internet without affecting existing traffic.

To investigate whether the fairness property of SCTP still holds in a network containing high bandwidth-delay product paths, Alamgir et. al. [7] compared the congestion control mechanisms of TCP and SCTP in satellite environment, which is typical of this kind of network. The simulation scenario is shown in Fig. 6. The study presented a detailed case study on the retransmission policies of the two protocols and showed that SCTP achieves a better performance than TCP even when both the protocols share a satellite path fairly. The improvement of throughput, which was reported to be up to 30.6%, resulted from different retransmission mechanisms of TCP and SCTP during congestion avoidance (see Sec. II-C).

B. Multi-homing

The main application of SCTP multi-homing is in high-availability environments to achieve fast recovery over fault conditions. In [8], the effect of SCTP multi-homing in the recovery of SS7 network linkset failure was investigated. The transport-layer seamless switch-over based on multi-homing

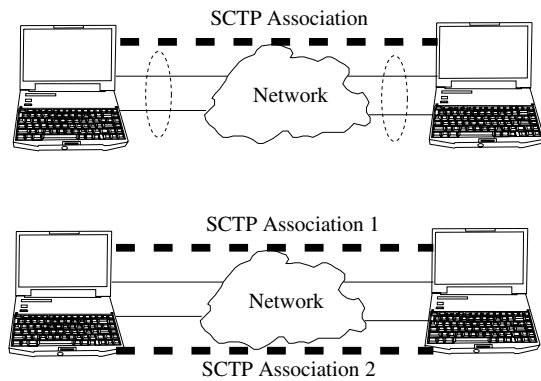


Fig. 7. Two approaches for link fail-over protection.

and the user-layer switch-over were compared. The two approaches are shown in Fig. 7.

In the top part of the figure, only one association is setup across the two available links. In the lower part of the figure, two associations are established, one association for each of the link paths. The first approach is called *transport-layer* because the failover is done by the SCTP layer and it's transparent to user applications; while the second approach requires the application to be aware of the failover, thus called *user-layer* switchover. It was reported that the first approach offered a smoother transition by keeping the average segment delay during failover at a much lower value than the second approach. The multi-homing feature of SCTP can thus help endpoints to quickly detect link failures.

C. Multi-streaming

For applications that have independent elements to be transported, such as multimedia, SCTP's multi-streaming feature helps to segment the various elements into separate streams and eliminate the HOL effect as shown in Sec. II-B. In [9], Caro et. al. showed the ability of SCTP in reducing the latency for streaming multimedia in high-loss environments. The experiment used the standard GIF compression format and eight parallel streams for the transmission of images. The study involved a network with a loss rate of 10% and 9.6kbps links. The experimental results show that multi-streaming of SCTP makes the network throughput degrade slower than TCP with an increasing loss rate. Moreover, the user satisfaction on the quality of the delivered multimedia increased with this feature. This effect is a result of partial-order delivery (maintain segment sequence only within streams but not within the overall association) of SCTP when losses occurred during the transmission.

Similarly, it was shown for the first time in [10] that multi-streaming results in higher goodput than using single streams when the receiver buffer is constrained as in the case of wireless handheld devices. The study also demonstrated that the multi-streaming feature of SCTP results in reduced buffer requirements at the receiver in the presence of losses in a satellite network. The above advantages makes SCTP an attractive transport protocol for wireless handheld devices.

D. Application in Wireless/Mobile Environment

SCTP is not limited to signaling transport only. There are research activities on the application of SCTP in other arenas, wireless/mobile application being one of the typical examples. Like TCP, SCTP is also designed with wired network in mind. a number of problems arise from wireless communications, one of which is the spurious timeout and retransmission caused by sudden long delays. In one of our previous papers [11], the effect of delay spike on SCTP in a wireless mobile environment was studied. It was shown that, like TCP, SCTP also suffers a go-back-N behavior after the delay spike. The work, however, showed that SACK can be used to make SCTP free of Spurious Fast Retransmissions. In a lossy network with small link bandwidth and large *rwnd* endpoint, SCTP performs better than TCP in the presence of delay spikes.

In [12], the performance of SCTP in Mobile-IP networks has been investigated. It was shown using *ns-2* simulations that SCTP can achieve a better performance than TCP-Reno and TCP-SACK by exploiting SCTP's support of large number of GapACK blocks in its SACK chunks. The better performance is achieved due to expediting of the error discovery and segment retransmissions. Simulation results show that the throughput improvement is especially prominent when the bottleneck link bandwidth is low.

The possibility of using SCTP multi-homing to reduce the network load after a Mobile IP handover has also been evaluated [13]. This can be achieved through assigning two address to the mobile host, one is the home address, and the other is the care-of address. The home address is kept unchanged throughout the whole life of the SCTP association, while the care-of address will be assigned by the current point of attachment to the network. During data transmission, the home address is always used to locate the mobile host, and the care-of address is used for the rest of the time. This eliminates the adverse effect of triangle routing encountered by Mobile-IP.

Mobile ad-hoc network is also one of the hottest topic in wireless research. Ye et. al. evaluated SCTP's performance in wireless multi-hop networks in [14]. They studied the impact of various IEEE 802.11 parameters on the throughput of SCTP, especially in the presence of hidden node or exposed node problems. They proposed an algorithm to overcome the *small window syndrome* that occurs when the receiver window is too small.

In summary, the research endeavors in SCTP over wireless networks are aimed at exploiting SCTP's current capabilities, or design new features that can make SCTP well suited for wireless channel characteristics and mobility issues arising from 3G and beyond wireless networks.

IV. SCTP PRODUCTS

In this section, we describe a number of SCTP products that are currently available for research work or commercial use.

A. Reference Implementation

Implementing SCTP in the operating system kernel, instead of in the user space, opens the way for SCTP to become a major transport protocol competing with TCP. The kernel reference implementation in several popular Unix operating systems

are available at [15]. These include the implementation over BSD/OS 4.3, FreeBSD 4.7, NetBSD 1.6, and OpenBSD 3.2.

B. SCTP Patch for the ns-2 Simulator

Ns-2 is a discrete event simulator targeted at networking research. It has become one of the most popular research tools in the networking area. *Ns-2* provides substantial support for simulation of transport, routing, and multicast protocols over wired and wireless (local area and satellite) networks. SCTP patch for the *ns-2* simulator has been contributed by a group at the University of Delaware [16]. The patch provides the core SCTP features (as specified in RFC 2960 [1]) including multi-streaming, multi-homing, congestion-control, chunk bundling, etc. The following are some of the limitations of the patch:

- The 4-way handshake which is used to initialize an association is done as simply as possible, and no actual tag and cookie information is exchanged between endpoints.
- A proper shutdown is not done yet. The association is terminated abruptly when the simulated connection ends.
- A multi-homed node is not implemented with actual multiple interfaces or multiple IP addresses. Each multi-homed node is actually made up of more than one node. There is a *core node* and multiple *interface nodes* are used to simulate the interfaces with the links between the nodes being hard-coded as wired links. This severely impacts the flexibility in performing simulations of SCTP over mobile environments. The authors of this paper are currently working on a new SCTP patch for *ns-2* which will remove some of the above limitations.

C. Linux Kernel SCTP (LKSCPTP)

This project is an open source implementation under GNU GPL (General Public License) to provide an SCTP module in Linux kernel [17]. The LKSCPTP project migrated to SourceForge in 2001, and now it provides support for Linux kernel 2.6.0-test4. The implemented features in this project include: association setup and takedown, sequenced delivery within streams, unordered messages within streams, data fragmentation and reassembly, congestion control, heartbeat, chunk bundling, packet validation, multi-streaming, multi-homing with failover, IPv4 and partial IPv6 addressing support, CRC32C checksum, and UDP-style sockets API. The implementation still needs to complete some important tasks such as add ICMP error handler, add IPv4-mapped-IPv6 address support, dynamically add/delete IP in an association, full IPv6 support, support large number of active associations simultaneously.

D. Commercial Products

Most of the commercial products which implement SCTP are geared towards provided a signaling transport solution. SCTP can transport various SS7 protocol types, such as MTP3, ISUP, SCCP, TCAP, etc. Fig. 8 shows an architecture where SCTP is used to transport TCAP messages. In this figure, TCAP resides in the Application Server Process (ASP), and the SS7 SCCP-User Adaptation (SUA) layer completes the adaption

from TCAP to SCTP. The transport of the SS7 signaling message over IP network occurs between the ASP and Signaling Gateway (SG), then the message is transferred through the Interworking Function (IWF) to the SS7 network.

Four of the available commercial SCTP products are discussed below:

- Cisco implemented SCTP as a part of its IP Transfer Point (ITP) product family, and was distributed in the Cisco IOS Software Releases 12.2 [18]. MTP-User-Peer-to-Peer Adaptation layer (M2PA) is used to adapt higher-layer protocols onto SCTP, and the expected traffic transported is mainly SS7 signaling messages.
- APS-SCTP/TTM [19]: the Adax SCTP/T software module is part of the Adax Protocol Software (APS) product family that has been designed for signaling transport. Adax SCTP/T provides a signaling framework that enables IP telephony networks to achieve the same levels of service quality and reliability as those expected from a Public Switched Telephony Network (PSTN).
- Signalware NexGen [20]: This software uses SCTP as the transport protocol to enable voice and data networks to converge, and allows carriers to offer enhancements such as Voice over the Internet and Internet call waiting.
- DC-SCTP [21]: The DC-SCTP product provides a flexible SCTP source code package to build devices for transport POTS signaling protocols (SS7, ISDN, etc) over IP networks.

V. ISSUES AND CHALLENGES

SCTP is a relatively new protocol, and still needs a lot of work to be refined. Following are two of the major issues that need to be addressed:

A. Meeting the Reliability Requirements of the SS7 Standard

SS7 network is stringent on the reliability requirements, and there is much concern regarding functional specifications for overcoming linkset failures and congestion in signaling network. Here, linkset is defined as the set of all links between two Signaling Point (SP) in as SS7 network. Some of the major requirements are summarized below [22]:

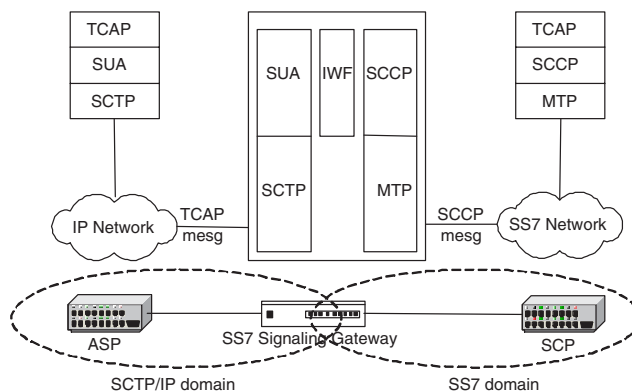


Fig. 8. Transport TCAP messages using SCTP

- The time needed to switch to another link when link failure occurs should be less than 800ms;
- The availability of communication service between two signaling points should be at least 99.998%, or a maximum downtime of 10 minutes per year;
- No more than one in 10^7 messages may be lost due to failure in the Message Transfer Part (MTP) layer;
- No more than one in 10^{10} messages may be delivered out-of-sequence to the user part due to failure in the MTP layer.

There is still no large scale experimental result available to verify that the current SCTP standard meets these requirements. A lot of simulation and experimental work still need to be carried out in this regard.

B. Performance in Wireless environments

SCTP is based on congestion control and retransmission schemes which are similar to those of TCP. SCTP and TCP are both designed with wireline environments in mind; they assume all losses are caused by congestion, and RTT changes slowly and gradually. However, wireless mobile networks encounter higher Bit Error Rates (BER) and more frequent delay spikes [11] than wireline networks. This will cause SCTP to back-off unnecessarily and result in poor throughput. Currently, there is no significant finding that can solve this kind of problem.

C. Dynamic Address Reconfiguration

The dynamic addition/deletion interface provides a graceful method to modify interfaces to an existing association. It is very useful in mission-critical applications or mobile environments by supporting service reconfiguration without interrupting on-going data transfers. This option needs to define new chunk types and parameter types, and is still at the stage of an IETF draft [23]. More work needs to be done in the test and implementation of this option to make it work as expected.

VI. CONCLUSIONS

The Stream Control Transmission Protocol (SCTP) is being standardized by IETF as a reliable transport protocol to address a number of limitations of TCP. Due to its attractive features such as multi-homing and multi-streaming, SCTP has received much attention from the research community and industry. In this paper, we first summarized several key features of SCTP, followed by a comprehensive categorized survey on the very recent research activities on SCTP. We also discussed the state-of-the-art products that are available for SCTP users. Finally, with a view to stimulating further research, we identified several issues and challenges that still need to be addressed to improve the performance of SCTP and use it for various applications in a Internet.

REFERENCES

- [1] R. Stewart and Q. Xie et. al., "Stream control transmission protocol." IETF RFC 2960, October 2000.
- [2] R. Stewart and C. Metz, "SCTP: New transport protocol for TCP/IP," *IEEE Internet Computing*, vol. 5, no. 6, pp. 64–69, November/December 2001.

- [3] K. Fall and S. Floyd, "Simulation-based Comparisons of Tahoe, Reno, and SACK TCP," *ACM Computer Communications Review*, vol. 26, no. 3, pp. 5–21, July 1996.
- [4] W. R. Stevens, *TCP/IP Illustrated, Volume 1 (The Protocols)*, Addison Wesley, November 1994.
- [5] R. Braden et. al., "Requirements for Internet hosts – communication layers." IETF RFC 1122, October 1989.
- [6] A. Jungmaier, "Performance evaluation of the stream control transmission protocol," *Proceedings of the IEEE Conference 2000 on High Performance Switching and Routing*, Heidelberg, Germany, pp. 141–148, June 2000.
- [7] R. Alamgir, M. Atiquzzaman, and W. Ivancic, "Effect of congestion control on the performance of TCP and SCTP over satellite networks," *NASA Earth Science Technology Conference*, Pasadena, CA, June 2002.
- [8] A. Jungmaier, E.P. Rathgeb, and M. Tuxen, "On the use of SCTP in failover-scenarios," *International Conference on Information Systems, Analysis and Synthesis*, Orlando, Florida, pp. 363–368, July 2002.
- [9] A.L. Caro, P.D. Amer, P.T. Conrad, and G.J. Heinz, "Improving multimedia performance over lossy networks via SCTP," *ATIRP 2001*, College Park, MD, March 2001.
- [10] M. Atiquzzaman and W. Ivancic, "Evaluation of SCTP multistreaming over satellite links." NASA Technical Report, 2002.
- [11] S. Fu, M. Atiquzzaman, and W. Ivancic, "Effect of delay spike on SCTP, TCP Reno, and Eifel in a wireless mobile environment," *International Conference on Computer Communications and Networks*, Miami, FL, pp. 575–578, October 2002.
- [12] S. Fu and M. Atiquzzaman, "Improving end-to-end throughput of Mobile IP using SCTP," *Workshop on High Performance Switching and Routing*, Torino, Italy, pp. 171–176, June 2003.
- [13] J. Noonan, P. Perry, and J. Murphy, "A study of SCTP services in a Mobile IP network," *IT&T Annual Conference*, WIT, Ireland, October 2002.
- [14] G. Ye, T. Saadawi, and M. Lee, "SCTP congestion control performance in wireless multi-hop networks," *MILCOM2002*, Anaheim, California, pp. 934–939, October 2002.
- [15] R. Stewart, *SCTP reference implementation page*. <http://www.sctp.org/index.html>.
- [16] *NS-2 SCTP Module Home Page*. <http://pel.cis.udel.edu>.
- [17] *SCTP for the Linux Kernel (LKSCTP)*. lksctp.sourceforge.net/index.html.
- [18] *Cisco products*. <http://www.cisco.com/en/us/products/index.html>.
- [19] *Adax homepage*. <http://www.adax.com/aps-sctp.html>.
- [20] *Ulticom Nexgen*. www.ulticom.com/html/english/products/nexgen.asp.
- [21] *DC-SCTP*. www.dataconnection.com/products/PRODSidx.htm.
- [22] International Telecom. Union, "Signalling system NO.7–message transfer part signalling performance." ITU-T Recommendation Q.706, March 1993.
- [23] R. Stewart, M. Ramalho, and Q. Xie et. al., "Stream control transmission protocol (sctp) dynamic address reconfiguration." www.ietf.org/internet-drafts/draft-ietf-tsvwg-addip-sctp-06.txt, September 2002.