Performance Evaluation of Transport Protocols in IP/MPLS Networks

Developed By:
Hafiz Zulfiqar Hussain
258-FAS/MSCS/F05

Supervised By:
Mr. Qaisar Javaid

Department of Computer Science
Faculty of Basic and Applied Science
International Islamic University Islamabad
2009

International Islamic University, Islamabad

Dated: --------------------

Final Approval

It is stated that we have read the thesis titled as "**Performance Evaluation of Transport Protocols in IP/MPLS Networks**" submitted by Hafiz Zulfiqar Hussain Reg # 258-FAS/MSCS/F05. It is our judgment that this project is of standard to warrant its acceptance by the International Islamic University, Islamabad, for the Degree of MS in Computer Science.

Project Evaluation Committee

External Examiner
Dr. Muhammad Zubair
Assistant Professor
Department of Computer Science
Riphah International University, Islamabad.

Internal Examiner
Prof. Dr. Muhammad Sher
Chairman, Department of Computer Science
Faculty of Basic and Applied Sciences
International Islamic University, Islamabad.

Supervisor
Mr. Qaisar Javaid
Assistant Professor
Department of Computer Science
Faculty of Basic and Applied Sciences
International Islamic University, Islamabad

Declaration of Originality

I, Hafiz Zulfiqar Hussain Reg # 258-FAS/MSCS/F05 student of MS (CS) in International Islamic University Islamabad, declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person or our self, nor material which to a substantial extent has been accepted for the award of any other unit, degree or diploma of a university or any other institute of higher learning, except where due acknowledgement is made in the text.

I have made and retained a copy of this original assignment.

SIGNATURE:

Hafiz Zulfiqar Hussain

A dissertation submitted to the

Department of Computer Science

International Islamic University Islamabad
As a partial fulfillment of requirements for the award of
The degree of

MS in Computer Science

Dedication

I dedicate this research project to our beloved Parents, to my Family Members.

Acknowledgement

At very first, I bestow all praises, acclamation and appreciation to Almighty Allah, the most merciful and compassionate. The most Gracious and Beneficent, Whose bounteous blessings enable me to pursue and perceive higher ideals of life. All praises for His Holy Prophet Muhammad (SAW) Who enabled me to recognize our Lord and creator and brought to me a real source of knowledge from Allah, The Quran, Who is role model for me in every aspect of life. Secondly I must mention that it was mainly due to my family's moral and financial support during my entire academic career that enabled me to complete my research project. I would like to thanks to my teachers and especially consider it a proud privileged to express my gratitude and deep sense of obligation to my reverend supervisor **Mr. Qaisar Javaid** (Assistant Professor DCS, IIUI) for his dexterous guidance and kind behavior during the project. I also would like to admit to my truly friends Mr. Zeeshan Shafi Khan (PHDCS, IIUI) and Mr. Rashid Mahmood (MSCS, IIUI) that helped me in every difficulty. I once again would like to admit that my owe all my achievement to my most loving parents who mean most to me, for their prayers are more precious them any treasure on earth.

Hafiz Zulfiqar Hussain

Project in Brief

Project Title:	**Performance Evaluation of Transport Protocols in IP/MPLS Networks**
Organization:	**International Islamic University, Islamabad (IIUI).**
Developed by:	**Hafiz Zulfiqar Hussain (258-FAS/MSCS/F05)**
Supervised by:	**Mr. Qaiser Javaid Assistant Professor Department of Computer Science Faculty of Basic and Applied Sciences International Islamic University, Islamabad**
Starting Date:	**March 2008**
Completion Date:	**June 2009**
Tool Used:	**Network Simulator NS-2 C++ MS Office**
Operating System:	**Linux Fedora Core 6**
System Used:	**Pentium IV (1.7GHz Genuine Intel) RAM 1024 MB HD 40 GB**

Abstract

Multi-Protocol Label Switching is the best technique for efficient utilization of network resources, with a small overhead labels. The reliability features of Stream Control Transmission Protocol (SCTP) and traffic engineering (TE) procedure improves performance needs of real time data requirements that need heavy overheads in Internet Protocol (IP) based networks. MPLS based setup has comparatively better Quality of Services (QoS) than the IP based systems.

The Internet Protocol is the dominant protocol in computer networks today. MPLS adds the flavor of virtual circuits in computer networks. We have to evaluate the efficiency of two different version of Transmission Control Protocol i.e TCP Vegas, TCP new Reno and new emerging transport protocol SCTP under MPLS technology. Average delay, throughput, channel wastage and average packet delivery are performance parameters before and after rerouting scenarios of FTP and CBR network traffic with variable bandwidth. The results showed that in scenario of delay TCP Vegas performs better with minimum average delay while in case of throughput, channel wastage and packet delivery SCTP perform better than other protocols used in this research. Results have been obtained through simulations over Network simulator.

List of Figures

List of Table

Table of Contents

Contents Page #

1. Introduction

Next generation Networks are progressively using the Multi-Protocol Label Switching technology as a source for creating the converged Networks. This makes available both the layer 2 and layer 3 properties. It minimizes the operational expenses of the network. MPLS uses IP based network control protocols that help in designing the IP-based networks. We want to examine the performance of Transport layer Protocols in the IP/MPLS network, and also hypothesis that transport protocols (TCP Reno and SCTP) perform better in this simulation environment than the other networks.

1.1 Multi-Protocol Label Switching (MPLS)

A framework that facilitates the effective routing, forwarding and switching of data streams through the network is called MPLS. It handles the issues related to data rate, reliability, scalability and quality of service of traffic in communication network. MPLS facilitate the bandwidth and service related issues of IP communication networks. It solved the issue related to scalability and routing. It is used to carry different types of traffics including IP packets, ATM, Synchronous Optical Network (SONET) and Ethernet frames. It is a technology which combines the network-layer routing with label-swapping model.

1.1.1 MPLS Functions

MPLS perform the following functions:

- It specified the mechanism used to handle the flow of data rates between different software applications and hardware systems.
- It is independent from data link and network layer.
- It represents the IP to a secured and fixed address.
- It facilitates the Asynchronous Transfer Mode, frame relay and IP protocols.
- It provides interfaces to routing protocols like RSVP, CR-LDP, OSPF, RSVP-Tunnel and BGP.

1.1.2 MPLS Class of Service

MPLS class of service enables network administrators to provide differentiation services (QoS) across an MPLS network, networking requirements can be fulfilled by defining the detailed class of service for each packet by means of the priority bit in each packet. MPLS CoS provides the differentiated services:

1.1.3 MPLS Labels

A label provides a way on which data packets travel. A label existed in a MPLS shim with other fields and this shim lies between the link layer and network layer headers. The values of packets are used to search the adjoining routers by those routers who used as receiver. After assigning the label to the packet, the packet passes through the central point on label switching.

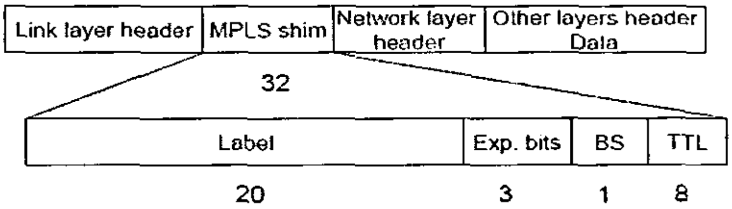

Figure 1.1: MPLS Label Format [2]

In the above figure each label values having the following fields.

- First filed contains the label that having a values of 20-bit.
- A 3-bit experimental bit, often used for Quality of Service or Class of Service in second field.
- Third filed of this header has one bit bottom of stack (BS) value. If it has a value, then it means it has the final value.
- Final value of this header is an eight bit value that represents the time to live fields.

1.1.3.1 Label assignment

This process depends on the underlying standards. Those include

- Single network destination routing.
- Traffic Engineering
- Virtual Private Network

- Quality of Service
- Multicasting

1.1.3.2 Label Creation Methods

There are number of ways to create the labels which are given as under.

- Topology based Method
- Traffic based Method
- Request based Method

The request and topology based techniques show the behavior of connection of label control, while the methods of data traffic are related with data driven binding techniques. Label Edge Router is a router that works at the edge of the networks. The entry and exit point in MPLS network is called label edge router. Label edge routers have more than one port connected to different networks likewise ATM, Ethernet and Frame Relay. Position of label edge routers and label switch routers are given as under.

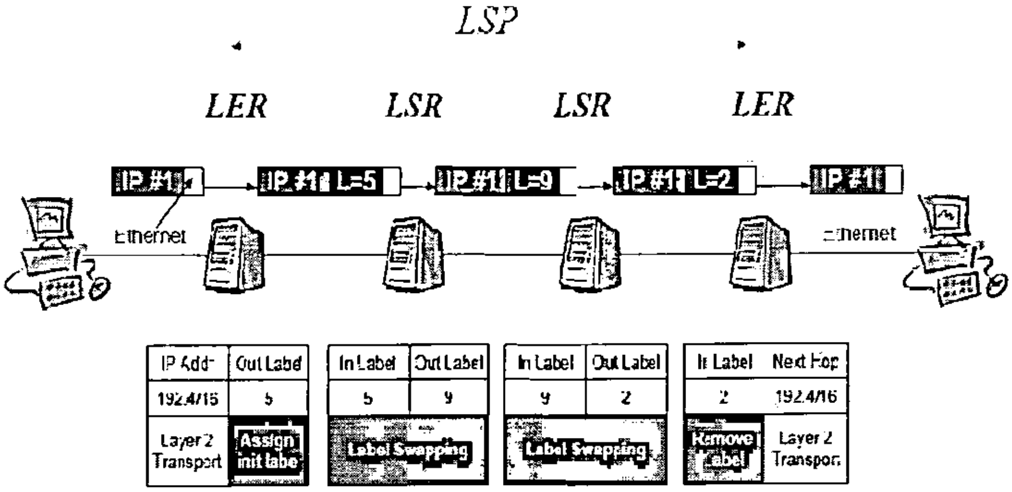

Figure 1.2: Position of LER & LSR. [2]

1.1.4 Label Switch Paths

A path through which more than one label switch routers are communicate, called label switch router. MPLS provide the following two ways to setup a Label switch path.

- **Hop by hop routing**

  This required a technique that is identically used in IP networks

- **Explicit routing.**

Here the LSR with entry point specifies the path for ER-LSR traverse. Resource also keeps the QoS.

For different reasons the label switched paths are designed by network administrators, such as to create IP VPN or to create router traffic along specified paths through network. At each node from source to destination, the label switch paths have a number of labels. There are a number of label distribution protocols used in MPLS environment such as Resource Reservation Protocol, and Constraint base Routing LDP and many more on routing protocols. Extensions of the base Label Distribution Protocol support explicit routing and for the purpose of explicit routing we use the CR-LDP. A number of techniques for the exchange of labels are introduced [2].

- Label Distribution Protocol
- RSVP,CR-LDP
- Border Gateway Protocol (BGP)-external label.
- Protocol independent multicast (PIM).

1.1.5 Label Distribution Protocol (LDP)

The communication knowledge for the division of label binding to label switch router in MPLS environment done by the LDP[4]. This is used for the distribution of label binding information to Label Switch Routers in an MPLS network. The connections between two LSP are created among the LDP having equal standards in MPLS networks. There are following types of LDP messages that existed [2]

- Discovery messages.
- Session messages.
- Advertisement messages.
- Notification messages.

1.1.6 Traffic Engineering

Traffic engineering (TE) is a technique that improves the utilization of network resources by attempting to create a consistent traffic throughout network. The significant outcomes of this technique is the prevention of congestion on anyone path. It does not inevitably select the minimal path length between the two systems. Although the starting and ending node of the MPLS domain are same yet they have follow the different paths. In MPLS, traffic engineering is inherently provided using explicitly routed paths. The LSPs are created independently, by specifying different paths that are based on user defined policies. There are two approaches RSVP and CR-LDP are used to provide dynamic QoS and TE in MPLS networks [2].

1.1.7 MPLS Operations

MPLS domain performs the following operations. This process shows the behavior of data packets travel through MPLS domain [2].

- Create the table at each router.
- Create the Label and distribute it.
- Label Switch Path creation.
- Packet forwarding
- Label insertion or search the table.

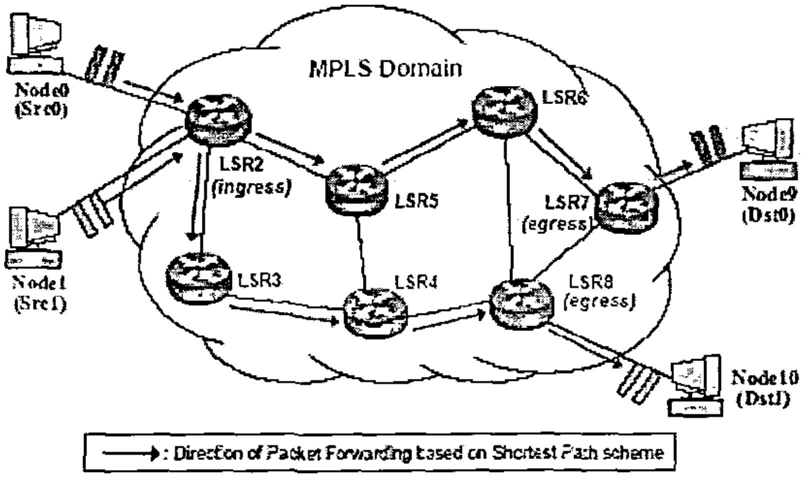

Figure 1.3: MPLS Network Domain [3]

Figure1.3 shows that node0, node1, node9 and node10 are the IP nodes and other nodes (LSR2, LSR3, LSR4, LSR5, LSR6, LSR7, and LSR8) are MPLS nodes. Sr0 agent is attached with node0 that sends data packets towards agent dst0 attached with the node9, while the node1 has src1 agent that sends data towards agent dst1 attached with the node10. Due to the packet forwarding techniques, data is travel on the shortest path from src0 to LSR 2-5-6-7 and packets are passed from src1 to dst1 through LSR2-3-4-8 [3].

All the data is not transferred along the same path in MPLS when the source system pushes its data traffic towards destination. It has limited property of MPLS that without mentioning the middle way routers; it can easily control the data packet path. This can be done by making the cavities through middle layer routers that can consist of more that one segments. This idea is used in furnishing virtual private networks (VPN) of MPLS.

1.1.8 MPLS Protocol Stack Architecture

These components can be divided into the following parts [2].

- Network routing protocols.
- The label switching network.
- The layer edge forwarding networks.
- The signaling protocol for distribution of label.
- Label schematics and granularities.
- Compatibilities with many data link layer forwarding, like, Point to Point, Frame relay and ATM.
- Traffic engineering.

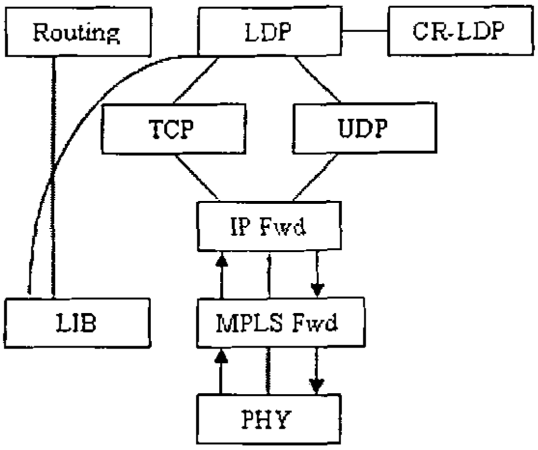

Figure 1.4: MPLS Protocol Stack [2].

Figure 2.4 shows the protocols that used in MPLS manipulation operations. The routing prototype may be used border gateway protocols (BGP), open shortest path first (OSPF) and asynchronous transmission mode (ATM). For reliable data transmission from one label switch (LSR) to other, the label distribution protocol (LDP) prototype uses the TCP. It also created the label information base (LIB) table. The LDP uses user datagram protocol (UDP) for discovery phase of operation. The IP forwarding looks the adjoining hop by making comparison of largest physical locations in table. The MPLS fwd unit makes a comparison of an outgoing port with a label. The layer in boxes with arrow lines may be implemented in hardware for effective and quick operations [2].

1.1.9 MPLS Applications

MPLS is a newer area in the switching network. It provides the following application [2].

- MPLS strengthen the packet switching and forwarding efficiency in a network.
    - o It strengthens the performance of the network by enabling the routing and switching at wireless access rates.
    - o MPLS improves the packet delivery by using the layer-2 switching domain.
    - o It gives the permission for easy to implement.
- It maintain the CoS and QoS for service differentiation.
    - o It combines the distribution for explicit and constraint based routing.
    - o TE has been used by the MPLS to establish the path setup and service securities.
- MPLS support scalability of network.
    - o It must be used to ignore the layer 2 overlay connected with meshed IP-ATM networks.
- MPLS integrate the Asynchronous Transfer Mode and Internet Protocol in the network.
    - o It can be efficiently uses ATM switch hardware, by joining the two separate domains.
    - o It provides a connection between access IP and core ATM networks.

- MPLS helps to build an expandable VPN with traffic engineering capability.
- It facilitates IP over SONET integration in optical switching Networks.

1.2 Transport Layer Protocols

Transport layer provides the three protocols; we want to uses SCTP (Stream Control Transmission Protocol) and TCP variant new Reno. We have chosen the SCTP for several reasons. First it is developed for time critical applications and also it has the capability to replace both the TCP and UDP because it has many more features that are not found in both these protocols. SCTP is reliable Transport Protocol to transmit SS7 signaling message on IP based networks. TCP is byte oriented while the SCTP is message oriented. In TCP all the data is transmitted in the form of bytes while in SCTP the data is treated in the forms of blocks. We also hypothesize that TCP New Reno performs better that other TCP Variants (Reno, Tahoe, Vegas) that previously worked.

1.2.1 Transmission Control Protocol (TCP)

The Transmission Control Protocol standard is defined in the RFC Standard 793 by the IETF. The three-way handshake in TCP is a method used to establish and turn down network opened connections. This handshaking technique is called 3-way handshake or more efficiently called SYN, SYN-ACK, and ACK. Figure 1.5 shows the three way handshaking communication between the sender and receiver.

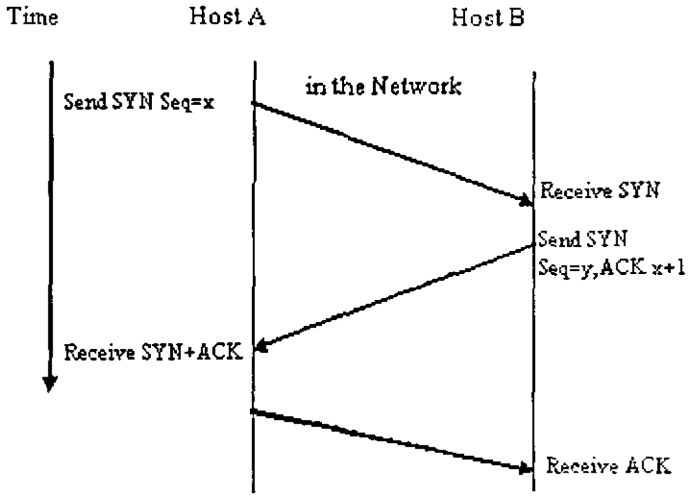

Figure 1.5: TCP Three way handshaking [4]

1.2.2 TCP Variants

TCP has a number of variants but we only discuss two of them, those are given as under.

1.2.2.1 TCP New Reno

TCP Reno's fast recovery and retransmit algorithms are well organized in dealing with the congestion packet drop. It does not care if just one packet is lost in a congestion window. In case of multiple packets drop, TCP Reno will retransmit the first packet for which it got the duplicate acknowledge from destination side, but then it will close the Fast Recovery phase. After knowing that there is more drop packets, TCP Reno wills again go to fast recovery state. The constantly restarting of the Fast Recovery state affects the performance of this protocol.

TCP New Reno tries to fix the problem by residing in the Fast Recovery stage until there are excellent dropped packets. That fact is known by the reception of the partial acknowledge. Partial Acknowledgement is the acknowledge to the first packet resend to the Fast Recovery phase that has not informed that all the packets were sends earlier to the Fast Recovery phase. This implies that the resend packets were not the only packet lost in that window. TCP New Reno resides in the Fast Recovery as long as partial acknowledgements are accepted.

1.2.2.2 TCP Vegas

TCP Vegas is a TCP algorithm that handles the network congestion by working on the packet delay and not on the packet losses. TCP congestion control algorithm includes the three mechanisms like Slow Start, Congestion Avoidance and Fast Retransmit. According to previous research that TCP Vegas has higher throughput values than the TCP Reno, which currently used in internet today. Therefore the researcher's emphasizes on the TCP Vegas to be implemented in working environment as compared to other versions of Transmission Control Protocol.

The congestion Avoidance algorithm of TCP Vegas depends on following parameters and their calculations. First we find out the Expected throughput values by dividing the congestion window with Base round trip time.

$$\text{Expected} = \text{CWND}/\text{BaseRTT}$$

Here CWND represents the congestion window size while the BaseRTT is the initial calculated Round Trip Time value.

After this we find out the actual throughput value by dividing the congestion window with round trip time.

$$\text{Actual} = \text{CWND}/\text{RTT}$$

Here CWND represents the congestion window size while the RTT is the value of component informed during the last Round Trip Time value.

Now we have to calculate the difference of these values by

$$\text{Diff} = (\text{Expected} - \text{Actual}) * \text{Base RTT}$$

If the difference is less than a threshold value α this congestion window goes to increased. If the difference is a less is greater threshold β then we says that the congestion window going to be decreased. Other wise there is no change in threshold values. The above mentioned process is used to arrange the values are given as under:

$$wnd = \begin{cases} Cwnd+1 & \text{If } diff < \alpha \\ Cwnd & \text{If } \alpha \leq diff \leq \beta \\ Cwnd-1 & \text{If } diff > \beta \end{cases}$$

The problem with the TCP Reno's congestion avoidance algorithm is that it leave gaps for other connections to be survived but the TCP Vegas is very sensitive in this case that it tries to fill up these spaces and not leave them for wastage.

It is a need to produce the quality of fairness between the TCP Vegas and TCP Reno for implementation of TCP Vegas in real world network scenarios.

1.2.3 Stream Control Transmission Protocol (SCTP)

Stream control Transmission Protocol (SCTP) improves upon TCP and UDP by combining the components of each. It is a protocol for transporting of public switching telephone network (PSTN) signals over an internet protocol. The developers of protocols say that Stream Control Transmission Protocol is probably used for larger scenarios, sincluding data multi-streaming; there was not any requirement of the Transmission Control Protocol. This protocol has connectional and reliability features that enables reliable data transfer over IP-based networks that provides many properties of Transmission Control Protocol such as sequencing and fragmentation. Stream Control Transmission Protocol ejects many of the running cost includes in TCP, in this response the factor of delays is arises. It also affords the facility of many more aspects that strengthen its transportation [8].

SCTP is considered as a bridge between the SCTP user application and a connectionless packet network of IP. Stream Control Transmission Control Protocol facilitates many basic features for its one end point to other one. SCTP also provides additional features as compared to TCP. SCTP detects the data corruption, data lost and data redundancy, and uses the selected retransmit data to correct the corrupted or lost data.

In an Internet Protocol stack, TCP, UDP and SCTP exist with many other transport layer protocols. The said protocol facilitates all properties that already existed in the user datagram protocol and the transmission control protocol. SCTP minimizes many deficiencies exhibited in transmission control protocol and it takes the best properties implemented in user datagram protocol (UDP).

SCTP provides the following features:

- It provides application protocol data unit bundling and fragmentation.
- SCTP reduces the delay by transmission of data into more than one data streams.
- It also notifies non-redundant and error free data transfer.
- SCTP provides the both ordered and unordered and full duplex data transmission.
- It supports the explicit congestion notification.
- It provides the facilities of multi-streaming and multi-homing that are not existed in TCP and UDP.
- It makes better the disadvantage of SYN-flooding.
- It gives the facility of path MTU discovery.

SCTP also includes many more features, like packet sequence numbers, checksum and selectively data resending, loss of data, data corruption detection, and of its duplication.

SCTP enables the communication of two endpoints to reduce Synchronous-flooding attacks, and to recognize unnecessary data packets. It also provides many congestion avoidance mechanisms to overcome data reduction in a changeable scenario. It provides the proper data correction mechanisms to escape non essential data transmission retransmission [9].

1.2.3.1 Limitations of TCP and UDP

Transmission control and user datagram protocols are used as protocol of network layer. But the data communication services provided by them are deficient to meet the needs of large area of technological applications, for example the telecommunication and multimedia applications. They need a healthy protocol that facilitates the reliability of TCP and flexibility of UDP, for transmission of data between source and destination.

There are following limitations exist in TCP:

- **Sequential Data Transmission:** TCP gives reliability of data transfer, but the data is transferred in sequential form. Many applications also require reliable and non sequential transportation of data. Although some applications require not a full ordering of data yet the ordered data is only handled in the inner small parts of data. In addition, a problem is created by sending the data from different sources for one destination, in this situation the data is transmitted only from one source while the data from other sources is blocked and creates the Head of line blocking. This problem is solved by the virtual output queues by creating the virtual data travel paths.
- **Transmission of data in Streams:** TCP transmits data in the form of streams. To describe their messages the applications add their marking of records. For the assurance of data delivery in specified the application uses PUSH flag in data packet.
- **Connection Oriented Nature**: It is a connection oriented protocol; every system in protocol has a NIC, for the creation of an association between two systems. If the connection is disconnected, data becomes unavailable until the host will reconnect.
- **SYN Flood Attacks**: Another disadvantaged of TCP is that it is defenseless to DoS attacks. These attacks occur by the sending the data packets by malevolent system with a fake address and it forwarded multiple TCP SYN information to the accused system. In

these attacks every time the accused host received a new synchronous flood packet. At last a situation come when the stack is full with these fake SYN packets and at this the system cannot in the condition to handle the genuine SYN packets.

Following are the limitations exist in User Data Protocol:

- **Unreliable Data Transfer:** Due to connectionless property the UDP has no reliability of data transfer in his connection; there is no guarantee that packet has reached his respected destination.

- **Absence of Congestion Control:** For the path congestion detection there is not any congestion control algorithm existed in UDP. Due to this a multitude data packets go into an already congested network. This responds data in the form of destruction.

- The execution of UDP causes additional overhead and problems in the applications, when accurate instructions of data certainty are effectuation in application.

1.2.3.2 SCTP in IP Stack

An IP stack comprised of several layers with the performance of their distinctive operations. The IP stack is consists of the given below layers of IP stack with their functionality.

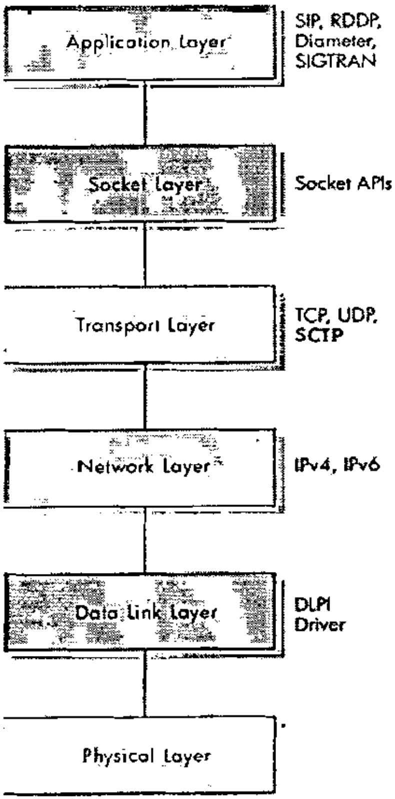

Figure 1.6: Internet protocol stack [9]

An IP stack comprised of several layers and each layer perform distinctive operations. Following are the layers in IP stack and their operations:

- **Physical layer:** This solves the issues related to physically data transmission on the network devices.
- **Data link layer:** It handles the issues related to data exchanged over devices. The detection and removal of wrongness faced by the physical layer is also the responsibility of this layer.
- **Network layer:** In this layer IP is used as routing protocol at when the data is transmitted from sender to receiver in the network.
- **Transport layer:** This layer facilitates delivery of data between senders to receiver system with the help of services used in the network layer. The said layer has two important protocols, the TCP and the UDP and SCTP. As we know previously TCP provides sequential and accurate data transmission through flow control and error recovery methods, while UDP is a connectionless and message oriented protocol. SCTP is also used as data transmission protocol.
- **Application Layer:** This layer enables the application programs to transmit data on network through an interface provided by this layer. Socket layer is used as an interface to correspond with the transport layer.

1.2.3.3 Connection Setup in SCTP

The transport layer protocol like transmission control protocol and stream control transmission protocol start a new connection with a handshake process. TCP uses a three-way while SCTP uses a four-way handshake process for the creation of new connection.

1. In this process system A starts a connection by sending an initial data intimation the system B.

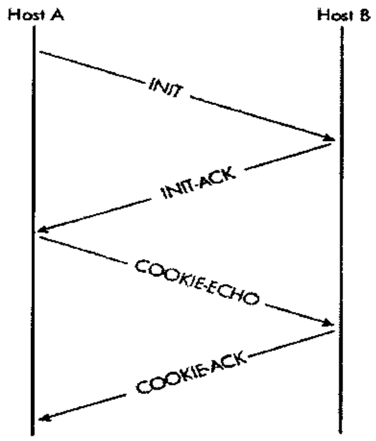

Figure 1.7: Four way handshake process in SCTP [9].

2. System B response with acknowledge of his init message. Here is a cookie field that has a timestamp to avoid playback attacks using old cookies and signature for authenticity. So the connection is established for communication between two systems.

3. To overcome this delay process, SCTP allows to interchanging of data in the COOKIE-ACK and COOKIE-ECHO data chunks [9].

1.2.3.4 SCTP Packet

SCTP sends data in the form of messages and each of them have one or more data packets. SCTP packet format is given in below figure 1.8.

The SCTP packet possesses data chunks and common headers. This header has contained the under mentioned data facts:

- For multiplexing of SCTP connections, the source and destination ports.
- SCTP has a verification field of 32-bit that protects in competition the addition of a wrong message into the SCTP connection.
- For the purpose of error detection SCTP uses a 32-bit checksum. This checksum can be a 32-bit CRC value.

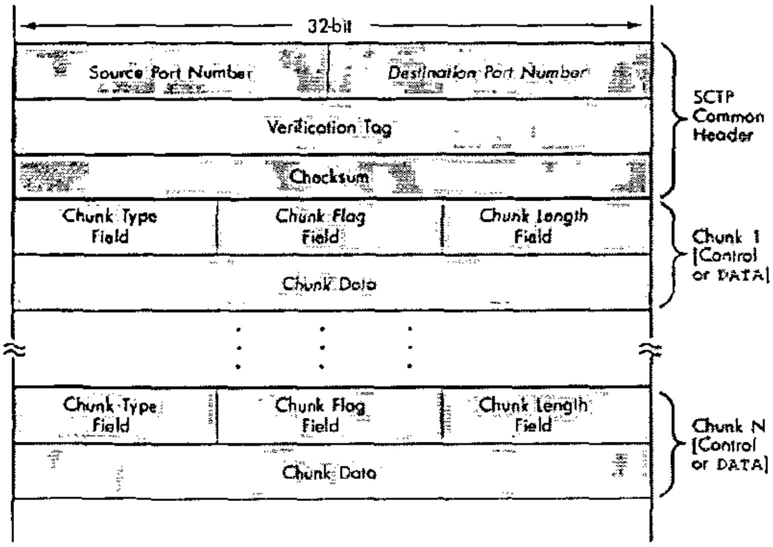

Figure 1.8: SCTP packet format [9].

A chunk can be a DATA or a control. A control chunk contains distinct parameters and flags, dependent on the chunk type. The DATA chunk contains flags to control segmentation of data, and parameters for Payload Protocol ID, SSN, SID and the TSN.

1.3 Contribution of this Dissertation

After evaluating these variants of TCP, no further research is made to test latest flavor of the TCP and other emerging transport layer protocol. TCP variants like New Reno can perform better as compared to TCP Vegas. Moreover a SCTP which is relative to TCP and also have many extra features as compared to TCP. So SCTP can be a good choice for MPLS as compared to three evaluated TCP variants. With traffic engineering it looks promising to study the performance achieved by TCP and its variants and as well as SCTP. We evaluate the performance of SCTP, TCP New Reno and TCP Vegas Protocols, developed a comparison of both these protocols and also analyses the best one in the MPLS Network environment. The simulation will perform in Network Simulator 2 (NS-2) tool which was specially design for the simulation of network protocol.

1.4 Dissertation Organization

My thesis is planned as follows. In Chapter 1, I have given the brief introduction of protocols that I used in this research, and the contribution of my work in this area of research. In Chapter 2, is about the literature survey according to this technology. In Chapter 3, the problem statement and purposed solution of this problem will be discussed. Chapter 4 describes the system design and methodology, In Chapter 5, introduction of the Network Simulator will be discussed while in Chapter 6 we analyze the three protocols under different scenarios and network parameters in the form of tables and graphs. Chapter 7 is about the conclusion and future work in this area of research will be discussed; finally references will be added.

2. Literature Survey

Given below papers have discussed the transport protocols implemented in different environment especially the performance analysis of the SCTP, TCP and TCP variants by using different network performance parameters. Although directly related work has not been found previously, yet the performance of Transport Protocols (SCTP and TCP Variants) have analyzed in different network environments.

2.1 Previous Work

We categories the previous work according to technologies, in first we discuss the previous work related to the multi-protocol label switching, then the performance of transport layer protocol and at last related work of transport layer protocols in MPLS environment has been discussed.

2.1.1 MPLS Survey

Dongli Zhang et al [17] analyzed the Quality of Service performance that includes the voice over IP and computerized video. Integrated audio and visual data will be transported in the combination of MPLS and IP systems. The combination of both MPLS technologies and QoS are considered to gives valuable results. If a path is congested then it cannot get the QoS because the data is lost due to congestion. MPLS Traffic Engineering creates an end to end data transmission path before the communication of data. MPLS-TE can not give the facility of QoS for differentiated services because it only reserves resource in one class. MPLS differentiated service aware TE made a MPLS-TE aware of Quality of Service, by mixing together the functionalities of both Traffic Engineering and differential services.

At the end the author shows that, according to the results of this simulation if bandwidth is reliable then it shows a good Quality of Service for different type of data traffics.

Chun-Choong Foo et al [1] review some problems about the performance of Mobile IP and control mechanism requires integrating MPLS and Mobile IP. Author proposed a technique to combine the Mobile IP and MPLS protocols. Mobile IP is an IETF protocol that permits the customers to go here and there but have the continuous IP network connectivity. This integration process strengthen the growing amount of data traveling process of Mobile IP by the advantage of MPLS features that have fast switching and higher values of growing amount of data. Authors have excluded the tunneling from IP to IP from Home agent to Foreign Agent on the bases of this MPLS and IP combination technology.

Gaeil Ahn et al [3] represents the design, and implementation of a MPLS simulator, which provides label swapping techniques, LDP, CR-LDP, and many kind of label distribution function. It is used by researchers to simulate how a Label Switch Paths (LSP) are created and ended and how these packets performed on the LSP. If we want to show how MPLS simulator behave, the basic MPLS functions interpret in MPLS standards is simulated; flow aggregation, label distribution schemes, LSP and ER-LSP tunnel. The simulator in this paper is not fully qualified; it has many deficiencies and requires a lot of improvements in it. For making a full finish product, it needs to be extended as RSVP and Quality of Service at its each ending point. The results are shown by graphical manner.

This paper by Wei Sun Praveen Bhaniramka Raj Jain et al [10] is about the quality of service of MPLS. Author makes a comparison of the UDP and TCP flows when they share MPLS traffic trunks or either a channel. As we know that in MPLS traffic non-shortest path channels can also used, due to this throughput of the network boost up with suitable MPLS traffic engineering. As in this research the TCP and UDP flows are combined with each other. Due to this mixing, the UDP flows increase their rates when TCP flows receive the minimize service. Also it has been seen that MPLS trunks should be implemented from initial to final stage by taking the advantages from traffic engineering. The advantages are eliminated, if some portion of the network is MPLS trunk-unaware.

In this paper Md. Arifur Rahmanl et al [19] said that MPLS is the fastest developing network to increase expansion of data rates, and service providing competencies. By using the virtual path capabilities, MPLS provides the different services across the Internet. Traffic engineering capabilities of MPLS has the ability to prevent the congestion and it effectively use the given bandwidth values. The exact results of MPLS can achieve by comparing it with given networks. RSVP, Traffic Extension RSVP and CR-LDP supporting the performance analysis of Quality of Service parameters. MPLS also applicable over the conventional systems to examine its efficiency.

2.1.2 Performance of Transport Protocols

In this paper, Grinnemo et al [14] describe that the SCTP is developed by the group of IETF; it is an emerging transport protocol for PSTN signaling data traffic. The influence of head of line blocking on TCP and SCTP has given uncertain results, in fact any important effect on transmission delay. Author has carried out a brief experimental result on the quantitative manner of HOLB. The study of this paper represents that although HOLB can actually acquire a considerable delay on a small part of the messages in an SCTP connection, it has only a lateral impulse on the average delay. In this paper we observed the change from 0 to 18 percent in average message transmission delay of using non arranged delivery as compared to arranged delivery of data. In addition, there was a big changed in between different tests results, which often made the impact of HOLB numerically import.

According to Chen Hui [18] Stream Control Transmission Protocol is appropriate for satellite transmission, because it has an augmented and valuable property of multi-homing and multi-streaming. But on the other hand the satellite networks have some characteristics those are not valuable for good data communications those included the large propagation delays, large corruption losses and bandwidth product. Due to this above mentioned characteristics, these transport protocols (TCP, UDP, and SCTP) doesn't perform good over the satellite network links. In this paper authors proposed a new technique including a new congestion control method and load sharing technique which flourishes the efficiency of SCTP over satellite networks. The results of this proposed technique is achieved by implemented it in the NS with an SCTP module. The analysis of this simulation gives a look that the efficiency of SCTP is improved.

According to Armando L. Caro Jr et al [20], this paper is about the impotency of congestion control algorithms of Stream Control Transmission Protocol. This aggrades its performance due to more than one packet lost in a single window. A New Reno SCTP is a SCTP variant introduced three congestion control algorithms. First we changed the HTNA algorithm of SCTP, which guaranteed there have no any delay in fast retransmits. Second, a Fast Recovery mechanism that resemble to that of New-Reno TCP is integrated to save multiple congestion window shortening in a single RTT. Third and last in which the author introduces a new technique that prevent congestion window to be enhanced in the Fast Recovery. Experimental results show that New Reno SCTP performs better and it fit to Additively Increase Multiplicative Decrease rules. It also compares with two variants of SCTP with TCP SACK and New Reno TCP under different traffic conditions. Author shows that New-Reno SCTP functions significantly better than New-Reno TCP, maintains unchangeable behavior related to TCP SACK, and is as vigorous to multiple shortening in a window.

According to Rajesh Rajamani et al [11], this paper is related to the performance of transport protocols for web based data. Hyper Text Transfer Protocol (HTTP) protocol is used for this purpose. In this data is taken from the web server on the request of the clients. Due to the message-oriented and reliable nature SCTP was very suitable to transport PSTN signals over IP. SCTP removes the idea of data transmission streams from an association that has shared characteristics to that of a TCP. The data passed over these streams are in sequenced form. The data transmissions from different streams are partially sequenced over the association and it can lesser the delay due to the hindrance of HOL blocking.

According to Jinyang Shil[13], this paper represents the performance of SCTP in Wireless multi-access networks and also proposes an effective SCTP load sharing improvements. The performance of SCTP has examined in multi-access scenarios with suppositional analyses and simulations have performed in NS and Linux-kernel experiments. As we know SCTP having a useful multi-homing technique and the transport layer remedy for multi-access can chose to the infect network conditions, SCTP is capable to bring up throughput for the synchronous multi-access, execute seamless behavior during the straight up handover between dissimilar networks, and provide better strongly and connectivity than other remedy in the wireless multi-access scenarios.

In this paper author ISHTIAQ Ahmed et al [12], discuss about the performance of SCTP an emerging transport protocol, which integrated good properties of TCP and UDP. SCTP is a message oriented, reliable protocol providing multi-homing and multi-streaming as well. The congestion control techniques of SCTP are more or less relevant to that of Transmission Control Protocol. The performance of SCTP over the satellite links and internet is improved as relevant with TCP. The congestion control technique of SCTP over high latency broadband networks required more elegance if more than one packet losses on a data transmission link. Here author introduced a new congestion control mechanism for SCTP and measure its performance.

In this paper Andreas Jungmaier et al [15] discuss the efficiency of SCTP in wide area network. It is used for the movement of Public Switching Telephone Network (PSTN) signaling information over an IP network. In this paper the authors explain SCTP and its implementation. Moreover, they examine how the protocol works in a wide area network, particularly when compared it with TCP. The desire outcomes were gained in a test-bed comprising of two local networks which are interconnected through a competitor of a WAN.

2.1.3 Transport Protocol in MPLS Network

In this paper M. Saeed Akbar et al [16], present the experimental results of TCP variants in MPLS with consideration on TCP Reno, TCP Vegas and TCP Tahoe. It has been proved that TCP supports reliable data communication under different network conditions. Different variations of TCP show variable degree of flexibility in IP networks. MPLS traffic engineering methods has potency to supplying the services of QoS. Experimental analysis shows that Reno and Tahoe perform worse while TCP Vegas shows good results after a short duration variable delays in the initial periods of data transmission. The fixed delay in MPLS makes TCP Vegas a desired protocol for medium level networks.

2.2 Limitations of Previous Work.

The study of this literature shows that SCTP performs better as compared to the TCP and its variants implemented in different network environments. But problem is that there exist no any scenario of transport protocols especially SCTP implemented in MPLS network environment .Now we check the efficiency of SCTP and TCP New Reno and TCP Vegas in MPLS network by using different quality of service parameters especially delay, throughput, and channel utilization and packet delivery. We hypothesize that STCP performs better in MPLS network as compared to the other networks by using above mentioned parameters.

Richard J. La [26] discusses the few issues of TCP Vegas that have a serious influence on the functionality of this protocol.

a) Rerouting.
b) Stability
c) Low retransmission of packets.

The last contradictory property is analyzed also by Mo et al. [27]. They show that due to the assertive nature of TCP Reno, when size of buffer are greater, then TCP loses to TCP Reno that cover up the available buffer space that forces the TCP Vegas to move back.

3. Requirement Analysis

3.1 Problem Definition

MPLS networks are introduced to get better QoS in real IP network. Traffic engineering is one of such technology used for this purpose. Transport protocols like TCP (and its variants) and SCTP include congestion control techniques that depend on the network status, whereas UDP blasts away as fast as specified. Performance of TCP Reno, TCP Tahoe and TCP Vegas on IP-MPLS networks has analyzed [2]. With different traffic load parameters, it has been experienced that TCP Reno and Tahoe failed to take benefits of MPLS features where as TCP Vegas has shown promising results.

After evaluating these three variants of TCP, no further research is made to test latest flavor of the TCP and other emerging transport layer protocol. TCP variants like New Reno can perform better as compared to TCP Vegas in some situations. Moreover a SCTP which is relative to TCP and also have many extra features as compared to TCP variants. So SCTP can be a good choice for MPLS network as compared to three evaluated TCP variants. With traffic engineering it looks promising to study the performance achieved by TCP, its variants as well as SCTP.

There are few issues of TCP Vegas that have a serious influence on the functionality of this protocol.

a) Rerouting.

b) Stability

c) Low retransmission of packets.

3.2 Proposed Work

We evaluate the performance of transport protocols like TCP New Reno, TCP Vegas and SCTP in IP/MPLS networks. This scenario can be seen from under depicted TCP /IP Protocol architecture.

The current and proposed scenario of the Transport layer has given below. These figures are only the pictorial representation of my research works. If we see in figure (a), in current scenario, it shows the previous work done on three variants of TCP those are TCP Reno, TCP Tahoe, and TCP Vegas in IP/MPLS environment. In current scenario analysis of TCP variants have been performed to their utilization under different traffic scenarios. The behaviors of TCP Tahoe, TCP Reno and TCP Vegas have been studied under different traffics. It has been analyzed that TCP Tahoe and TCP Reno fails to take advantage of MPLS features, while TCP Vegas exhibit the best performance as compared to other analyzed flavors of TCP

But in proposed scenario we evaluate TCP New Reno, TCP Vegas and SCTP a new emerged transport layer protocols in figure 2(b). We intend to include TCP New Reno and new emerging transport protocol Stream Control Transmission Protocol. As we know that SCTP was designed with limitations of TCP (like Head-of-Line blocking, TCP data streaming, TCP connection failure, TCP SYN flood attacks and Address shortcoming in TCP) in mind. Several studies have been carried out for comparisons of SCTP with variants of TCP. It can also be studied that with traffic engineering, protocols like TCP and SCTP able to cover up sending rates in comparison to UDP and still being internet traffic friendly. Speed of SCTP can be a good choice for MPLS as compared to three evaluated TCP variants.

Current Scenario	**Proposed Scenario**
Application layer	Application layer
Transport layer (TCP Tahoe, TCP Reno, TCP Vegas)	Transport layer (TCP New Reno, TCP Vegas and SCTP)
Internet layer	Internet layer
Network access layer	Network access layer
Physical layer	Physical layer

Table 3.1: Current and proposed scenario.

We have detailed comparison of SCTP with other transport layer protocols like TCP and UDP, from which we easily analysis that why we take SCTP protocol for our problem scenario.

3.3 SCTP Features

The IETF group developed the SCTP locate the limitations in TCP and UDP. From the given table we can easily analysis the benefits of SCTP over TCP and UDP [9].

Sr.No.	Features	SCTP	TCP	UDP
1	Allow half closed connections	Not	Yes	N/A
2	Application PDU bundling	Yes	Yes	No
3	Application PDU fragmentation	Yes	Yes	No
4	Congestion Control	Yes	Yes	No
5	Connection oriented	Yes	Yes	No
6	Explicit Congestion Notification (ECN) Support	Yes	Yes	No
7	Flow Control	Yes	Yes	No
8	Full Duplex data transmission	Yes	Yes	Yes
9	Multi-homing	Yes	No	No
10	Multi-streaming	Yes	No	No
11	Ordered data delivery	Yes	Yes	No
12	Unordered data delivery	Yes	No	Yes
13	Partial reliable data transfer	Optional	No	No
14	Path Maximum Transmission Unit discovery	Yes	Yes	No
15	Preservation message boundaries	Yes	No	Yes
16	Protect against SYN flooding Attacks	Yes	No	N/A
17	Pseudo header for checksum	Uses Vtag	Yes	Yes
18	Reach-ability Check	Yes	Yes	No
19	Reliable data transfer	Yes	Yes	No
20	Selective Acknowledgements	Yes	Optional	No
21	Time wait state	For Vtag	For 4-tuple	N/A
22	4-way handshake	Yes	No	No
23	Message Oriented	Yes	No	

Table 3.1: Comparison among transport protocols [9]

4. System Design and Methodology

4.1 Simulation Architecture

In given below proposed system is describe by using the flow diagram.

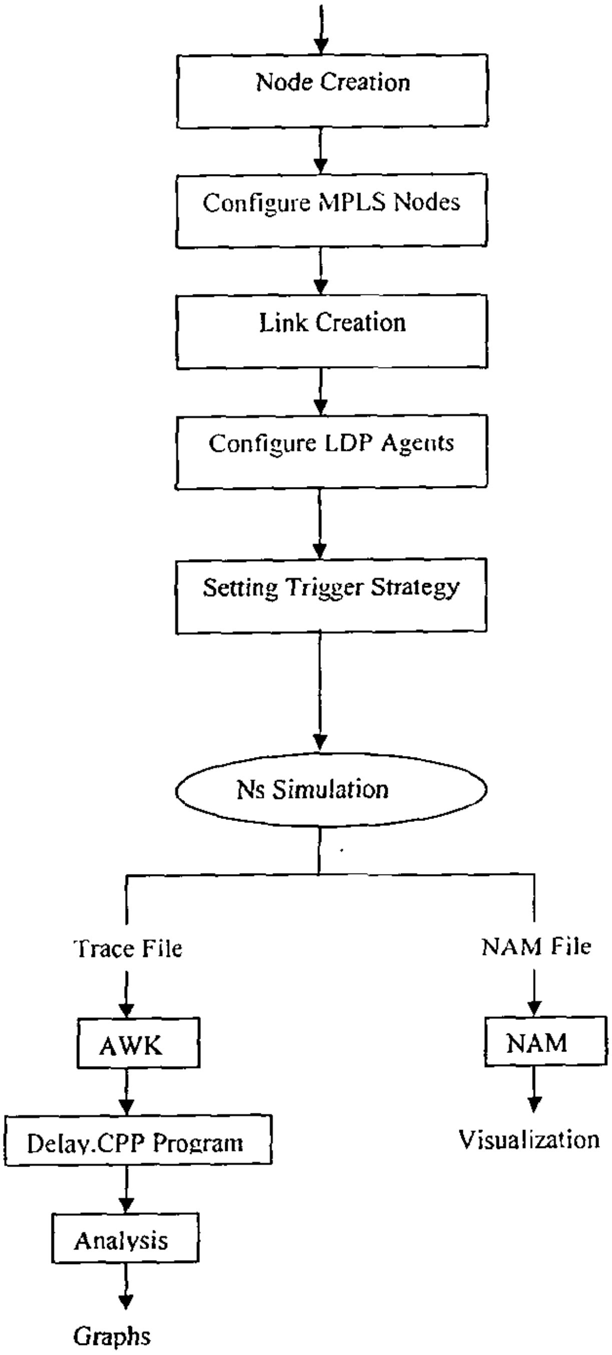

Figure 4.1: Block diagram of implementation

Figure 4.1 describes the flow of system design; in this process first the node is created, after this MPLS is configured on these nodes. After this the link is created between these links, and then the LDP agent is configured on these nodes, after this process the triggering strategies are implemented on these links. The whole process goes to the network simulator that divides it into two files; those are NAM file and Trace file. Experimental results have fetched from the Trace files by using the AWK scripts, then analysis this data and implemented them in the form of graphs. The NAM file is used for visualization of this topology in pictorial form.

4.2 Simulation Topology

Figure 4.2 illustrates the topology diagram of basic MPLS functions and IP network for the experimental situation of this simulation. For purpose of label distribution among the nodes, MPLS uses the label distribution protocols. This topology comprise of 15 nodes, those divides into further two domains those are IP and MPLS. The area of MPLS enable nodes are comprised of 11 nodes, those are listed as from LSR2 to LSR11. The IP domain consists of 4 nodes, labeled as Node0, Node1, Node13 and Node14 are connecting with ingress and egress router respectively and those connected with the MPLS domain. In the topology src0 agent attached to IPNode0 and src1 agent is attached to IPNode1 acts as sender node while dst0 agent is attached with IPNode13 and dst1 agent is attached to IPNode14 act as receiver node. Packet forwarding scheme in this topology is based on the distance vector routing protocol, all the packets are travel on the two designated routes to investigate the throughput, average delay, channel wastage and packet delivery ratio of SCTP, TCP New Reno and TCP Vegas on the MPLS network with variable bandwidth and packet size of 1000bytes. During the different experiments, channel has variable bandwidth. The traffic types FTP and CBR are used here with the 1000bytes packet size and with 500 seconds of simulation time with and with out rerouting scenarios.

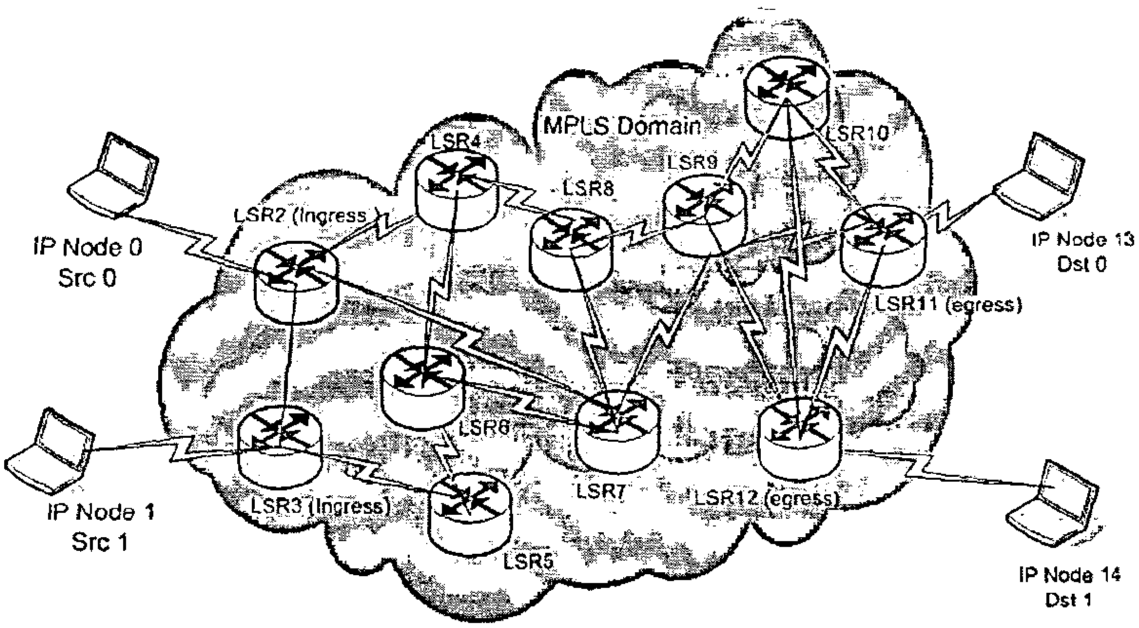

Figure 4.2: Proposed scenario of the implementation

4.3 Layered and Simulation Parameters

For the measurement of three transport protocols in IP/MPLS network environment, different layered and simulation parameters are used.

4.3.1 Layered Parameters

Here we describe the specification of different OSI layers used in our simulation environment.

Physical Layer: MPLS routers are used for physically connectivity of interfaces.

Data Link Layer: It handles the issues related to data exchanged over network devices. The detection and removal of wrongness faced by the physical layer is also the responsibility of this layer.

Network Layer: IP is used as internet layer protocol. Layer 3 routing protocol DSV is used for routing of traffic on different paths as the requirement of the situation during and before the link breakage.

Transport Layer: transport layer protocols are the major part of my thesis. I have used TCP variants like TCP New Reno, TCP Vegas and SCTP.

Application Layer: Here FTP and CBR are used as traffic type with packet size of 1000 bytes with variable bandwidth.

4.3.2 Simulation Parameters

Given below table 4.1 gives a looks on the parameters used in this simulated environment.

Parameter Name	Values
Traffic Type	FTP and CBR
Bandwidth	Vary from 1 to 8 MB
Packet Size	1000 bytes
Simulation Time	500 Sec
Link Type	Full duplex
No of IP Nodes	4
No of MPLS Nodes	11
Routing Protocol	DV
Triggering Strategy	Control Driven
Agent Type	LDP

Table 4.1: Simulation Parameters

4.4 Traffic Model

The communications between the two nodes are mainly take place in FTP and CBR. For simulation results we used the FTP and CBR traffics running over the TCP connection. These traffic types creates the TCL scripts existed in network simulator used to generate the packets.

4.5 Performance Metrics

We have to choose the following parameters to check the performance of a network.

4.5.1 Throughput

Throughput can be defined as the amount of data transferred from source to destination in unit time interval. It can also be defined as average rate of successfully delivery of data over a link. It can be calculated as by dividing the transfer size of window with the transfer time to get the throughput in mega, kilo or simply bits per second.

Throughput = Transfer size / Transfer time

4.5.2 Delay

The difference in travel times of different rate of occurrences of values in a signal. If the frequencies reach their destinations at different times, the signal distortion and errors can be occurred.

$$Mean\,Delay = \frac{\sum_{i=1}^{n} Delay\,Of\;Packet}{n}$$

Delay of Packet Transfer

Time from the packet is transmitted to the time the packet is received called delay of packet.

Packet Transfer delay consists of four following parts:

1. **Processing Delay:**

It can be defines as the time taken by the routers to process the header of packet. It determines output channel and check error of bits.

2. **Queuing Delay :**

It can be defined as the time taken by the packets sit in routing queues. It relay on congestion of router and duration for transmission of output channel.

3. **Transmission Delay:**

The time takes to push the packet's bits onto the channel. It can be calculated by this formula.

Transmission delay = L/R

4. **Propagation Delay:**

Time taken by the signals to pass through the channel path is called propagation delay. Mathematically it is written as

Propagation delay = d/s

4.5.2 Channel Utilization

Channel utilization can be defined as a link in the network is loaded or not.

We can be calculated as under.

Utilization % = (data in bits x 100) / (bandwidth x time interval)

5. Implementation

We have carried out our simulation, to analyze the performance of transport protocols using different performance parameters in different scenarios using NS-2 (version 2.27) [24] simulator.

5.1 NS-2 Simulators

NS-2 is a simulation tool used to simulate different network protocols using object oriented technique. NS is implemented, written in C++, at the front end OTcl parser is used. The simulator contains a chain of class hierarchy in C++ and a similar chain of class hierarchy is followed in the OTcl parser. From the user prospective, two hierarchies have a close relation with each other, in interpret and compile hierarchy one to one correspondence is there. The root of hierarchy is Tcl object. As for as end user is concerned the end user have to create new objects in the predictor; the new objects are surrounded within the predictor, which are in compiled hierarchy are mirror by relevant object.

To work for two different tasks, NS has two languages, first to deal with simulation of protocols it required a system language to work with bytes, packet headers and after powerful processing these are put into algorithm to run over bulky data. Run time for these tasks is more significant than turnaround time. Second in the network simulation scenarios require quick configuration of some parameters in these situations iteration time is more considerable than run time. So NS provides the structure in which we can run simulations for real time networks for analysis of different scenarios and different parameters.

Some people are still confused that why NS uses two language. The answer is very clear simply said "OTcl is used only one time to set different parameters like delay, queuing etc but if you want some special to do rather than existing parameters that are supported by OTcl then you will require the use of C++ which will permit you to create new objects". That's why two languages are used by NS. There is a large amount of classes defined in ns-2. Out of which six classes are more frequently used in ns:

1. Tcl
2. TclObject

3. TclClass
4. TclCommand
5. EmbeddedTcl
6. InstVar

Figure 5.1 describes the simplified view from user perspective.

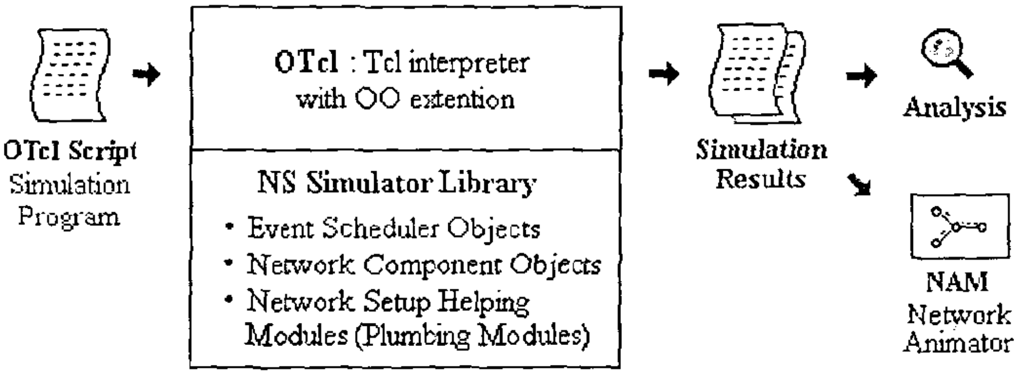

Figure 5.1: Simplified user's view of NS-2 [24]

5.1.1 TCL interpreter:

Ns-2 uses two languages which are entirely different from each other to make the two languages understandable for each other some sort of parser is required which could make possible the communication of the two languages. TCL interpreter is used for this purpose Tcl is used between the communication of OTcl and C++. Toolkit command scripts are designed to solve the different topologies. The objects in C++ that do not need to be controlled in a simulation or internally used by another object do not need to be linked to OTcl. Likewise, an object (not in the data path) can be entirely implemented in OTcl. Figure [4.2] shows an object hierarchy example in C++ and OTcl. One thing to note in the figure is that for C++ objects that have an OTcl linkage forming a hierarchy, there is a matching OTcl object hierarchy very similar to that of C++ as shown in fig 5.2.

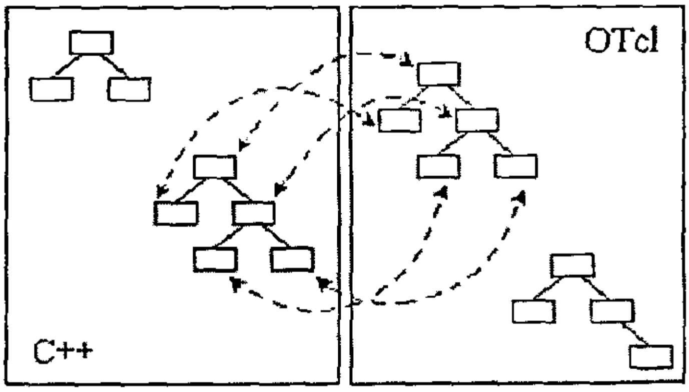

Figure 5.2: C++ and OTcl: The duality [24]

5.1.2 Network Animator (NAM)

It is used for graphical visualization of different network scenarios which are created by user. It provides the visual images of packet flows with different colors, node movements, packet queues, link between nodes, wireless nodes transmission range, drop packets and etc. NAM is a tool to visualize the imitations and traces for real world packets which is based on Tcl/TK. The theme to build NAM was to take the imitation and traces so that these results could be used in different visualization situations. The NAM when runs generate a file which could be used later if required. The advantage of NAM is that the generated file is of some significant size for some large simulations.

Trace file is required before ones can use NAM to visualize the simulations. More often trace files produced by NS, however many application can generate NAM trace file. When one run the NAM file one can see the topology design flow of packets in different direction depending upon the topology packets which are dropped due to some reasons can also be visualized from NAM visualization. All this can be seen in a separate window.

Following OTcl codes are used to set node attributes, these are methods of the class Node:

```
$node color [color]          ; # sets the color of node
$node shape [shape]          ; # sets the shape of node
$node label [label]          ; # sets the node label
```

```
$node label-color [lcolor]                  ; # sets the color of label
$node label-at [ldirection]                 ; # sets the position of label
$node add-mark [name] [color] [shape]       ; # mark adds to node
$node delete-mark [name]                    ; # delete the mark from node
```

5.1.3 User Interface

When starting with network animator, firstly it will create the NAM console window as shown in figure 5.3. You can have more than one animations running under the one NAM instance. At the top of NAM windows, there is a menu bar. That have 'File' and 'Help' menus. Under the 'File' menu, a command of 'New' used for creating a ns topology using this NAM editor , and also have an 'Open' command which permit you to open existing trace files, and a 'WinList' command that popup a window, that have the list of all recently opened trace files, and a 'Quit' command which close NAM file. The 'Help' menu have a very small number of popup help screen and a command to show copyright and version information [24].

When a trace file loads into NAM, an animation window will show. It has a 'Save layout' command which saves the existent layout into a file and a 'Print' command which allow prints the current layout.

The 'Views' menu has 4 buttons [24] *as shown in fig 5.3:*

- **New view button:** It creates a new display view of this current animation. User can also can scroll and magnify on the new view.
- **Show monitors checkbox:** If this box is checked, then it will display a pane at the bottom end of window, where monitors will be show their display.
- **Show auto layout checkbox:** In case of checked box is checked, then it will display a pane at the lower end of this window, which has the input boxes and an automatic layout adjustments button.
- **Show annotation checkbox:** If this box is checked, it will displays a list box at the lower side of this window, which use to list annotations in the moving upward direction of time.

- Under the menu bar, a control bar is existed, that contains 6 buttons, a small scale and labels.
- **Button 1 («) - Rewind**. When this button clicked, the animation time will go backward at the ratio of 25 times the update time of screen.
- **Button 2 (<) - Backward play.** When this button clicked, animation will be played towards back with the decreasing of time.
- **Button 3 (square) - Stop**. When this button clicked, the animation will pause.
- **Button 4 (>) - Forward play.** When this button is clicked, the animation will be played forward with increasing of time.
- **Button 5 (») - Fast Forward.** When this button is clicked, animation time will increases at the rate of 25 times opened screen update rate.
- **Button 6 (Chevron logo)** - Close the animation window.

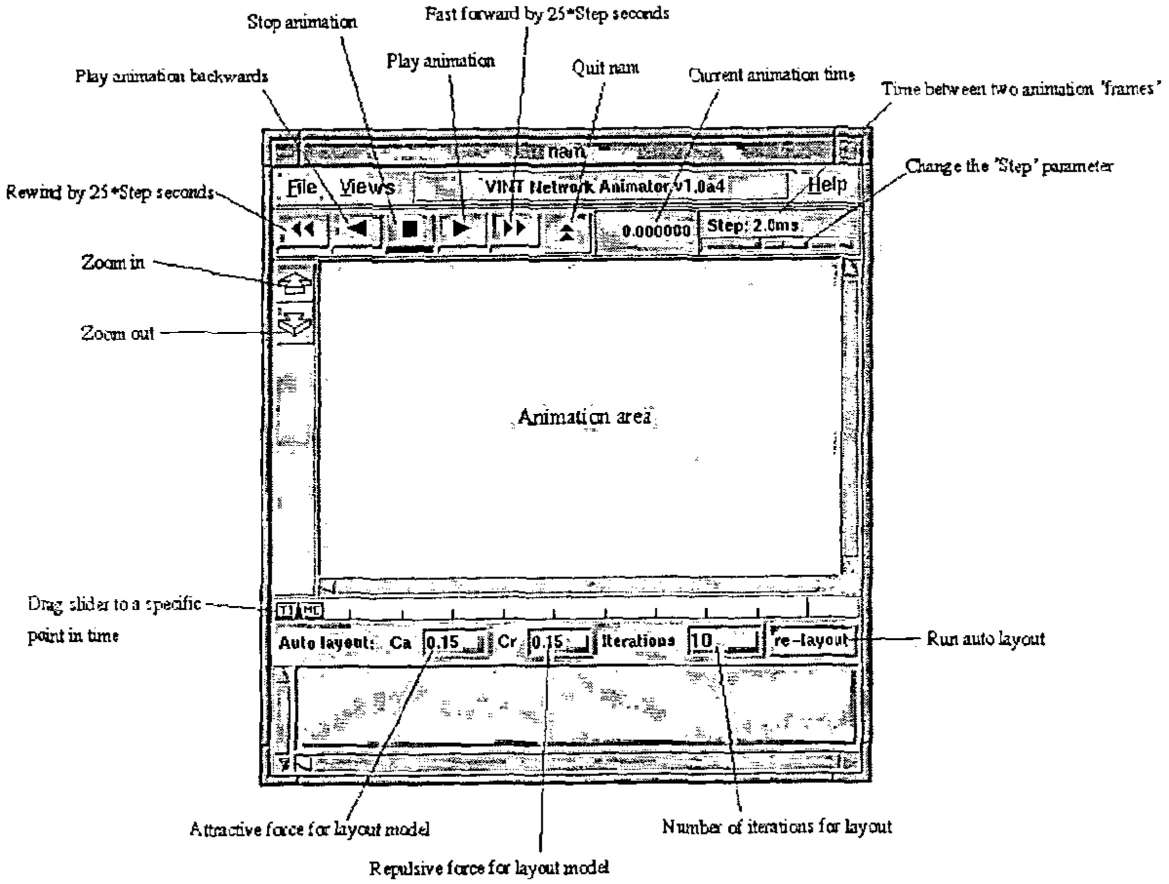
Stop animation
Fast forward by 25*Step seconds
Play animation
Quit nam
Current animation time
Time between two animation 'frames'
Play animation backwards
Change the 'Step' parameter
Rewind by 25*Step seconds
Zoom in
Zoom out
nam
File
Views
VINT Network Animator v1.0a4
Help
0.000000
Step: 2.0ms
Animation area
Drag slider to a specific point in time
Auto layout: Ca 0.15 Cr 0.15 Iterations 10 re-layout
Run auto layout
Attractive force for layout model
Repulsive force for layout model
Number of iterations for layout

Figure 5.3: NS-2 user interface [24]

Time label – it show the time of current animation

Rate Slider – it manages the screen update rate. The current rate is displayed in the label upward the slider.

Under the control bar, there is main display that possesses a main view pane with two panning scroll bars and tool bar. All the views are created by menu command 'Views/New view' will have three components. The tool bar consists of two zoom buttons. For zoom in, a button with an up arrow is used, and button with a down arrow is specifying for zooms out. The two scroll bars are used to contain the main animation view. The main view pane will appears an information window by clicking the left button on any of the objects. There is a 'display' button in the appeared window, for packet and agent objects. Clicking that button will take out the monitor pane, and add a monitor to that object. There will be a button 'Graph' for link object, clicking on that button will appear another window. Currently we have only packets and agents have monitors. A packet monitor only displays the id, size, and sent time. When the packet arrives at its destination side, but the monitor will arises a message that the packet is invisible. An agent monitor displays the name of the agent, and variable traces if they associated with this agent [24].

Under the monitor pane there is a time slider. It looks like a scale ruler, with a label 'TIME' which can be protracted along the ruler, used to set the simulation time. The left border of the slider shows the initial event time in the trace file and the right border shows the final event time. Same effect as Rewind or Fast Forward will show by clicking the left button on the ruler, but it depending on the clicking position of ruler. Automatic layout pane may be hidden or visible. If visible, it is under the time slider. It has one relay out button and three input boxes. The labeled input boxes let user to arrange the number of iterations and two automatic layout constants during next layout. When user press Enter or click the 'relay out' button, then that number of iterations will be accomplished. The lower parts of the NAM window is an annotation list box, where annotations are shown. By clicking on marginalia in the list box will brings NAM to the time when it is recorded. When pointer is in the list box, it will stop the animation by clicking the right button and carrying a pop up menu with three options buttons: Info, Delete, and Add [24].

5.1.4 Trace Data Analyzers

There are number of ways to analyze trace file generated from Network Simulation. There are following four methods that are mostly to examine the trace file.

1) XGraph

It is an X-Windows application that contains the derivatives, animations, interactive graphing and plotting.

To use XGraph in Network Simulator-2 the executable file can be called within a TCL Script. Then it loads a graph showing the information visually of the trace file produced from the simulation.

2) Trace Graph

It is used to analyze the trace file that runs under Windows, UNIX, Linux systems and it requires Matlab 6.0 or higher version. Trace graph supports the following trace file formats.

- Wired
- Satellite
- Wireless

3) AWK Scripts with Microsoft word

It is shell scripting language that derives data from trace file according to the need of user and formulates these extracted data. Then Microsoft Excel draw graphs according to data provided. It can support any trace file format.

4) User built-in code

In this method a user builds its own code to extract, compute data and draws it into the graphical format. This code can be created in any programming language like C++ and java. It can support any trace file format.

5.2 Characteristics of NS-2

NS-2 can perform the following features in simulations [24].

1) It gives Router queue Management Techniques DropTail, RED, CBQ,

2) It gives the feather of multicasting

3) We can develop the Simulation of wireless networks.

- Developed by Sun Microsystems + UC Berkeley (Daedalus Project).
- IEEE 802.11 can be simulated, Ad-hoc protocols such as DSDV, TORA, DSR and AODV and Mobile IP.
- Terrestrial (cellular, ad-hoc, GPRS, WLAN, BLUETOOTH), satellite.

4) It provides the Traffic Source Behaviors like WWW, VBR, and CBR.

5) It provides the routing mechanism

6) It supports the Transport Agents like TCP and UDP.

7) Network topologies.

8) Packet flow mechanism

9) Applications- Telnet, FTP, Ping.

10) It provides the Tracing Packets on all or specific links.

5.3 Operating Systems for NS-2

NS can be used on the following platforms [24]:

- Linux (RedHat 9, Enterprise Edition, FEDORA 4 or above)
- UNIX (Free BSD, SunOS, Solaris).
- Microsoft Windows platform with Cygwing emulator.

5.4 Potential Benefits

1) **Economy and ease of installation** are important benefit while using NS-2 simulations. Because we can not physically implements the network scenarios because it requires a lot of resources for simulations

2) **Speed** is also another important factor, which strengthens us to use the NS-2. Because physical simulation take much time for work processing. And changes in NS-2 are also faster and easier than real world scenario.

3) **Low space** is requires as relative to physical networks. Because in physical networks, a lot of machines, other network components and power cables are involves that requires a lot of space, while in simulation scenarios, one have to only installed network simulator on a system.

4) **Open source and free software:** There are also other simulators exist like OPNET, which is very costly as compared to NS-2. The const of research version of OPNET is more than Rs. 320000. While NS-2 has not any cost and freely available on Internet.

5.5 Limitations

As we study the benefits of the Network Simulator, but it also has some limitations, those are given as under [24].

1) NS-2 offers interesting features but it is difficult to work in NS-2 environment for new user.
2) As the NS–2 is requires a vast memory, so there is lots of problem arises during the simulation of large network and as the number of nodes are increases, the processing time of simulation is also increases.
3) We have significant confidence in NS-2; it is not a complete product, but it is the consequence of a continuous attempt of development and research.
4) Users of NS-2 are responsible for verifying that their simulations are not disqualified by bugs.
5) Bugs in the NS–2 software are still being explored and removed.
6) Tolerate to debug NS-2 source code when necessary.
7) More complex simulations scenarios may need changes to NS source code, this is also difficult.
8) Debugging is very complicated process so there is quite knowledge of C++ and Otcl languages are required.

5.6 Implementation Detail

The following is the code written in TCL (Tool Command language) for construction the MPLS and IP network as described in the figure 4.3.

First we have to create a simulator object as,

```
set ns [new Simulator]
```

Then we have to open different trace files for the recording of different events during the whole simulation time.

```
set nf [open mpls.nam w]
$ns namtrace-all $nf
set f0 [open mpls.tr w]
$ns trace-all $f0
```

IP and MPLS node are created as follow.

```
#se MPLS nodes
set node0   [$ns node]
$node0 shape square
$node0 color blue
$node0 label Src0
set node1   [$ns node]
$node1 shape box
$node1 color red
$node1 label Src1
$ns node-config -MPLS ON
set LSR2   [$ns node]
set LSR3   [$ns node]
set LSR4   [$ns node]
set LSR5   [$ns node]
set LSR6   [$ns node]
set LSR7   [$ns node]
set LSR8   [$ns node]
set LSR9   [$ns node]
set LSR10   [$ns node]
set LSR11   [$ns node]
set LSR12   [$ns node]
$ns node-config -MPLS OFF
set node13  [$ns node]
```

```
$node13 shape box
$node13 color blue
$node13 label Dst0
set node14 [$ns node]
$node14 shape box
$node14 color red
$node14 label Dst1
```

The next lines create links between the nodes. Each node is connected with duplex link with bandwidth 1Mb to 8 Mb, delay of 10ms and a DropTail queue mechanism.

```
# Define links, bandwidth 8Mb, delay 10ms, queue management DropTail
$ns duplex-link $node0 $LSR2  8Mb 10ms DropTail
$ns duplex-link $node1 $LSR3  8Mb 10ms DropTail
$ns duplex-link $LSR2  $LSR3  8Mb 10ms DropTail
$ns duplex-link $LSR2  $LSR7  8Mb 10ms DropTail
$ns duplex-link $LSR2  $LSR4  8Mb 10ms DropTail
$ns duplex-link $LSR3  $LSR5  8Mb 10ms DropTail
$ns duplex-link $LSR4  $LSR6  8Mb 10ms DropTail
$ns duplex-link $LSR4  $LSR8  8Mb 10ms DropTail
$ns duplex-link $LSR5  $LSR6  8Mb 10ms DropTail
$ns duplex-link $LSR6  $LSR7  8Mb 10ms DropTail
$ns duplex-link $LSR7  $LSR8  8Mb 10ms DropTail
$ns duplex-link $LSR7  $LSR9  8Mb 10ms DropTail
$ns duplex-link $LSR7  $LSR10  8Mb 10ms DropTail
$ns duplex-link $LSR8  $LSR9  8Mb 10ms DropTail
$ns duplex-link $LSR9  $LSR10  8Mb 10ms DropTail
$ns duplex-link $LSR9  $LSR12  8Mb 10ms DropTail
$ns duplex-link $LSR10 $LSR11  8Mb 10ms DropTail
$ns duplex-link $LSR10 $LSR12  8Mb 10ms DropTail
$ns duplex-link $LSR11 $LSR12  8Mb 10ms DropTail
$ns duplex-link $node13 $LSR11  8Mb 10ms DropTail
```

```
$ns duplex-link $node14 $LSR12  8Mb 10ms DropTail
```

The next lines configure the LDP agents on all MPLS nodes.

```
for {set i 2} {$i < 13} {incr i} {
        set a LSR$i
        for {set j [expr $i+2]} {$j < 13} {incr j} {
                set b LSR$j
                eval $ns LDP-peer $$a $$b
        }
        set m [eval $$a get-module "MPLS"]
        $m enable-reroute "new"
}
```

The next lines are the code for setting the triggering strategy for LSP establishment to control-driven trigger.

```
Classifier/Addr/MPLS set control_driven_ 1
Classifier/Addr/MPLS enable-on-demand
Classifier/Addr/MPLS enable-ordered-control
```

The next lines of the code specify the agent type and traffic type. We use three types of agents, SCTP and TCP New Reno and TCP Vegas. Traffic type is FTP and CBR and attaches it to the Node 0 and Node 1.

```
set sctp1 [new Agent/SCTP]
$ns attach-agent $node0 $sctp1
$sctp1 set dataChunkSize_ 968
$sctp1 set packetSize_ 968
$sctp1 set mtu_ 1000
$sctp1 set class_ 0

set sctpsink1 [new Agent/SCTP]
$sctpsink1 set useDelayedSacks_ 0
```

```
$ns attach-agent $node9 $sctpsink1

# connect both agents
$ns connect $sctp1 $sctpsink1

set ftp1 [new Application/Traffic/FTP]
$ftp1 set packetSize_ 1000
$ftp1 attach-agent $sctp1

$sctp2 set class_ 1
set sctpsink2 [new Agent/SCTP]
$sctpsink2 set useDelayedSacks_ 0
$ns attach-agent $node10 $sctpsink2
# connect both agents
$ns connect $sctp2 $sctpsink2
```

The next lines show the event scheduling

```
$ns at 1.0 "$cbr1 start"
$ns at 498.0 "$cbr1 stop"

$ns at 1.0 "$cbr2 start"
$ns at 498.0 "$cbr2 stop"

# Calls the procedure "finish"
$ns at 500.0 "finish"

# The last line finally starts the simulation
$ns run
```

The following lines are awk script run on the trace files and generate our required results.

```
exec awk {
    {
if (($1=="+" && $3==0 && $4==2 && $5=="sctp" && $6==1000 &&  $9=="0.0" &&
$10=="13.0" ) || ($1=="r" && $3==11 && $4==13 && $5=="sctp" && $6==1000 &&
$9=="0.0" && $10=="13.0" )) {

        print $1, $2, $12
    }
     }
      } mpls.tr > sendreceive1.tr
```

6. Testing and Performance Evaluation

6.1 Source Configuration

There are some points discussed for source configuration of the simulation scenarios.

1. Since TCP is used to establishment the link between the source and destination, SCTP configured to use only one stream.
2. Payload for protocol is 980 byte.
3. Here we take SCTP for ordered delivery of data.
4. Receiver window for protocol was set to the maximum allowed 65536 bytes.

6.2 Simulation Results

We have considered different scenarios for analysis of our simulation firstly we take

6.2.1 Case 1: with FTP Traffic

Traffic Type: FTP

Packet Size: 1000 bytes

Bandwidth: vary from 1 to 8 Mb.

6.2.1.1 Average Delay

As we run two session of traffic, one from src0 and second src1, we take the average delay by applying some calculations on these values.

Bandwidth	SCTP (sec)	TCP New Reno (sec)	TCP Vegas (sec)
1 Mb	0.5076	0.4774	0.1274
2 Mb	0.2541	0.2511	0.0924
3 Mb	0.1713	0.1767	0.0781
4 Mb	0.1427	0.1409	0.0691
5 Mb	0.1192	0.1179	0.0681
6 Mb	0.1233	0.1180	0.0932
7 Mb	0.0981	0.0936	0.0639
8 Mb	0.0791	0.0752	0.0633

Table 6.1: Average delay with FTP traffic

Table 6.1 shows the numerical values of average delay for SCTP, TCP New Reno and TCP Vegas protocols in seconds. Simulations are run with bandwidth from 1 to 8 MB for 500 seconds.

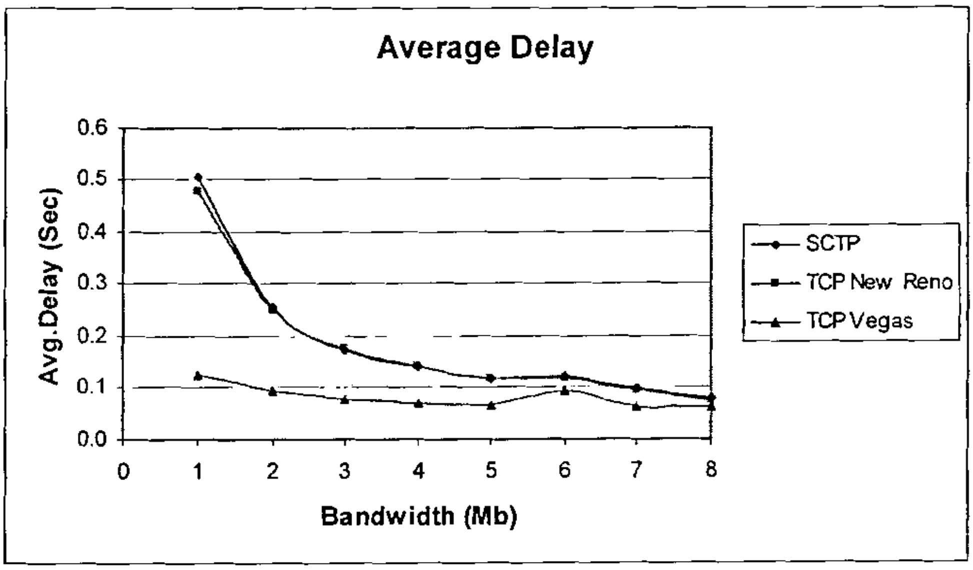

Figure 6.1: Average delay with FTP traffic

Figure 6.1 show average delay of SCTP, TCP New Reno and TCP Vegas with variable bandwidth and traffic type FTP. On x-axis we take the bandwidth values and on y-axis we place the average delay. These three protocols show different behavior at the initial bandwidth values, TCP Vegas has lower delay while SCTP and TCP New Reno has approximately same average delay values, as the value of bandwidth increases gradually the average delay of these three protocols decreases. Above mentioned graph shows that TCP Vegas has lower delay value as compared to SCTP and TCP New Reno.

6.2.1.2 Throughput

Table 6.2 shows the throughput values in kilo bits per second of three concerned protocols. From this table we easily analysis that which protocols has the higher throughput values.

Bandwidth	SCTP (kbps)	TCP New Reno (kbps)	TCP Vegas (kbps)
1 Mb	999.416	997.68	992.88
2 Mb	1990.864	1989.024	1985.44
3 Mb	2982.576	2983.232	2976.832
4 Mb	3975.232	3974.848	3965.616
5 Mb	4962.608	4959.264	4952.624
6 Mb	5955.84	5958.368	5946.336
7 Mb	6948.352	6951.104	6941.296
8 Mb	7940.336	7943.456	7926.432

Table 6.2: Throughput with FTP Traffic

Table 6.2 shows throughput values calculated by given below formula. It can be calculated by packet sending rate from source to destination divided to its transfer time.

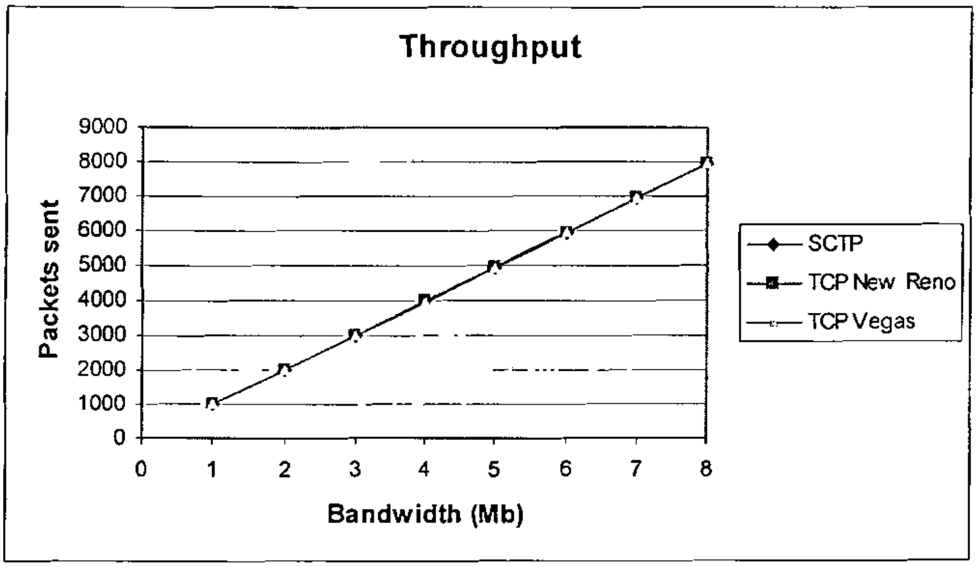

Figure 6.2: Throughput with FTP traffic.

Figure 6.2 shows the throughput analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type FTP. On x-axis we take the bandwidth values and on y-axis we place the total no. of packet sent from the source to destination. Figure shows that as the value of bandwidth increases, throughput is also increase. If we see the closer look with the help of above mentioned table, SCTP shows the higher throughput values. So after analyzing these three protocols, it has been observed that SCTP has higher throughput as compared to other two protocols.

6.2.1.3 Channel Wastage

Channel utilization of SCTP, TCP New Reno and TCP Vegas can calculate by simulation given as under.

Now from this formula we can calculate the channel wastage during data transmission.

Bandwidth	SCTP (%)	TCP New Reno (%)	TCP Vegas (%)
1 Mb	0.1584	0.2320	0.7120
2 Mb	0.4568	0.5488	0.7280
3 Mb	0.5808	0.5589	0.7722
4 Mb	0.6192	0.6288	0.8596
5 Mb	0.7478	0.8147	0.9475
6 Mb	0.7360	0.6938	0.8944
7 Mb	0.7378	0.6985	0.8386
8 Mb	0.7458	0.7068	0.9196

Table 6.3: Channel wastage with FTP traffic

Table 6.3 shows the numerical analysis of channel wastage of these three transport layer protocols. Simulations are run with bandwidth range from 1 to 8 Mb for 500 seconds.

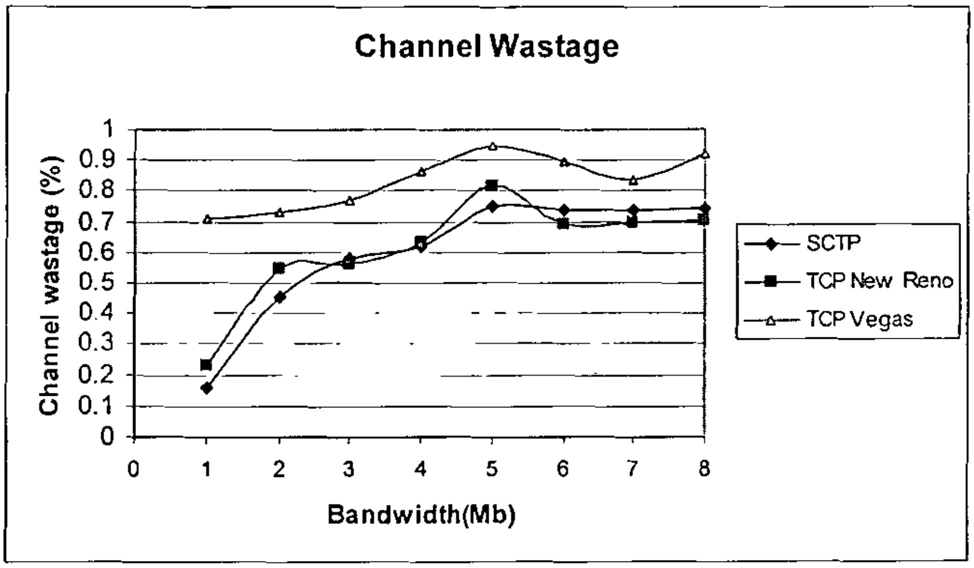

Figure 6.3: Channel wastage with FTP traffic.

Figure 6.3 shows the channel wastage analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type FTP. On x-axis we take the bandwidth values and on y-axis we place the channel wastage in percentage.

Figure shows that as the value of bandwidth increases the value of channel wastage is also increases. There is no big difference of channel wastage between TCP new Reno and SCTP. On the initial and middle stages SCTP has lower channel wastage, while on the final stages TCP New Reno has lower values. After analyzing this graph, we can say that TCP New Reno has performs better while TCP Vegas perform worse in the case of channel wastage.

6.2.2 Case 2: With CBR traffic

Now we have to change the traffic type with CBR (Constant Bit Rate) with packet size 1000 bytes and variable bandwidth from 1 to 8 Mb for the analysis of our simulation results.

6.2.2.1 Average Delay.

Table 6.4 shows the average delay of SCTP, TCP New Reno and TCP Vegas with variable bandwidth and CBR traffic type.

Bandwidth	SCTP (sec)	TCP New Reno (sec)	TCP Vegas (sec)
1 Mb	0.5057	0.4774	0.1274
2 Mb	0.2081	0.2511	0.0924
3 Mb	0.1490	0.1767	0.0781
4 Mb	0.0897	0.1409	0.0740
5 Mb	0.0690	0.1180	0.0644
6 Mb	0.0670	0.1180	0.0629
7 Mb	0.0660	0.0936	0.0618
8 Mb	0.0649	0.0752	0.0608

Table 6.4: Average delay with CBR traffic

Table 6.4 contains the statistical analysis of delay of three transport layer protocols. Simulations are run with bandwidth from 1 to 8 Mb for 500 seconds.

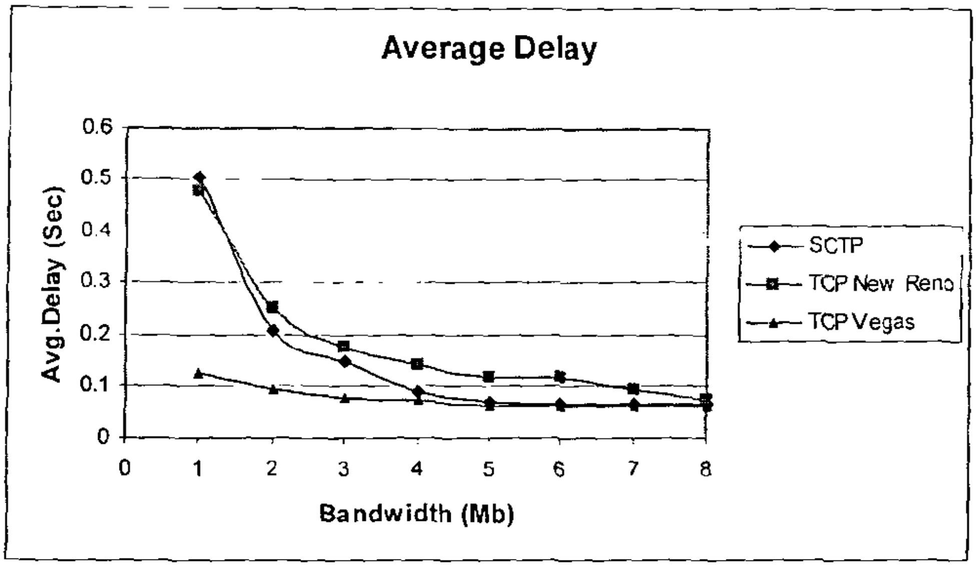

Figure 6.4: Average delay with CBR traffic

Figure 6.4 shows the average delay of SCTP, TCP New Reno and TCP Vegas with variable bandwidth and CBR traffic. On x-axis we take the bandwidth values and on y-axis we place the average delay. These three protocols show different behavior at the initial bandwidth values, TCP Vegas has lower delay while SCTP and TCP New Reno has different have average delay values, as the value of bandwidth increases gradually the average delay of these three protocols decreases. Above mentioned graph shows that TCP Vegas has lower delay as compared to SCTP while SCTP has lower average delay than TCP New Reno.

6.2.2.2 Throughput

Table 6.5 show the throughput values in kilo bits per second of three concerned protocols. From this, it can be easily analyzed that which protocols has the higher throughput.

Bandwidth	SCTP (kbps)	TCP New Reno (kbps)	TCP Vegas (kbps)
1 Mb	1000	1000	996.88
2 Mb	1975.856	1997.12	1993.44
3 Mb	2991.216	2995.104	2988.816
4 Mb	3987.52	3990.992	3974.864
5 Mb	4555.072	4978.976	4241.088
6 Mb	4625.008	5982.368	4241.088
7 Mb	4658.064	6979.088	4241.088
8 Mb	4698.112	7975.472	4241.088

Table 6.5: Throughput with CBR traffic

Table 6.5 contains the statistical analysis of throughput of three transport layer protocols. Simulations are run with bandwidth from 1 to 8 Mb for 500 seconds.

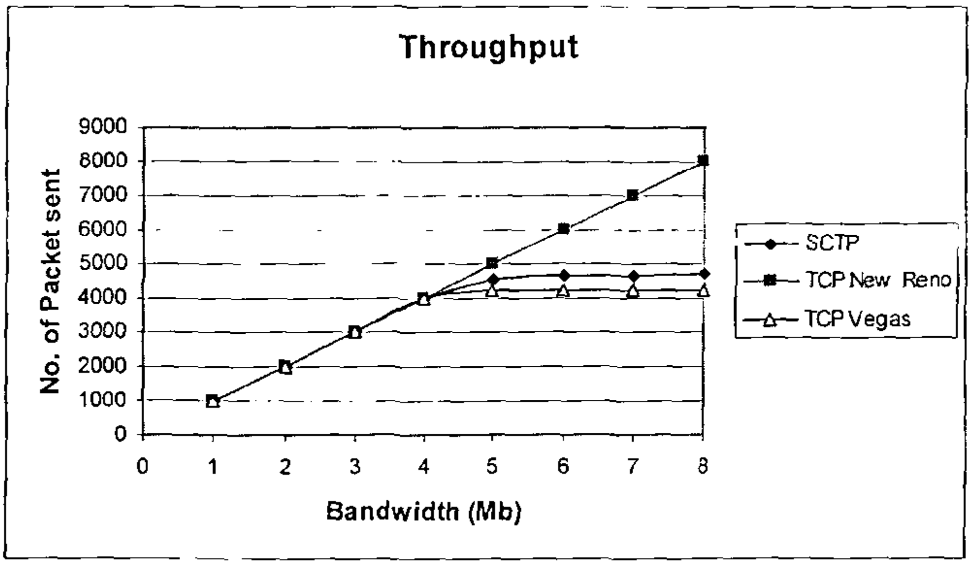

Figure 6.5: Throughput with CBR traffic.

Figure 6.5 shows the throughput analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and CBR traffic. On x-axis we take the bandwidth values and on y-axis we place the total no. of packet sent from the source towards destination. Figure shows that as the value of bandwidth increases, throughput is also increase. If we see on the above graph, as the value of bandwidth increase from 1 to 4 Mb, all these protocols have same throughout values but after the 4 Mb bandwidth the SCTP has increase with the same position while after 4Mb the values of TCP Vegas and TCP New Reno decreases till 8 Mb. So after analyzing these three protocols it has been observe that SCTP has higher throughput as compared to other two protocols.

6.2.2.3 Channel wastage

Channel utilization of SCTP, TCP New Reno and TCP Vegas with packet size 1000 bytes can be calculates by simulation given as under.

Bandwidth	SCTP (%)	TCP New Reno (%)	TCP Vegas (%)
1 Mb	0.0032	0.7936	0.3120
2 Mb	1.2072	0.1440	0.3280
3 Mb	0.2928	0.1632	0.3728
4 Mb	0.3120	0.2252	0.6284

5 Mb	8.8985	0.4204	15.1782
6 Mb	22.9165	0.2938	29.3152
7 Mb	33.4562	0.2987	39.4130
8 Mb	41.0377	0.3066	46.9864

Table 6.6: Channel wastage with CBR traffic

Table 6.6 contains the statistical analysis of channel wastage of three transport layer protocols. Simulations run with bandwidth 1 to 8 Mb for 500 sec.

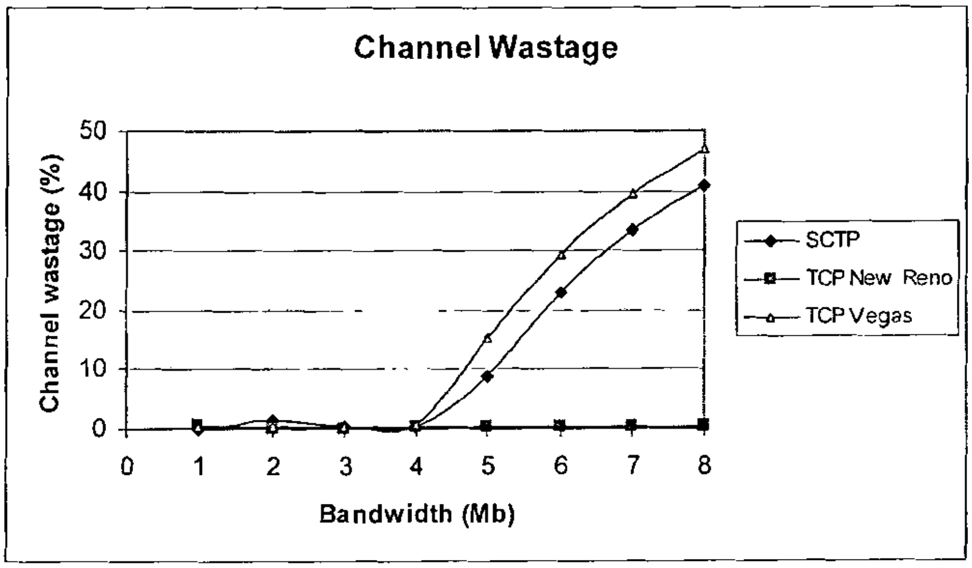

Figure 6.6: Channel wastage with CBR traffic.

Figure 6.6 shows percentage channel wastage analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type CBR. On x-axis we take the bandwidth values and on y-axis we place the channel wastage in percentage.

Above figure shows that as the value of bandwidth increases from 1 to 4 Mb, the value of channel wastage is 0% of these three protocols. But from 4 to 8 Mb bandwidth the channel wastage of TCP New Reno remains 0% whiles the values of SCTP and TCP Vegas goes upwards. TCP Vegas has higher channel wastage than SCTP and TCP New

Reno. After analyzing this graph we can say that TCP New Reno has performs better while TCP Vegas perform worse in the case of channel wastage.

6.2.3 Rerouting with FTP Traffic

When a link failed in a network, the traffic that uses the failed link must change its path to reach its destination: the data traffic rerouted from a primary path to a backup path. These paths can be fully disjoint or partially combined. Rerouting can be use in both circuit and packet switching networks. We take FTP traffic with packet size 1000 bytes and variable bandwidth.

6.2.3.1 Average Delay

As we run the two session of traffic one from src0 and second src1, we take the average delay by applying some calculations on these values.

Bandwidth	SCTP (sec)	TCP New Reno (sec)	TCP Vegas (sec)
1 Mb	0.5166	0.3954	0.1228
2 Mb	0.2301	0.1809	0.0918
3 Mb	0.1509	0.1166	0.0807
4 Mb	0.1258	0.0986	0.0738
5 Mb	0.1166	0.0933	0.0667
6 Mb	0.1135	0.0951	0.0649
7 Mb	0.1148	0.0852	0.0638
8 Mb	0.1172	0.0795	0.0655

Table 6.7: Average delay with rerouted FTP traffic

Table 6.7 contains the statistical analysis of average delay for SCTP, TCP New Reno and TCP Vegas protocols with variable bandwidth and FTP traffic with simulations time 500 seconds in case of rerouting.

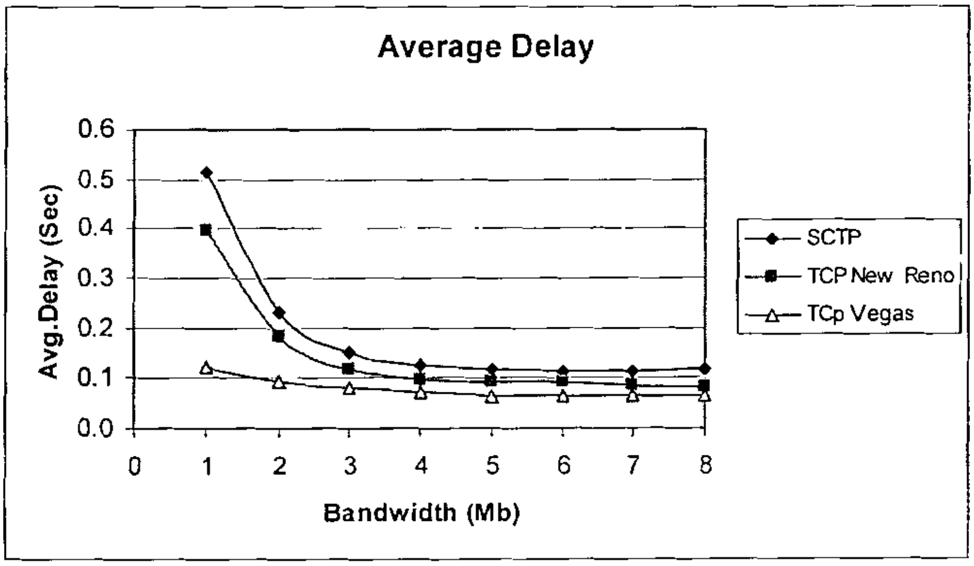

Figure 6.7: Average delay with rerouted FTP traffic.

Figure 6.7 gives a look of delay of SCTP, TCP New Reno and TCP Vegas with variable bandwidth and FTP traffic. On x-axis we take the bandwidth values and on y-axis we place the average delay in seconds. these three protocols show different behavior at the initial bandwidth values, TCP Vegas has lower delay while SCTP and TCP New Reno have different average delay values, as the value of bandwidth increases gradually the average delay of these three protocols is decreases. Above mentioned graph shows that TCP Vegas has lower delay value as compared to SCTP, TCP New Reno and SCTP has higher average delay than TCP New Reno.

6.2.3.2 Throughput

Table 6.8 shows the throughput values in kilo bits per second of three concern protocols. From this table we easily analysis that which protocols has the higher throughput values.

Bandwidth	SCTP (kbps)	TCP New Reno (kbps)	TCP Vegas (kbps)
1 Mb	861.856	847.632	348.288
2 Mb	1686.224	1582.08	728.864

3 Mb	2372.432	2244.16	548.224
4 Mb	2758.992	2633.248	743.344
5 Mb	2990.048	2847.808	1173.488
6 Mb	3190.384	3148.592	1287.472
7 Mb	3291.168	3271.296	1186.320
8 Mb	3483.008	3559.808	1096.640

Table 6.8: Throughput with rerouted FTP traffic

Table 6.8 shows the throughput values calculated by the given below formula. It can be calculate by the rate of packet send from source to destination with its transfer time in seconds of SCTP, TCP New Reno and TCP Vegas with variable bandwidth for throughput analysis.

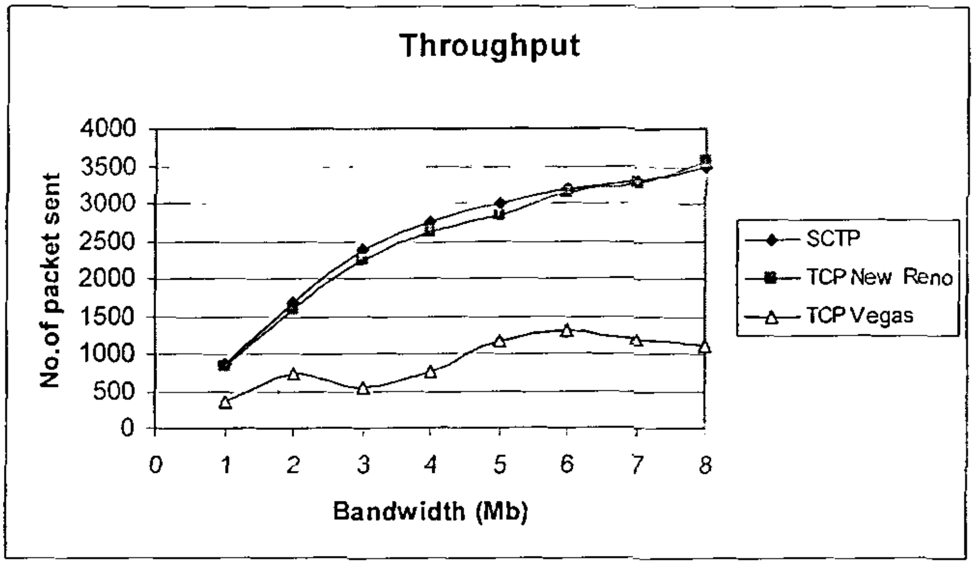

Figure 6.8: Throughput with rerouted FTP traffic.

Figure 6.8 shows the throughput analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type FTP. On x-axis we take the bandwidth values and on y-axis we place the total number of packet sent from the source to destination. Figure shows that as the value of bandwidth increases, throughput of SCTP and TCP New Reno is also increase while the throughput value of TCP Vegas is not increases with the same scenario. So after analyzing these three protocols it has been observe that SCTP has higher throughput as compared to other two protocols.

6.2.3.3 Channel Wastage

Channel utilization of SCTP, TCP New Reno and TCP Vegas can be calculated by simulation results given as under.

Now from this formula we can calculate the channel wastage during data transmission.

Bandwidth	SCTP (%)	TCP New Reno (%)	TCP Vegas (%)
1 Mb	13.8144	15.2368	65.1712
2 Mb	15.6888	20.8960	63.5568
3 Mb	20.9189	25.1946	81.7258
4 Mb	31.0252	34.1688	81.4164
5 Mb	40.1990	43.0438	76.5302
6 Mb	46.8269	47.5234	78.5421
7 Mb	49.8370	50.4312	82.5243
8 Mb	55.4624	55.5024	86.2920

Table 6.9: Channel wastage with rerouted FTP traffic

Table 6.9 contains the statistical analysis of channel wastage of three transport layer protocols. Simulations run with bandwidth from 1 to 8 Mb for 500 seconds.

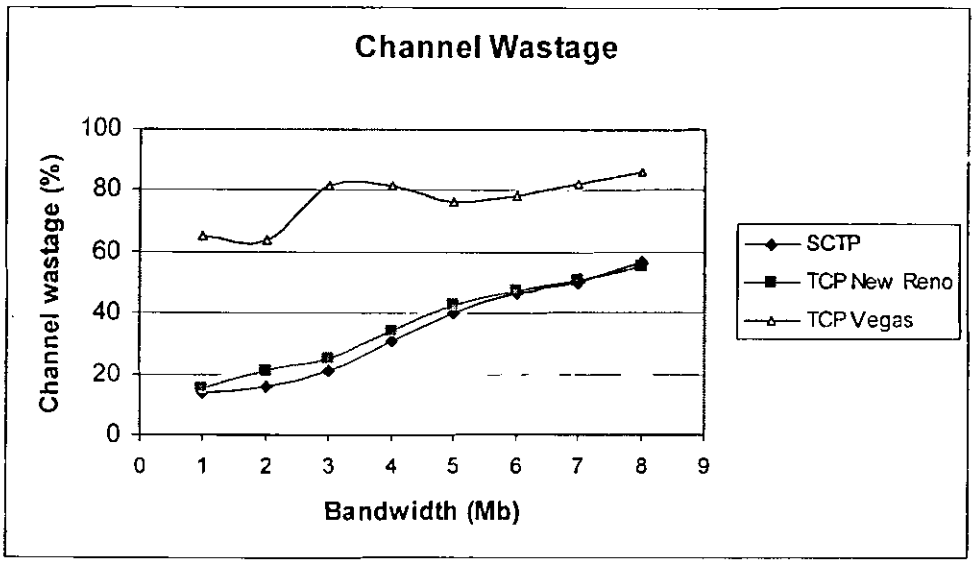

Figure 6.9: Channel wastage with rerouted FTP traffic.

Figure 6.9 shows the channel wastage analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type FTP. On x-axis we take the bandwidth values and on y-axis we place the channel wastage in percentage.

Above figure gives a look that, as the value of bandwidth goes higher the value of channel wastage is also increases. If bandwidth goes higher, the value of SCTP and TCP Vegas is gradually increases but the TCP Vegas has higher channel wastage than SCTP and TCP New Reno. After analyzing this graph we can say that SCTP has performs better in the case of Channel wastage.

6.2.4 Rerouting with CBR Traffic

We take CBR traffic with packet size 1000 bytes and variable bandwidth from 1 to 8 Mb.

6.24.1 Average Delay.

Table 6.10 shows the average delay of SCTP, TCP New Reno and TCP Vegas with variable bandwidth.

Bandwidth	SCTP (sec)	TCP New Reno (sec)	TCP Vegas (sec)
1 Mb	0.5148	0.3955	0.1228
2 Mb	0.2289	0.1812	0.0917
3 Mb	0.1487	0.1169	0.0807
4 Mb	0.1173	0.0988	0.0714
5 Mb	0.1091	0.0941	0.0689
6 Mb	0.1053	0.0952	0.0663
7 Mb	0.1034	0.0853	0.0651
8 Mb	0.1001	0.0795	0.0641

Table 6.10: Average delay with rerouted CBR traffic

Table 6.10 contains the statistical analysis of average delay for SCTP, TCP New Reno and TCP Vegas protocols with variable bandwidth and CBR traffic with simulations time 500 seconds in case of rerouting.

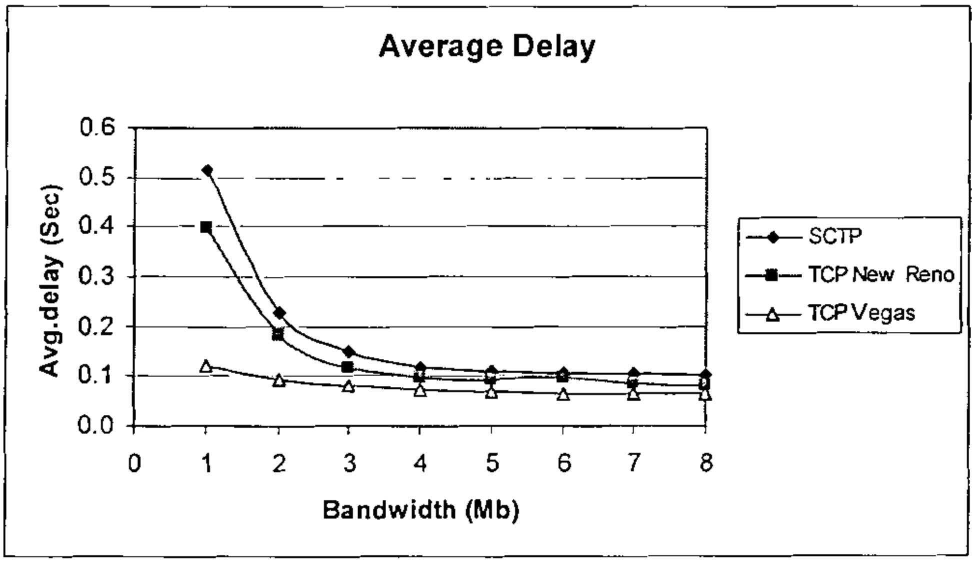

Figure 6.10: Average delay with rerouted CBR traffic

Figure 6.10 shows the average delay of SCTP, TCP New Reno and TCP Vegas with variable bandwidth and CBR traffic in case of rerouting. On x-axis we take the bandwidth values and on y-axis we place the average delay in seconds. These three protocols show different behavior till the 4Mb bandwidth values, but from 4Mb bandwidth value TCP Vegas has lower delay while SCTP and TCP New Reno have approximately same average delay values. If we see a closer look, then above mentioned graph shows that TCP Vegas has lower delay values as compared to SCTP while TCP New Reno has lower average delay than SCTP.

6.2.4.2 Throughput

Table 6.11 shows the throughput values in kilo bits per second of three concern protocols. From this it can be easily analyze that which protocols has the higher throughput values.

Bandwidth	SCTP (kbps)	TCP New Reno (kbps)	TCP Vegas (kbps)
1 Mb	870.672	851.568	352.288
2 Mb	1710.224	1589.424	736.864
3 Mb	2416.272	2256.048	549.216
4 Mb	2835.264	2649.216	1086.608
5 Mb	3005.856	2877.712	1038.352
6 Mb	3104.656	3172.592	1405.936
7 Mb	3187.656	3267.592	1305.936
8 Mb	3235.104	3391.808	1287.056

Table 6.11: Throughput with rerouted CBR traffic

Table 6.11 contains the statistical analysis of throughput for SCTP, TCP New Reno and TCP Vegas protocols with variable bandwidth and CBR traffic with simulations time 500 seconds in case of rerouting.

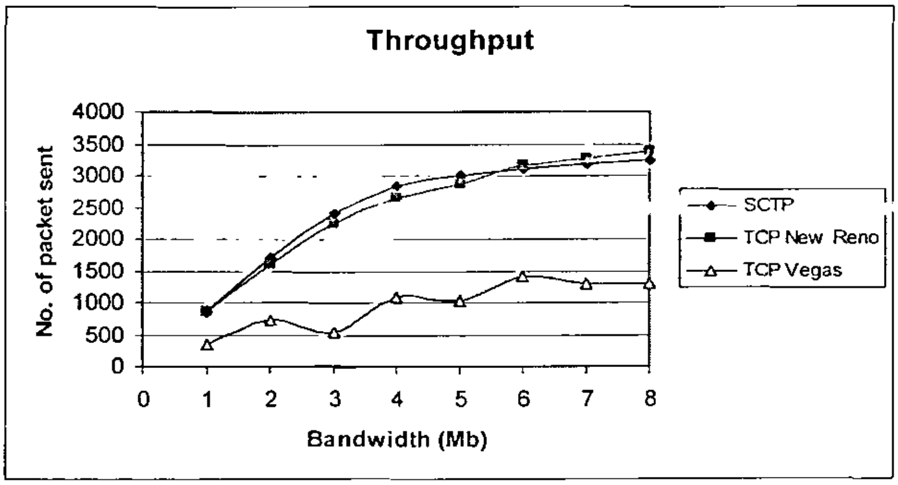

Figure 6.11: Throughput with rerouted CBR traffic

Figure 6.11 shows the throughput analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type CBR. On x-axis we take the bandwidth values and on y-axis we place the total number of packet sent. It shows that with the increase of bandwidth, throughput of SCTP and TCP New Reno is also increase while the throughput value of TCP Vegas is not increases with the same scenario. So after analyzing these three protocols it has been observed that SCTP has higher throughput as compared to other two protocols.

6.2.4.3 Channel wastage

Channel utilization of SCTP, TCP New Reno and TCP Vegas with packet size 1000 bytes can calculated by simulation given as under.

Bandwidth	SCTP (%)	TCP New Reno (%)	TCP Vegas (%)
1 Mb	12.9328	14.8432	64.7712
2 Mb	14.4888	20.5288	63.1568
3 Mb	19.4576	24.7984	81.6928
4 Mb	29.1184	33.7696	72.8348
5 Mb	39.8828	42.4457	79.2329
6 Mb	48.2557	47.1234	76.5677
7 Mb	54.2120	51.0236	78.1263
8 Mb	59.5612	55.1024	83.9118

Table 6.12: Channel wastage with rerouted CBR traffic

Table 6.12 contains the statistical analysis of channel wastage for SCTP, TCP New Reno and TCP Vegas protocols with variable bandwidth and CBR traffic with simulation time of 500 seconds in case of rerouting.

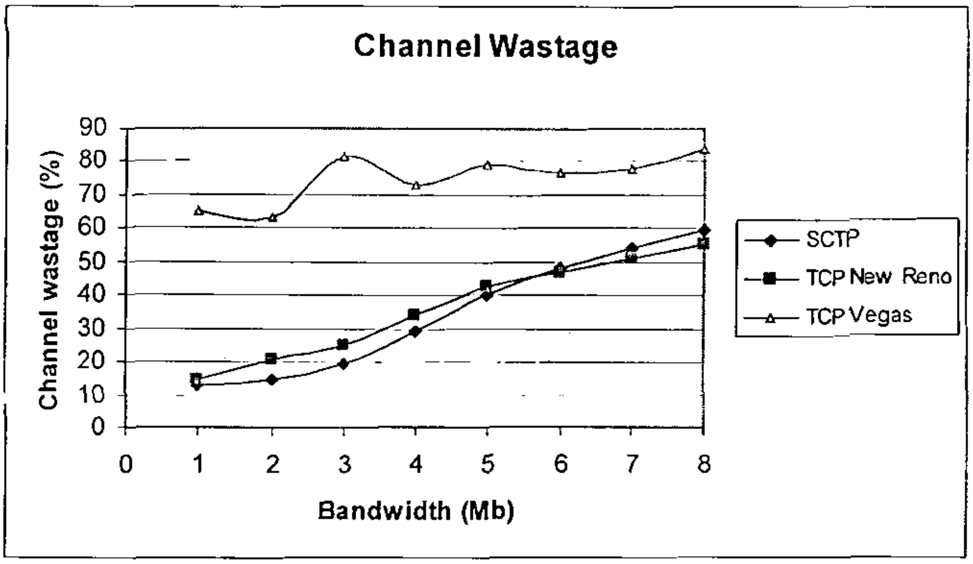

Figure 6.12: Channel wastage with rerouted CBR traffic

Figure 6.12 shows the channel wastage analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and CBR traffic in case of rerouting. On x-axis we take the bandwidth values and on y-axis we place the channel wastage in percentage. If value of bandwidth goes higher, then value of SCTP and TCP Vegas is gradually increases but the TCP Vegas has higher channel wastage than SCTP and TCP New Reno. After analyzing this graph we can say that SCTP performs better in the case of Channel wastage at the initial stage while at the finishing stage TCP New Reno performs better than SCTP.

6.2.4.4 Packet delivery Analysis with FTP traffic

Given below graph shows the behavior of three protocols with FTP traffic before and after rerouting scenarios.

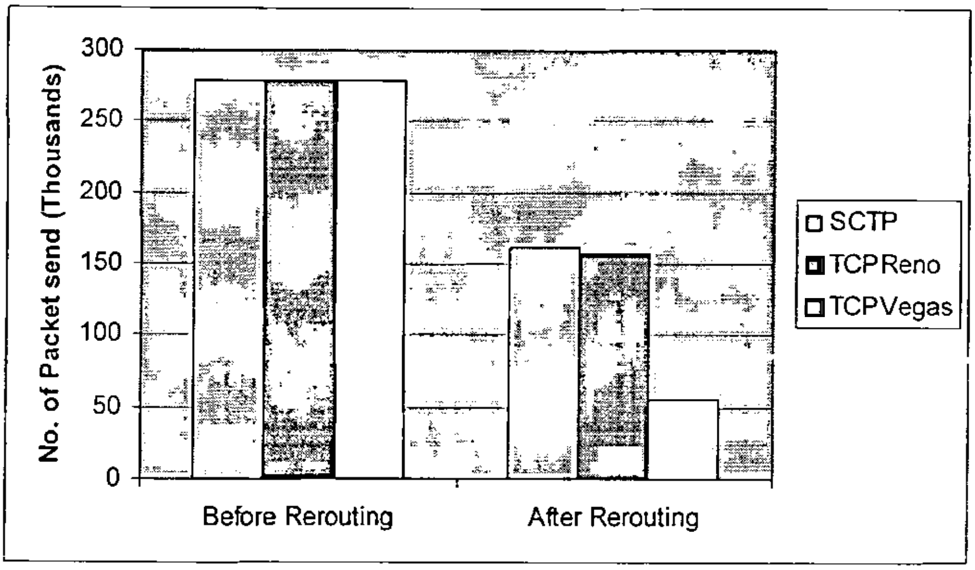

Figure 6.13: Packet delivery analysis with FTP traffic

Figure 6.13 shows the packet delivery analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type FTP. On x-axis we take the before and after rerouting values and on y-axis we place the number of packets sent in thousands. Before rerouting the packet transmission ratio is higher than the after rerouting scenario.

6.2.4.5 Packet delivery Analysis with CBR traffic

Given below figure shows the behavior of three protocols with FTP traffic before and after rerouting scenarios.

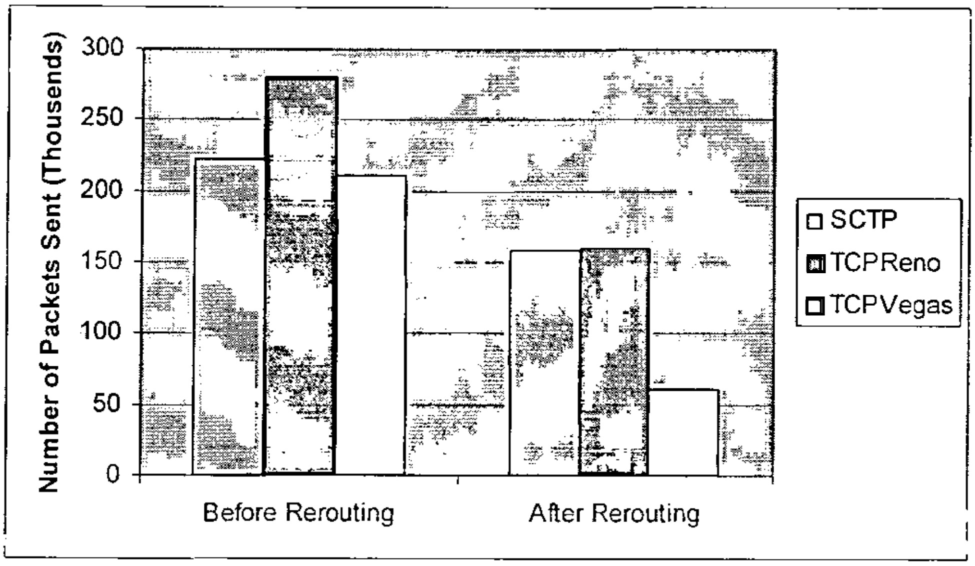

Figure 6.14: Packet delivery analysis with CBR traffic

Figure 6.14 shows packet delivery analysis of SCTP, TCP New Reno and TCP Vegas with variable bandwidth, packet size 1000 bytes and traffic type CBR. On x-axis we take the before and after rerouting values and on y-axis we place the number of packets sent in thousands. Before rerouting the packet transmission ratio is higher than the after rerouting scenario.

7. Conclusion and Outlook

7.1 Conclusion

Multi-Protocol Label Switching is a best technique for efficient utilization of network resources with small overhead labels. The reliability feature of SCTP and Traffic Engineering (TE) procedures improves the performance needs of real time data requirements that require heavy overheads in IP based networks. MPLS based setup has comparatively best QoS than the IP based systems. As the efficient utilization, traffic engineering of MPLS and reliability of different transport protocols improves the different QoS parameters.

SCTP improves upon TCP and UDP by combining the components of each. It is a protocol for transporting of PSTN signals over an internet protocol. The developers of protocols say that Stream Control Transmission Protocol is probably used for larger scenarios, including data multi-streaming; there have not any requirement of TCP.

In this research, quantitative analysis of TCP variants TCP Vegas, TCP New Reno and the new transport protocol SCTP have been performed for FTP and CBR traffic under different bandwidths and constant packet size. The quality of service parameters analyzed in this research is average delay, throughput, packet delivery and channel wastage with variable network bandwidth.

This study evaluated that TCP Vegas exhibits minimum average delay mostly in all cases of before and after rerouting in comparison of TCP New Reno and SCTP while SCTP has perform better in case of throughput, channel wastage and packet delivery in all bandwidth before and after rerouting scenarios. All these three protocols almost send the same number of packet with different bandwidth and 1000 bytes packet size. It is observed that STCP show some consistent behavior when the bandwidth increases.

7.2 Future work

In future we would try to improve this work in different aspects. Few of our future aspects for the advancement are given below.

1) Theses transport protocol can be analyzed with multiple traffic flows and the behavior of protocol when on link break in network and all traffic will be routed on the different suitable path.
2) Network topology can also be changed by adding more label switch routers (LSR) and IP nodes.
3) It can also be analyze with different bandwidth and packet size.
4) We can add the bottleneck scenario.

References

[1] Zhong Ren, Chen-Khong Tham, Chun-Choong Foo, Chi-Chung Ko. "Integration of Mobile IP and MPLS" IEEE ICC 2001, vol. 7, pp 2123-2127, Helsinki, Finland, June 2001.

[2] http://www.iec.org. Trillium MPLS web proforum tutorials.

[3]Gaeil Ahn and Woojik Chun, "Design and Implementation of MPLS Network Simulator Supporting LDP and CR-LDP" Networks, 2000, (ICON 2000), Proceedings. IEEE, International Conference 2000, pp: 441-446 ISBN: 0-7695-0777-8. 2000, IEEE.

[4]http://www.tcpipguide.com/free/t_TCPConnectionEstablishmentProcessTheThreeWayHandsh-3.htm.

[5]Yi-Cheng Chan, Chia-Tai Chan,Yaw-Chung Chen and Cheng-Yuan Ho "Performance Improvement of Congestion Avoidance Mechanism for TCP Vegas", Parallel and Distributed Systems, 2004. ICPADS 2004. Proceedings. Tenth International Conference pp: 605- 612, ISBN: 0-7695-2152-5, 7-9 July 2004 IEEE.

[6] Joel Sing and Ben Soh, "TCP New Vegas: Performance Evaluation and Validation", Proceedings of the 11th IEEE Symposium on Computers and Communications (ISCC'06) 0-7695-2588-1/06 2006 IEEE.

[7] Mo, J. La, R.J. Anantharam, V. Walrand, J. "Analysis and comparison of TCP Reno and Vegas" Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies. INFOCOM. '99.

[8] RFC 2960.

[9] SCTP Programmer's Guide HP-UX 11i v2, HP Part Number: 5992-0620, September 2007.

[10] Wei Sun Praveen Bhaniramka Raj Jain "Quality of Service using Traffic Engineering over MPLS: An Analysis" Local Computer Networks, 2000. Proceedings. 25th Annual IEEE Conference, pp: 238-241, ISBN: 0-7695-0912-6 2000 IEEE.

[11] Rajesh Rajamani, Sumit Kumar, Nikhil Gupta "SCTP versus TCP: Comparing the Performance of Transport Protocols for Web Traffic" Computer Sciences Department, University of Wisconsin-Madison, July 22, 2002 IEEE.

[12] ISHTIAQ Ahmed, OKABE Yasuo KANAZAWA Masanori "Improving Performance of SCTP over Broadband High Latency Networks" Local Computer Networks, Proceedings 28th Annual IEEE International Conference 2003, pp: 644- 645, ISBN: 0-7695-2037-5 20-24 Oct. 2003 IEEE.

[13] Jinyang Shi, Yuehui Jin, Wei Guo, Shiduan Cheng, Hui Huang, Dajiang Zhang "Performance Evaluation of SCTP as a Transport Layer Solution for Wireless Multi-access Networks" Wireless Communications and Networking Conference.2004 Vol: 1, pp: 453- 458, ISBN: 0-7803-8344-31. 21-25 March 2004, IEEE.

[14] Grinnemo ,TietoEnator AB, Lagergrens gata, Torbj¨orn Andersson, Kaniken¨asbanken, Anna Brunstrom, Karl-Johan "Performance Benefits of Avoiding Head-of-Line Blocking in SCTP" Autonomic and Autonomous Systems and International Conference on Networking and Services, ICAS-ICNS 2005, IEEE Joint International Conference, page 44-44, ISBN: 0-7695-2450-8, 23-28 Oct. 2005, IEEE.

[15] Andreas Jungmaier1, Germany Michael Schopp, Michael Tiixen, Siemens AG "Performance Evaluation of the Stream Control Transmission Protocol", Electro - technical Conference, 2006 Mediterranean, pp: 781-784, ISBN: 1-4244-0087-2, 16-19 May 2006, IEEE.

[16] M. Saeed Akbar, Syed Zubair Ahmed, M. Abdul Qadir "Quantitative Analytical Performance of TCP Variants in IP and MPLS Networks" Multitopic Conference, INMIC '06. IEEE, pp 331-336, ISBN: 1-4244-0795-8, Dec 2006.

[17] Dongli Zhang and Dan Ionescu "QoS Performance Analysis in Deployment of DiffServ-aware MPLS Traffic Engineering" Software Engineering, Artificial Intelligence, Networking, and Parallel/Distributed Computing, SNPD 2007. Eighth ACIS International Conference, IEEE, vol. 3, pp 963-967, ISBN: 978-0-7695-2909-7, July 30 2007-Aug. 1 2007.

[18] Chen Hui 1, Wang Jun 2, Feng Xiaolin 1, "New Strategy of Improving Stream Control Transmission Protocol Performance over Satellite Link." Control Conference 2008, CCC 2008, 27th Chinese, IEEE, pp 307-310, ISBN: 978-7-900719-70-6, July 16-18, 2008.

[19] Md. Arifur Rahman1, Ahmedul Haque Kabir1, K. A. M. Lutfullah1, M. Zahedul Hassan2, M. R. Amin1 "Performance Analysis and the Study of the behavior of MPLS Protocols" Computer and Communication Engineering, 2008. ICCCE 2008, International Conference, pp: 226-229, ISBN: 978-1-4244-1691, 13-15 May 2008, IEEE.

[20] Armando L. Caro Jr., Keyur Shah, Janardhan R. Iyengar, Paul D. Amer "SCTP and TCP Variants: Congestion Control under Multiple Losses" Protocol Engineering Lab Computer and Information Sciences University of Delaware Randall R. Stewart.

[21]http://en.wikipedia.org/wiki/Packet_transfer_delay

[22]http://en.wikipedia.org/wiki/Bandwidth_(computing)

[23] Anjali Agarwal "Quality of Service (QoS) in the New Public Network Architecture"

[24]Kevin fall and Kannan Varadhan, editor, NS-documentation http://www.isi.edu/nsnam/ns/doc/ns_doc.pdf , August 2008.

[25]Microsoft Press® Computer and Internet Dictionary © & (P) 1997, 1998, 1999 Microsoft Corporation.

[26]http://www.eecs.berkeley.edu/~ananth/19992001/Richard/IssuesInTCPVegas.pdf.

[27] J. Mo, R.J. La, V. Anantharam, and J. Walrand, "Analysis and Comparison of TCP Vegas", Available at http://www.path.berkeley.edu/ hyongla, June 1998.

Acronyms

MPLS:	Multi-Protocol Label Switching
IP:	Internet Protocol
ATM:	Asynchronous Transfer Mode
SONET:	Synchronous Optical Network
SCTP:	Stream Control Transmission Protocol
TCP:	Transmission Control Protocol
UDP:	User Datagram Protocol
QoS:	Quality of Service
NS-2:	Network Simulator-2
OSPF:	Open Shortest Path First
RSVP:	Resource Reservation Protocol
LER:	Label Edge Router
LSR:	Label Switch Router
FEC:	Forward Equivalence Class
CoS:	Class of Service
TTL:	Time to Live
VPN:	Virtual Private Network
LSP:	Label Switch Path
ER LSR:	Explicit Routed LSR
LDP:	Label Distribution Protocol
CR-LDP:	Constraint base Routing LDP
PIM:	Protocol Independent Multicast
BGP:	Border Gateway Protocol
TE:	Traffic Engineering
NAM:	Network Animator
LIB:	Label Information Base
PHY:	Physical
RFC:	Request for Comments
IETF:	Internet Engineering Task Force
OTcl:	Object–Oriented Tran scripting Language
SYN:	Synchronization
ACK:	Acknowledgement
RTT:	Round Trip Time
SS:	Slow Start
CA:	Congestion Avoidance
Sstresh:	Slow Start Threshold
Cwnd:	Congestion Window
AIMD:	Additive Increase Multiplicative Decrease
MTU:	Maximum Transmission Unit
HOL:	Head of Line
DoS:	Denial of Services
INIT-ACK:	Initialization Acknowledgement

CRC:	Cyclic Redundancy Check
TSN:	Transmission Sequence Number
SSN:	Stream Sequence Number
SID:	Stream Identifier
Rwnd:	Receiver Window
SACK:	Selective Acknowledgement
PDU:	Protocol Data Unit
ECN:	Explicit Congestion Notification
HA:	Home Agent
FA:	Foreign Agent
HTTP:	Hyper Text Transfer Protocol
PSTN:	Public Switched Telephone Network
VOIP:	Voice over IP
VBR:	Variable Bit Rate
HTNA:	Highest TSN Newly Acknowledged
OTCL:	Object Oriented Toolkit Command Language
WLAN:	Wireless LAN
DSDV:	Destination Sequence Distance Vector
TORA:	Temporally Ordered Routing Algorithm
DSR:	Dynamic Source Routing
AODV:	Ad hoc On-demand Distance Vector
WWW:	World Wide Web
CBR:	Constant Bit Rate
FTP:	File Transfer Protocol
BSD:	Berkley Software Distribution
IEEE:	Institute of Electrical and Electronics Engineering