Dynamic IP reconfiguration in Stream Control Transmission Protocol

Aparna Kanungo *Janani.T* *P.Narayanasamy*
(appu_kg@hotmail.com) *(janani_krsna@yahoo.com)* *(sam@annauniv.edu)*
School of Computer Science & Engineering
Anna University, Chennai , INDIA – 600 025.

Abstract

Multihoming is the ability of a single end-point to support multiple IP addresses. Existing transport protocols like TCP and UDP do not support multihoming. However, SCTP provides a transparent support for communication between two end-points of which one or both can be multihomed. Dynamic IP Address Reconfiguration is the technique that has been used to implement this unique feature of SCTP. This capability is useful in building redundant paths or deleting inactive paths, without disturbing the established association and hence provides applications with transport level fault tolerance. It has the potential of providing robustness against network failures. SCTP-Aware application in turn helps in exploiting the multihoming and streaming feature of SCTP. The simulations of this extension of SCTP have been carried out in NS-2 and the performance has been studied graphically.

Keywords: SCTP, multihoming, association, ASCONF

1. Introduction

The transport level provides end-to-end communication between processes executing on different machines. This layer is takes care of the Quality of Service, flow and congestion control. Two of the most popular transport protocols used in the Internet are TCP and UDP.

TCP provides the upper layer with transport services such as data reliability, data sequence preservation, flow and congestion control. However, TCP is byte-stream-oriented where one stream is a sequence of bytes. It combines reliable data transfer with strict order-of-delivery which causes head-of-line blocking, especially for those applications that require only a reliable transfer with unordered delivery. Also the byte stream oriented nature requires applications to add their own record marking.

UDP is a connectionless transport protocol, which provides best-effort data delivery and preservation of message boundaries as its main services. But UDP is unreliable thus application programs using UDP must handle problems of the lost, delayed or duplicated data.

SCTP (Stream Control Transmission Protocol) is a new transport protocol that provides a reliable connection-oriented mechanism. The decisive difference to TCP is multi homing and the concept of several streams (messages) within a connection. SCTP can be used as the transport protocol for applications where monitoring and detection of loss of session is required. This paper discusses how multihoming (through dynamic IP address reconfiguration technique) can be used to enhance network fault tolerance.

This paper discusses on the advantages of the multihoming feature of SCTP. Section II explains this unique feature of SCTP in detail. Section III gives the technique by which multihoming can be introduced in SCTP. Section IV provides the detailed design of both dynamic IP reconfiguration as well as the SCTP-Aware application. Details of the SCTP-Aware application are given in Section V. Finally; Section VI presents the results and analysis.

2. Stream Control Transmission Protocol

SCTP is a reliable transport protocol operating on top of a potentially unreliable connectionless packet service such as IP. It offers acknowledged error-free non-duplicated transfer of datagrams (messages). Detection of data corruption, loss of data and duplication of data is achieved by using checksums and sequence numbers. A selective retransmission mechanism is applied to correct loss or corruption of data.

2.1 Multi Homing

The essential property of SCTP is its support of multi-homed nodes, i.e. nodes that can be reached under several IP addresses. If the SCTP nodes and the according IP network are configured in such a way that traffic from one node to another travels on physically different paths when different IP addresses are used, associations become tolerant against physical network failures. For this, both the peers exchange the set of IP addresses through which they can be reached, during the initialization phase.

2.2 Path Monitoring

An SCTP instance monitors all transmission paths to the peer instance of an association using HEARTBEAT chunks. These chunks are sent over all paths and based on the interval between the transmission of HEARTBEAT chunk and reception of HEARTBEAT-ACK chunk, the activeness of a path can be determined.

2.3 Path Selection

At the set-up of an SCTP association, one of the IP addresses from the returned list is selected as initial primary path. Data chunks are transmitted over this primary transmission path by default. In case of retransmissions, other active paths can be used.

3. Dynamic IP Address Reconfiguration

To extend the utility and application scenarios of SCTP, optional extensions are introduced that provide SCTP with the ability to reconfigure IP address information on an existing association. Dynamic IP Address Reconfiguration is:

- A graceful method to add/delete to the interfaces of an existing association
- A method for an endpoint to request its peer to set the primary destination address

These extensions enable SCTP to be utilized in the following applications:

- For computational or networking platforms that allow addition removal of physical interface cards this feature can provide a graceful method to add to the interfaces of an existing association. For IPv6 this feature allows renumbering of existing associations.
- This provides a method for an endpoint to request that its peer set its primary destination address. This can be useful when an address is about to be deleted, or when an endpoint has some predetermined knowledge about which is the preferred address to receive SCTP packets upon.

SCTP is used for dynamic address reconfiguration due to two features provided by it:

- An end-point can use multiple IP-addresses for one connection
- An end-point can change these multiple addresses dynamically without affecting the established association.

4. Design of Dynamic IP Address Reconfiguration in NS-2

NS2 does not support multihoming. A node in NS2 is assigned a single IP address. Hence, to implement multihoming in NS2 each multihomed host is made to consist of:

- A core node representing a multihomed host
- A set of interface nodes, each node representing one IP address for the multihomed core

The core node maintains two associated Linked Lists, one containing the list of interface nodes through which it can send data, and other containing the list of destination interface nodes through which it can reach the other end.

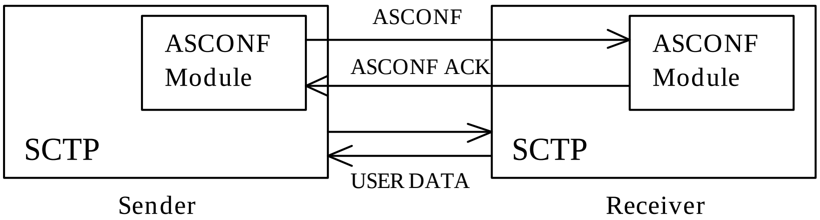

Figure 1. Sender and Receiver Interaction

The design is a sender/receiver scenario as shown in Figure 1.where one node assumes the role of the sender, which informs the receiver about the dynamic addition/deletion of IP address. The sender creates an ASCONF (Address/Stream Configuration Change) Chunk with a list of add/delete parameters.

When an end-point receives an ASCONF Chunk from the remote peer, it has to identify the association the ASCONF Chunk is associated with. If the association is unidentified, then the chunk is discarded. After identification and verification of the association, the processing of the ASCONF chunk begins at the receiver end. If the parameter being considered is an 'add' parameter then the address specified is simply added to the destination list of the receiver. However, in case of the 'delete' parameter, the receiver has to check for two error conditions:

- Deletion of the last IP address
- Deletion of the source IP address

After both these error conditions have been checked, the specified address is deleted from the receiver's destination list and an ASCONF-ACK chunk is generated with either success report (if the addition/deletion operation was done successfully) or error report (if an error condition was present during the deletion operation) and sends it to the sender end.

When the ASCONF-ACK arrives at the sender end, it checks for the success/error report for each of the parameters that it sent. If the parameter report indicates success, the corresponding operation i.e. the addition/deletion of the address to the interface list of the sender is done. However, if the report indicates a failure, then the sender does not effect the change.

Since the host can be reached through more than one interface, NS-2 implementation of SCTP gives the option of choosing:

- The source interface through which the packet can be transmitted from the sender, an option termed as "Force Source".
- The primary destination interface through which the packets can be delivered to the destination core.

5.SCTP -Aware Application

The legacy NS-2 applications are predominantly made for the environment where TCP is the transport layer protocol. So they don't have the capability to exploit the unique features of SCTP. Hence SCTP-Aware application modules were introduced, which make use of SCTP's way of sending packets as streams instead of as bytes as done in TCP. By using SCTP -Aware applications one can specify:

- Number of Outgoing streams
- Reliability level of the message
- Unordered delivery of stream is needed or not
- Whether a stream can be specified unreliable or not

This application also provides the user with the flexibility to dynamically reconfigure IP addresses through the application itself. This can be helpful when the user wants to use a particular path for security reasons or when the path's activeness needs to be tested.

6. Results and Analysis

The result of the simulation was obtained using the following:

- XGRAPH demonstrating the performance of SCTP graphically
- Debug file showing a step-by-step execution of each module and other relevant information
- Visualization of Dynamic Addition/Deletion of interfaces through Network-Animator (nam) tool of NS-2

The graphs generated by XGRAPH demonstrated SCTP's ability to transfer packets through the best path. The HEARTBEAT Chunks were used for this purpose thus ensuring that the paths through which the HEARTBEAT-ACK was received were active and the rest were inactive. Thus the performance of each interface nodes was measured by its effective response to Data and HEARTBEAT Chunks.

The introduction of a new agent that supports SCTP in NS-2 has made it possible for a node to support both SCTP and TCP at its transport layer. Thus a node can establish an SCTP association to another SCTP-Aware NS-2 node. At the same time it can also communicate with other legacy NS-2 nodes, which are not SCTP-Aware, using the traditional TCP as it transport protocol. Thus SCTP can be smoothly integrated into the existing NS-2 environment.

The performance of SCTP was compared with that of TCP by varying various parameters for both such as:

- The packet size
- MTU of the path
- Channel properties
- Number of streams
- Applications

7. Conclusion

The study of SCTP's performance in different scenarios indicates that SCTP's behavior is comparable to that of TCP when used in a normal single-homed host. The addition of "Dynamic IP address reconfiguration using ASCONF Chunks" has given SCTP greater flexibility in choosing its paths dynamically, without affecting the established association. SctpApp service when used in the Application Layer helps in deploying these features and extensions of SCTP to the maximum extent. Multi-streaming feature provides the important advantage of avoiding head-of-line blocking. Altogether, multi-homing and multi-streaming features give SCTP lot of flexibility and improved performance. The enhancement of this work is to use this SCTP for various wireless mobile applications.

REFERENCES

[1] M. Riegel, M.Tuexen: "Mobile SCTP," <draft-riegel-tuexen-mobile-sctp-00.txt>, Jan 2002

[2] R.R.Stewart, Q.Xie, M.Tuexen, I.Rytina: "SCTP Dynamic Addition of IP Addresses," <draft-ietf-tsvwg-addip-sctp-04.txt>, Jan 2002

[3] L. Coene et al.: "SCTP Applicability Statement,"<draft-ieft-sigtran-sctp-applicability- 08.txt>, Nov 2001

[4] R.R.Stewart et al.: "Stream Control Transmission Protocol," RFC 2960, Nov 2000

[5] Armando L. Caro Jr., Janardhan Iyengar: "SCTP Module for NS-2 Simulator," http://pel.cis.udel.edu, Oct 2002