An Experimental Study of SCTP in NS2

Md. Ibrahim Chowdhury (Corresponding author)

Dept. of Computer Science and Engineering, City University

Bangladesh

E-mail: ibrahimiiuc@gmail.com

Imran Hossain Jony

Dept. of Electrical and Electronics Engineering, City University

Bangladesh

E-mail: jony263@yahoo.com

Abstract

SCTP (Stream Control Transport Protocol) is a reliable transport layer protocol which provides acknowledged, error-free and non-duplicated transfer of message. It combines the advantages of TCP and UDP. In additional to that, SCTP has many attractive characteristics such as multi-homing, multi-streaming and partial data reliability. This paper thoroughly discusses about the advantages of SCTP. We have experimented the SCTP endpoints data transmission by using NS2 (Network Simulator 2) tool including multi-homed SCTP nodes data transmission. The result shows that an SCTP node in NS2 cannot be truly multi-homed. Hence, NS2 still lacks of full implementation of SCTP features.

Keywords: SCTP, NS2.

1. Introduction

Nowadays fast and reliable exchange of control information works as a backbone of telecommunication networks. One of the key issues in mobility management in wireless networks is signalling (exchange of control information) between diverse network entities [1]. Traditional Signalling System Number 7 (SS7) [2] has the greatest influence in bearing the signalling traffic for telecommunication networks. But the problem lies in SS7 signalling network is that it requires a very expensive dedicated network infrastructure. The standard-based solution of this problem is the Stream Control Transmission Protocol (SCTP) [3]. In this paper, we have studied the main features of SCTP like "SCTP Association", "Multi-homing", "Multi-streaming" etc., and experimented SCTP's common features by using the Network Simulator (NS2) tool.

1.1 Stream Control Transmission Protocol (SCTP)

Stream Control Transmission Protocol (SCTP) is a Transport Layer protocol, was introduced by IETF workgroup SIGTRAN in September 2007 (RFC 4960). It provides almost all service features of Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) [4]. And it also facilitates some more advantages like multi-homing, multi-streaming, which makes it exceptionally good for handling with challenging problems in transport layer whereas TCP and UDP could not fulfil the current need of mobility models.

SCTP is reliable as any data that is transferred must be acknowledged. It must retransmit if the data is not acknowledged [5]. It also supports unreliable data transmission as UDP. A comparison between SCTP, TCP and UDP is presented in Table 1.

Table 1. Comparisons of SCTP features with TCP and UDP [6]

Protocol Feature	SCTP	TCP	UDP
Reliable data transfer	Yes	Yes	No
Partial reliable data transfer	Yes	No	No
Connection oriented delivery	Yes	Yes	No
Congestion control and avoidance	Yes	Yes	No
Path MTU discovery and message fragmentation	Yes	Yes	No
Message bundling	Yes	Yes	No
Multi-homing	Yes	No	No
Multi-streaming	Yes	No	No
Ordered data deliver	Yes	Yes	No
Unordered data delivery	Yes	No	Yes
Path reachability check	Yes	No	No

1.1.1 SCTP Association

It is a protocol relationship between two SCTP endpoints and state information including Verification Tags and current active set of Transmission Sequence Numbers (TSNs) etc. The

end points of an association can be uniquely identified by the transport addresses used by in the association. SCTP closes of an active association on request from the SCTP user. SCTP also allows ungraceful termination on request from the user [7].

A peer endpoint in a transmitting endpoint considers available for receiving user messages is an active destination transport address. A SCTP packet contains a chunk header which consists of data chunks and control chunks to places messages and control information. SCTP congestion window (CWND) is a number of limited data in bytes of a SCTP variable that a sender can send to a particular destination transport address before receiving an acknowledgement [7] [8].
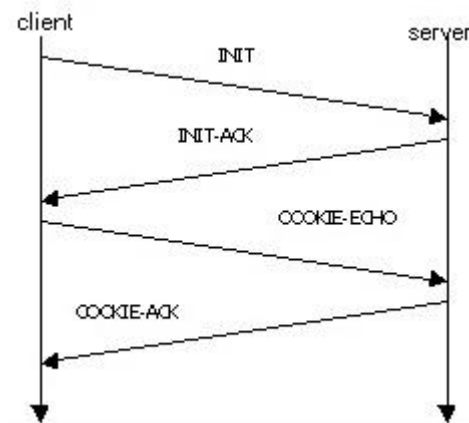

Figure 1. SCTP Association

A SCTP Association establishment start by a client sends an INIT chunk to the server. Then the client enters the COOKIE-WAIT state after client starts the T1-init timer. The server responds with an INIT-ACK chunk. This INIT-ACK chunk contains a state cookie. It comprises a Message Authentication Code (MAC), along with a cookie creation time stamp, the state cookie lifetime, and other information of association establishment. The MAC is provided by the server which is a secret key only known to it. After receiving the INIT-ACK signal, the client sends a COOKIE-ECHO response by echoes the state cookie. The server then verifies the state cookie using the secret key. Then it allocates the resources for the association by sending a COOKIE-ACK response acknowledging the COOKIE-ECHO signal, and moves the association to ESTABLISHED state [9]. With the use of a cookie, SCTP defends against denial of service attacks. The cookie is bundled with the INIT-ACK from the server to the client. The server does not record the association or keep a transmission control block (TCB), rather it derives the TCB from the cookie, which is sent back from the client inside the COOKIE-ECHO. Since it has no knowledge of the association till the client responds with a COOKIE-ECHO, it becomes resilient to denial of service attacks [9].

1.1.2 Multi-homing

Standard SCTP introduces a new feature called multi-homing. Multi-homing allows the use of multiple source-destinations IP addresses for a single association between two SCTP endpoints.

To support multi-homing, the endpoints of a transport protocol must have more than one transport layer addresses. These transport layer addresses are the different paths of the peer towards the endpoint with the multiple transport addresses [10]. Multi-homed nodes can be simultaneously connected through multiple end-to-end paths to increase resilience to path failure. For instance, users might be simultaneously connected through dial-up/broadband or via multiple wireless technologies such as 802.11b and GPRS [11].
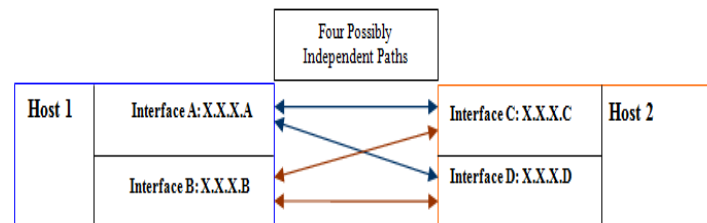

Figure 2. Network level fault tolerance of Multi-homing.

Figure 2 shows that in case of a failure on the path to one of the IP addresses, the SCTP sender may use another of the receiver's addresses. In RFC 2960 [12], SCTP multi-homing support appends the following essential functions to the protocol:

1. One path is considered primary within an association i.e., one of the IP addresses assigned to the receiver endpoint of the association is selected as the primary address. An endpoint should always transmit on the primary path to its peer endpoint if it is not specified otherwise.

2. When acknowledging arriving chunks, the respond chunks (acknowledgements) should be transmitted through the same path that was used by the received chunks. If the receiver acknowledges several data chunks received in packets from different source addresses at once, the acknowledgement chunk may be sent to any of these source addresses.

3. When retransmitting a data chunk to a multi-homed peer, the receiver should choose another destination address than the one to which the original data chunk was sent. If a chunk is not acknowledged and has to be retransmitted, there might be a network failure. In addition, when an endpoint receives a duplicate data chunk, the endpoint may vary the destination address and not use the source address of this duplicate data chunk to send an acknowledgement.

4. The heartbeat mechanism helps the implementation of multi-homing in SCTP. It detects failures in idle paths (path that are not being used) and endpoints and therefore can detect if a destination address is active or passive. Heartbeat chunks are sent periodically to all idle destinations, and a counter is maintained on the number of Heartbeats sent to an inactive destination without receipt of a corresponding Heartbeat Acknowledgement. When this counter exceeds a configured maximum, that destination address is declared inactive. So when the need is to use another IP address than the primary address to reach a multi-homed endpoint, SCTP knows which other addresses are active or not and thus, can avoid using another path that has a failure.

1.1.3 Multi-streaming

SCTP facilitates the multi-streaming functionality which allows independent delivery among data streams. The application data can be partitioned into multiple streams. These portions or data chunks are formed inside an SCTP packet and each packet can contain multiple data chunks from different applications. The chunks header contains Transmission Sequence Number (TSN), Stream ID and Stream Sequence Number (SSN) that can offer independent delivery of each stream to the application [13].

Multi-streaming is a unidirectional dataflow of multiple streams of an SCTP association. Order of data is preserved as intra stream such that each stream has its own sequencing space. Incoming data from the upper layer application can be multiplexed onto an association in SCTP. When a segment of a certain stream is lost, following segments of the same stream will be stored in the receiver's stream buffer until the source retransmits the lost segment. Yet, other streams can still be passed to the upper-layer application. Multi-streaming elude the Head-of-line blocking (HOL) exists in TCP (Transmission Control Protocol), where a single stream loss can cause subsequent streams to also be delayed. HOL effect does not affect the SCTP association due to individual streams [11].
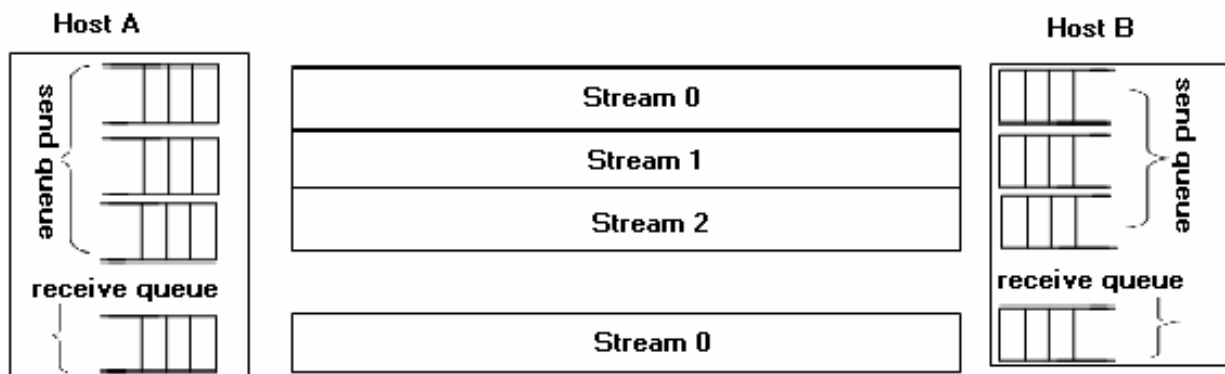

Figure 3. Multi-streaming Scenario.

Figure 3 describes a multi-streamed association between hosts A and B. In the example, host A requested three streams to host B (numbered 0 to 2), and host B requested only one stream to host A (numbered 0).

1.1.4 SCTP Packet Structure

The unit of data delivery across the interface between SCTP and the connectionless packet network (e.g. IP). An SCTP packet includes the common SCTP header, possible SCTP control chunks, and user data encapsulated within SCTP DATA chunks. [4]. Common header inhabits first 12 bytes of SCTP packet which consists of Source port, Destination port, Verification tag and Checksum. The remaining part of the SCTP packet occupies Chunk type, Chunk flags, Chunk length and Chunk value [14]. An SCTP packet is composed of a common header and chunks. A chunk contains either control information or user data. Figure 4 shows SCTP packet format.

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                         Common Header                         |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Chunk #1                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             ...                               |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                           Chunk #n                            |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```
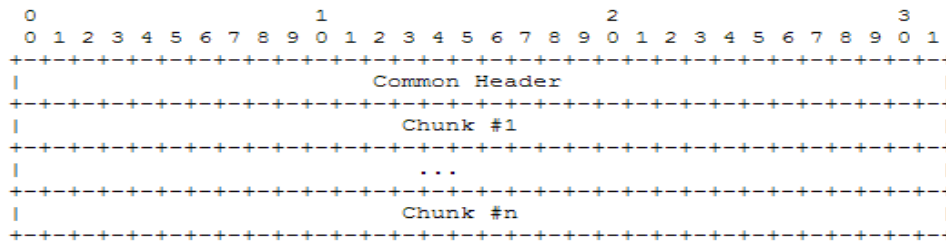
Figure 4. SCTP packet format.

Multiple chunks can be bundled into one SCTP packet up to the MTU size, except for the INIT, INIT ACK, and SHUTDOWN COMPLETE chunks. These chunks must not be bundled with any other chunk in a packet.

2. Simulation with NS2

This section describes simulation components and their structures to build a script in NS2. These NS2 components included within are nodes, links, Simple Link objects, packets, agents, and applications.

2.1 Structure of Wired and Mobile nodes

NS2 associates two kinds of nodes for wired and wireless environment. The wired nodes consist of an entry point and two classifiers named Address Classifier and Port Classifier. The configuration of a mobile node is same like wired nodes. In a mobile node, packets arrive through the entry point to the address classifiers to check the address. The packets are then flow through the port classifiers to the routing agents. Routing agent processed the packet via some layer functionality. Thus the mobile node can be treated as mobile because of the routing type defined in the Tcl script [15]. An agent of a mobile node is responsible for packet generations and receptions. It is also responsible for maintaining the traffics like CBR (Constant Bit Rate), FTP (File Transfer Protocol) etc. Routing agents (DSDV, DSR, and AODV) are configured multi hop routes for packets [15]. Simulation of wireless networking in NS2 has various modules; such as mobile node, Ad-hoc Routing (DSR, DSDV, AODV), MAC 802.11, Radio Propagation Model and Channel [15].

2.2 Trace File

The trace file format is used to trace files produced by the Trace class. Figure 5 illustrates nine (9) trace entries of a SCTP simulation, from them, the event column has three enqueue operations ("+"), two dequeue operations ("-"), three receive events ("r"), and one drop event. In the second column, the simulation time is in seconds at which each event occurred. The next two columns indicate trace happenings of two nodes. In the next two fields, the type of a packet and their size is displayed. SCTP packet trace contains five (5) extra column fields than a TCP trace. Among these, the chunk type indicates some alphabetic flags like I, D, S, H, B. The flag I indicate an association initiation control chunk like INIT, INIT-ACK, COOKIE-ECHO, and COOKIE-ACK. The D, S flags indicate a DATA and a SACK chunk.

The rest chunks H, and B are indicates a HEARTBEAT chunk and HEARTBEAT-ACK chunk successively [15].

Event	Simulation time	From node	To node	Packet type	Packet size	Flag	Chunk Type	Flow ID	Source ID	Destination ID	TSN		stream ID	SSN	Packet ID
+	0.5	1	4	sctp	56	-----	I	0	1.0	4.0	1	-1	4	65535	65535
-	0.5	1	4	sctp	56	-----	I	0	1.0	4.0	1	-1	4	65535	65535
r	0.700896	1	4	sctp	56	-----	I	0	1.0	4.0	1	-1	4	65535	65535
+	0.700896	4	1	sctp	56	-----	I	0	4.0	1.0	1	-1	5	65535	65535
....	....	....	....	....	....	....	....	....	....	....	....	....	....	....	....
r	1.550304	1	4	sctp	1500	-----	D	0	1.0	4.0	1	2	9	0	1
+	1.550304	4	1	sctp	1500	-----	S	0	4.0	1.0	1	2	11	65535	65535
....	....	....	....	....	....	....	....	....	....	....	...	....	....	....	....
r	19.303264	4	1	sctp	56	-----	H	0	4.0	1.0	1	-1	232	65535	65535
+	19.303264	1	4	sctp	56	-----	B	0	1.0	4.0	1	-1	337	65535	65535

Figure 5. Trace File format.

2.3 Results and Analysis

The SCTP module for NS-2 currently supports the features in the following sections of RFC4460 [4].

- Normal Establishment of an Association (rudimentary handshake)

- Path Verification

- Transmission of DATA Chunks

- Acknowledgement of Reception of DATA Chunks

- Management Retransmission Timer

- Multi-homed SCTP Endpoints

- Stream Identifier and Stream Sequence Number

- Ordered and Unordered Delivery

- Report Gaps in Received DATA TSNs

- SCTP Slow-Start and Congestion Avoidance

- Endpoint Failure Detection

- Path Failure Detection

- Path Heartbeat (without upper layer control)

The unreliable data mode extension (U-SCTP) is also supported [4]. The SCTP module has NS-2 upper layer API support but it still lacks many features. It supports all legacy NS-2

applications, but they obviously aren't able to utilize all SCTP's features. NS2 supports multi-homing features of SCTP which motivated us to choose this as a simulation tool.

According to NS2 architecture, a SCTP enabled node cannot be multi-homed. A multi-homed node actually be the combination of three nodes such that a ''core node'' and a couple of ''interface nodes'' to simulate the interfaces. All these nodes are SCTP enabled.
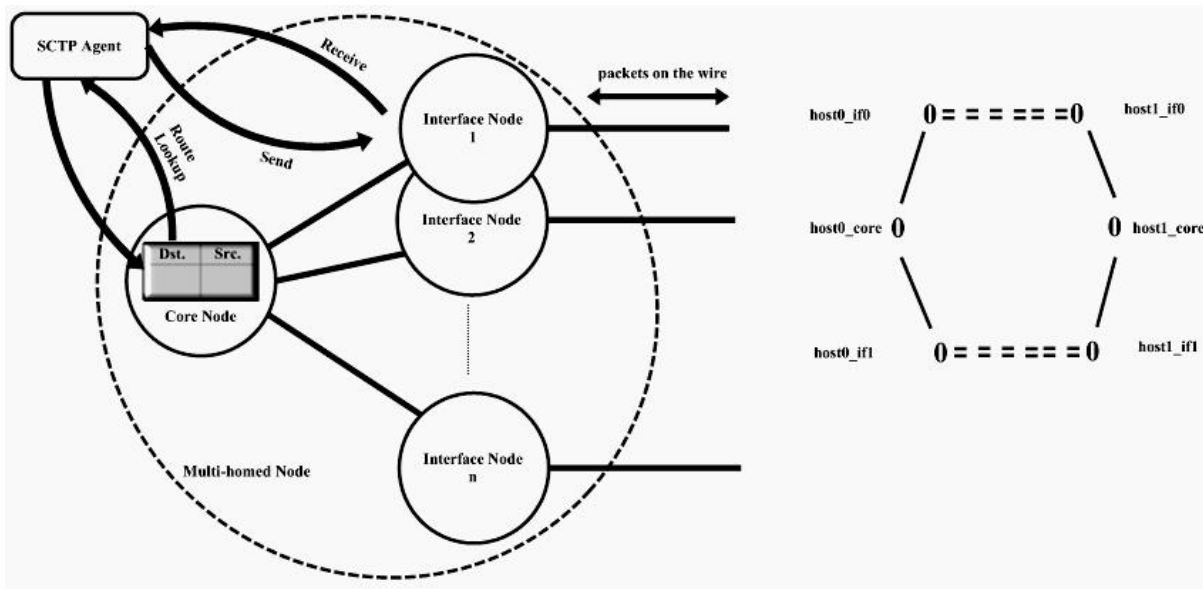

Figure 6. (a) Multi-homed node of SCTP, (b) Visualized Multi-homed structure.

But, traffic flows only through the interfaces nodes. The core node has connection with these interface nodes and is used only for routing. The connection of core node with the interfaces is a unidirectional link. No traffic flows through the core node. Figure 6 demonstrates and visualizes the multi-homed architecture different features of multi-homing, where two SCTP endpoints, each with 2 interfaces, are directly connected between each pair of interfaces. In the middle of the association, a change primary is done [15]. The following section helps to understand data transmission on SCTP nodes.

2.3.1 NS2 Simulation of SCTP endpoints

A reflective process model has been used by The SCTP agent establishes an association using a four-way handshake, but the handshake is kept simple and does not strictly conform to RFC2960. The handshake does not exchange tags, and the INIT and COOKIE-ECHO chunks are not used to update the RTT. Instead, RTT estimation begins with the first DATA chunk. Currently, the SCTP agent does not perform a proper shutdown. The association is abruptly terminated when the simulated connection ends. A shutdown procedure may be added in a future release.

According to the visualized structure of SCTP, we know that an SCTP has two interface nodes: if0 and if1. In Figure 7, we can see the trace dynamics of SCTP interface (if0). Here from the beginning, an association initiation (by an association initial control chunk I) of SCTP packet containing packet size 56 arrives at 0.5s. The first receive event occurs at

approximately 0.7s. Since, an association initiation (INIT) is a control chunk, in TSN, Stream ID or and SSN fields of the trace lines of these chunks are numbered as undefined (-1 or 65535). While ns-2 starts numbering packets at 0, the SCTP module starts numbering DATA chunk TSNs at 1 and assigns undefined TSN values to non-DATA chunks (i.e., INIT, SACK, HEARTBEAT, etc.). In addition, the 4 packets exchanged during the association initialization are not included in the traces and do not show up in the graphs, but they are still counted in the packet enumeration. This information is important when doing things like specifying a drop list for the ErrorModel object. For example, if a List is specified for the ErrorModel object, packet 2 actually refers to the first SCTP packet with DATA chunk(s) [16].

```
+ 0.5 1 4 sctp 56 -------I 0 1.0 4.0 1 -1 4 65535 65535
- 0.5 1 4 sctp 56 -------I 0 1.0 4.0 1 -1 4 65535 65535
r 0.700896 1 4 sctp 56 -------I 0 1.0 4.0 1 -1 4 65535 65535
+ 0.901792 1 4 sctp 36 -------I 0 1.0 4.0 1 -1 6 65535 65535
- 0.901792 1 4 sctp 36 -------I 0 1.0 4.0 1 -1 6 65535 65535
r 1.102368 1 4 sctp 36 -------I 0 1.0 4.0 1 -1 6 65535 65535
+ 1.302944 1 4 sctp 1500 -------D 0 1.0 4.0 1 1 8 0 0
- 1.302944 1 4 sctp 1500 -------D 0 1.0 4.0 1 1 8 0 0
+ 1.302944 1 4 sctp 1500 -------D 0 1.0 4.0 1 2 9 0 1
- 1.326944 1 4 sctp 1500 -------D 0 1.0 4.0 1 2 9 0 1
r 1.526944 1 4 sctp 1500 -------D 0 1.0 4.0 1 1 8 0 0
r 1.550944 1 4 sctp 1500 -------D 0 1.0 4.0 1 2 9 0 1
+ 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 3 11 0 2
- 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 3 11 0 2
+ 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 4 12 0 3
+ 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 5 13 0 4
- 1.775712 1 4 sctp 1500 -------D 0 1.0 4.0 1 4 12 0 3
- 1.799712 1 4 sctp 1500 -------D 0 1.0 4.0 1 5 13 0 4
r 1.975712 1 4 sctp 1500 -------D 0 1.0 4.0 1 3 11 0 2
r 1.999712 1 4 sctp 1500 -------D 0 1.0 4.0 1 4 12 0 3
r 2.023712 1 4 sctp 1500 -------D 0 1.0 4.0 1 5 13 0 4
+ 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 6 15 0 5
- 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 6 15 0 5
+ 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 7 16 0 6
+ 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 8 17 0 7
- 2.22448 1 4 sctp 1500 -------D 0 1.0 4.0 1 7 16 0 6
- 2.24848 1 4 sctp 1500 -------D 0 1.0 4.0 1 8 17 0 7
r 2.42448 1 4 sctp 1500 -------D 0 1.0 4.0 1 6 15 0 5
+ 2.42448 1 4 sctp 1500 -------D 0 1.0 4.0 1 9 19 0 8
- 2.42448 1 4 sctp 1500 -------D 0 1.0 4.0 1 9 19 0 8
+ 2.42448 1 4 sctp 1500 -------D 0 1.0 4.0 1 10 20 0 9
r 2.44848 1 4 sctp 1500 -------D 0 1.0 4.0 1 7 16 0 6
- 2.44848 1 4 sctp 1500 -------D 0 1.0 4.0 1 10 20 0 9
r 2.47248 1 4 sctp 1500 -------D 0 1.0 4.0 1 8 17 0 7
r 2.64848 1 4 sctp 1500 -------D 0 1.0 4.0 1 9 19 0 8
+ 2.649248 1 4 sctp 1500 -------D 0 1.0 4.0 1 11 23 0 10
```
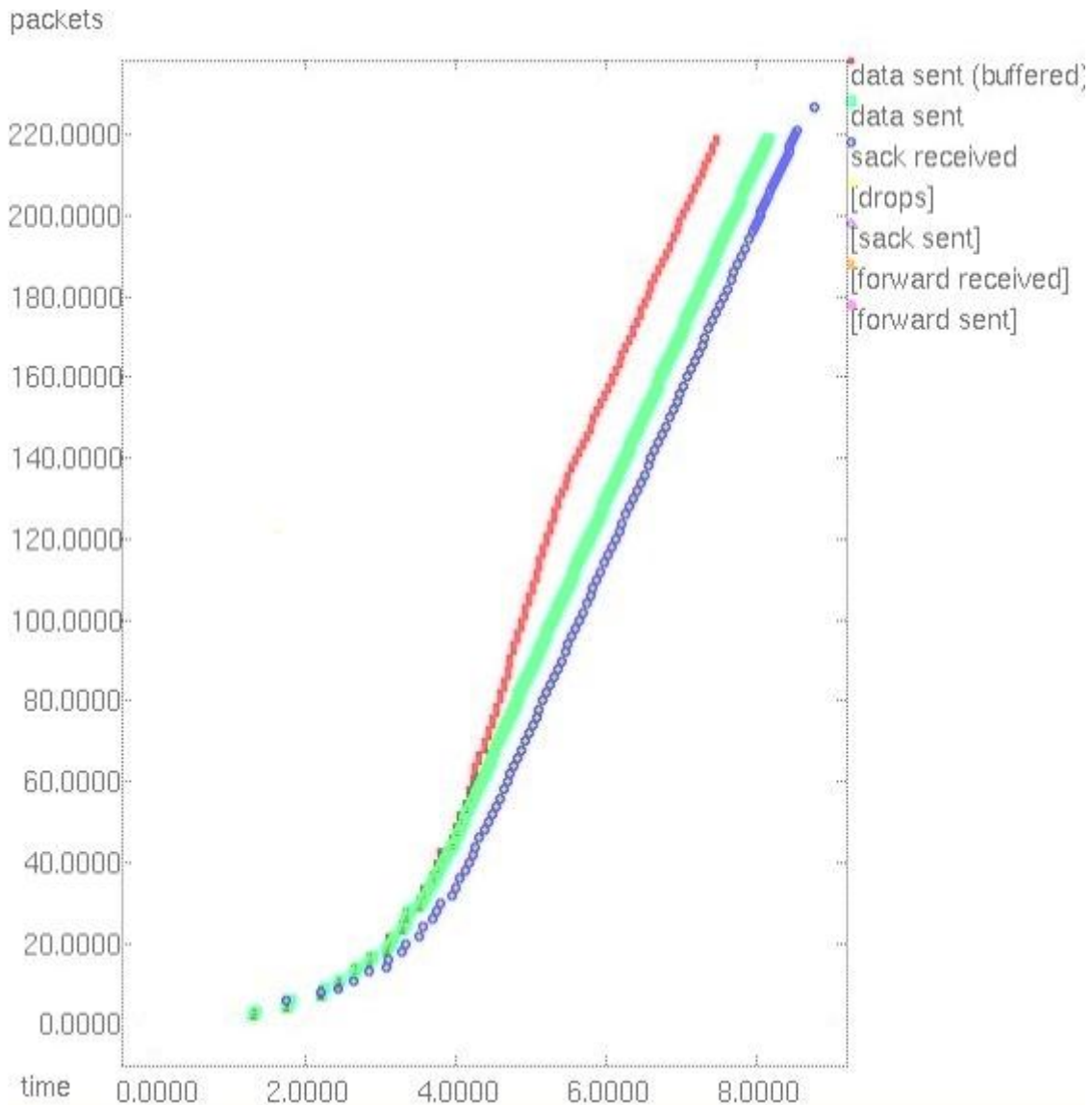
Figure 7.Trace file sample SCTP interface (if0).

Figure 8. XGRAPH view of SCTP node data transmission of if0.

In Figure 8, (the XGRAPH view of SCTP node data transmission) it is seen that with respect to time the SCTP packets' data sent, data sent (buffered) and sack received are exponentially increasing. Rarely any packet drop seen during the simulation. We can observe a pretty fast data transmission has occurred in SCTP. In this graph, SCTP packets limit is shown till 220.000 and time limit is approximately 8.0000.

In Figure 9, we can see that the trace dynamics of SCTP interface (if1). Here from the beginning, an association initiation (by an association initial control chunk I) of SCTP packet containing packet size 1500 arrives at 7.5s. The first receive event occurs at approximately 7.7s. Also, we observe that all the chunks are data chunks and traffic flows through source ID 2.0 and destination ID 5.0.

We can observe in Figure 10 that, there are less data packets are sent through SCTP node of if1 compared to if0. The reason, if0 used to transmit most of the traffic in the simulation. Here, we can observe that, for if1 SCTP packets limit is shown till 260.0000 and time limit is 10.0000. Only packets sent and received after 220.000 are observed because in the earlier graph (Figure 9), we noticed that most of the SCTP packets are sent through if0.

2.3.2 Multi-homing simulation on NS2

An SCTP agent simultaneously resides on all these nodes (i.e., the core and interface nodes), but actual traffic only goes to/from the interface nodes. Whenever the SCTP agent needs to send data to a destination and does not know which outgoing interface to use, the agent initially consults with the core node for a route lookup.

```
+ 7.515552 2 5 sctp 1500 -------D 0 2.0 5.0 1 219 318 0 218
- 7.515552 2 5 sctp 1500 -------D 0 2.0 5.0 1 219 318 0 218
+ 7.515552 2 5 sctp 1500 -------D 0 2.0 5.0 1 220 319 0 219
- 7.539552 2 5 sctp 1500 -------D 0 2.0 5.0 1 220 319 0 219
r 7.739552 2 5 sctp 1500 -------D 0 2.0 5.0 1 219 318 0 218
r 7.763552 2 5 sctp 1500 -------D 0 2.0 5.0 1 220 319 0 219
+ 7.940384 2 5 sctp 1500 -------D 0 2.0 5.0 1 221 335 0 220
- 7.940384 2 5 sctp 1500 -------D 0 2.0 5.0 1 221 335 0 220
+ 7.940384 2 5 sctp 1500 -------D 0 2.0 5.0 1 222 336 0 221
+ 7.964384 2 5 sctp 1500 -------D 0 2.0 5.0 1 223 338 0 222
- 7.964384 2 5 sctp 1500 -------D 0 2.0 5.0 1 222 336 0 221
- 7.988384 2 5 sctp 1500 -------D 0 2.0 5.0 1 223 338 0 222
r 8.164384 2 5 sctp 1500 -------D 0 2.0 5.0 1 221 335 0 220
r 8.188384 2 5 sctp 1500 -------D 0 2.0 5.0 1 222 336 0 221
r 8.212384 2 5 sctp 1500 -------D 0 2.0 5.0 1 223 338 0 222
+ 8.365216 2 5 sctp 1500 -------D 0 2.0 5.0 1 224 358 0 223
- 8.365216 2 5 sctp 1500 -------D 0 2.0 5.0 1 224 358 0 223
+ 8.389216 2 5 sctp 1500 -------D 0 2.0 5.0 1 211 359 0 210
- 8.389216 2 5 sctp 1500 -------D 0 2.0 5.0 1 211 359 0 210
+ 8.403616 2 5 sctp 1500 -------D 0 2.0 5.0 1 212 360 0 211
+ 8.413216 2 5 sctp 1500 -------D 0 2.0 5.0 1 213 361 0 212
- 8.413216 2 5 sctp 1500 -------D 0 2.0 5.0 1 212 360 0 211
+ 8.427616 2 5 sctp 1500 -------D 0 2.0 5.0 1 214 362 0 213
- 8.437216 2 5 sctp 1500 -------D 0 2.0 5.0 1 213 361 0 212
+ 8.451616 2 5 sctp 1500 -------D 0 2.0 5.0 1 215 363 0 214
- 8.461216 2 5 sctp 1500 -------D 0 2.0 5.0 1 214 362 0 213
+ 8.475616 2 5 sctp 1500 -------D 0 2.0 5.0 1 216 364 0 215
- 8.485216 2 5 sctp 1500 -------D 0 2.0 5.0 1 215 363 0 214
+ 8.499616 2 5 sctp 1500 -------D 0 2.0 5.0 1 217 365 0 216
- 8.509216 2 5 sctp 1500 -------D 0 2.0 5.0 1 216 364 0 215
+ 8.523616 2 5 sctp 1500 -------D 0 2.0 5.0 1 218 366 0 217
- 8.533216 2 5 sctp 1500 -------D 0 2.0 5.0 1 217 365 0 216
+ 8.547616 2 5 sctp 1500 -------D 0 2.0 5.0 1 225 367 0 224
- 8.557216 2 5 sctp 1500 -------D 0 2.0 5.0 1 218 366 0 217
- 8.581216 2 5 sctp 1500 -------D 0 2.0 5.0 1 225 367 0 224
r 8.589216 2 5 sctp 1500 -------D 0 2.0 5.0 1 224 358 0 223
```

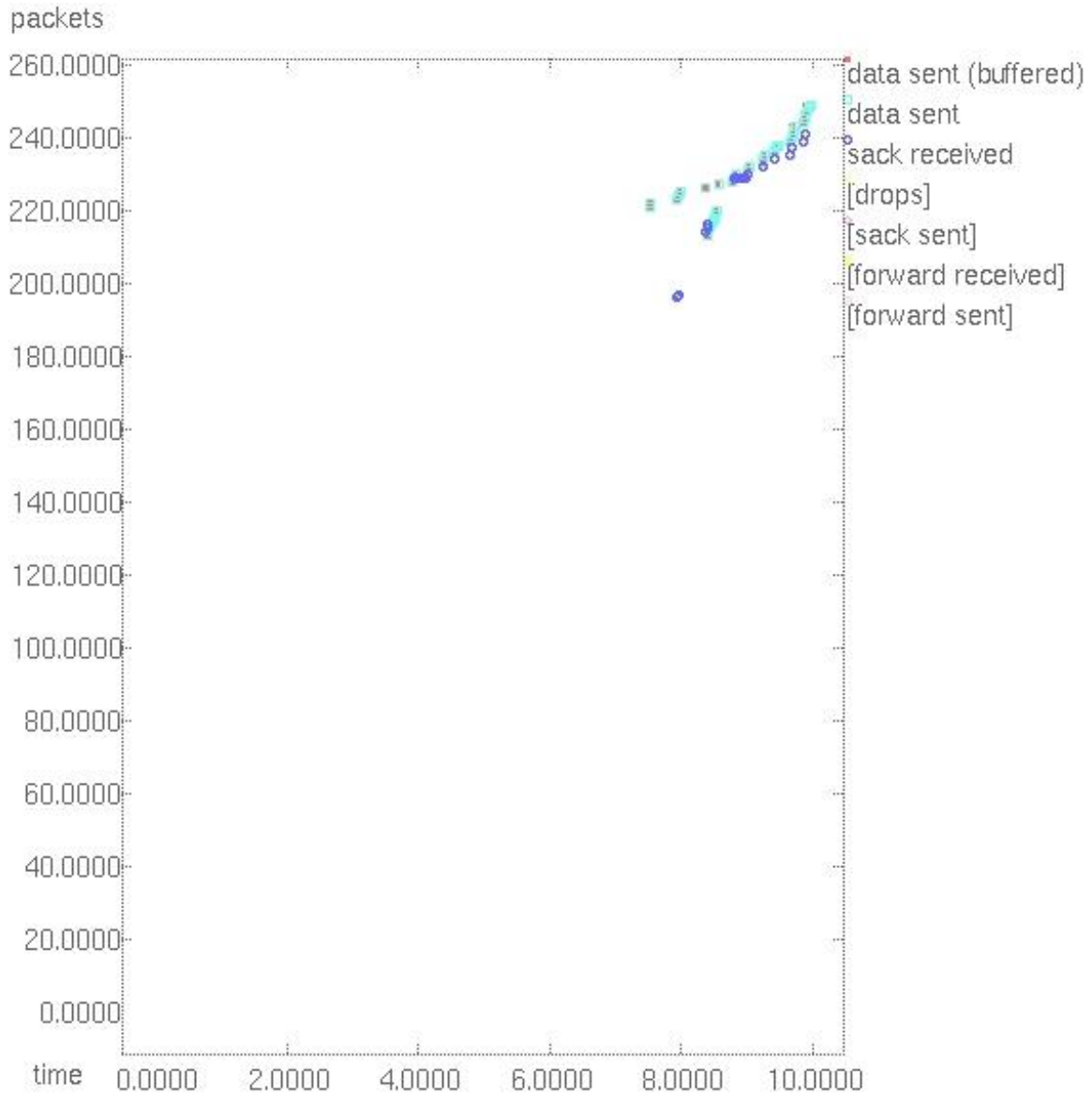Figure 9. Trace file sample SCTP interface (if1).

Figure 10. XGRAPH view of SCTP node data transmission of if1.

Then, the SCTP agent performs the send from the appropriate interface node. Incoming data is received at one of the interface nodes directly and passed up to the SCTP agent. This solution is applicable to any transport protocol that requires multi-homing functionality in NS2 [15]. The core node is used for routing and is connected to each interface node via a unidirectional link towards the interface node. Traffic does not traverse this link. Instead, these links are used to dynamically determine which outgoing interface node to use for sending to a particular destination. Figure 11 gives us some idea about the traffic flow. We can see in Figure 12, that a multiple SACK chunks (S). These acknowledge about the reception of multiple Data chunks (D).

```
                    +- iface 1     O=====
                    |                   /
    SCTP Agent      |         core   O
                    |                   \
                    +- iface 2     O=====
```
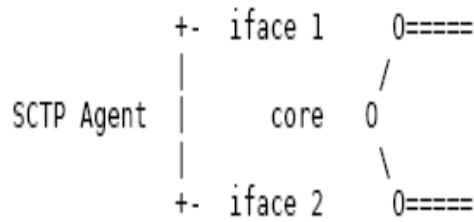
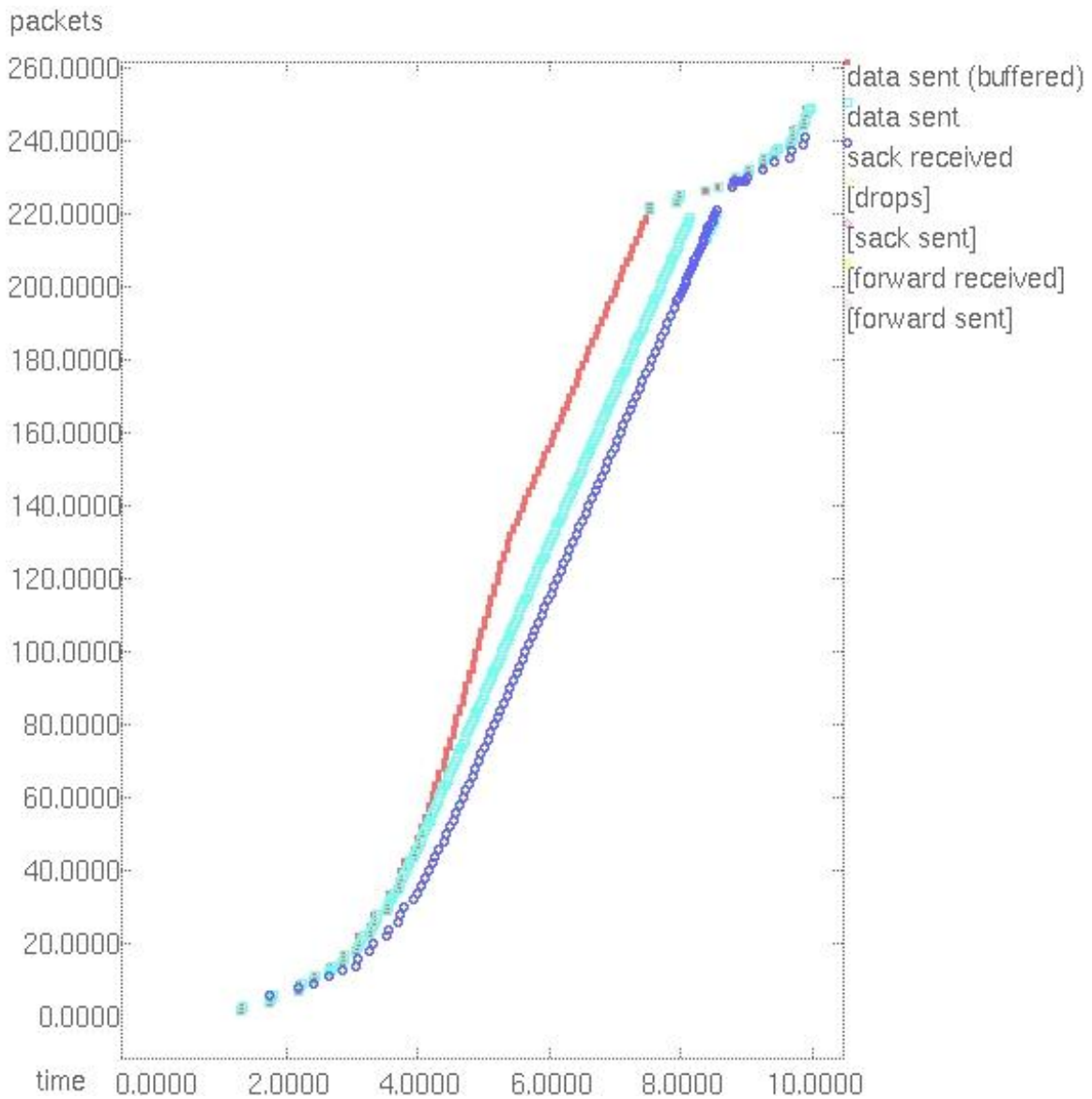Figure 11. Visualized SCTP Agent with a core node and two interface nodes [16].

Figure 12. Multi-homed SCTP node sample trace file.

```
+ 0.5 1 4 sctp 56 -------I 0 1.0 4.0 1 -1 4 65535 65535
- 0.5 1 4 sctp 56 -------I 0 1.0 4.0 1 -1 4 65535 65535
r 0.700896 1 4 sctp 56 -------I 0 1.0 4.0 1 -1 4 65535 65535
+ 0.700896 4 1 sctp 56 -------I 0 4.0 1.0 1 -1 5 65535 65535
- 0.700896 4 1 sctp 56 -------I 0 4.0 1.0 1 -1 5 65535 65535
r 0.901792 4 1 sctp 56 -------I 0 4.0 1.0 1 -1 5 65535 65535
+ 0.901792 1 4 sctp 36 -------I 0 1.0 4.0 1 -1 6 65535 65535
- 0.901792 1 4 sctp 36 -------I 0 1.0 4.0 1 -1 6 65535 65535
r 1.102368 1 4 sctp 36 -------I 0 1.0 4.0 1 -1 6 65535 65535
+ 1.102368 4 1 sctp 36 -------I 0 4.0 1.0 1 -1 7 65535 65535
- 1.102368 4 1 sctp 36 -------I 0 4.0 1.0 1 -1 7 65535 65535
r 1.302944 4 1 sctp 36 -------I 0 4.0 1.0 1 -1 7 65535 65535
+ 1.302944 1 4 sctp 1500 -------D 0 1.0 4.0 1 1 8 0 0
- 1.302944 1 4 sctp 1500 -------D 0 1.0 4.0 1 1 8 0 0
+ 1.302944 1 4 sctp 1500 -------D 0 1.0 4.0 1 2 9 0 1
- 1.326944 1 4 sctp 1500 -------D 0 1.0 4.0 1 2 9 0 1
r 1.526944 1 4 sctp 1500 -------D 0 1.0 4.0 1 1 8 0 0
r 1.550944 1 4 sctp 1500 -------D 0 1.0 4.0 1 2 9 0 1
+ 1.550944 4 1 sctp 48 -------S 0 4.0 1.0 1 2 10 65535 65535
- 1.550944 4 1 sctp 48 -------S 0 4.0 1.0 1 2 10 65535 65535
r 1.751712 4 1 sctp 48 -------S 0 4.0 1.0 1 2 10 65535 65535
+ 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 3 11 0 2
- 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 3 11 0 2
+ 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 4 12 0 3
+ 1.751712 1 4 sctp 1500 -------D 0 1.0 4.0 1 5 13 0 4
- 1.775712 1 4 sctp 1500 -------D 0 1.0 4.0 1 4 12 0 3
- 1.799712 1 4 sctp 1500 -------D 0 1.0 4.0 1 5 13 0 4
r 1.975712 1 4 sctp 1500 -------D 0 1.0 4.0 1 3 11 0 2
r 1.999712 1 4 sctp 1500 -------D 0 1.0 4.0 1 4 12 0 3
+ 1.999712 4 1 sctp 48 -------S 0 4.0 1.0 1 4 14 65535 65535
- 1.999712 4 1 sctp 48 -------S 0 4.0 1.0 1 4 14 65535 65535
r 2.023712 1 4 sctp 1500 -------D 0 1.0 4.0 1 5 13 0 4
r 2.20048 4 1 sctp 48 -------S 0 4.0 1.0 1 4 14 65535 65535
+ 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 6 15 0 5
- 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 6 15 0 5
+ 2.20048 1 4 sctp 1500 -------D 0 1.0 4.0 1 7 16 0 6
```

Figure 13. XGRAPH view of Multi-homed SCTP nodes data transmission.

Figure 13, shows the data transmission of Multi-homed SCTP nodes. We can observe that, all the interface nodes (if0 and if1) are shown together with SCTP packets limit of 260.0000 and time limit 10.000 packets sent and received till 220.000 are following exponential pattern with respect to time. But, after SCTP packet size crossed 220.000, they are received and acknowledged approximately at the same time. It shows that an SCTP Agent node in NS2 tool cannot be actually multi-homed. An SCTP node cannot transmit data simultaneously by both interface nodes. In NS2, it can either use interface node no. 0 (i.e., if0) or interface node no. 1 (i.e., if1) at the same time.

3. Conclusion

NS2 allows the general support for logically multi-homing nodes that have a multi-homed SCTP transport layer. But, it is scrutinized from the results that in NS2 data cannot transmit simultaneously from both interfaces of SCTP agent. We did not observe any heartbeat chunks being sent or acknowledged in the trace of multi-homed SCTP node simulated in NS2. The original infrastructure of NS2 does not support multiple interfaces for a single node. To overcome this limitation, NS2 allows the general support for logically multihoming nodes that have a multihomed transport layer, such as SCTP where multihomed node is actually made up of more than one node. This complexity can torn down into different results in practical solution scenarios where a single node is needed to have multiple interface for real multihoming support for experiments like simultaneous mobility.

Therefore, we can conclude that, there remains lots of scope to be investigated and development needed on NS2 to do full utilization of SCTP features. An improved version of NS-2 (NS-3) is currently under development. Hence we suggest to the developers of NS-3 to notice our findings and improve the current version of NS-3.

References

[1] T. Ravier, R. Brennan, and T. Curran, *"Experimental studies of SCTP multihoming,"* First Joint IEI/IEE Symposium on Telecommunications Systems Research, Dublin, Ireland, Nov 27, 2001.

[2] ITU-T Recommendation Q.700: *Introduction to CCITT Signalling System* No. 7, International Telecommunication Union, Geneva, March 1993.

[3] R. Stewart, Q. Xie, et al., *"RFC 2960: Stream Control Transmission Protocol",* The Internet Society, 2000.

[4] Stewart R., *"Stream Control Transmission Protocol"*, RFC 4960, September 2007.

[5] 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.

[6] R. Stewart and P. D. Amer, "Why is SCTP needed given TCP and UDP are widely available?" Sep. 2007. Available at: http://www.isoc.org/briefings/017/ (Last Access August 2014)

[7] Erwin P. Rathgeb. SCTP for beginners [online]. Available: http://www.bandwidthco.com/whitepapers/netforensics/sctp-mpls/SCTP%20for%20Beginners.pdf (Last Access August 2014)

[8] K. J. Lee, S. S. Nam, and B. I. Mun, *"SCTP efficient flow control during handover,"* in *Proc. IEEE WCNC*, New Orleans, LA, Apr. 2006, pp. 69–73. http://dx.doi.org/10.1109/10.1109/WCNC.2006.1683443

[9] SCTP Association Startup and Shutdown. Available at: http://publib.boulder.ibm.com/infocenter/aix/v6r1/index.jsp?topic=/com.ibm.aix.commadmn/doc/commadmndita/sctp_startup.htm (Last Access August 2014)

[10] M. Riegel, M. Tuxen, N. Rozic, and D. Begusic, *"Mobile SCTP transport layer mobility management for the Internet,"* in *Proc. SoftCOM*, Oct. 2002, pp. 305–309.

[11] S. Fu and M. Atiquzzaman, *"SCTP: State of the art in research, products, and technical challenges," IEEE Commun. Mag.*, vol. 42, no. 4, pp. 64–76, Apr. 2004. http://dx.doi.org/10.1109/MCOM.2004.1284931

[12] R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, V. Paxson, *"RFC 2960: Stream Control Transmission Protocol",* The Internet Society, 2000.

[13] Peyman Behbahani, *Connection Robustness for Wireless Moving Networks Using Transport Layer Multi-homing.* Doctoral thesis. City University London. (2010).

[14] A.L., Jr. Caro, J.R. Iyengar, P.D. Amer, Ladha Sourabh, G.J. II Heinz, K.C. Shah, *"SCTP: a proposed standard for robust Internet data transport," Computer*, vol. 36, no. 11, pp. 56-63, Nov. 2003. http://dx.doi.org/10.1109/MC.2003.1244536

[15] NS Manual. Available at: http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf (Last Access August 2014)

[16] NS-2 SCTP Module. Available at: http://www.eecis.udel.edu/~nekiz/sctp (Last Access August 2014)

Copyright Disclaimer