MEE08:

Evaluation of UDP and SCTP for SIP-T and TCP, UDP and SCTP with constant traffic

Muhammad Sarfraz Zafar (761114-P271)
Muhammad Shoaib Gill (811026-P578)

This thesis is presented as part of Degree of
Master of Science in Electrical Engineering

Blekinge Institute of Technology
November 2008

Blekinge Tekniska Högskola
SE–371 79 Karlskrona, Sweden
Tel.vx 0455-38 50 00
Fax 0455-38 50 57
School of Engineering
Department of Electrical Engineering
Supervisor Dr David Ermaan
Examiner Dr David Ermaan

Abstract

In recent years, Voice over IP (VoIP) has gained a lot of popularity. Signaling being an important part of VoIP has been addressed by the (IETF) SIGTRAN working group to meet Quality of Service as given by Public Switched Telephone Network (PSTN), so that both PSTN and VoIP can co-exit and work together in a seamless manner

SIP (Session Initiation Protocol) developed by IETF for VoIP signaling is a communication control protocol capable of running on different transport layers, e.g., TCP, UDP or SCTP. Today's SIP application is mostly operating over the unreliable transport protocol UDP. In lossy environment such as wireless networks and congested Internet networks, SIP messages can be lost or delivered out of sequence. The SIP application then has to retransmit the lost messages and re-order the received packets. This additional processing overhead can degrade the performance of the SIP application. Therefore to solve this problem, the researchers are looking for a more appropriate transport layer for SIP. SCTP, a transport protocol providing acknowledged, error-free, non-duplicated transfer of messages, has been proposed to be an alternative to UDP [1] and TCP [2]. The multi-streaming and multi-homing features of SCTP are especially attractive for applications that have stringent performance and high reliability requirements and an example is the SIP proxy server.

In this research, we have analyzed the performance offered by SCTP for SIP message delivery in the perspective of historic research work as well as determined call setup time using UDP and SCTP by simulating SIP traffic in Network Simulator-2 (ns-2). We also evaluate TCP, UDP and SCTP traffic with constant bit rate of traffic through ns-2

Acknowledgments

All praise to Almighty ALLAH, the creator of creators, the most beneficent the most merciful whose shower like blessings ever on mankind and on us all time through our life.

We say much thanks to our beloved supervisor Dr. David Erman, for providing us an open mindedness to carry out our research. We are thankful to all our friends who supported us and more to those who did not. Above all, we dedicate this report to our beloved parents for their consistent efforts for our progress, patience, support and love.

Table of Contents

Chapter 1

Introduction

Introduction

With the acceptance that PSTN and IP based telephony are to co-exit for a fairly long period of time, telecommunication service providers tend to transform their traditional circuit-switched networks into packet-based networks. IP base networks that smooth the progress of new services that put altogether data, voice, and video information. The main motivating force of this deployment is that it is economical and also associated with converged data and voice networks.

This chapter explains a traditional circuit-switched telephone network, and then a short explanation of IP based / packet-based network which is called as IP Telephony.

1.1 A Telephone Network

The PSTN is a network of word's public circuit switched network like the internet is a network of word's public IP based, which is called packet switched. Originally it was fixed line analog and now completely digital and includes both mobile and fixed line. Figure 1.1 exemplifies circuit-switched networking among a PBX and a cellular network. A PBX which is a private telephone network is using in an industrious organization. The PBX users share some outside lines for placing telephone calls outer to PBX. MSC (mobile switching center), BSS (Base station sub system) and a cell phone are the part of cellular network. The BSS is responsible for radio links and related tasks and MSC routes all kind of receiving and transmitting calls and also responsible for assigning user channels. The MSC and PBX are connected via T1 or T3 circuits. The T1/T3 circuits carry signaling and voice messages.

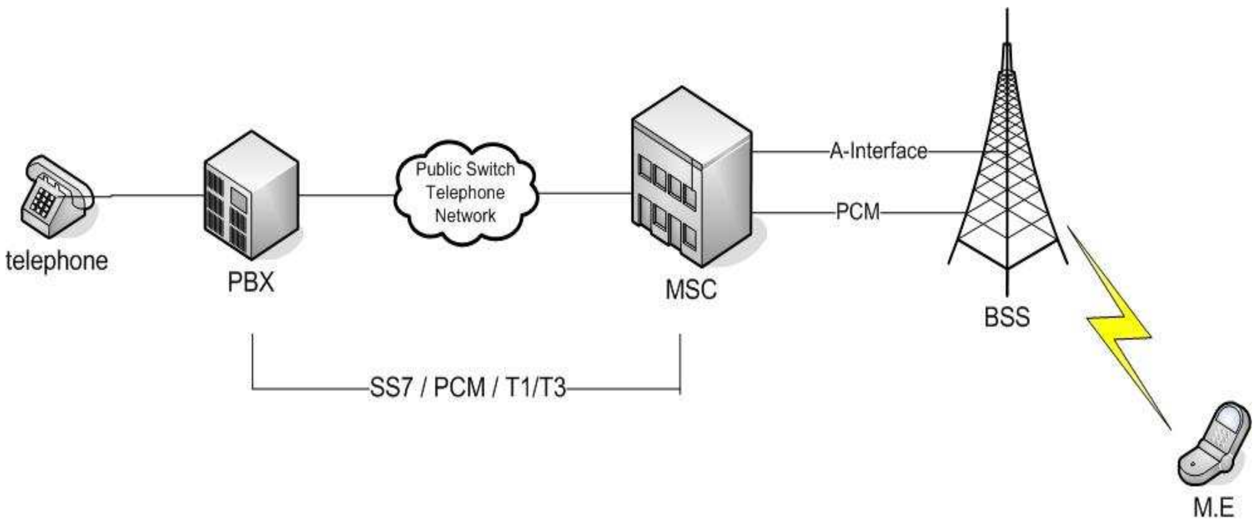


Figure 1.1: Telephone network / Cellular network -- Circuit-switched networking.

The philosophy of circuit is that when a call setup between both sides in a telephone networks, the connection establishes for complete session mean for total duration of the call and the connection is called a circuit and this network is called circuit- switched network. It is the base of the Public Switched Telephone Network (PSTN) [3]. Circuit switched network throw away a lot of bandwidth and capacity of the network is limited. While talking with each other in a phone discussion, when one person is talking, the other one listens. So only half of the connection uses at any time slice. There comes some occasions when no one talks for some time.

1.2 Internet Telephony

Internet telephony is a transport of telephone calls over internet, base on IP and Voice over IP is IP telephony. In this procedure voice calls are transmitted over a packet switched network. The most noteworthy advantage of IPT is the saving of money and easy implementation of innovative services (value added services). In money saving low capital and operating costs are considered. The efficient use of bandwidth always remained an advantage of VoIP. Internet Telephony Service Providers (ITSP) uses a single transportation for providing both, Internet access and Internet telephony. Only data oriented switches could be installed for switching data as well as pocket-sized voice. Multiplexing of data and voice, result in better bandwidth utilization. Not only the service providers but clients also take advantages of lower cost in IP telephony.

In packet switched networks several user share the same physical connection. The communication channels are break downed in small chunks of data called packets in IP networks and cells in ATM networks. IP packets are routed all the way through a network based on the destination address contained within the packet header. CELP is a data compression techniques use for reduction of the size of packet. CELP basically is a speech coding algorithm originally opposed by M.R Schroeder and B S Atal in 1985. In this scenario voice first coded and then compressed. After compression packets creates and then added a header and placed on the network. Besides good utilization of network resources, by integrating voice and data on a single converged network, companies save long distance changes, wiring fees, and administrative costs. VoIP bypass long distance calls by transporting calls over an IP network. IP telephony allows service providers to offer advanced services that increase revenue. An example of this is the integrated messaging that provides voice mail, email and fax access from centralized location using multiple devices such as Personal Computer phone and Personal Digital Assistant. NGN (next generation network) is a broad term to describe some key architectural evaluation in telecommunication core and access networks. NGN idea is that one network transports all information and services (voice, data, all sort of media and video) by encapsulating these into packets. NGN separates control path and voice path as shown in Figure 1.2. The devices of circuit switched networks are replaced by soft switches. A soft switch is a software-based entity that provides call control functionality [4]. One new unit called media gateway is launched into the network. The media gateway works like a bridge between both network, and is controlled by softswitches using Media Gateway Control Protocol (MGCP). It converts the voice (PCM) signals into packets and vice versa. RTP (Real time protocol) is used for transporting voice packets in an IP network. The call control signaling protocol can be either SS7 or SIP protocols.

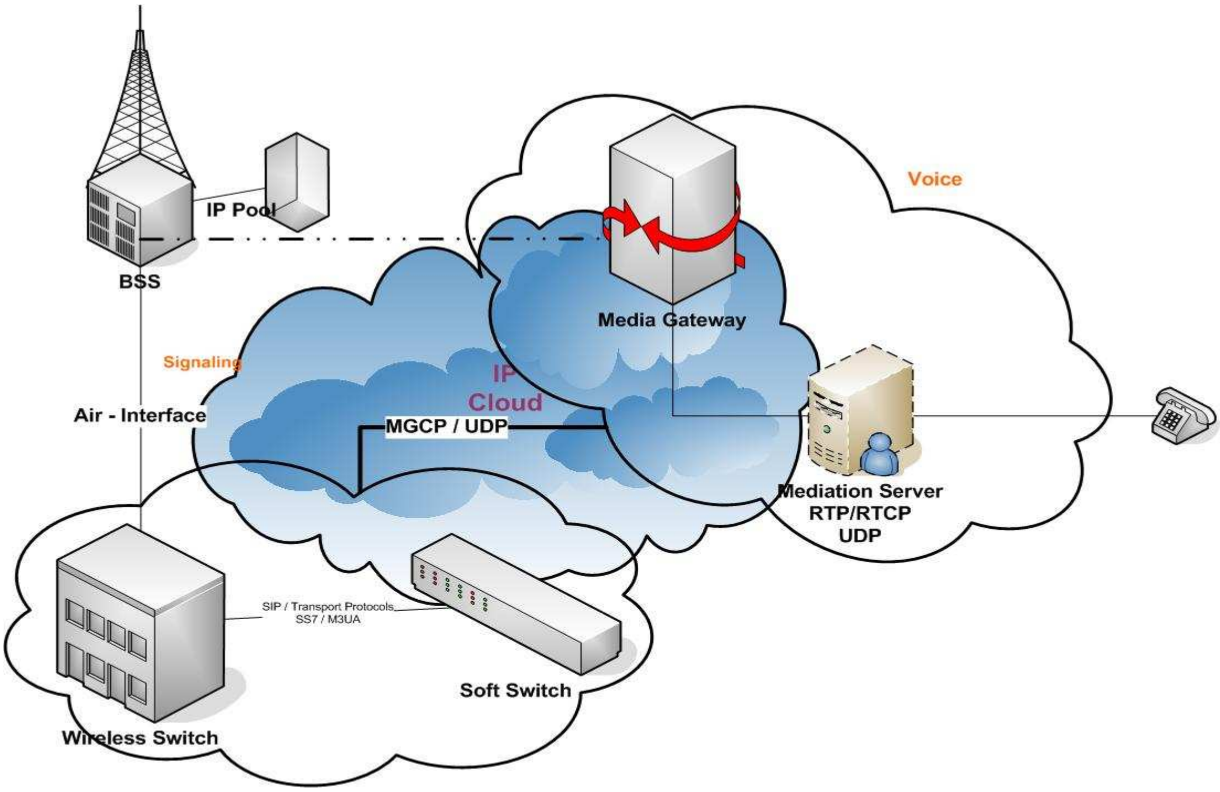


Figure 1.2: Telephone network / Cellular network -- Packet-switched networking.

Benefits for separation of signaling and voice path [5]

- Independency: Different traffic always has different network requirements. For example consistency, reliability and throughput and thus must employ in parallel.

- To spread work load between two networks or some other resources to get optimal resource utilization, maximum throughput and minimum response time. Therefore signaling and voice traffic should be dispersed between multiple soft switches and media gateways based on traffic load.

- If one media gateway system fails then voice traffic can be redirected or deviated from one gateway to another same like signal messages can be detoured from the active softswitch to the standby.

- Switches and gateways can be added or removed from the network based on the number of subscribers in the network, without interrupting the ongoing services. Through this technique network is scalable.

1.3 Computer Telephony Integration (CTI)

During the last 20 years, business users wanted to find integration for their telephony and computing systems. One can only visualize the vast demand for this, assuming that computer and communications resources and finance are in the same sphere. If some one has all in an integration manner then the savings are countless. Every organization want that there should be integrated resources for taking benefits of a single system in a combined infrastructure [5].

A time ago, voice communication department and Management Information Systems (MIS) department have been operating separately. And their operating funds and staff extend beyond in certain areas. The level of overlap varied by organization, but the overlap itself was the alarming thing. Since the meetings and convergence began almost a decade ago, the primary moves in these operations were somewhat striking. These moves were not likely to explode in themselves, but additionally with a gradually rolling wave. In many site within the organization, these changes were almost with out observations.

Business has attained passion and a pace that is extraordinary and yes it is amazing thing that it is speedup. The competition in business and changes in technology are in a speedy way and growing exponentially. So organization has started to locate a way how to find a better means of dealing and commerce with their customers and to answer the requirements of clients for growing productivity, decreasing expenses, and enhancing an aggressive frame over the existing market sector. Maintaining pace with the industry and uphold the competitive edge has turned into a crucial factor in the survivability of any business sector, organization or industry [5].

1.4 Telephony Signaling

A main question in integration of VoIP and PSTN is of signaling. Both networks uses different sort of signaling, witch requires to be mapped into the form, which used by other network when necessary to navigate the links of that network. Following table shows the signals used by both type of networks and their equivalent at either sides.

IP	PSTN
INVITE TRYING	IAM (Initial Address Message)
RINGING	ACM (Address Complete Message)
OK	ANM (Answer)
BYE	REL (Release)
OK	RLC (Release Complete)

Table 1.1: PSTN and SIP (IP) equivalent signals

Soft switches are uses to map signals into each other forms. The soft switches are capable of identifying both form of signals and translating them to the other form used by target network.

1.5 Project Motivation

SCTP was developed by IETF SIGTRAN Working Group to carry telephony signals on IP from networks like SS7 [6]. The main propose of design consideration was to beat the limitations of TCP and UDP as signaling carrier. Since its close similarities with TCP in clogging and flow control it has undergone a lot of studies and investigations in terms of performance evaluation and judgment with TCP.

SIP is most fruitful signaling protocol now days. SIP can operate on UDP, TCP or SCTP. UDP provides unreliable and untrustworthy datagram service [1], and relies on the application layer for error control, detection of message repetition, duplication, and retransmission of lost messages.

On the other side, TCP gives error and flow control [2]. However, its strict byte order delivery creates performance issues. It also suffers from other downsides as mentioned in [7]. SCTP overcomes some of the limitations of TCP and SCTP also provides a reliable datagram transport mechanism. SCTP also provides features which required by a SIP system such as multi stream message passing for performance, cookie mechanism for security, and multi homing for fault tolerance and high availability [7].

The selection of protocols is influenced by the fact that SCTP, TCP and its all variants form one category of protocols (reliable, have flow and congestion control, connection oriented) whereas UDP is a protocol without connection orientation, without flow and congestion control. Thus UDP has minimum of overhead, but retransmissions have to be implemented in application layer witch could be a major disadvantage. So the comparison I made here is between UDP and SCTP.

The reasons for selection of SCTP are the features like message orientation and unordered delivery, witch are highly desirable factors while transporting VoIP signals.

1.6 Project Scope

There have been numerous research efforts regarding the performance of SCTP when compared with TCP and UDP. In this report we have included and reviewed various performance studies carried out for SCTP, in particular the ones those focuses the Session Initiation Protocol; and voice communication trends in today's IP network and its signaling requirements. We have also compared SIP message delays when these are transported over UDP and SCTP in SIP-T scenarios. We are also analyzed the traffic when it runs over TCP, UDP and SCTP with constant level and compared the results.

The order of the report is well thought-out as follows. An introduction is in Chapter 1 and Voice over IP, Real time protocol and Session Initiation Protocol (SIP) are in Chapter 2. Chapter 3 comprises of the analysis of TCP, UDP and SCTP in term of packet loss, delays and throughput. In Chapter 4 the simulation setup and results are given for SCTP and UDP w.r.t SIP-T and Finally, the conclusion and remarks are in Chapter 5.

Chapter 2

Session Initiation Protocol, SIP-T, Signaling and RTP

Session Initiation Protocol and Signaling

Session Initiation Protocol has gained a lot of popularity for carrying signaling information for multimedia applications including

- Voice-over-IP
- Video Conferencing
- Distance Learning etc. [5]

Historically H.323 [8] has been used for such purpose, but due to certain limitations and drawbacks in it SIP was proposed and widely accepted. Since than, the popularity of SIP as signaling standard for wider domains. Its is already accepted by 3GPP (Third Generation Partnership Project) [8] as a signaling protocol for its IP Multimedia Subsystem (IMS), as well as expected to be dominant signaling protocol for wireless and wired networks in future [9]

2.1 Primary Concepts

Session Initiation Protocol (SIP), developed by the Internet Engineering Task Force (IETF), is a contral protocol which creates, modify and terminate session with one or more participnts and this session can be an Internet call, multimedia conference session, or multimedia distribution. The IETF RFC 3261 defines this protocol [10]. SIP is a lightweight protocol because it requires very few messages, called methods, for managing a basic session. These methods are INVITE, BYE, ACK, REGISTER, OPTIONS, CANCEL and INFO. When a user wants to join a session an INVITE method is uses. It is similar to IAM message in SS7 ISDN User Part for Public Switched Telephone Network. For termination for an established session BYE method is used and ACK confirms that a caller has received a final response to an INVITE and a user agent uses the REGISTER method to notify a SIP network of its current location. The OPTIONS routine is used to query a user agent or server about its competencies and discover its current availability. The CANCEL method is making use for ending a pending request.

For holding a mid call information INFO method is used. SIP uses Session Description Protocol (SDP) [11] to describe the session. In the case of video, audio, or multimedia session, the session information will be used for setting up an RTP stream, running on UDP that in turn operates on IP. SIP is not dependent of transport layer, i.e., it can be used over UDP, TCP, or SCTP. In SIP case when using UDP, messages may be lost or received out of sequence. SIP, therefore, uses its own reliable mechanisms via retransmission timers, command sequence (CSeq) numbers, and positive acknowledgments. The reliable mechanisms will be discussed later in the report. Following Figure 2.1 shows commonly used Internet multimedia protocol stack.

Internet Layer	IP
Transport Layer	TCP / SCTP / UDP
Application Layer	H.323 / SIP / RTP

Figure 2.1: Internet multimedia protocol stack

2.2 SIP Design and Architecture

A SIP system has two components, user agents and network servers. One of two the user agent is end system and both components be active on behalf of someone, who wants to participate in calls. User agent have both things a protocol client called user agent client (UAC), and a protocol server called UAS and UAC is used to initiate a call, while the UAS is used to answer a call. The existence of both in a user agent enables peer to peer operations.

SIP offers two different types of network servers which consists of a proxy and a redirect server in which SIP proxy server receives requests and checks where to send the requests, and then forwards the request to the next server on behalf of the user and the redirect server receives the requests but relatively passing these onto the next server, it sends a response back to the caller indicating the address of the called user. The caller then links the called party at the next server directly. The main task of a SIP network server is to do the work of DNS server mean it provides name resolution and user location. Like Hyper Text Transfer Protocol (HTTP), SIP user is identified using Uniform Resource Locator (URL). The SIP network server looks up the URL either from the local database or remote location server, and finds the exact location (IP address) of the user.

A proxy server is of two types a state-ful or state-less. When a proxy server is state-ful then a proxy server remembers the incoming requests that generate outgoing requests. A stateless proxy not remembers all information once an outgoing request is produced. The proxies those accept TCP connections or SCTP associations must be state-ful. Or else, if the proxy were to lose a request, the TCP or SCTP client would never retransmit it. Therefore, in this task we assume a state-ful proxy server. Figure 2.2 shows SIP architecture and demonstrate the basic message flow for setting up a session.

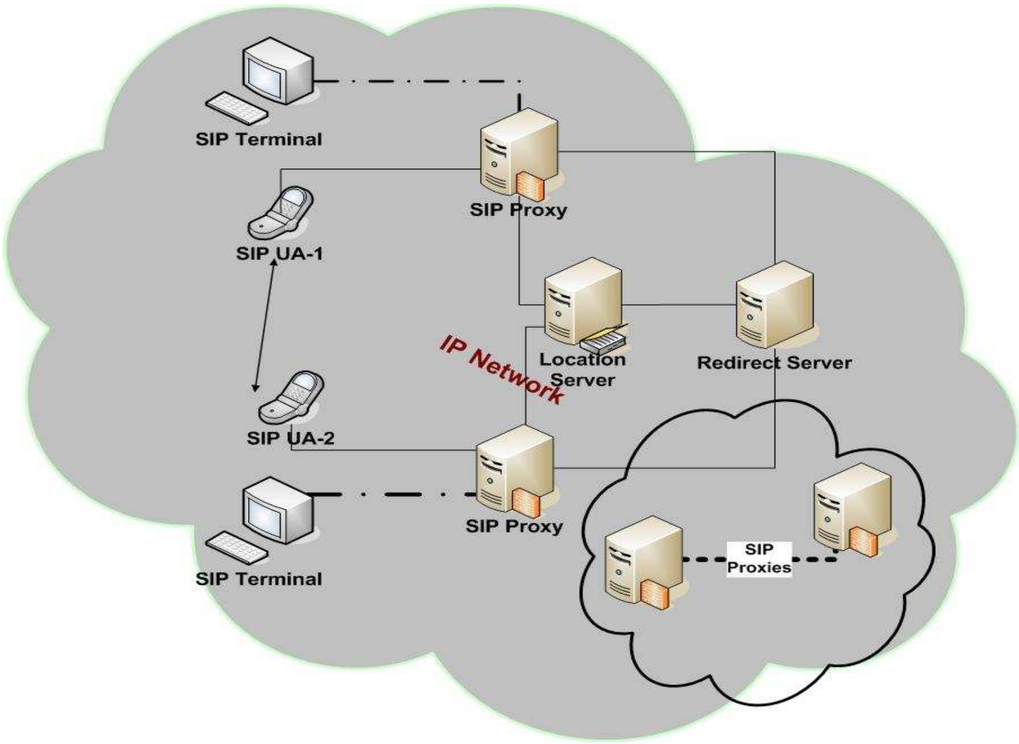


Figure 2.2

Below is the detailed of SIP session of Fig 2.2

.

- SIP UA-1 sends INVITE request to SIP proxy server.

- In case of failure of resolving called party's location, it diverge the request to the redirect server.

- Redirect server get the address of the next proxy server and based on this address, the proxy server send the request to the next SIP proxy server.

- In next step proxy server ask the registrar, location server for the address of the next hop and server return the address of next hop,

- This time, the proxy is familiar the location of the called party and forks the invitation request to SIP UA-2.

- In next some steps the acknowledgment message from UA-2 will be returned to UA-1 using the same path as the request message.

- Once the call setup process is completed, the session will be directly established between UA-1 and UA-2. For IP network, the voice/audio stream can be carried over IP using RTP.

Along with other servers in a SIP network, there is a SIP Registrar shown in Figure 2.3. SIP registrar permits mobility hold within the SIP network. When a SIP user agent moves to a new location, it will register its new location with the Registrar, which in turns updates the location database via Location Server. Upon consulting the Location Server, the network server will then know how to route the new incoming calls to the new location. SIP Network Server and SIP Registrar are usually implemented on the same machine.

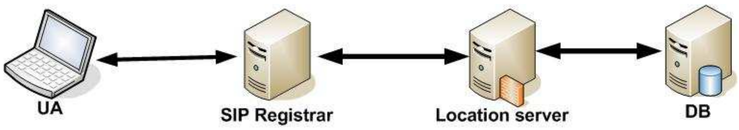


Figure 2.3: SIP registrar tracks SIP UA current location.

- The User A sends a registration message to the Registrar.
- The Registrar keeps the record of registration information in location service.
- As information is stored, Registrar sends the proper response back to the UA the user agent.

2.3 SIP Messages

A Session Initiation Protocol is a message that is either a request from a client to a server or vice versa. A SIP request message begins with a request line, response message begins with a status line, followed by header fields, and a message body that is optional, the request line as well as the header field defines the nature of the call in terms of services, addresses, and protocol features. Message body is not dependent of the SIP protocol and can have an arbitrary content. For multimedia application, the message body usually contains Session Description Protocol (SDP) that defines the session information. Below are sample SIP request and response messages. The descriptions of each field in the SIP messages are defined in SIP specifications [10].

2.4 Message Flow

The following section describes the examples of SIP message flows. When a user agent starts up or moves to a new location, it must register its location with the SIP Proxy Server before accepting any incoming calls. Figure 2.4 depicts a sample

SIP registration call flow. Both UA (1 and 2) register their location with the Proxy Server. Upon successful registration, the Proxy Server sends the success response 401 OK to the user agents.

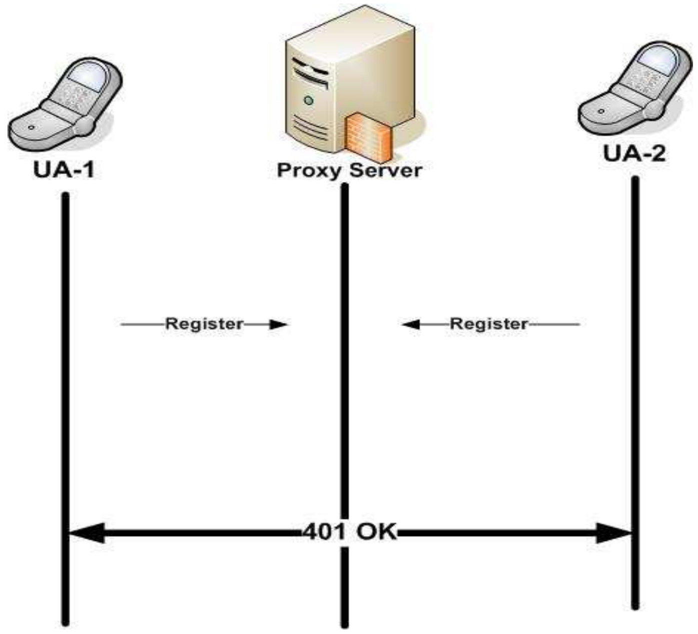


Figure 2.4 Registration message flow diagram.

If user 1 knows the IP address of user 2, it can directly send the INVITE user 2. In reality, this may not be the case because IP addresses are often dynamically assigned due to the shortage of IPv4 addresses but in IPv6 scenario can be different.

The peer may also move to different placements with time. So a proxy server is in need to route the requests.

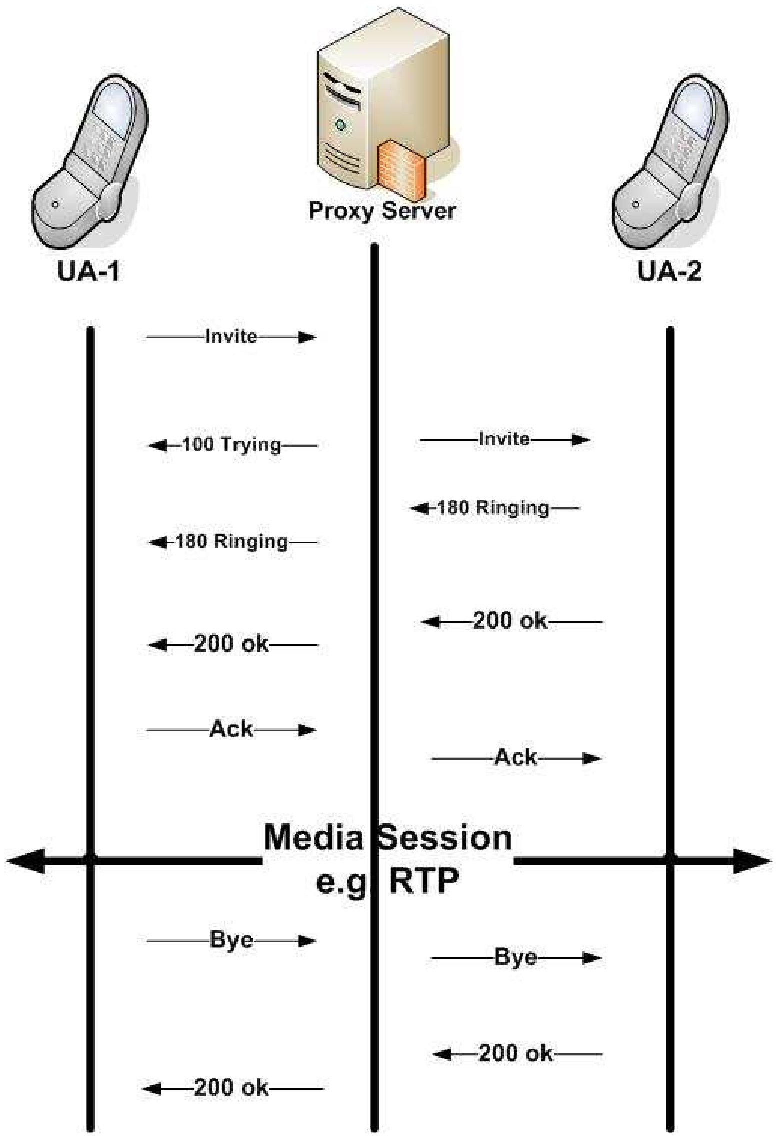


Figure 2.5 Call setup and message flow

Figure 2.5 describe a SIP call establishment and termination via SIP proxy server. User 1 does not know exactly where User 2 is at present logged in, and therefore sends the INVITE method with User 2's SIP URL to the SIP Proxy Server. The Proxy server sends TRYING response to User 1 as it takes time to perform a name resolution of User 2's SIP URL. After getting the IP address (current location) of User 2, the Proxy Server route the INVITE request to User 2. User 2 returns RINGING response to indicate that the INVITE has been received and that alerting has taken place. The Proxy Server routes the RINGING response to User 1, and User 1 should play the ringing tone. User 2 sends OK to the Proxy Server to accept the call. UA 1 acknowledges the final responses to the INVITE request by sending ACK to User 2 via the proxy server. User 1 and User 2 can now start the media session, for example, sending and receiving RTP streams.

To terminate the call, User 1 sends BYE to User 2 via proxy server. User 2 responds with OK, and the media session will be closed. To differentiate the second OK response (for BYE request) from the first OK response (for INVITE request), the BYE and the INVITE requests have different command sequence number (CSeq), but they have the same Call-ID because they belong to the same call session.

2.5 SIP for Telephony (SIP-T)

The SIP-T specified detailed in RFC 3372 [12], SIP-T specifies the SIP standards when public switch telephone network works with an IP backbone in an integrated approach. The scenarios which are possible for PSTN-IP networks co-working are the followings.

2.5.1 SIP-T Flows

The possible scenarios for SIP-T are described in subsections below.

2.5.1.1 SIP linkage

(PSTN - IP - PSTN) SIP bridging is that set-up where a SIP network provides a connection between two fragments (Nodes) of the PSTN that is called as 'SIP bridging'. When a call that is intended to go for the SIP network initiate in the PSTN, an SS7 ISDN user part (ISUP) message will ultimately be received by the gateway, which is the spot of interlink with the PSTN network and this gateway is as of the viewpoint of the SIP protocol the user agent client for this call arrangement demand as a request. Conventional (usually) SIP routing that is used in the IP network is to find out the suitable position, where there can be a termination (in this occurrence a gateway) and for set up a SIP dialog and start cooperation for a media session among the starting point from where the call originate and the endpoints where the calls terminate.

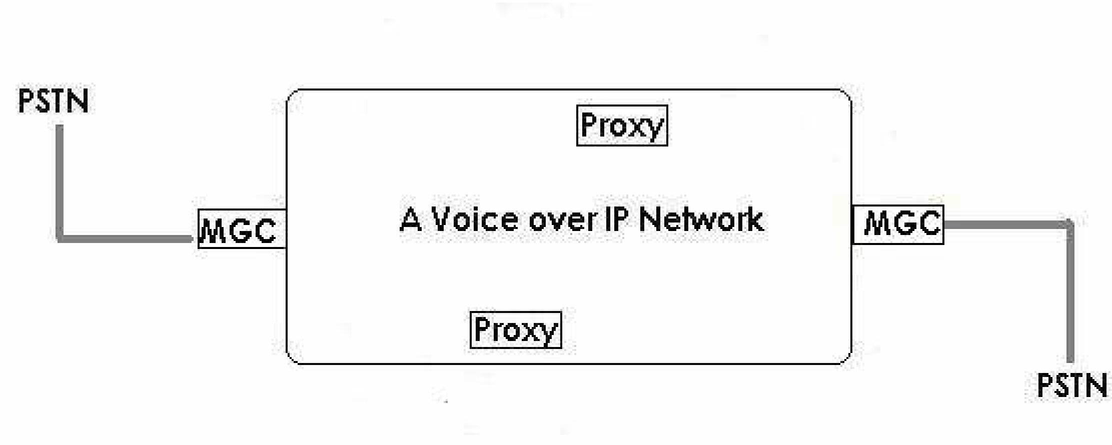


Figure 2.6: PSTN – PSTN (Originating and termination) (SIP Bridging)

The gateway known as egress indicate a signal ISUP to the PSTN, using again any ISUP that is currently here in the SIP request it obtains as proper. In SIP bridging when there is a call from PSTN originator to PSTN terminator; it needs to make available a definite level of feature transparency.

Following the SIP bridging, call-flow

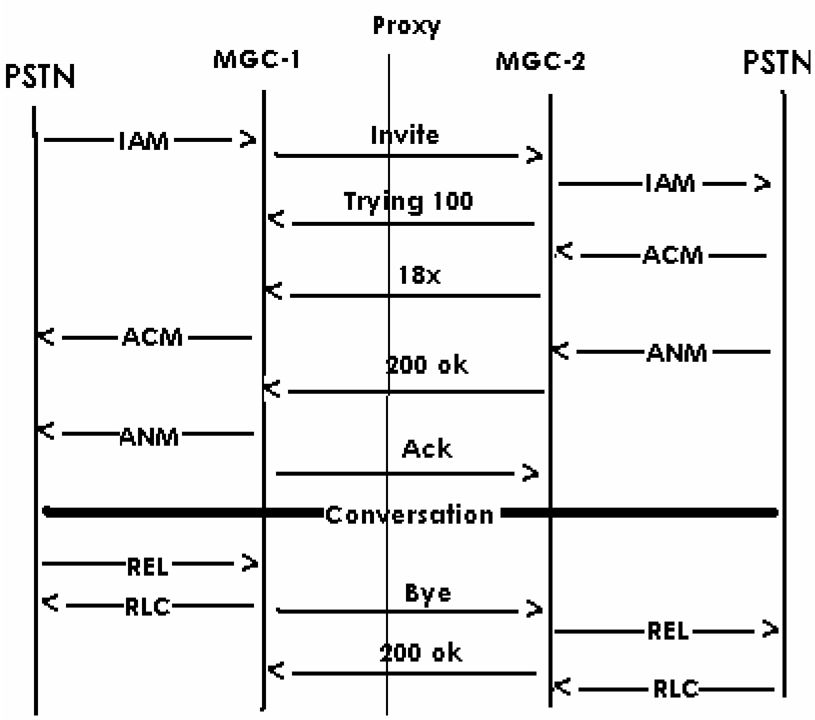


Figure 2.7: PSTN origination - PSTN termination (Message Flow)

2.5.1.2 PSTN – IP (Origination and Termination)

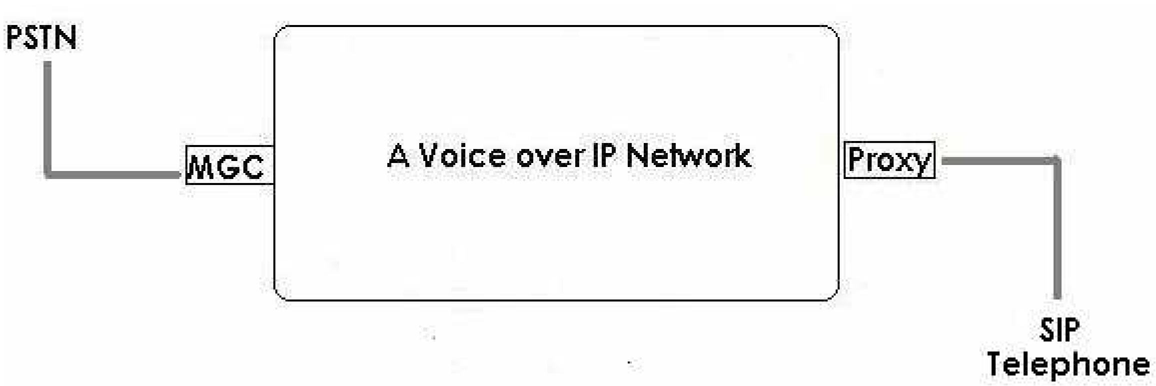


Figure 2.8: PSTN origination - IP termination

In figure 2.8 we see that the PSTN end is the originator for the call setup and the call is terminating at the SIP phone end. A simple call-flow describing the ISUP and SIP signaling in the figure below for a PSTN (call originating) and terminating at a SIP endpoint.

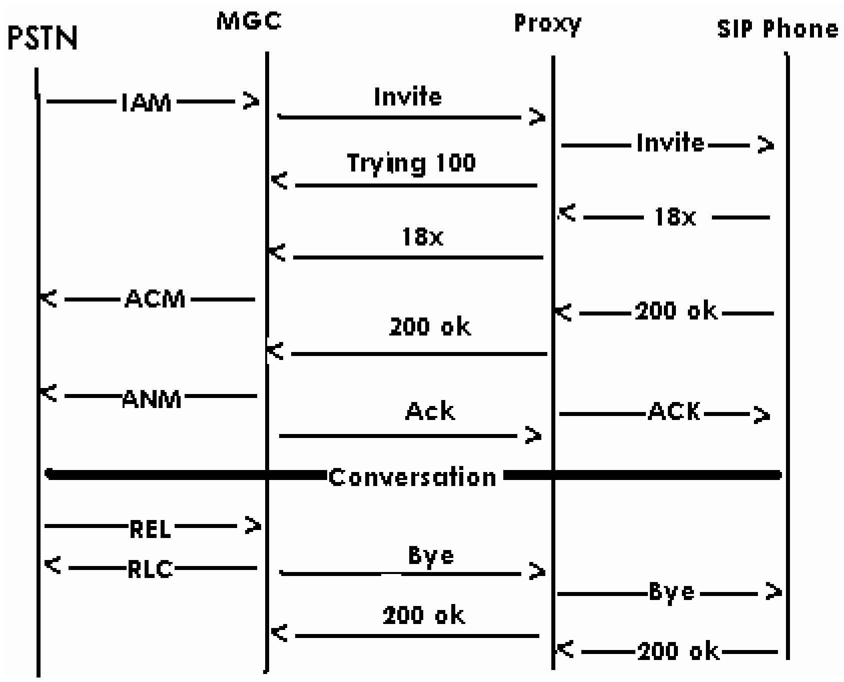


Figure 2.9: Originating from PSTN terminating at IP

2.5.1.3 IP origination - PSTN termination

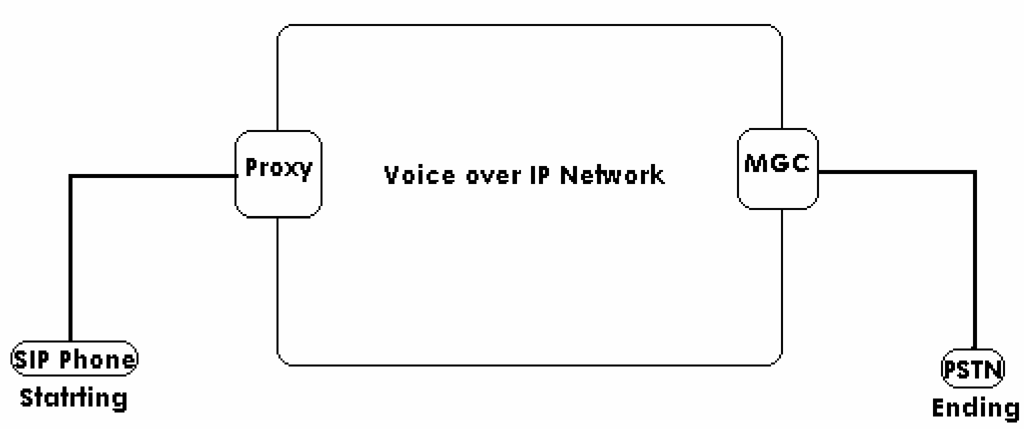


Figure 2.10: IP origination - PSTN termination

In figure 2.8 PSTN was the originator for the call and the call was terminating at the SIP phone. And in figure 2.10 the call is originating from a SIP phone and it is terminating at the PSTN end. One thing we see here that unlike in above two flows, there is thus no ISUP encapsulation in the request – for deriving the values for ISUP parameters, the gateway where the call is terminating only performs the transformation on the SIP headers. A simple call flow diagram can be seen below in figure 2.11.

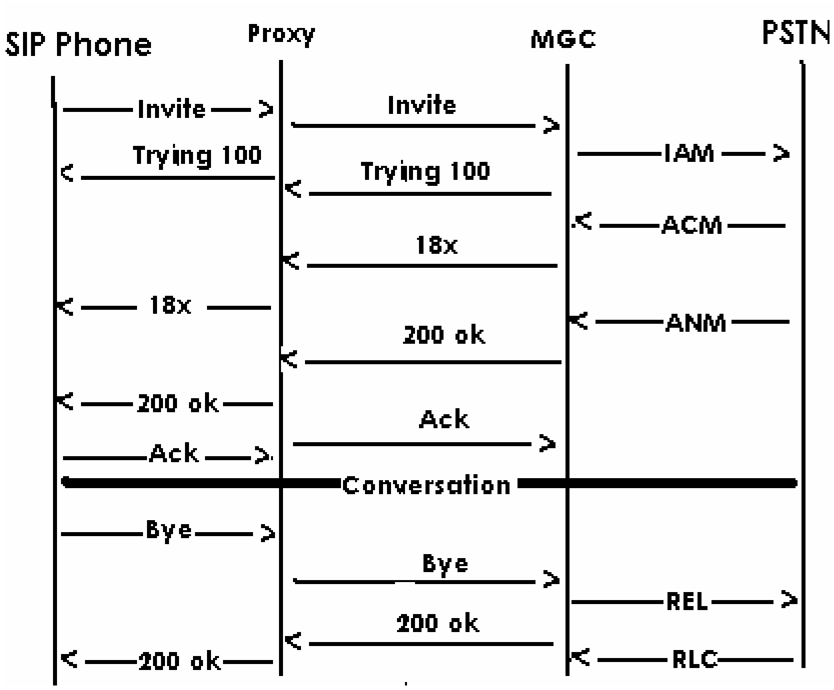


Figure 2.11: IP origination - PSTN termination (Message Flow)

2.5.2 Roles of SIP-T and its Conduct

There are three different types of elements from a practical prospective in a session initiation protocol, voice over IP network.

SIP / VoIP network: Interconnection with the PSTN:

1 SIP signaling, Originators
2 SIP signaling, Terminators
3 From the originator to the terminator, there are intermediates, for routing SIP requests.

2.5.2.1 The starting point (Originator)

The generation of SIP call setup requests (invites) is the responsibility of the originating User Agent Client. In the PSTN network when a call initiate, User Agent Client will be the gateway. If UAC is not the gateway then some local SIP ends is the UAC. In both cases the originator can never foresee or find that what will be the terminator. That is whether the last destination of call is in a Public switch telephone network or in a SIP network.

If the calls are starting in the PSTN network then the gateway which is the originating gateway does some important and essential steps for save the ISUP information (related data). Originator preserves all this data through encapsulation process and encapsulate in SIP requests it makes. Then it devises the headers of SIP Invite request. It does all this from ISDN-user part parameters which it obtained from PSTN networks. For instance setting may "To: header field in the Invite request to dialed no. the called party of the acknowledged ISUP IAM

SIP phone starts the Voice over IP calls in some cases. When a SIP phone originate the call it sends the request to the proxy known as SIP proxy, which is liable for steering request to a proper destination. At the UAC there is no ISUP message to encapsulate because there is no PSTN interface. The call can ended up in the telephone network and it must signal ISUP to happen, the starting node has no way to foresee this and it will unwise to want that all SIP Voice over IP agents have ability for producing ISUP. Therefore IP ends have no responsibility to produce encapsulated ISUP. So the originator needs to make the SIP signaling when carry out ISUP encapsulation and translation. When it is possible (it means here that when the PSTN is the originator of calls).

2.5.2.2 The Ending point (Terminator)

SIP calls consumer is known as SIP-T terminator. A typical SIP user agent is a terminator and it is the standard. It can be a gateway that works with PSTN or it can be a SIP phone. When the calls terminate at PSTN end the egress gateway ended up all the calls at its PSTN interface. For signaling, the terminator produce ISUP suitable from incoming SIP message to PSTN.

The main Terminator requirements are as follow:

Typical (standard) SIP processing
Explanation and interpretation for encapsulated ISUP and it is only for gateways,
The other requirement should be the support for multipart MIME and Graceful handling of unknown MIME content and it is for non-gateways only.

2.5.2.3 Agents: Intercede / Intermediates

The purpose of intercedes are to route the messages to each another. It also routes the gateways and SIP messages. Intermediates works like proxy servers. The forwarding conclusions for SIP are makes by proxy servers. Each proxy server plays individual role for forwarded decision. These decisions are based on a variety of headers values or routable basics calls elements. When a request comes at servers that has to forward, for this some extra features are commenced by SIP-T. These extra features show the way to new features and requirements for intercede. For a clear concept of SIP-T, the noticed thing was the feature clearness, transparency of ISUP and it did. And this thing (the clear image of transparency) can be seen by the compatibility among the ISUP variants of the starting and ending points of PSTN interfaces.

The ISUP variants those encapsulated with requests might be the interests of proxies. When the call ended up at any point which gives a very good closeness to the final destination is also a key consideration. The preference among the outcome of results is a substitution among simplicity of process and charge. The requirement of getting a reasonable rate may say that a SIP-T call covers dissimilar PSTN interfaces (SIP bridging between different gateways that don't hold any ISUP variants in common).

2.5.2.4 Behavioral Requirements Summary

If an ISUP request is acknowledge at SIP-T starting point that is also a gateway then it performs encapsulation as well as translation ISUP. It doesn’t care in spite where the originator makes an idea that the request will ended up.
The ended point that is the terminator if doesn’t come into know ISUP, it doesn’t pay

attention when it completes typical SIP progressing. And if it understands the ISUP and require to send a signal to the PSTN then use again the encapsulated ISUP and if it be aware of the variant also.

These are the some steps that the terminator should perform.

- First of all from the message it takes out the ISUP and uses this ISUP as a message pattern. One thing should be noted here that if no ISUP exit in the message then the gateway should use a recognized template.

- The headers of SIP requests are translated by the terminators into ISUP parameters that overwrite any values in the message pattern.

- Terminator also applies any local rules and procedures for counting parameter.

- The call must be routed by the intercedes that based on the selection of routable components in the SIP headers.

2.5.3 Parts and content elements of SIP-T Protocol

Following sections describes the parts of SIP-T that make available the protocol purpose involved by the requirements.

2.5.3.1 The kernel (The most essential SIP part)

The SIP-T known as sip for telephony uses the methods and procedures of SIP, defined by RFC 3261 which are followings.

SIP Requests:

There are six fundamental request or method types.

INVITE that is session establishment
ACK that Acknowledge an INVITE request
BYE that terminate a session
CANCEL that cancels establishing of a session
REGISTER that communicates user location, host name and IP.
OPTIONS that corresponds information about the potentials of the calling and receiving SIP phones.

SIP responses:

There are six classes by which SIP requests are answered

1xx these are informational responses, one is 180, which means ringing
2xx are success responses
3xx are redirection responses
4xx are request failures
5xx are server errors
6xx are global failures.

2.5.3.2 Encapsulation

The most key requirement of SIP-T is encapsulation (putting up in a nut shell, summing up) of the PSTN signaling. SIP-T uses multiple MIME bodies to make possible SIP messages to hold multiple payloads (Session Description Protocol or ISDN User part etc.). The ISUP MIME type make possible receiver to be familiar with the ISUP type in the most speedy possible way.

2.5.3.3 Interpretation

Translation consists of all side of signaling protocol conversion among SIP and ISUP and there are fundamentally two parts to the problem of translation:

1. Message Mapping of ISUP and SIP:

This explains a mapping among ISUP and SIP at the message stage. In SIP-T deployments gateways are trusted through the assignment of making a definite ”integrated services digital network user part” message for each SIP message received and vice versa, it is very important to state the set of laws that oversee the mapping among ISUP and SIP messages.

2. Mapping of ISUP parameter and SIP header:

A SIP request that is used to make out a telephone call should carry information that facilitate it to be suitably routed to its target mean ending points or destinations by proxy servers in a SIP network. An example if this can be given that, a call starting user dialed a telephone number. It is essential to regulate a set of practices that identify the process for transformation of information from ISUP to SIP. The example of this can be that a called party number in an ISUP IAM have to map on the SIP. This matter turn into naturally extra complex by the reality that the headers of a SIP request particularly an INVITE may be transformed by mediators and that as a result, the SIP headers and encapsulated ISUP bodies come to state contradictory values in fact, a component of the encapsulated ISUP may be turn into unrelated.

2.5.3.4 Support for mid call signaling

SIP does not contain at all supply for carrying out any mid call control information that is produced all through a session. The INFO routine supposed to be used intended for this purpose. It should be noted here that even so that INFO is not fit for supervision and managing overlap call.

2.6 RTP Real time protocol

Real-time Transport Protocol (RTP) is defined in RFC 3550 [27] and is a simple protocol to perform real-time data. It is also a protocol, PSTN (RTP Control Protocol), which provides feedback from other members of the RTP session QOS connection parameters. The RTP protocol is independent of protocol and network transport protocol that runs. Typically, RTP runs on top of UDP / IP, which is also used in this project. The RTP protocol is described first and second PSTN protocol is described below.

Many packets arrive out of service because the transmission of RTP packets over UDP / IP does not guarantee the correct arrival. Detection of an out-of-order received packets; the packets can be re-adapted for mode. The first order is a random number, which is reinforced by the next RTP packet. Timestamp field is used to determine at what time a package. The time stamp value is not unity, but the solution of this meter must be high enough to ensure a package is played. Real-time audio, the meter is usually increased each sample. This means that if an RTP packet contains data of 160 samples of audio (which says nothing about the size in bytes of charge), the count rose to 160 after the RTP packet.

Chapter 3

Simulation-1
TCP, UDP and SCTP

SIMULATIONS

3.1 Introduction

All the simulations are carried out using network simulator (ns-2) [13]. In second step of simulations the traffic is with constant bit rate. And our transports are TCP, UDP and SCTP. And we measured the delay, packet loss and through which are all Quality of services. The packet size is 1000 bytes and channel capacity is 0.2 Mb. We added some awk script and some C code for final results.

3.2 Simulation Scenarios

In order to analyze the packet loss, delays and average throughput offered by transport protocols some scenarios are simulated. One is to study the effect of traffic on delays and second to observe the effect of various packet loss conditions on delays.

3.3 Transmission Protocols Simulations

3.3.1 TCP

The packet size is 1000 byte and channel capacity is 0.2 Mb. The tcl script runs over ns and we found a .trace file and a nam file then added some awk script and found required trace file. Then got the results and generate the graphs shown in fig 4.2.

```
set ns [new Simulator]

$ns colour 0 blue
$ns colour 1 red
{
set n0  [$ns node, node 1]
set n1  [$ns node, node 2]
set n2  [$ns node, node 3]
set n3  [$ns node, node 4]
}
set f[open tcp.trc w]
$ns trace-al $f,
```

```
set nf[open tcp.nam w]
$ns namtrace-all $nf,




$ns dvplex-link $n0, $n1 0.2Mb 10ms DropTail
$ns duplex-link $n0, $n2 0.2Mb 10ms DropTail
$ns duplex-link $n1, $n3 0.2Mb 10ms DropTail
$ns duplex-link $n2, $n3 0.2Mb 10ms DropTail


set tcp [new Agent/TCP]
$tcp set class_1
set sink [new Agent/TCP Sink]
$ns atach-agent $n0 $tcp
$ns atach-agent $n3 $sink
$ns conect $tcp, $sink


set cbr {new Application / Traffic / CBR.
$cbr set packatSize_ 1000
$cbr attach-agent $tcp

$ns at 1.0 "$cbr start"
$ns at 29.0 "$cbr stop"


$ns at 30.0 ‘finish’

proc finsh {}
      globel ns f nf
      $ns flesh-trace
      close $f,
      close $nf,

      puts ‘runing nam...’
############################################
exec awk {
      {
if ($1="+" && $3==0 && $4==1 && $5=="tcp" && $6==1040) ||  ($1=="r"
&& $3==1 && $4==3 && $5=="tcp" && $6==1040)  {
```

```
              print $1, $2, $11
      }
       }
         } tcp.tr > sendreceive.tr
##############################################
      exec ./delay sendreceive.tr res1.tr res2.tr


      exec nam tcp.nam &
      exit 0
}

$ns run
```

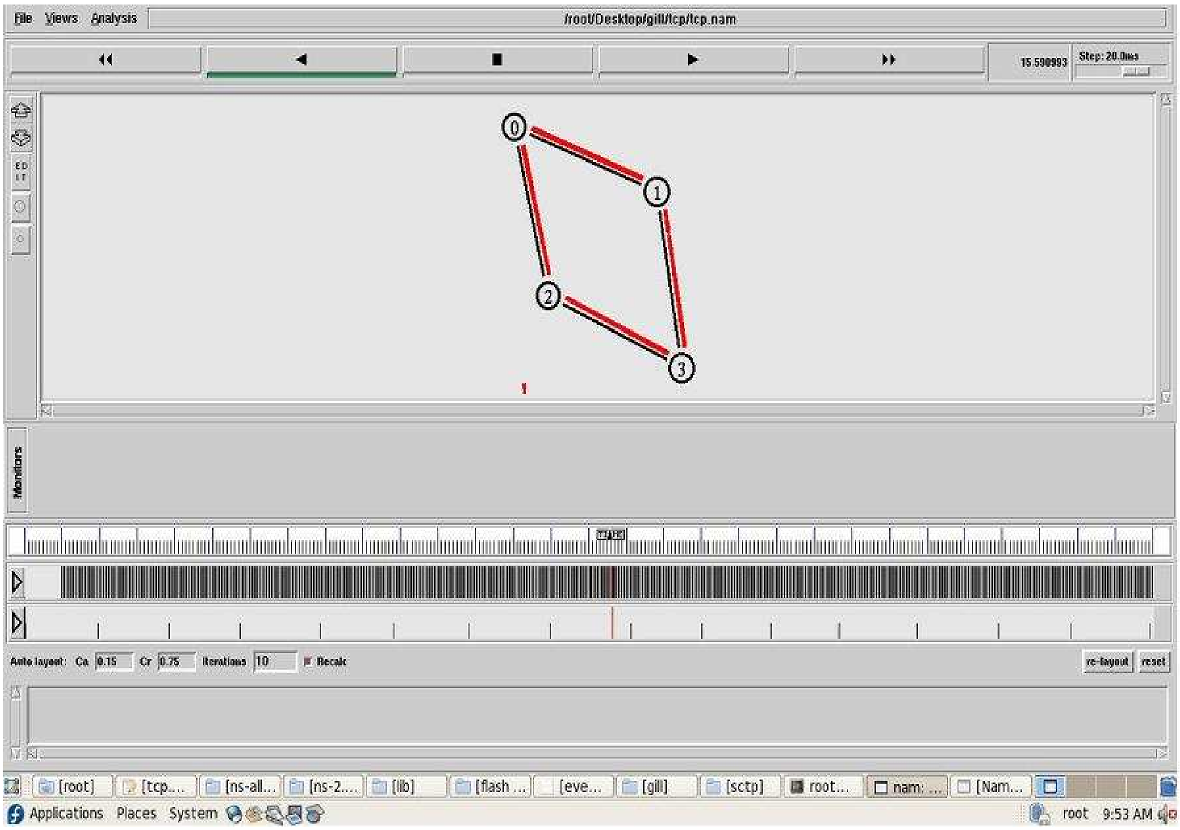


Figure 4.1 TCP.nam

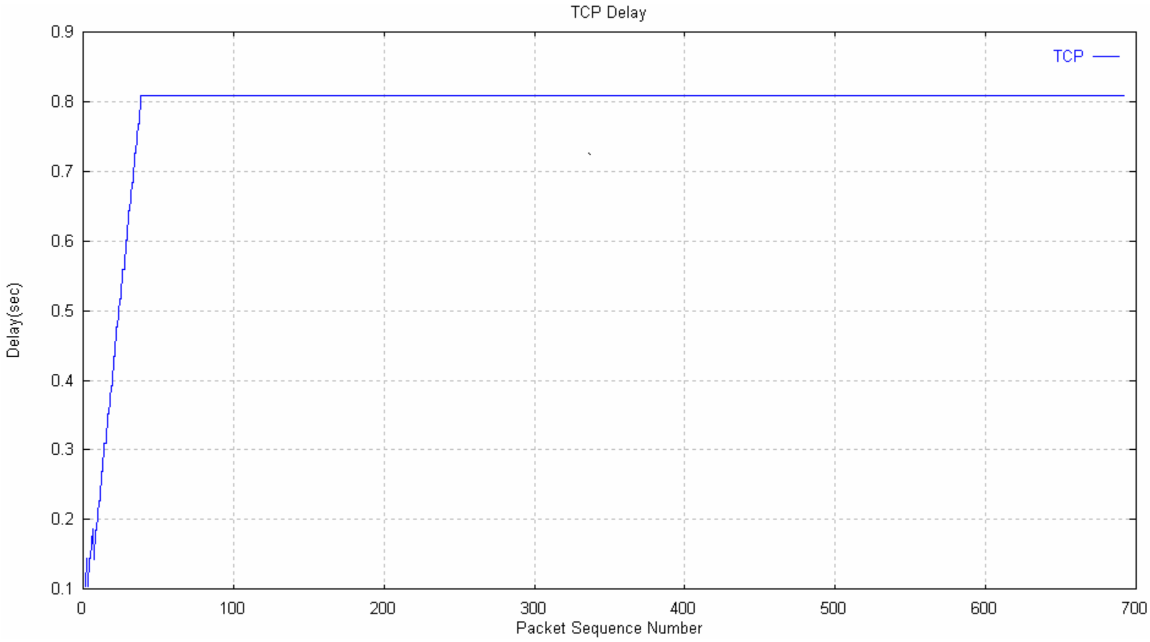


Figure 4.2 TCP delays

The time at witch a message is en-queued at transport layer at node 0 is subtracted from the time when it is delivered to application at node 3 to get delay for a particular request.

Number of packet read: 1405
Number of packet sent: 712
Number of packet received: 693
Packet lost: 19
Average delay of packets: 0.787624
Variance of delay is: 0.010288
% Throughput: 97.331460

3.3.2 UDP

The packet size is 1000 byte and channel capacity is 0.2 Mb. The tcl script runs over ns and we found a .trace file and a nam file then added some awk script and found required trace file. Then got the results and generate the graphs shown in fig 4.4.

```
set ns [new Simulator],

$ns color 0 blue
```

```
$ns colour 1 red
$ns colour 2 white

{
set n0 [$ns node, node1]
set n1 [$ns node, node2]
set n2 [$ns node, node3]
set n3 [$ns node, node4]
}

set f [open udp.trc w]
$ns trace-all $f,
set nf [open udp.nam w]
$ns namtrace-all $nf,


$ns duplax-link $n0 $n1 0.2Mb 10ms DropTail
$ns duplex-link $n0 $n2 0.2Mb 10ms DropTail
$ns duplex-link $n1 $n3 0.2Mb 10ms DropTail
$ns duplex-link $n2 $n3 0.2Mb 10ms DropTail


set udp [new Agent/UDP]
$udp, set class_ 1
set sink [new Agent / Null]
$ns attach-agent $n0 $ udp
$ns attach-agent $n3 $sink
$ns connect $udp $sink

set cbr [new Application / Trafic / CBR]
$cbr set packetSize _ 1000
$cbr attach-agent $udp

$ns at 1.0 "$cbr start"
$ns at 29.0 "$cbr stop"

$ns at 30.0 "finish,"

proc finish {},
      globol ns f nf
      $ns flush-trace
      clase $f
      close $nf
```

```
      puts "running nam..."


######################################################
exec awk {
      {
if {($1=="+" && $3==0 && $4==1 && $5=="cbr" && $6==1000) ||  ($1=="r"
&& $3==1 && $4==3 && $5=="cbr" && $6==1000)}  {

              print $1, $2, $11
      }
       }
         } udp.tr > sendreceive.tr
########################################################
      exec ./delay sendreceive.tr res1.tr res2.tr
      exec nam udp.nam &
      exit 0
}

$ns run
```

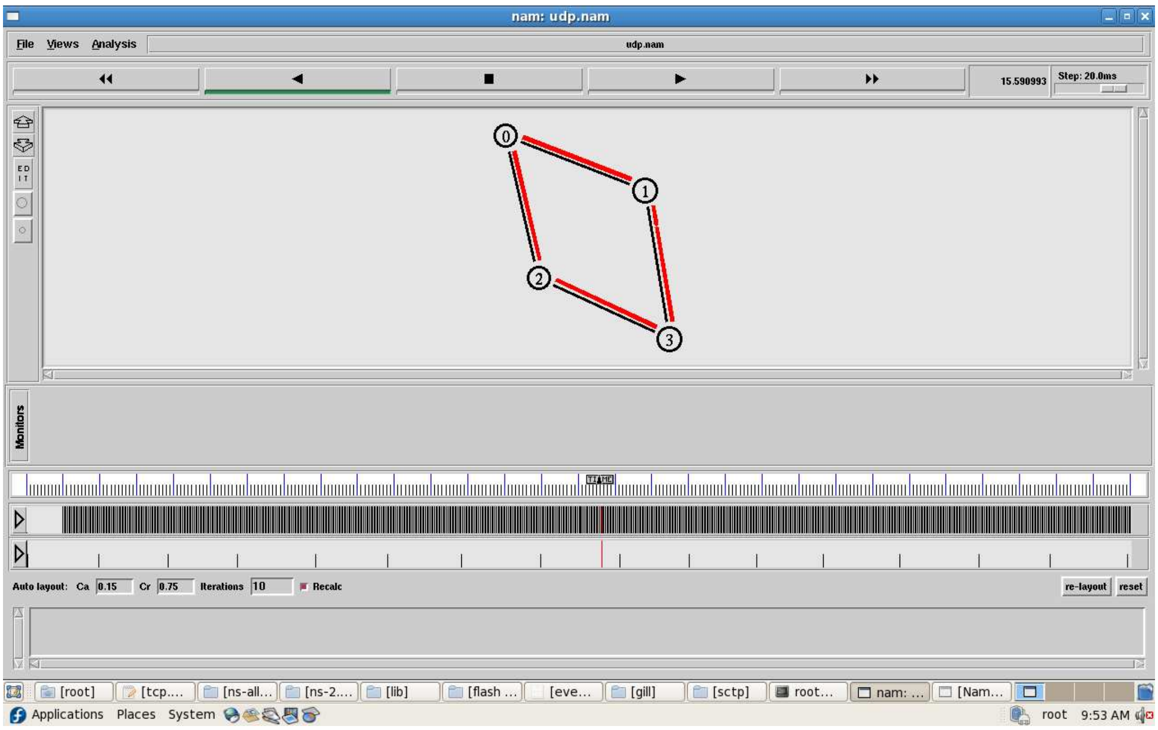


Figure 4.3 UDP.nam

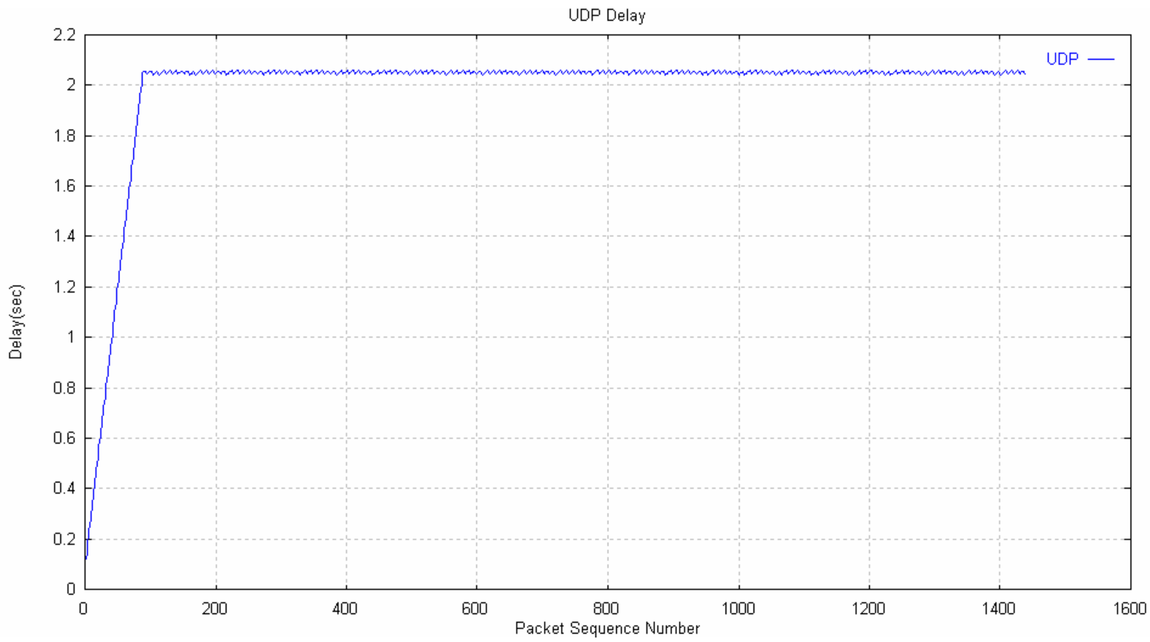


Figure 4.4 UDP delay

We can see here that the delay is increasing in term of time w.r.t TCP that 0.82 and in UDP it is upto 2.05.

Number of packet read: 2292
Number of packet sent: 1569
Number of packet received: 723
Packet lost: 846
Average delay of packet: 1.930832
Variance of delay is: 0.142850
% Throughput: 46.080503

3.3.3 SCTP

The packet size is 1000 byte and channel capacity is 0.2 Mb. The tcl script runs over ns and we found a .trace file and a nam file then added some awk script and found required trace file. Then got the results and generate the graphs shown in fig 4.6.

```
Trace set show_sctphdr_ 1
```

```
set ns [new Simulator]

$ns colour 0 blue
$ns colour 1 red
$ns colour 2 white

set n0, [$ns node, 1]
set n1, [$ns node, 2]
set n2, [$ns node, 3]
set n3, [$ns node, 4]

set f [opn sctp.trc w]
$ns trace-all $f
set nf [open sctp.nam w]
$ns namtroce-all $nf

$ns duplax-link $n0 $n1 0.2Mb 10ms DropTail
$ns duplex-link $n0 $n2 0.2Mb 10ms DropTail
$ns duplex-link $n1 $n3 0.2Mb 10ms DropTail
$ns duplex-link $n2 $n3 0.2Mb 10ms DropTail


set sctp [new Agent/SCTP]
$ns attach-agent $n0 $sctp


set sctpsink [new Agent/SCTP]
$sctpsink set use DelayedSacks _ 0
$ns attach-agent $n3 $sctpsink

# connect both agents
$ns connect $sctp $sctpsink


set cbr [new Application/Traffic/CBR]
$cbr set packetSize_ 1000
$cbr attach-agent $sctp

$ns at 1.0 "$cbr start"
$ns at 29.0 "$cbr stop"


$ns at 30.0 "finish",
```

```
proc finish {} [
      global ns f nf
      $ns flush-truce
      close $f
      close $nf

      pvts "running nam..."

############################################
exec awk {
      {
if  {($1=="+"  &&  $3==0  &&  $4==1  &&  $5=="sctp"  &&  $6==1048)  ||
($1=="r" && $3==1 && $4==3 && $5=="sctp" && $6==1048)}  {

              print $1, $2, $12
      }
       }
         } sctp.tr > sendreceive.tr
##############################################
      exec ./delay sendreceive.tr res1.tr res2.tr
      exec nam sctp.nam &
      exit 0
}

$ns run
```

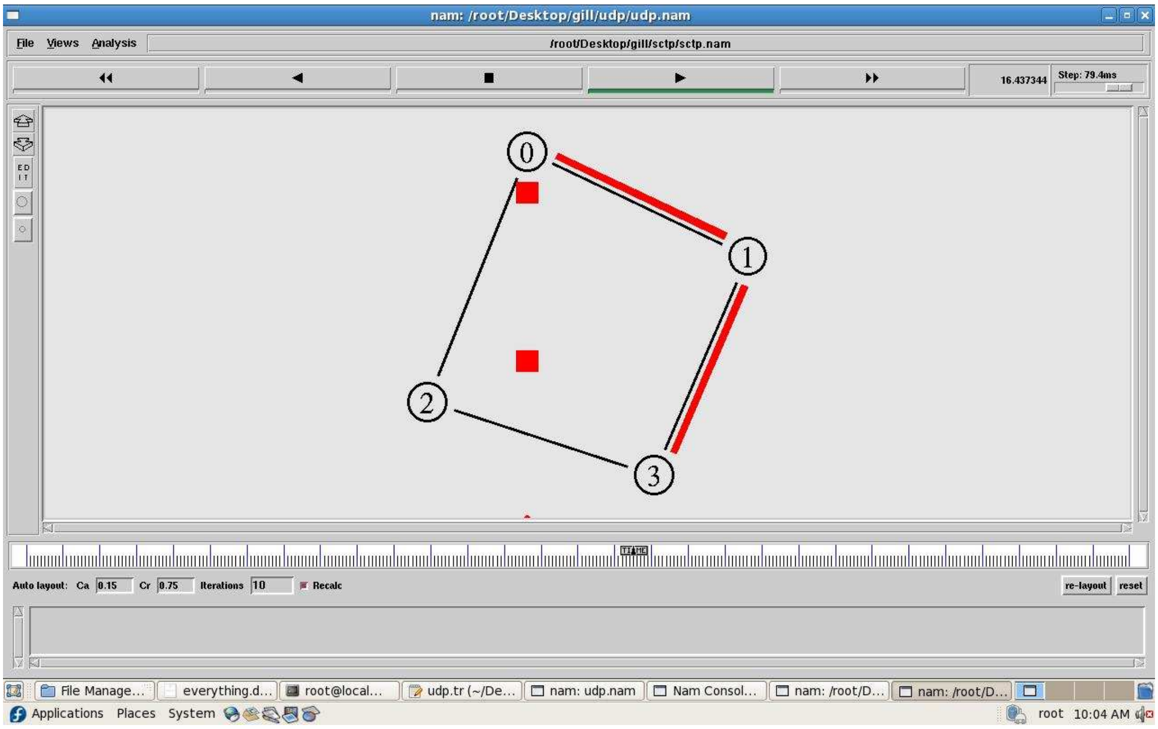


Figure 4.5 SCTP.nam

Number of packet read: 1437
Number of packet sent: 750
Number of packet received: 687
Packet lost: 63
Average delay of entries: 1.680858
Variance of delay is: 0.315453
% Throughput: 91.6

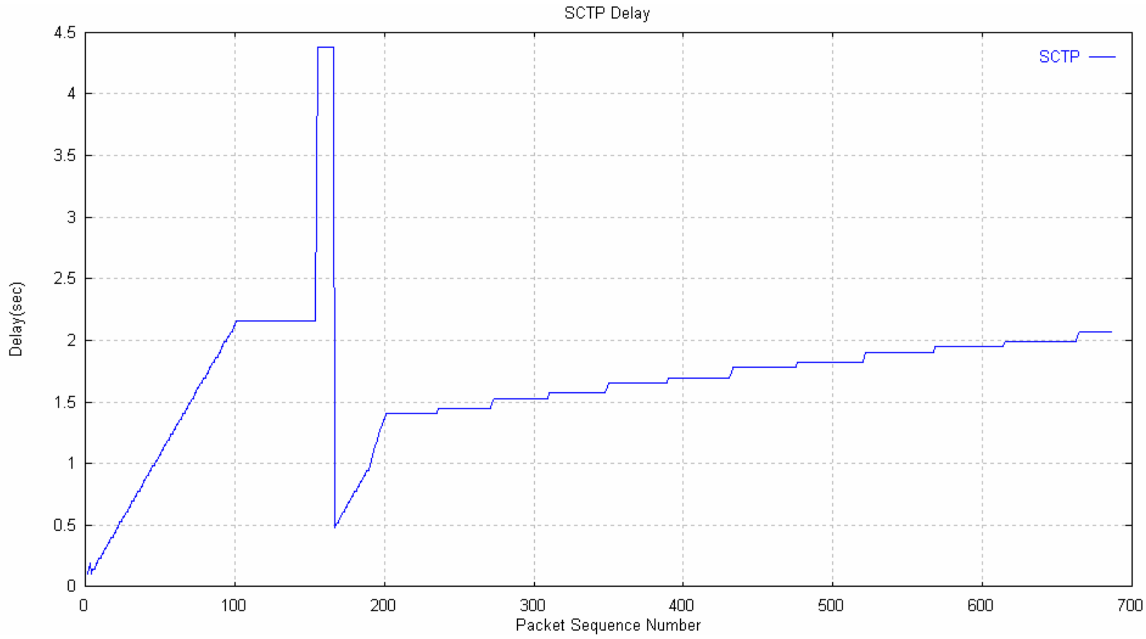
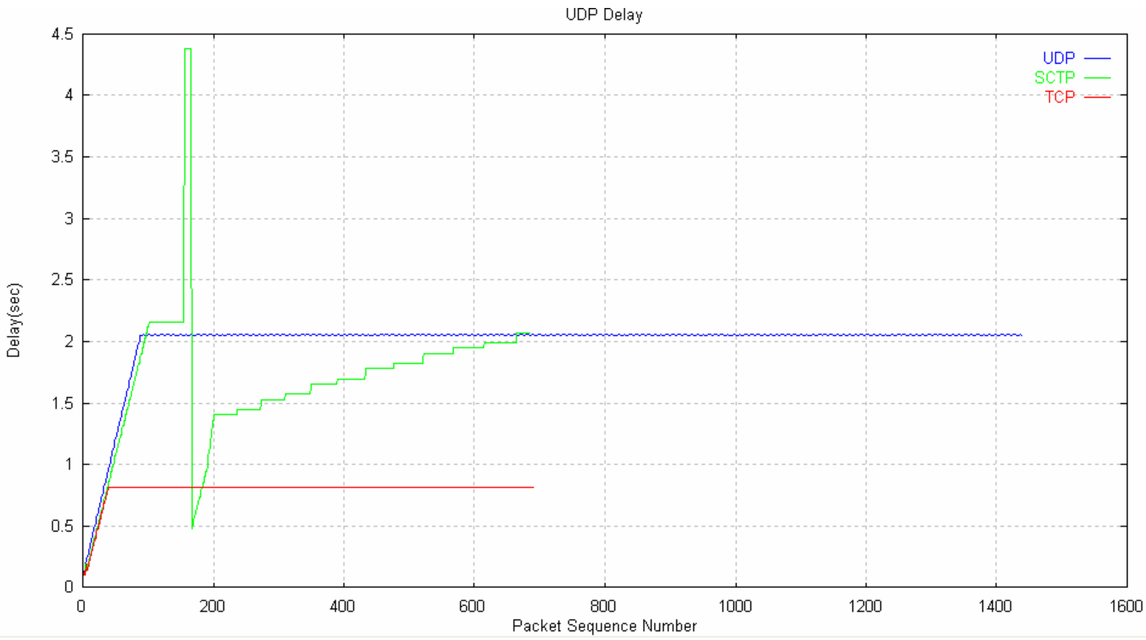


Figure 4.6 SCTP delays

We can see that delays are increasing up till packet no 100 that is 2.3 then constant to 150 then 150 to 160 it boosted up 4.4 then minimize to 0.5 then it is increasing slowly till last packet.

3.4 Comparison of TCP, UDP and SCTP



Protocol	Sent	Receive	Packet loss	Avg. Delay	Variance of delays	% Throughput
TCP	712	693	19	0.7876	0.01028	97.331
UDP	1599	723	846	1.9308	0.1428	46.080
SCTP	750	687	63	1.6808	0.3154	91.6

Depending upon the bandwidth, if we do more results, packet lost decrease. And SCTP looks in a better manner.

Chapter 4

Simulation-2

SIMULATIONS

4.1 Introduction

All the simulations are carried out using network simulator (ns-2) [13]. To add SIP support in ns-2 we have used the sip patch developed by Michele Fasciana [14], having the capability of running SIP over both UDP and SCTP including application layer retransmissions for SIP running over UDP. It behaves as a SIP entity to setup bidirectional SIP dialogs. We have used this patch to simulate SIP traffic that originates traverses and terminates on the internet.

Figure 4.1 shows the network topology we used for the simulations. This is similar topology used by Lulling et al. [11] and also resembles with the one used by Camarillo et al. [7] Nodes 1 and 2 are buffer limited droptail routers. Nodes 4 and 5 are TCP source and sink respectively carrying FTP traffic, used in specific scenario. Nodes 0 and 3 simulate SIP entities (namely MCG's) exchanging SIP messages in an active session. Link between nodes 1 and 2 is the only bottleneck link on witch traffic from both sender endpoints 0 and 4 competes in relevant experiment. Since the actual scenario could involve larger number of intermediate hops the bandwidth on the bottleneck link is chosen to provide meaningful observations. The delays values are chosen so as to make total delay 45 ms witch represents the transmission delay between a SIP server in US and another some where in Europe.

The case simulated for SIP messages exchange involves congestion and traffic competition problems, since it carries the PSTN signaling over IP without the involvement of any endpoint being on IP. Traffic generation is achieved using a stationary Poission model (generally used for telephony) generating SIP messages of 578-bytes at node 0. INVITE requests arriving at node 0 are forwarded to node 1, witch are responded firstly by 100 Trying on successful reception and later 180 Ringing response is returned when the ring is received from PSTN domain. Requests are generated at a rate of 120/s causing 50% of the link utilization. This is because in case of SIP-T an MGC is not just liable to send INVITE requests but also ACK/200OK, so the application traffic generation rate is double the call rate i.e. 240/sec

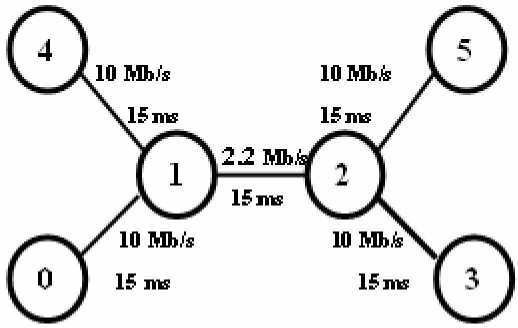


Figure 4.1: Network Topology

The basic observation is the delay of INVITE and relevant Ringing response using UDP and SCTP under different conditions. Throughput is also calculated in addition to delays with continuous competition of traffic on bottleneck link and varying buffer sizes at router.

4.2 Simulation Scenarios

In order to analyze the delays offered by transport protocols two basic scenarios are simulated. One is to study the effect of cross traffic on delays and second to observe the effect of various packet loss conditions on delays.

4.2.1 Competing Traffic

This test is designed to evaluate the effect on delays caused by the cross-traffic between node 4 and 5 for making both node 0 and node 4 to compete for the bandwidth on the bottleneck link.

SIP messages are generated by proxy at node 0 as mentioned previously and are forwarded to node 3 immediately. The FTP cross traffic at node 4 is generated using TCP Reno as this is considered to be the most apposite scenario for such purpose.

The time at witch a SIP message INVITE is enqueued at transport layer at node 0 is subtracted from the time when it is delivered to application at node 3 to get delay for a particular INVITE request.

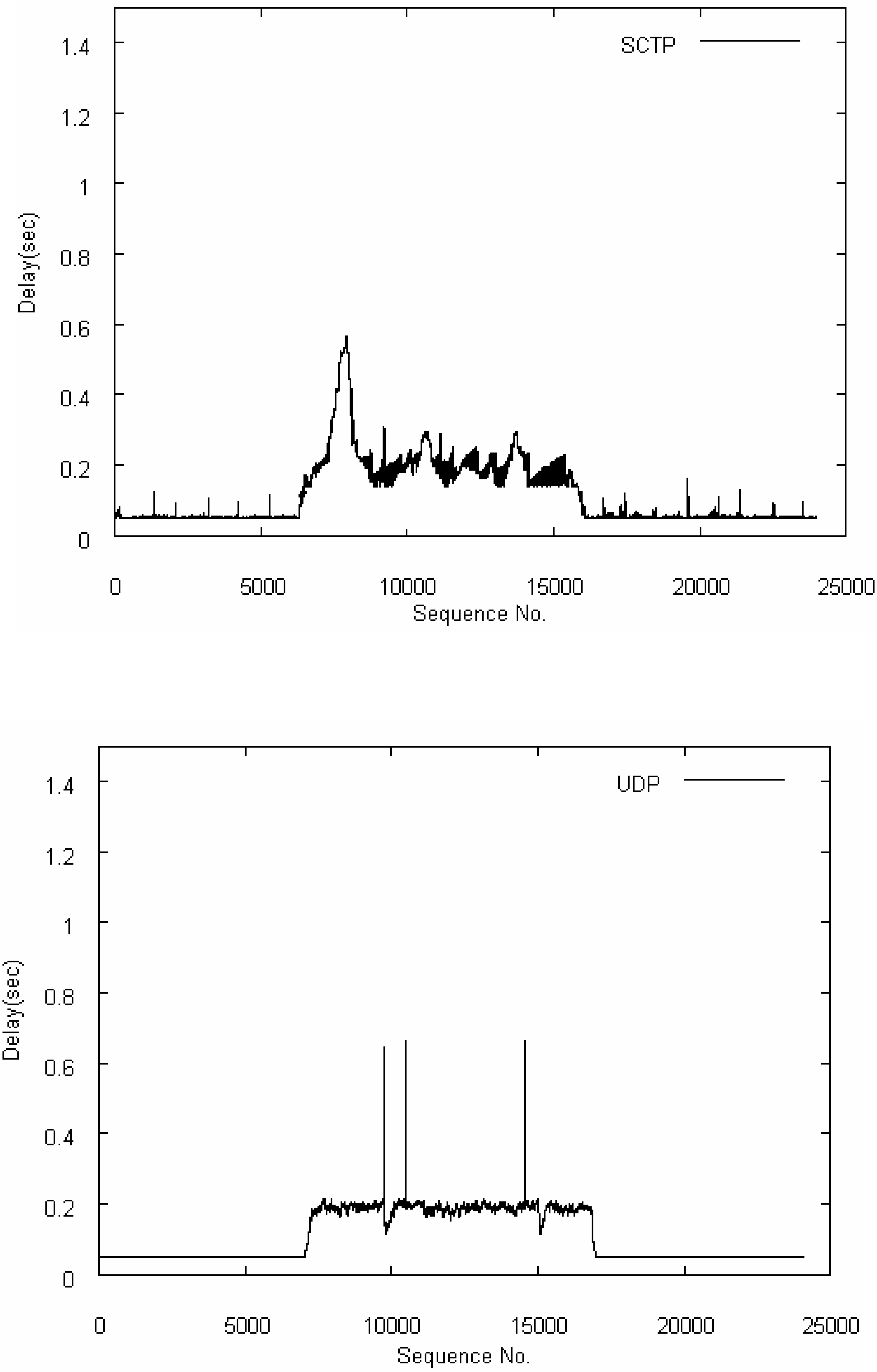


Figure 4.2: Effect of Competing Traffic

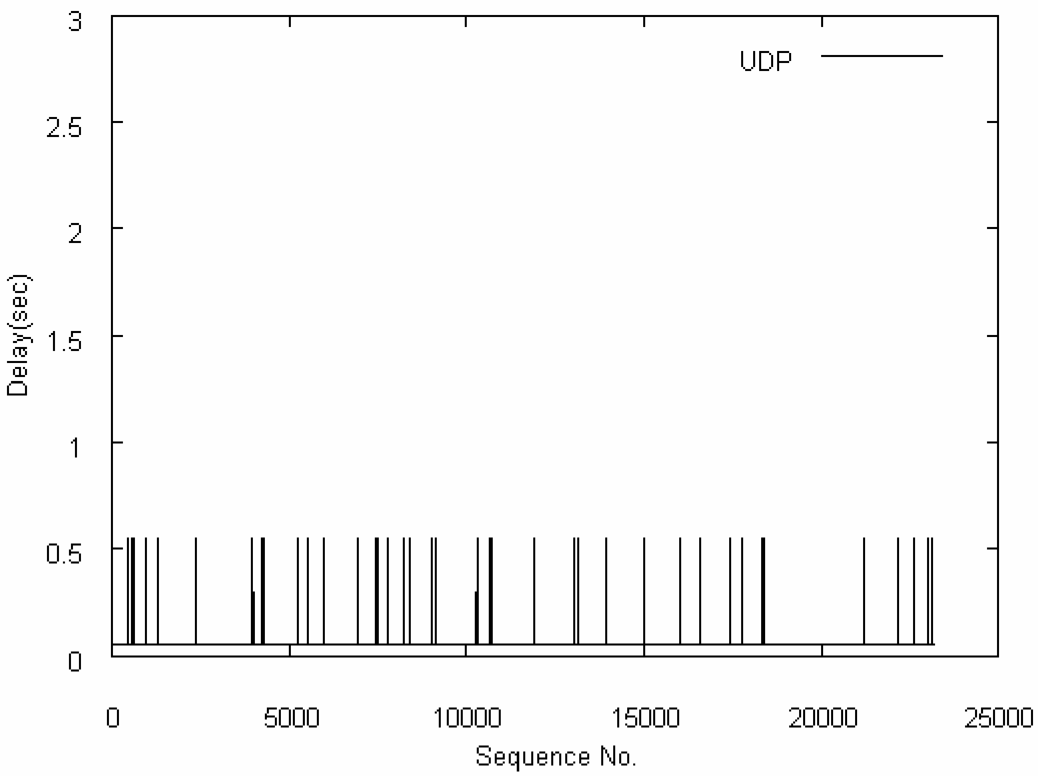
FTP traffic at node 4 using TCP Reno is generated for only a specific period of time. Results for this test are shown in figure 4.2 X-axis shows the Message Sequence Number whereas y-axis shows the time taken by INVITE SIP messages to reach the destination. It is visible that there is not any significantly big difference in delays achieved by both transport protocols.

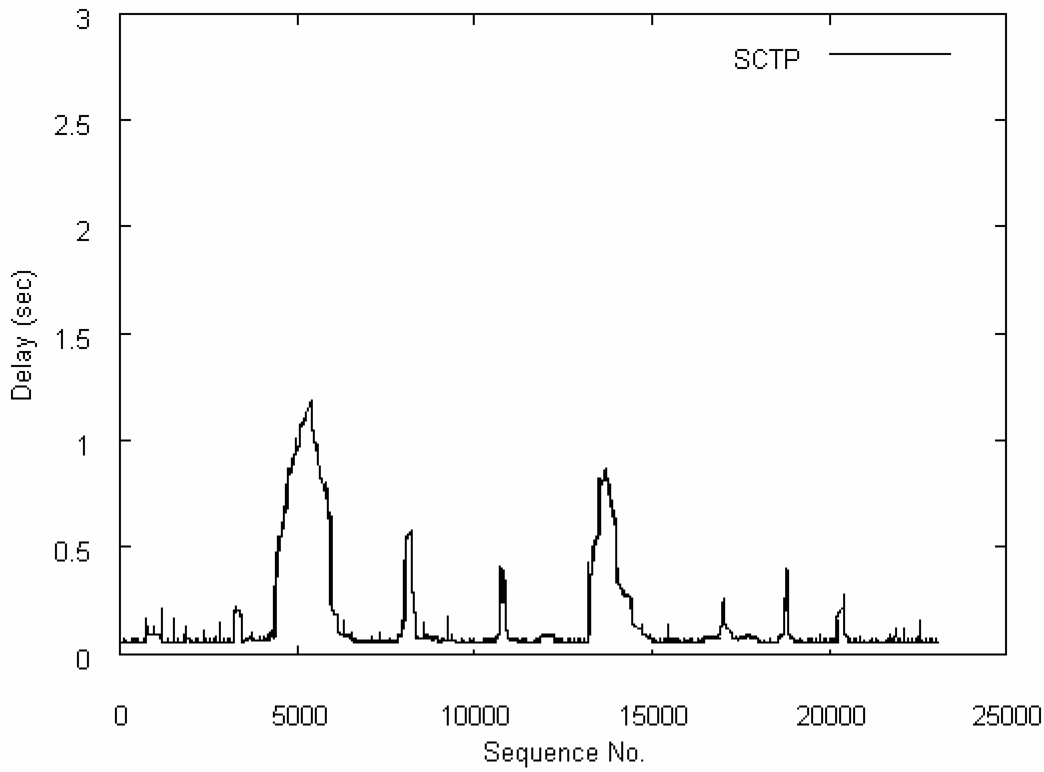
SCTP does face a situation where the packets see delay up to 800 ms with a bit of consistency, for a pretty short time though. In case of UDP delays remain consistent as it is a very basic attribute of UDP. It shows big bars only in case of retransmission for the packets witch had to be re-transmitted. Probably this is the biggest disadvantage of using UDP alongside many benefits witch are particularly favorable for the case of telephony signaling.

The general result from this experiment is that both UDP and SCTP perform well although the delays seen by SCTP are a touch on the higher side but still not significantly alarming, considering the overheads it has to see.

4.2.2 Packet Loss Effect

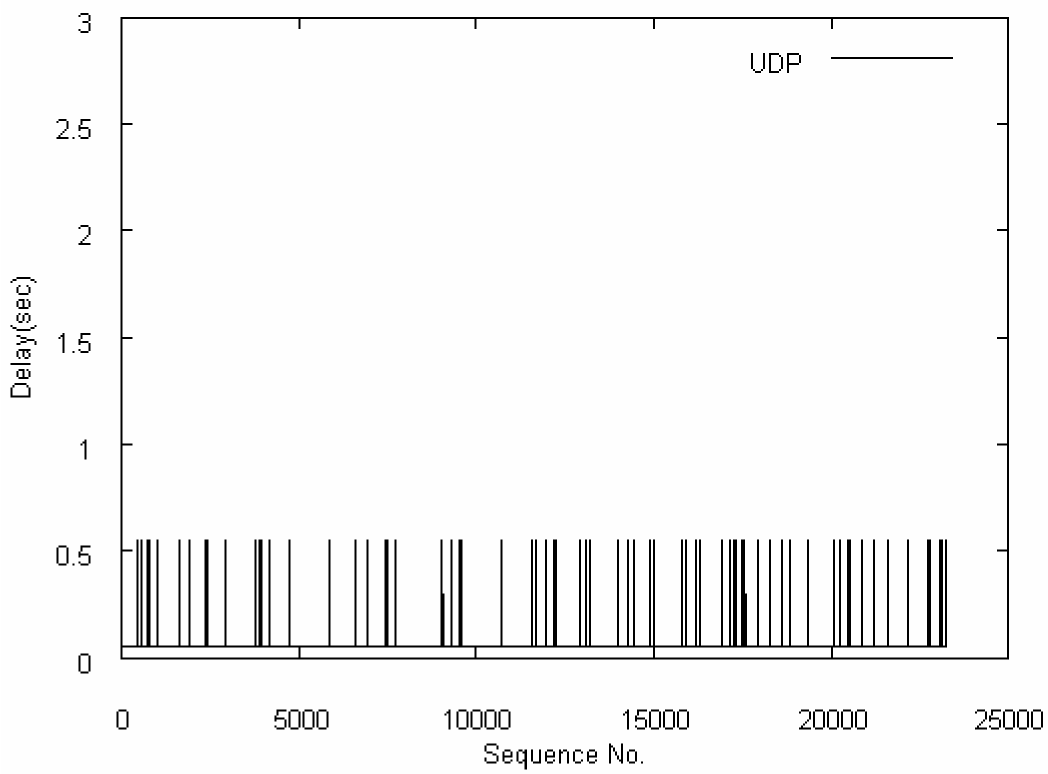
In order to compare the performance of UDP and SCTP under packet loss conditions, three simulations are run for both UDP and SCTP. The random packet loss rates 0.2, 0.3 and 0.3% are simulated at node 1.





The metrics used for delay calculations are same as described in the previous experiment. The results (shown in figure 4.3, 4.4 and 4.5) are interesting and distinguishing the performance given by both the protocols.

It can be seen in figure 4.3 that in case of 0.1% packet loss SCTP behavior remains in check. It does reach the limit of beyond 1 sec but it recovers after each abnormal event fairly soon. Whereas UDP has to rely on application layer retransmission witch lifts the delay to up to 600 ms each time a packet is dropped. Although the peak delay met by SCTP is more than that met by UDP, still SCTP is competitive enough to be considered as a potential candidate as on the whole the performance is not deteriorated totally.



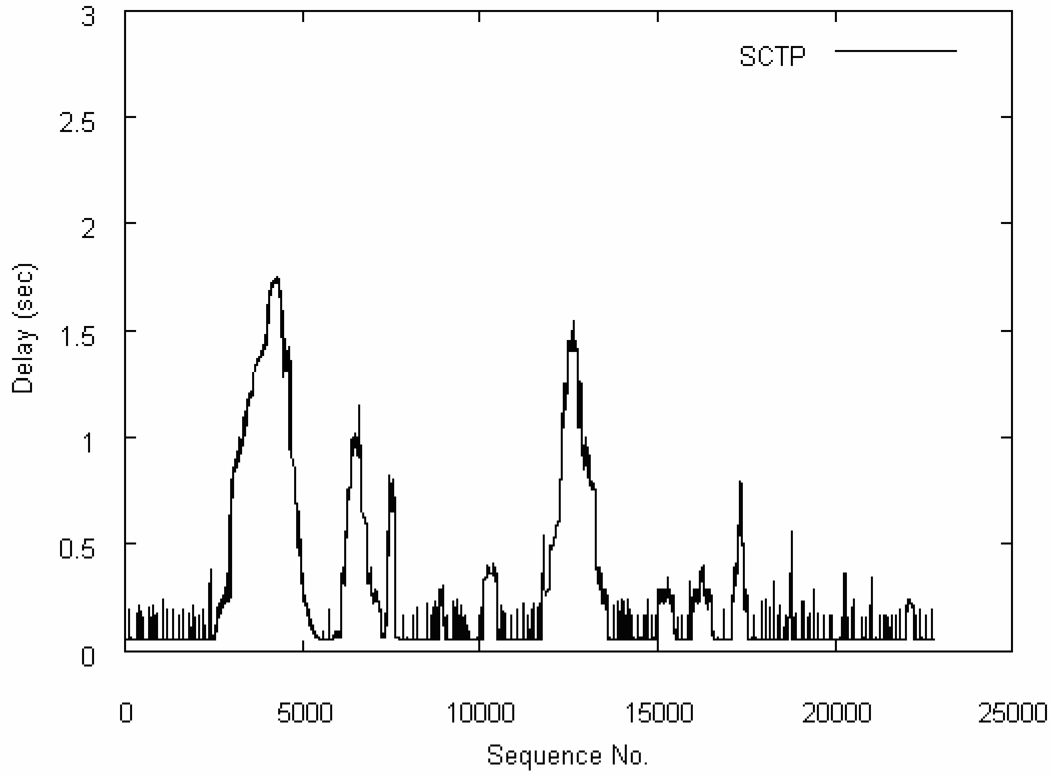
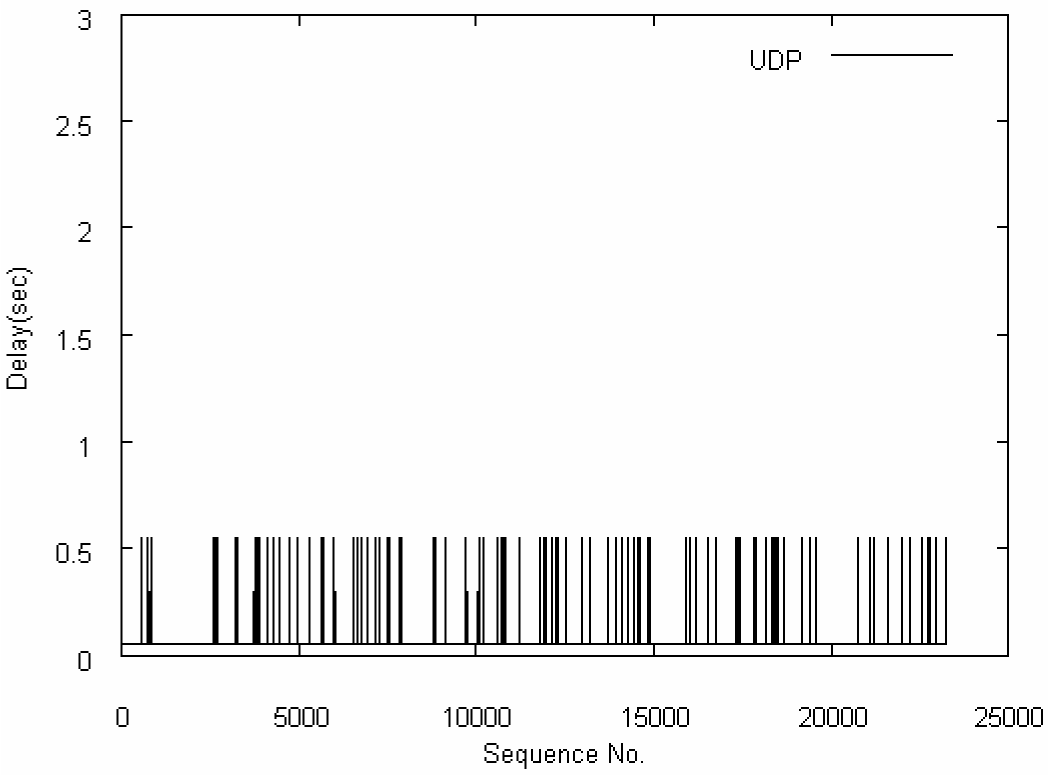


Figure 4.4 Effect of 0.2% random packet loss

Figure 4.4 shows the effect on performance of SCTP and UDP with a packet loss of 0.2%. Here we see that the delay starts mounting and the maximum delay extent reached by SCTP here is round 1.8 seconds mark, witch is pretty high especially considering the real time delivery needs of telephony signaling. UDP again relying on application layer retransmissions rarely had to retransmit a packet. The matter of fact is that 0.2% packet loss means that 2 packets out of every 1000 packets are lost. So out of some 23,000 packets UDP requires some 46 plain retransmissions, witch in case of SIP-T may be a little unevenly distributed in INVITE and ACK/200OK and it may well not be fully uniform, but still it does not put a significant effect on its overall performance.

On the other hand the same effect observable in SCTP is very different and significantly bulk. This rises from the fact that application layer keeps sending packets at regular intervals but transport layer hold these packets in its buffer and does not send these straight away as the sending is constrained by congestion window. Since the call rate is kept fixed, so any lost packets causes the congestion window to be halved witch further ruins the matters as INVITE is not the only contributing message, congestion window is effected by ACK/200OK as well..

Similarly in 0.3% loss rate the results go beyond 2.5 seconds mark. So the conclusion that can be drawn from this study of packet loss is that SCTP cannot live increasing packet loss, especially which occurs in bursts.



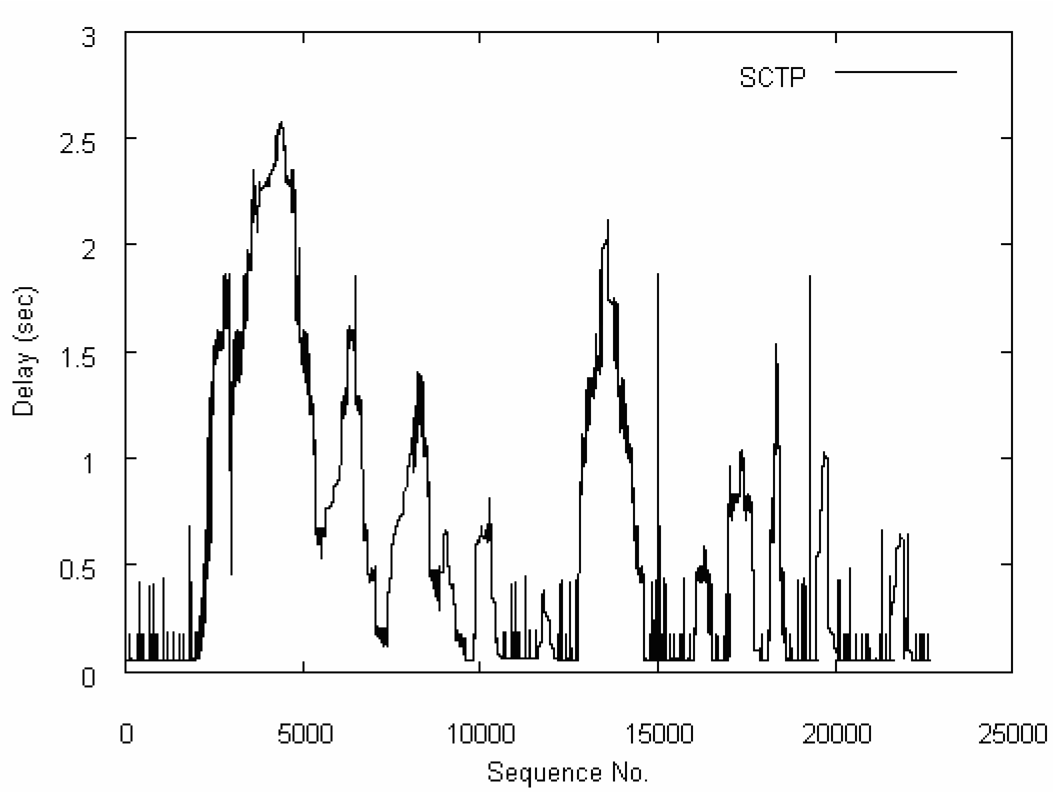


Figure 4.5 Effect of 0.3% random packet loss

The main difference here in SIP-T scenario and SIP Proxy is that in SIP Proxy the only message sent from sender end is INVITE and it receives the provisional responses “Trying” and “Ringing” from the recipient, whereas in case of SIP-T all the SIP messages regarding session establishment, management and terminations are dealt by an MGC. So here the normal message in addition to INVITE is ACK/200OK. The loss of ACK/200OK will make SCTP suffer more ion terms of delays as the packet loss increases and congestion window suffers more and more. To accommodate this packets are dropped at transport layer then, witch turns out to be a real performance blow.

4.2.3 Throughput

This experiment is designed to find the throughput achieved by UDP and SCTP by simulating continuous effect of cross traffic. The FTP traffic generated at node 4 uses TCP Reno as transport protocol. The other varying parameter used here is buffer size of router attached to node 0.

The graph below shows the throughput achieved by SCTP and UDP with buffer sizes 5, 20, 50, 100, 150, 200 and 250. It is again clear the UDP shows dominance especially with decreasing buffer size. UDP is able to send almost all the required number of messages within due time whereas throughput for SCTP varies greatly, being on the lower side with lesser buffer sizes and increases with increasing buffer sizes.

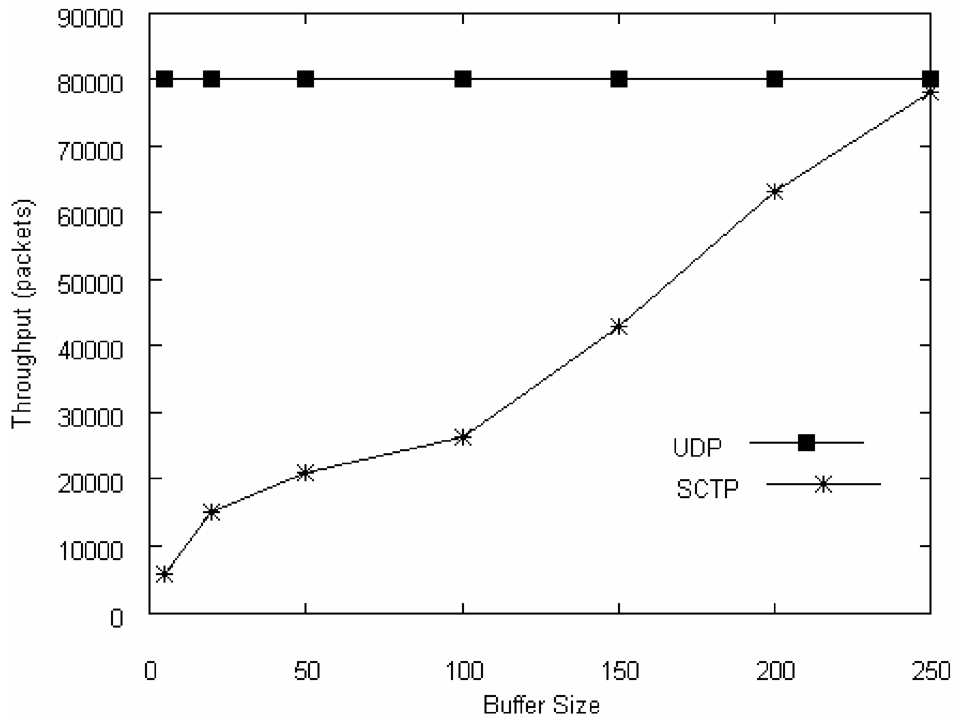


Figure 4.6: Throughput achieved by UDP and SCTP with varying buffer sizes facing continuous competition on bottleneck link

Again it is visible the UDP performs better and the reason in quiet obvious for this that UDP has no transport overheads, no flow control, no congestion control, no slow starts. It works in a flat out manner, keeping the thing very simple and straight forward, although not friendly with internet traffic of today with TCP's share of more than 80%.

Chapter 5

Conclusions

Conclusions

In first step we conducted some results to compare the performance of TCP, UDP and SCTP with traffic analysis. We kept the packet size of 1000 bytes and run the simulation with constant bit rates over all transport protocols and we had the channel capacity 0.2 Mb. As shown in figure when comparing the results we can see that TCP is performing the best with least number of packet loss then SCTP and then UDP. SCTP is best effort because its multi homing and multi association and packet delivery acknowledgement is time consuming but researches are going on and in future SCTP would be with some more advantages over TCP.

Secondly we conducted some experiments to compare the performance of SCTP and UDP for SIP dialog between two SIP entities, UDP being a dominantly used protocol whereas SCTP being a protocol designed for signaling. The main idea of this work is to present the idea of performance evaluation of transport protocols for session initiation protocol in the context of SIP-T using an MGC for bridging the IP and PSTN networks and mapping between ISUP and SIP messages.

First test was designed to compare the performance of two protocols, with cross traffic on bottleneck link. We observed more or less satisfactory performance in competing traffic with UDP and SCTP, although UDP has an edge being free from all sorts of transport overheads. But in the case of packet loss, where SCTP suffers a bit of delay variations UDP suffers from the effect of application layer retransmission.

Second experiment was designed to monitor the packet loss effect on the performance of protocols under study. With increasing effect of packet loss the performance of SCTP undergoes a severe degradation. This degradation in case of SIP-T (MGC) is greater in magnitude compared to the SIP Proxy case. UDP on the other hand keeps a consistent behavior as the packet drop has no effect to put on application. Same rate of packet loss in SCTP causes packets drops at transport layer and delays increase in a consistent manner. So it is very easily observable that SCTP has no comparison with UDP. Since internet traffic is burst in nature and it is difficult to predict traffic density and loss rates, in the same way it is not simple to give a clear verdict regarding choice of a transport protocol, but the kind of performance given by SCTP makes it clear that SCTP cannot be used with confidence since even the loss rate of 0.2% degrades it performance significantly compared to UDP.

Similarly throughput given by SCTP is significantly dependent on the buffer size and decrease in buffer size significantly degrades performance offered by SCTP. On the other hand UDP stands out even at a buffer size of 5 packets only.

So from this study it is concluded that in case of SIP-T SCTP is even worse performing compared to SIP Proxy scenario. UDP remains the most prolific protocol for carrying SIP messages.

Serious considerations are required to be made regarding protocol redesign that can best suitable to carry SIP signaling messages, considering one flaw of UDP, that it is not internet traffic friendly.

References

References

[1] J Postel User Datagram Protocol IETF RFC 768, August 1980

[2] J Postel Transmission Control Protocol IETF RFC 793, September 1981

[3] Howstuffworks "How IP telephony works": http://electronics.how-stuffworks.com/ip-telephony.htm (March 14, 2003)

[4] Softswitch consortium: http://www.softswitch.org

[5] Broadband Telecommunications Handbook, Rejis J. "Bud" Bates, 2nd Edition 2000, McGraw Hill Companies, Inc

[6] IETF Signaling Transport Working Group: "sigtran", www.ietf.org/html.charters/sigtran-charter.html

[7] R Stewart, Q Xie, K Morneault, C Sharp, H Schwarzbauer, T Taylor, I Rytina, M Kalla, L Zhang, and V Paxson Stream Control Transmission Protocol IETF RFC 2960, October 2000

[8] International Telecommunication Union (Visual telephone system and equipment for local area networks witch provide a non-guaranteed quality of service) Recommendation H.323, Telecommunication Standardization Sector of ITU, Geneva, Switzerland, May 1996

[9] 3GPP TSG SSA, IP Multimedia Subsystem (IMS) – Stage 2 (Release 5), TS 23.228.v5.6.0, 2002-09, www.3gpp.org

[10] ROSENBERG, J, et al: 'SIP Session Initiation Protocol'. KFC3261, IETF, June 2002

[11] M Handley and V Jacobson SDP session description protocol Request for Comments 2327, Internet Engineering Task Force, April 1998

[12] A Vemuri, J Peterson et al RFC 3372 SIP-T Context and Architectures September 2002

[13] G Camarillo, A B Roach, J Peterson et al RFC 3398

[14] R Alamgir, M Atiquzzaman, and W Ivancic, “Effect of congestion control on the performance of TCP and SCTP over satellite networks,” ESTC-2002, June 2002

[15] R Rajamani, S Kumar, and N Gupta SCTP versus TCP, Comparing the performance of transport protocols for web traffic, University of Wisconsin-Madison, May 2002 http://www.cs.wisc.edu/~raj/sctp/report.pdf (March 28, 2003)

[16] M Atiquzzaman, W Ivancic “Evaluation of SCTP Multistreaming overWireless/Satellite Links” 2003 IEEE

[18] KAME SCTP: www.sctp.be

[19] R Brennan, T Ravier, and T Curran, Experimental studies of SCTP multi-homing, Teltec DCU, Dublin 9, Ireland 2001 http://telecoms.eeng.dcu.ie/symposium/papers/B4pdf (March 28, 2003)

[20] M Mathis, J Mahdavi, S Floyd et al RFC 2018 - TCP Selective Acknowledgment Options October 1996

[21] Steven Cherry “VoIP is turning telephony into just another Internet application—and a cheap one at that” www.infineon.com/press/voip

[22] S Hezzler, M Welzl “Seamless transport service selection by deploying a middleware” Computer Communications 2006, Scince Direct

[23] G Camarillo, R Kantola H Schulzrinne, “Evaluation of Transport Prolocols for the Session Initiation Protocol“, IEEE Network, September 2003

[24] The Network Simulator, ns-2 www.isi.edu/nsnam/ns/

[25] M Lulling, J Vaughan (A simulation-based comparative evaluation of trensport protocols, for SIP) Computer Communications 29 2006 Science Direct

[26] X L Wang Et al “Applying PR-SCTP to Transport SIP Traffic” IEEE GLOBECOM 2005

[27] R Stewart, M Ramalho, Q Xie, M Tuexen et al RFC 3758 Stream Control Transmission Protocol (SCTP) Partial Reliability Extension May 2004

[28] G. D Marco, S Loretro “SCTP as a transport for SIP: a case study”,

[29] Linux Kernel Implementation of SCTP, lksctp.sourceforge.net

[30] L Alcuri, M Fasciana Transport Service for Session, Initiation Protocol in SIP-T Scenarios” IEEE Communications, 2004