Complete reliability, Chunk authentication, and Dynamic address reconfiguration feature implementation in SCTP

Heena Gupta, M.Tech. Student, Department of Computer Science and Applications
Neeraj Maglani, Assistant Professor, Department of Computer Science and Applications
Jagannath University
Jaipur, INDIA

E-mail: heenamitrc16@gmail.com

Abstract: The core SCTP protocol, including the extensions for partial reliability, partial chunk authentication, and dynamic address reconfiguration. The Stream Control Transmission Protocol (SCTP) is a general-purpose transport layer protocol providing a service similar to TCP — plus a set of advanced features to utilize the enhanced capabilities of modern IP networks and to support increased application requirements. While SCTP was standardized as an RFC several years ago, there is still significant ongoing work within the IETF to discuss and standardize further features in the form of protocol extensions. In this paper, we first introduce the SCTP base protocol and already standardized extensions. After that, we focus on the ongoing SCTP standardization progress in the IETF and give an overview of activities and challenges in the areas of security and concurrent multipath transport. In this we are going to implement the complete/full reliability of messages and also the chunk authentication and dynamic address configuration will also be improved.

Index Word SCTP, complete reliability, chunk authentication, Dynamic address reconfiguration.

I. INTRODUCTION

SCTP is a connection-oriented general-purpose transport protocol that preserves message boundaries. An SCTP connection, called an SCTP association, can be used on top of IPv4 and IPv6. One of the main design goals was the efficient transport of small messages in a network fault-tolerant way, important for transporting signaling messages. Another feature is TCP fairness. SCTP should behave in a fair way when it competes with TCP traffic. This is important for deploying SCTP in networks with TCP-based traffic. Therefore, SCTP adopts the congestion and flow control from TCP, which is described in more detail later. Chunk has a variable length and consists of a type, some flags, a length field, the actual value, and possibly padding, which ensures that the total length in bytes of a chunk is a multiple of four. Since this generic format is used by all chunks, a receiver can parse a received packet even if it does not support some of the received chunk types. Since how to handle unknown chunks is also defined, the packet format is extensible. User messages are put into DATA chunks, and the other chunks, so-called control chunks, are used for SCTP control information.

Small user messages are put into their own DATA-chunk, and multiple DATA-chunks are put into one packet. This so-called bundling of user messages allows the sending of multiple messages within one SCTP packet.

II. FEATURES OF SCTP

Multi-homing

Multi-homing can be achieved with multiple network interface cards on a single host to improve reliability and fault tolerance on a network. When a SCTP association is established between two multi-homed hosts it automatically selects one destination address as primary and others as secondary or backup. If the connection with primary destination address is failed, SCTP automatically switch to other available paths (backup destination address) and when the primary destination address is operational again SCTP switches back to primary destination address. Multi-homing can provide redundancy, high availability, fault-tolerance, and load balancing in any SCTP enabled network.

The most common Multi-homing setup is like as shown in the fig1. In a client/server environment to maintain high availability and to improve throughput, servers are equipped with multiple interfaces (in most cases) and clients with single interface. We can call it as asymmetric Multi-homing. There is another type of Multi-homing setup is possible as shown fig 2. Here in this cases both server and client is configured with multiple IP address, we can call this as symmetric Multi-homing. From the fig4 host A can be reached through IP1, IP2 and let us assume host B selected IP2 as primary address. Host B is configured with IP3, IP4 and Host A selected IP3 as primary address to reach Host B. Host A is unaware of the Primary address selected by Host B and same with Host B.

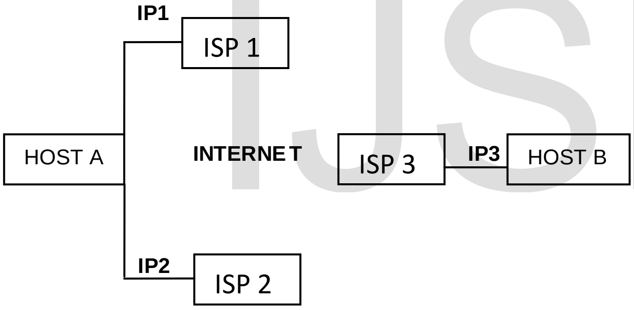

Figure1: Asymmetric Multi-Homing

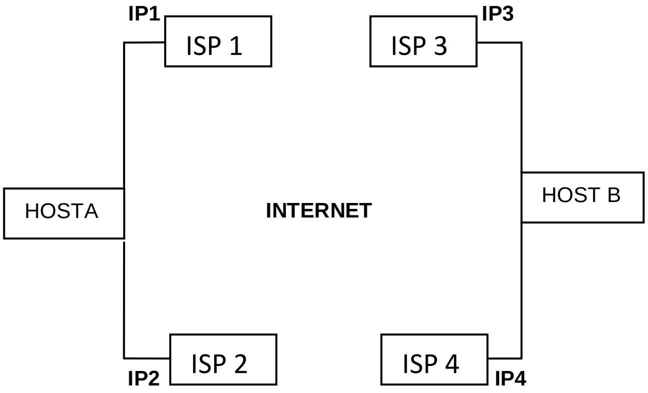

Figure2: Symmetric Multi-Homing

Multi-streaming

Data transmission between two SCTP end points is carried out using multiple logical streams within a single association as shown in Fig 3. The messages that are carried in different streams are independent of each other. Each logical stream is assigned with unique stream number and messages are delivered in sequential order to the application layer with respect to the corresponding stream only. So loss of messages in one stream does not affect the transmission of messages to the application in other streams except only when unordered delivery method is used. Multi-streaming solves the problem of HOL (Head of Line) by segmenting data stream into number of logical streams. It gives the flexibility to transfer messages of different applications on different streams.

When un-ordered delivery method is used messages are delivered to the upper-layer protocol as soon as they arrived at the receiver.

According to previous research on multi-streaming the possibility of higher throughput, minimum buffer requirements at the receiver also observed. The quality of multimedia applications can be improved to a greater extent by making use of SCTP Multi-streaming technique. SCTP allows an association to have multiple inputs or output streams up to a maximum of 65536.

II. THE PROBLEM TO BE SOLVED BY THE PROPOSAL FOR SCTP

Two protocol features were omitted in the base protocol specification of SCTP to ensure that the RFC was published on time: partial reliability and the reconfiguration of IP addresses being during the lifetime of the association. To overcome this limitation, a protocol extension called Dynamic Address Reconfiguration was specified in RFC 5061 [3]. This extension allows an SCTP endpoint to add or delete a local address and notify its peer to use a specific local address as the primary path. Thus, it is possible to reconfigure a host by adding or removing network interface cards or changing the IP configuration without disturbing existing SCTP associations. This is important for providing long living SCTP associations and was the original motivation to add this feature.

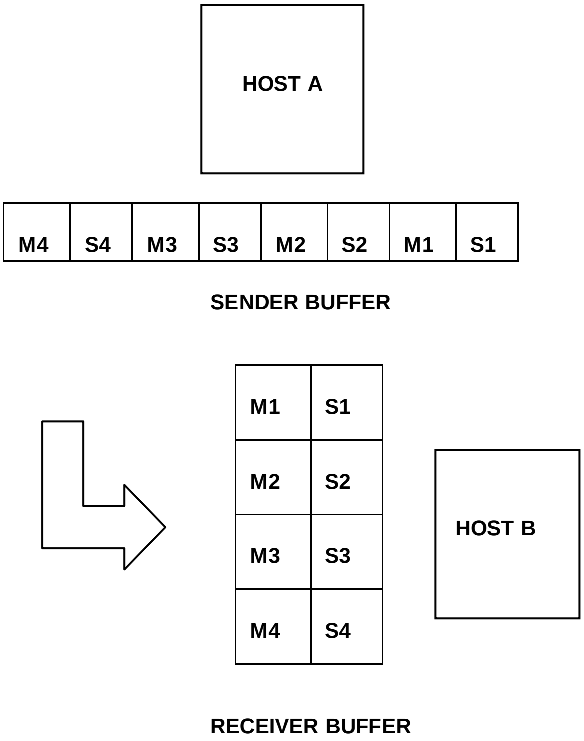

Figure3: SCTP Multi-Streaming

It should be noted that there are two other important usage scenarios for dynamic address reconfiguration. One is the possibility of supporting mobility at the transport layer. Transport layer mobility allows a transport connection to continue even when some IP-addresses of a host change.

The other useful application is setting up an SCTP association as single homed and then adding each additional local address one by one using dynamic address reconfiguration. Hence, it is possible to avoid IP-addresses being part of the SCTP packet, a feature making an SCTP-aware Network Address Translation (NAT) traversal possible [5][6]. One of the requirements for this extension was not to introduce any additional security issues to SCTP associations. To ensure this, another extension was necessary, which is introduced in the following subsection.

As mentioned earlier, SCTP has adopted the mechanisms for congestion and flow control from TCP. Yet again, the message orientation of SCTP has significant impact on the implementation of the algorithms. In the case of congestion control, the amount of data that may be sent is calculated as the difference between the congestion window and the number of outstanding bytes (i.e., the data that has been sent, but not yet acknowledged). The way these outstanding bytes are counted (i.e., whether the DATA-chunk header is included or not) is not specified in RFC 4960 [1]. In [4], we analyzed the impact of these options on the fairness toward TCP. Simulation results showed that the DATA-chunk headers definitely have to be considered to avoid unfairness toward TCP.

Comparing the flow control behaviour of various implementations showed that the sending of small messages might exhaust the receiver window before the advertised receiver window in the SACK-chunk is reduced to zero, which results in spurious retransmissions. This is caused by the storing of additional information for each incoming DATA-chunk, which is not announced.

III. METHODOLOGY

The Partial Reliability extension (PR-SCTP) is defined in RFC 3758[2] and enables the SCTP sender of a message to decide that a user message is not transmitted or retransmitted any more. Such a message is called abandoned. A specific method of abandoning messages is called a PR-SCTP policy. An important feature of PR-SCTP is that the receiver is policy agnostic. Only the sender implements a particular policy. This allows an implementer to deploy new policies easily, since only sender side modifications are necessary. During association setup, each SCTP endpoint provides a list of its addresses in the INIT- and INIT-ACK-chunk to the peer. For security reasons each endpoint first has to verify that the remote addresses really belong to the peer. This path verification uses so-called HEARTBEAT and HEARTBEAT-ACK-chunks. These chunks are also used to monitor remote addresses when no user traffic is sent to them to check their reachability. The base protocol uses the multiple remote addresses only for redundancy. This means that it typically sends all user messages to one remote address, the so-called primary address (Fig. 4). If messages have to be retransmitted due to timeouts, other remote addresses are used for the retransmission. After a number of consecutive message losses for a particular remote address, this address is considered unreachable and is not used for user message transport any more until it is reachable again.

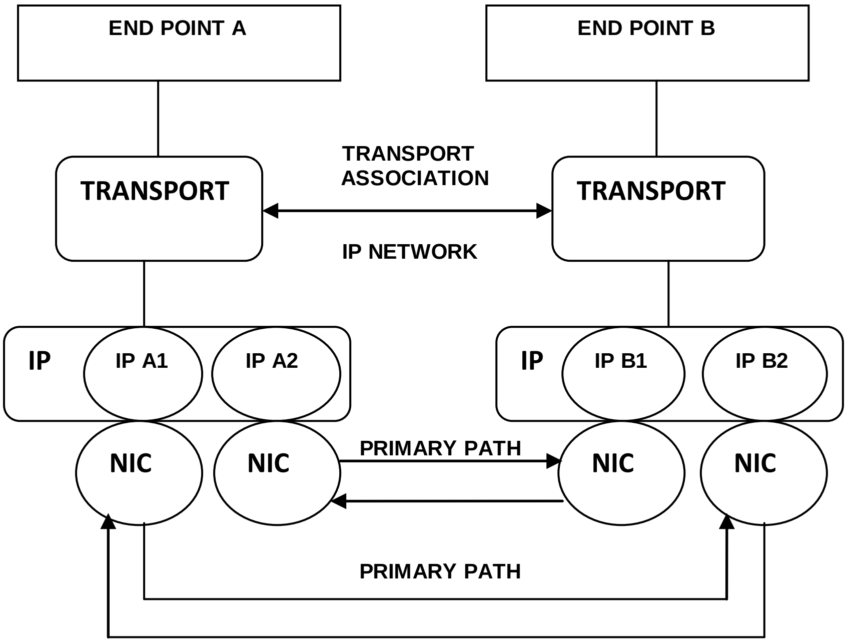

Figure 4: The principle of multihoming

REFERENCES

[1] 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, ” IEEE Communications Magazine • April 2011

[2] R. Stewart, “Stream Control Transmission Protocol,” IETF RFC 4960, Sept. 2007.

[3] R. Stewart et al., “Stream Control Transmission Protocol (SCTP) Partial Reliability Extension,” IETF RFC 3758, May 2004.

[4] I. Rüngeler, SCTP — Evaluating, Improving and Extending the Protocol for Broader Deployment, Dissertation, Univ. of Duisburg-Essen, Faculty of Econ., Inst. For Comp. Sci. and Business Info. Sys., Dec. 2009.

[5] M. Tüxen, R. Seggelmann, and E. Rescorla, “Datagram Transport Layer Security for Stream Control Transmission Protocol,” IETF RFC 6083, Jan. 2011.

[6] M. Tüxen et al., “Network Address Translation for the Stream Control Transmission Protocol,” IEEE Network, vol. 22, no 5, 2008, pp. 26–32.