The Voice over IP Challenge

An Empirical study of VoIP Deployment

and

Performance Challenges

A Project Report

by

Gurpreet Nanda
gnanda@ualberta.ca

Project Supervisor:
Dr. Mike MacGregor
Associate Professor
Faculty of Computing Science
University of Alberta
Edmonton, Alberta

Industry Partner:
Dr. Adil Kodian
General Manager
Rohit Connexion Ltd.
Edmonton, Alberta

Executive Summary

The Telecom sector is subject to a strict set of standards with regards to the performance and availability of its systems. It has therefore primarily been a domain of proprietary and standardized software and hardware. Traditionally telecom companies have resisted change.

With data overtaking Time Domain Multiplexed (TDM) voice as the primary component of traffic carried by the network, new innovations like Voice over IP (VoIP) are now gaining critical mass. VoIP enables carrying of voice over packet based data networks. Voice over IP (VoIP) is a disruptive killer application of the internet. In VoIP, voice is digitized, packetized and transported across long distances on the Internet. These packets are routed across the public internet to the destination where the reverse process is followed. In contrast, in traditional circuit switched telephony, also called Public Switched Telephony Network (PSTN) the process is to first create a temporarily dedicated circuit of 64kb/s between the two callers and route the digitized (but not packetized) voice data using non-multiplexed transport. VoIP trades off the high quality of using dedicated point to point resources by enabling packet level sharing of resources (by statistical multiplexing) between multiple users.

The key difference between VoIP and regular telephony therefore is this lack of resource sharing in the working capacity layer. VoIP reduces the cost of carriage of data by enabling the sharing of working channels via statistical multiplexing. Since VoIP uses existing Internet infrastructure, it enables low-cost creation and delivery of a variety of service types that can share common transport infrastructure and open standards. However, problems such as echo cancellation and reliability, which were solved for the regular telephony network a few decades ago now, need to be re-examined and essentially re-solved for the VoIP domain. VoIP also faces new problems due to queuing, propagation delay, bit errors, lack of synchronicity and so on. We hope to draw on the vast body of general knowledge and systems engineering experience with regular telephony to detect and suggest solutions to problems in VoIP.

In this project we simulate the performance of a specific VoIP system architecture using OPNET. We then follow it up with testing on a similarly configured production VoIP system setup at our Industry partner's premises. Specifically we study the jitter and echo problems related to high end-to-end delays in VoIP networks and their causes. Results show that the end-to-end delays can reach high values approaching 200 ms that cause jitter and echo related problems.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

PSTN	Public Switched Telephone Network
TDM	Time Division Multiplexing
ITU	International Telecommunications Union
LEC	Local Exchange Carrier
CO	Central Office
PBX	Private Branch Exchange
IETF	Internet Engineering Task Force
VoIP	Voice over Internet Protocol
RFC	Request for Comments
SIP	Session Initiation Protocol
RTP	Real-time Transport Protocol
UDP	User Datagram Protocol
RTCP	Real-time Transport Control Protocol
NAT	Network Address Translation
NAPT	Network Address Port Translation
STUN	Simple Traversal of UDP through Network Address Translators
COMEDIA	Connection Oriented Media
ICE	Internet Connectivity Establishment
SBC	Session Border Controller
TURN	Traversal using Relay NAT
SDP	Session Description Protocol
TCP	Transmission Control Protocol
IP	Internet Protocol
EC	Echo Canceller
AEC	Acoustic Echo Canceller
ACD	Automatic Call Distribution
DSP	Digital Signal Processing
CS-ACELP	Conjugate-Structure Algebraic-Code-Excited Linear-Prediction
UAC	User Agent Client
UAS	User Agent Server
QoS	Quality of Service
MGCP	Media Gateway Control Protocol
FTP	File Transfer Protocol

Chapter 1 Introduction

1.1 Problem Description

VoIP is a disruptive technology that has revolutionized the field of telecommunications with its flexibility, portability and myriad of possible services arising due to its use of open standards. However, there is a price to pay for that flexibility and open standards. Organizations anxious to take advantage of the cost savings associated with VoIP are jumping onto to the VoIP bandwagon without completely understanding the challenges associated with VoIP, leading to failed deployments or unsatisfactory performance. There is no doubt that the cost savings are there but there is a lot more to VoIP than that.

In this project, we research and highlight VoIP deployment challenges. We hope that the work done would help organizations looking at deploying VoIP technology to better understand the VoIP issues and make educated deployment decisions instead of just focusing on cost-savings.

1.2 VoIP Technology

1.2.1 Background in Telephony

The various technologies belonging to the traditional circuit switched telephony world are collectively referred to as PSTN (Public Switched Telephone Network). The digitized data between Local Exchange Carrier (LEC) switches also referred to as Central Office (CO) is carried over Time Division Multiplexed (TDM) based links. The transmission of this data is very fast and does not incur significant delays between the sender and the receiver. Various types of Codecs (Coder-decoder) are used to convert the subscriber analog signal to digital form by the CO but the most basic form is Pulse Code Modulation (PCM) in which the subscriber analog signal is sampled at a rate of 8000 samples per second and each sample is encoded using 8 bits.

The operation of these Codecs is very fast and does not contribute any delay. However, Echo is a concern over call paths involving long delays such as satellite and transcontinental calls, which is overcome by engaging echo-cancellers in these calls. A detailed analysis of echo is performed in Section 1.2.5 and Section 2.4.

Along with the transmission aspects of PSTN, there are a variety of call signaling standards/protocols used on the PSTN. Their discussion is beyond the scope of this report but essentially all have the basic function of setting up and tearing down a call, call accounting and with providing enhanced PSTN services.

Private Branch Exchanges (PBX) are also a component of PSTN. PBX are devices deployed by organizations to take advantage of the multiplexing capability of the devices and usually have proprietary signaling protocols created by vendors for their equipment. PBX is similar in function to a CO but differs greatly in the signaling and services that they provide to the internal subscribers in the organization but their basic function is call handling and processing. In this context, the IP counterparts of these PBX are also sometimes referred to as IP-PBX's. Their basic functionality remains the same – only the protocols and platforms have changed.

1.2.2 Packetizing Voice

VoIP technology basically involves digitizing the analog voice signals using a coding scheme also referred to as a codec, (optionally) compressing it and then further packetizing this encoded data for transmission using packet switched networks. It is notable that the encoding process is no different than the PCM process mentioned in Section 1.2.1. Going further, it is simple enough to say that voice is now a “packet”. However, this conversion of voice into a packet has many far reaching consequences in terms of voice call quality and reliable transmission.

Since VoIP leverages the benefits associated with the use of open TCP/IP standards for its operation, let us look at VoIP operation and how a VoIP packet is formed. After the analog voice signal has been digitized using a codec, a certain fixed amount of that encoded data is handed over to the lower layers of the TCP/IP stack for packetization. A RTP header is added along with the UDP, IP and Layer-2 headers.

Before we look at the VoIP packet it is important to look at the codec operation and the underlying Layer-2 technology being used for transmitting the VoIP packets. Most of the codecs developed for the TDM world also have been ported to the IP world as well and PCM based on ITU-T G.711 standard forms the basis of these codecs with other more advanced codecs using different algorithms for varying compression, speech quality and bandwidth usage. As mentioned in Section 1.2.1 PCM works by sampling an analog signal at the rate of 8000 samples per second at 8 bits per sample leading to a sample size of 125 μs and encodes it in a digitized format. Typically, 20 ms of this sample data is sent for packetization, which means there are (20/.125) or 160 samples in one packet. As each sample is encoded using 8-bits (1-Byte), we get a voice payload of 160 Bytes.

We also have to consider the Layer-2 header as well and this would vary depending upon the Layer-2 technology used.

Figure 1-1 shows how the VoIP packet looks like along with the header sizes if VoIP is deployed over Ethernet without any kind of compression:

Voice Payload 160 Bytes	RTP Header 12 Bytes	UDP Header 8 Bytes	IP Header 20 Bytes	Ethernet Header 14 Bytes

Figure 1-1 – VoIP Packet

1.2.3 VoIP Call Bandwidth and Codec Considerations

We'll compare two cases, one using ITU-T G.711 Codec (PCM) and the other using ITU-T G.729 Codec using Conjugate-Structure Algebraic-Code-Excited Linear-Prediction (CS-ACELP) algorithm.

1.2.3.1 ITU-T G.711 Codec

Using the VoIP packet structure in Figure 1-1and adding all the headers and the voice payload, we get 214 bytes as the size of VoIP packet. Since the bitrate of the G.711 codec is 64000 bits per second, the packets per second (pps) rate at which it generates VoIP packets is given by:

Codec bitrate/Voice payload size = 64000/120*8 = 64000/1280 = 50 pps.

Hence, the bandwidth requirements for a G.711 VoIP stream = 214*8*50= 85600 bps or 85.6 Kbps.

1.2.3.2 ITU-T G.729 Codec

G.729 compresses audio data in 10 ms periods producing a bit-rate of 8000 bits per second or 8 bits/millisecond, which means 10 ms represents 10 Bytes. Considering a case where 2 of these 10 ms samples are combined and packetized, the voice payload for G.729 is 20 bytes. After including the effects of header, we get bandwidth requirement of a G.729 VoIP stream over Ethernet as 29.6 Kbps

1.2.4 VoIP Signaling

Before the RTP/UDP/IP stream data can be exchanged as part of a VoIP call, the call needs to be setup. There are a number of VoIP Call Signaling protocols in use in the industry. The ITU-T H.323 and IETF SIP call signaling protocols are some of the popular standards based VoIP call signaling protocols. These two protocols are examples of peer-to-peer call signaling protocols, in which a VoIP end point can be both a client and server, which means they don't need a central call processing server for their operation. For example, in the case of SIP an endpoint can be both a User Agent Client (UAC) and User Agent Server (UAS).

There are other VoIP call signaling protocols as well, which are created by efforts of different vendors and are designed to work best with their equipment. It is notable here that the call signaling traffic is very minimal and occurs mostly during call setup and teardown. Hence, its effects can be ignored for VoIP call bandwidth calculations. 3 has some good explanation on the operation and scope SIP signaling protocol. It is worth mentioning that both UDP and TCP can be used for the transmission of the call signaling packets.

1.2.5 Delay, Jitter, Echo in VoIP

Delay:

As mentioned earlier, the fundamental difference between the operation of packet switched networks and the TDM circuit switched networks means that VoIP packets are now susceptible to delay while they traverse the various elements of the VoIP network. ITU-T has published a G.714 listed in [8] , which specifies that for toll-quality voice, the one-way delay must be below 150 ms.

Please note that speech quality tests are perceptual in nature and the perception varies from a person to person. Even though a one-way delay of 200 ms might be tolerable in certain deployments, there are so many transient and varying conditions in an IP network, it's advisable to keep the delay as low as possible. A detailed delay analysis is performed in Section 2.4 later on in this report.

Jitter:

Now let's suppose you have designed your network such that the one way delay in a VoIP call is below 150 ms. Is that enough? It turns out that it as important to have regular, steady stream of VoIP packets as it is to ensure that the one-way delay stays below 150 ms. The regular, steady stream means that the VoIP packets are arriving at a rate with a constant inter-arrival time between consecutive packets.

However, due to the very nature of packet switched networks, all these packets might not take the same time to travel from the sender to the receiver. This could be due to the fact that the two consecutive packets took different paths to get to the receiver or there were different levels of congestion at a certain node in the network, when these packets traversed that mode. Hence, it is unavoidable to have some variation in packet inter-arrival time. This variation in packet inter-arrival time is called Jitter. If the variation is too high, it can cause havoc to a voice conversation.

De-jitter buffers are available on VoIP endpoints to address this issue. Basically, they address the need to play back a continuous voice stream at the receiving by buffering enough voice data so that it can be played back without interruption to the listener. It is notable that the size of the de-jitter buffers has an impact on the end-to-end delay. Even though most VoIP endpoints support dynamic de-jitter buffer size, care should be taken to not have this de-jitter buffer size too high, otherwise the end-to-end delay would become too high and the voice quality would suffer.

Echo:

Echo is the audible leak-through of your own voice into your own receive path and one significant concern in Echo analysis is the round-trip delay in a voice call. The round-trip delay in a voice call is the time taken for the voice from the speaker's mouth to travel to the transmit path and through the source of the leak to the receive path and back to the speakers ear. The higher this round-trip delay, the bigger the concern it presents. Also, the louder the echo amplitude, the more annoying it is.

There are generally two types of echo; Hybrid echo also sometimes referred to as Line echo and Acoustic echo. We now discuss the causes of these two types of echo:

Hybrid echo - The traditional PSTN phone line is two wires, with both transmit and receive paths of the conversation on the single pair. The transmission network is built using 4-wire. It is notable that any digital link is also an equivalent 4-wire circuit even though it's physically a 2-wire circuit. A device called a hybrid transformer is used on the network to convert four wire to two wire, or back wherever that conversion occurs. It works on the principle of matching impedances and to provide isolation between the transmit and receive path signal. The problem is that no matter how well the hybrid is made, it does not always work perfectly. There's always an impedance mismatch which means that some of the transmit and receive signal gets mixed and sent back to the other side of the hybrid transformer. This is called sidetone. In almost every telephone device, some of the Tx signal is fed back into the earpiece so that you can hear yourself speaking

As mentioned earlier, the sidetone is always present. On a regular analog phone call, the signal leakage caused by the inability of the hybrid to match impedance perfectly is not audible because there's minimal round-trip delay. But in voice call involving delay inducing network elements, the round trip delay can be significant leading to audible echo.

Acoustic Echo – Acoustic Echo is caused by acoustic coupling of the signal through the air from a loudspeaker to a microphone. The hands-free speakerphones are notorious for this type of echo. Some poorly designed headsets can also cause acoustic echo.

In general, if the round-trip time is above 50 ms in a voice call, the echo becomes audible as indicated in [6]. A detailed echo analysis is performed in Section 2.4.

1.3 Overview of the remaining report

In Chapter 2, we analyze the network design and planning issues related to delay and NAT traversal along with the VoIP Gateway choices. In Chapter 3, we simulate and analyze the performance of different VoIP scenarios using OPNET and also talk about one approach to address OPNET simulation performance related issues. In Chapter 4, we verify the OPNET simulation results in our industry partner's VoIP network. Chapter 5 is about the conclusion of our findings and Chapter 6 talks about future work related to this project. Chapter 7 lists references and related documents.

Chapter 2 Network Design and Planning Issues

2.1 VoIP Network Design

The Figure 2-1 represents our reference network design and reflects the network of the industry partner's network.

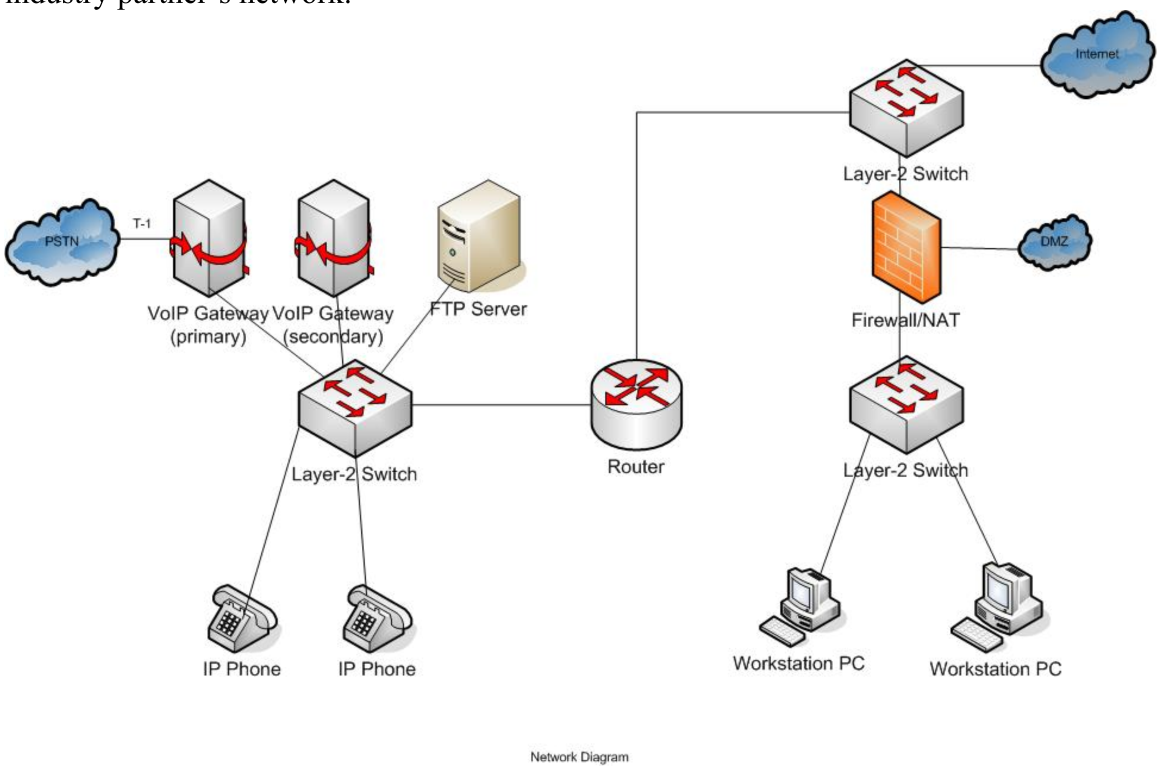

Figure 2-1 – VoIP Network Diagram

2.2 VoIP Bandwidth and Codec Considerations

The IP Phones in the VoIP network are configured to use the G.711 codec. The Layer-2 switch is Gigabit Ethernet and the VoIP network is connected to the PSTN using a T-1 PRI link.

2.3 Delay Analysis

ITU-T G.714 recommendation imposes a maximum total one-way packet delay of 150ms end-to-end for VoIP applications. We can break the end-to-end delay occurring in a VoIP call path into at least three different contributing components, which are as follows:

2.3.1 Sender Processing Delay

The delay introduced at the sender includes the following components:

a) Encoding,
b) Compression, and
c) Packetization delay

2.3.2 Network Delay

The delay introduced by the network includes the following components:

a) Insertion/serialization,
b) Propagation, and
c) Queuing delay

2.3.3 Receiver Delay

The delay introduced at the receiver includes the following components:

a) Buffering,
b) Decompression,
c) Depacketization,
d) Decoding, and
e) De-jitter buffer/Playback delay

For example, In case of G.729, if two 10 ms samples are placed in one VoIP packet, the delay at the sender will be 20 ms. An initial look ahead time of 5 ms occurs in case of G.729, giving the delay to be 25 ms at the source. The delay incurred at the receiver includes de-jitter buffer delay. The de-jitter delay is variable depending upon the implementation and usually is at most two packets in a network with minimal congestion. We'll use a value of 40ms in this case. When the receiver's processing delay of depacketization and decoding is added, we obtain a total fixed delay of 45ms at the receiver. Hence the delay to be introduced by the network should not exceed $(150 - 25 - 45)$ or 80ms.

2.3.4 Delay Budget Calculations:

For the Delay Budget calculation, we consider a voice call scenario, where an IP Phone calls a PSTN phone. Since we made the decision to deploy G.711 in our network, the following are the delay calculations introduced by various components:

2.3.4.1 Sender Processing Delay

The IP Phones are using the G.711 codec. Using the G.711 codec, 20 ms of voice is packetized in one VoIP packet. The look-ahead time is zero in case of G.711. At this point, I would like mention that this 20 ms delay is due to packetization as the VoIP endpoint <u>must</u> wait for 20 ms for the packet to be created. It is <u>not</u> due to the codec as codecs are quite fast in their operation.

2.3.4.2 Network Delay

The following components of network delay are calculated:

a) Insertion/Serialization Delay (X): This delay component is introduced at the VoIP Gateway. Since the connection from the VoIP Gateway to the PSTN is T-1, we calculate the insertion delay as below:
   The length of each VoIP frame, L = 216 bytes
   The Data rate offered by the T-1 line, R= 1.536 Mbps
   Hence, Insertion Delay, X = L/R = 216x8/1536000 = 0.001125 s = 1.1 ms
b) Propagation Delay (T_p): This delay component is the time it takes for the signal to propagate from one end to another. It is notable that this delay is a function of the distance between the routers and propagation speed and has nothing to do with packet length or data rate.

Assuming a phone call from IP Phone to a phone in the Greater Edmonton area, we can calculate the propagation delay as below:

Let's say "d" = distance of the propagation path
"s" = speed of propagation in the medium (copper) = 0.5 x speed of light

T_p =Propagation Delay from IP Phone to the VoIP Gateway(d1/s) + Propagation Delay from VoIP Gateway to the PSTN Phone(d2/s)

Further assuming for calculation purposes,

d1= 100m = length of the copper wire from IP Phone to the wiring closet

d2 = 100,000 m = length of the copper wire from the T-1 gateway to the remote PSTN phone.

Hence, T_p = 100/0.5x3e8 + 100,000/0.5x3e8 = (0000006666+.00066666) s = .667 ms

It is notable that the propagation delay is a small component of the total delay in a local call scenario. But for an international long distance call, the propagation delay would be significant. Generally, every 1000 Km of propagation path adds about 6 ms to the propagation delay.

c) Queuing Delay – Queuing delay is a big concern in converged packet switched networks. In our case since the VoIP network is physically independent of the IP data network, there is no queuing occurring at the network devices. The only queuing in this case is occurring at the VoIP Gateway. It is reasonable to include a value of less than 10 ms for the queuing delay. However, we consider a worst case scenario of Queuing delay of 10 ms.

2.3.4.3 Receiver Processing Delay

The most significant component of delay at the receiver is the De-jitter buffer delay. Typical values for the de-jitter buffer are between 30-50 ms. Although adaptive de-jitter buffers can be in the range of 100-200 ms. The IP Phones support an adaptive de-jitter buffer but for calculations we can use the standard of de-jitter buffer size as twice the voice information in each 20 ms VoIP packet i.e 40 ms.

The de-jitter buffer should be at least the size of one packet. Although it is not uncommon to see de-jitter buffer settings approaching 80 ms due non-ideal network conditions, high values certainly add to the overall delay budget and should be avoided.

Hence, the Delay Budget for the IP-to-PSTN Call Scenario is shown in Table 2-1:

Type of Delay	**Fixed Delay**	**Variable Delay**
Sender Processing Delay - Look ahead time - Encoding, compression and packetization	 0 ms 20 ms	
Network Delay - Insertion Delay - Propagation Delay - Queuing Delay	 1.1 ms .667 ms	 10 ms
Receiver Processing Delay - Depacketization and decoding - De-Jitter Buffer delay	 5 ms	 40 ms
Total Delay	26.767 ms	50 ms

Table 2-1 – IP Phone -to-PSTN Call Delay Budget Calculation

We notice from the above table that the End-to-end delay adds upto to about 77 ms, which is within the ITU-T G.714 recommendation of less than 150 ms. It is also notable that the design goal should be to minimize the network delay components so that network delay is less than 150-45-20 = 85 ms. In our case the network delay is approx. 12 ms, which is well below the 85 ms limit.

Similarly, the Delay Budget for an IP Phone-to-IP Phone call can be calculated as well s shown in the Table 2-2 The only difference is that the network delay is now the propagation delay within the VoIP network as no traffic crosses the VoIP Gateway.

Type of Delay	**Fixed Delay**	**Variable Delay**
Sender Processing Delay - Look ahead time - Encoding, compression and packetization	 0 ms 20 ms	
Network Delay - Insertion Delay - Propagation Delay - Queuing Delay	 .000667 ms	
Receiver Processing Delay - Depacketization (re-sequencing) and decoding - De-Jitter Buffer delay	 5 ms	 40 ms
Total Delay	25.000667 ms	40

Table 2-2 – IP Phone to IP Phone Call Delay Budget Calculation

We notice from the above table that the end-to-end delay adds upto to approx. 65 ms, which is within the ITU-T G.714 recommendation of less than 150 ms.

2.4 Echo Analysis

We started out with the task of locating the possible causes of the echo while keeping in mind that perceived audible echo at the talker side is due to a leakage problem on the listener party side as explained in [6]. The following presents an analysis of the possible causes:

2.4.1 IP Phones and Analog Phones

IP Phones: If the echo is only noticed by the talking IP Phone party in an IP Phone to PSTN call and there is no echo reported by the PSTN called party, then we can safely remove the IP Phones a cause. If some echo was occurring at the IP phone due to acoustic coupling, then the PSTN called party would also report the echo. Moreover, there was no echo reported while making IP phone - IP phone calls. Hence, the IP Phones were not a suspect in this case. It pointed out that the Acoustic Echo Cancellation (AEC) on the IP Phone was sufficient.

Analog Phones: Signal leakage also occurs at the hybrid in the analog phones at the PSTN called party side. As mentioned earlier, this signal leakage is always present and we have no control over this network characteristic. Hence, the inefficiency of the hybrid in the analog phones to match the impedance perfectly is certainly the cause of the signal leakage leading to audible hybrid echo.

Moreover, if the PSTN called party is using a speakerphone or a poorly designed headset that can lead to audible acoustic echo.

2.4.2 Hybrid Transformers

The Hybrid Transformers on the tail-circuit (PSTN called party side) could possibly lead to leakage causing echo. However, we have no control over this network characteristic.

2.4.3 VoIP Gateway

On one side, VoIP Gateway connects to the traditional voice world (in this case Digital T-1), and on another side to VoIP network. As the interface, the gateway needs to translate signaling messages between the two sides as well as compress and decompress the voice. Although in our case, there is no audio compression occurring as we are using G.711 codec but inherent delays due to translation are present.

The IP stack on the VoIP Gateway requires fixed spacing between the packets so that a steady stream can be constructed to be sent out to the receiving IP Phone. The typical solution to this is to implement a jitter buffer within the gateway. The jitter buffer deliberately delays incoming packets in order to present them to the IP stack at a regular rate.

Since digital segments of the network do not cause leaks as bits don’t leak between paths, adding routers (and IP/PSTN gateway routers) to the network does not cause echo but instead adds delay to the network—delay that can make a previously imperceptible echo perceptible. We concluded that even though the VoIP Gateway itself does not cause any signal leakage but it can certainly enhance the echo problem by adding to the round-trip delay. Hence, we decided to further investigate and test the delay introduced by the VoIP Gateway in Section 4.2.

2.4.4 Gigabit Ethernet Layer-2 Switches

We suspected that the Layer-2 switches could be buffering some frames which might add to the round-trip delay. Hence, we decided to further investigate the delay introduced by the Layer-2 switch in Section 4.2.

2.4.5 IP Phone to PSTN end-to-end voice call delays

Short round-trip delays in PSTN mean that even relatively loud echo in the PSTN remain imperceptible. Echo is not of a significant concern in traditional PSTN networks as Echo Cancellers are engaged in all long distance calls.

However, in VoIP networks an increase in the end-to-end voice call delay would also increase the round-trip delay and hence this was also identified as a concern as in VoIP networks, certain delays are inherently present such as the sender packetization delay of 20 ms, which cannot be avoided.

2.4.6 QoS Mechanisms

Deploying QoS measures in your network might reduce the end-to-end network delay especially if they are congested. The shorter the delay, the less annoying a given echo becomes. Since our VoIP network is physically separated from the data network and the only traffic it is carrying is the VoIP signaling and audio data, we concluded that deploying QoS mechanisms was not identified as concern.

2.5 Addressing the NAT/Firewall Traversal Problem

Problem Description:

NAT presents problems for VoIP Traversal. The problems arise from the fact that VoIP call signaling protocols such as SIP, H.323, and MGCP use different channels for call control and RTP audio exchange.

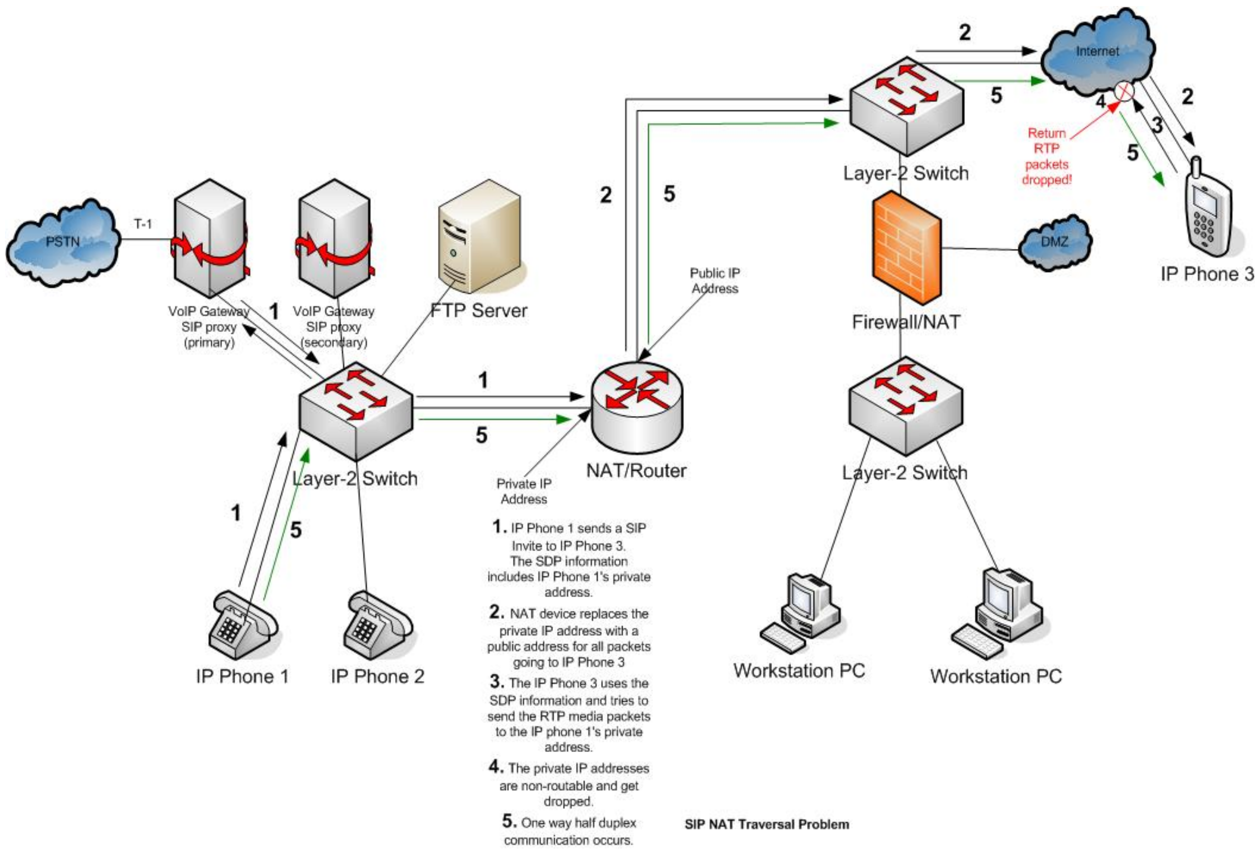

Figure 2-2 - SIP NAT Traversal Problem

As noted in Figure 2-2, the SIP User Agent Client (UAC) on IP Phone 1 sends out an INVITE message to the remote SIP User Agent Server (UAS) on IP Phone 3. The SIP signaling includes information about the private IP addresses and ports that the IP Phone 1 wants to use for the media exchange. When IP Phone 3 attempts to use these private IP addresses to setup a media path, the attempt fails as these IP addresses are non-routable. In these situations we end up with a one-way half-duplex communication.

2.5.1 NAT Traversal Problem Analysis

We first start with a discussion of the different types of Network Address Translation (NAT) and Network Address Port Translation (NAPT) methods in use and then address the issue of NAT/NAPT Traversal. It is notable here that the term NAT is used generically to represent both NAT and NAPT.

2.5.2 Types of NAT

2.5.2.1 One-to-One NAT

This is the most basic form of NAT in which each private IP address is mapped to a different public IP address. This is not a very practical solution for a typical network as it requires a large number of limited public IP addresses to operate.

2.5.2.2 Full Cone NAPT

Full cone NAT receives outgoing TCP or UDP streams from internal hosts and creates an entry in the NAPT table that maps a given internal IP address and port to a specific external IP address and port. It is notable that the NAPT entry is created irrespective of the final external public destination of packets in a given stream. In essence, the NAPT entry acts like a pinhole in the NAT device for incoming packets from external devices on the Internet.

2.5.2.3 Restricted Cone NAPT

Restricted Cone NAPT restricts incoming external (public) IP traffic on the basis of public IP address and does not check the port number. Only internal hosts can initiate a session through Restricted Cone NAPT.

2.5.2.4 Port-Restricted Cone NAPT

Port-Restricted Cone NAPT is identical to Restricted Cone NAPT except that it checks the public IP address AND the port number of the incoming public IP traffic and restricts traffic based on that. In this case, packets from external hosts must contain the source IP address and a port number previously used by an internal host.

2.5.2.5 Symmetric Cone NAPT

Symmetric NAPT assigns a unique external port to every session that an internal host initiates with an external host. External hosts can only initiate a session if their source address and port number were previously used by that internal host. A new entry is created in the NAPT table for traffic from internal hosts regardless of the destination.

2.5.3 SIP NAT/Firewall Traversal Methods

2.5.3.1 Simple Traversal of UDP through Network Address Translators (STUN)

As mentioned in the discussion of Full-cone NAT, a mapping entry is created in the NAPT table for each source IP and port, irrespective of the destination. If an internal host reuses a port to communicate with a different port on the same external host or even to a different external host altogether, the same entry will be used. This behaviour is used in the STUN approach for one possible solution to VoIP Traversal through NAT and is standardized as RFC 3489 listed in [5].

Every time the IP phone on the private side of a NAT device wants to send out an INVITE message with an IP address and port that cannot be reached through the NAT/NAPT function, it first sends a STUN query to a STUN server on the public side of the NAT device on the well-known STUN server port. The response from the STUN Server indicates the public source address and port in the NAPT entry on the NAT device. The IP Phone will now advertise the correct public IP address and port number combination in the INVITE message.

STUN is a very popular and practical solution as it works for Full Cone NAPT, Restricted Cone NAPT and Port-Restricted NAPT deployments because it does not involve upgrading/changing the NAT devices. Most IP Phone (both hardware and software based) support STUN capability. The IP Phones chosen for our deployment also support STUN.

The limitation of STUN is that it fails for Symmetric Cone deployments, where a new NAPT entry (Public IP and Port#) is created for each outgoing stream irrespective of the destination. So, in essence a new NAPT pinhole is created for every outgoing stream. Since, the STUN server has already provided information about the old NAPT pinhole to the internal SIP device and now the Symmetric Full Cone NAPT device has created a new NAPT entry, STUN fails in these scenarios.

2.5.3.2 Traversal using Relay NAT (TURN)

TURN is another NAT traversal approach that uses the TCP/UDP pinhole opened through the NAT function to solve the worst-case Symmetric NAT scenario. It works by setting up a bidirectional communication tunnel with a TURN Server in the public network.

The SIP device located in the private network which requires establishing a communication with the public internet first communicates with the TURN server using the TURN protocol and as a first step the SIP device requests an IP address and port for its own use on the TURN Server. The SIP device can then advertise this IP address and port to external SIP devices that need to communicate with it. When the TURN server receives these packets, it simply forwards these packets to the internal SIP device using the TURN protocol. It is notable that the TURN Server can traverse NAT function because it is based on a permanent TCP connection between internal SIP device and the TURN server or it uses symmetric UDP. TCP or Symmetric UDP (using keep-alive messages) is required in this setup because if that is not the case, then the NAPT entries would timeout and the NAPT device would fail to forward the packets to the internal host. It is also notable that the since all RTP audio data passes through the TURN server, it needs to be a high-performance server to avoid adding latency.

2.5.3.3 ICE (Internet Connectivity Establishment)

The Interactive Connectivity Establishment (ICE) tries to integrate the various SIP traversal approaches by extending the signaling capabilities in the Session Description Protocol (SDP, RFC 2327), which is used to set up SIP and other multimedia sessions, to all of the NAT traversal protocols (STUN or TURN) in order to determine the optimum means for communication.

Hence, it allows for all SIP devices to discover the NAT types and capabilities and use the best method. It is notable that this adds significant complexity in the SIP clients and also the SIP clients must support at least one SIP traversal method. The advantage being that it will work with all types of NATs.

2.5.3.4 Application Layer Gateway (ALG)

Application Layer Gateway is an enhanced Firewall/NAT device that is VoIP application aware, which means that it recognizes and understands VoIP signaling. It works by modifying the outgoing SIP signaling (or H.323, MGCP) from the internal SIP UAC to reflect the (correct) public IP addresses being used by the NAPT device. It usually requires replacing the existing NAT/Firewall although software upgrades might be available from the vendor. It is notable that this method does not require any changes on the SIP clients.

2.5.3.5 Session Border Controller (SBC)

Session Border Controller is a solution in which all outgoing SIP signaling messages and RTP audio data from the internal SIP UAC is directed towards the SBC signaling and media proxy (both functions on a SBC). In essence, the SBC maintains control of the signaling and media path.

As mentioned in [7] SIP signaling messages from the internal SIP UAC leave the private network using the source IP address as a public IP address and port assigned by the NAPT device. When the SBC signaling proxy receives this signaling message, a source address is assigned to this SIP UAC. The signaling proxy then sends a modified REGISTER message to the Call Agent indicating in the CONTACT and VIA fields that the SBC signaling proxy is the source. The Call Agent sends all signaling messages to the SBC signaling proxy after that.

When the internal SIP UAC initiates a call, the INVITE message is sent to the SBC signaling proxy. The signaling proxy then modifies the source address and SDP information to its own IP address as the return path for signaling and the address of the media proxy as the return path for the RTP audio data. The modified invite message is forwarded to the Call Agent, which thinks that the call originated from the SBC. The Call Agent then forwards the INVITE message to the receiving SIP endpoint. The ACK message from the receiving endpoint is sent to the SBC signaling proxy via the Call Agent. The signaling proxy then modifies the ACK message to change the IP address and port to that of the media proxy. This modified ACK message is sent to the internal SIP UAC and all subsequent RTP audio data is now directed through the media proxy.

2.5.4 NAT Traversal Method Choice

STUN approach for NAT traversal was chosen for the VoIP network for these reasons:

1) The IP Phones used in our deployment support STUN capability.
2) The NAPT device uses Full Cone NAPT method.
3) There are a number of STUN Servers available publicly that can be used in our solution.

2.6 VoIP Gateway

The VoIP network as shown in Figure 2-1 uses Switchvox Asterisk based VoIP Gateway/IP PBX product. Here are some of the Switchvox features:

1) Advanced features like a java based switchboard that works off a computer for the receptionist, advanced call center features, call queuing etc.
2) Extensive and reasonably documented API that can be used to write third party applications.
3) A significant portion of the ACD component is already implemented.
4) Provides for an easy mechanism to integrate with Vonage or Junction Networks, so that users can use Vonage to make long distance calls.

Nonetheless, here are some features that we found missing in Switchvox:

1) The ability to truly load share and traffic engineer at the packet level to ensure QoS.
2) No support for true peer to peer processing.
3) No support of G.729 out of the box. Purchase of additional licenses required.

Chapter 3 Simulation

3.1 Simulation Approach

As discussed in 2.3, G.714 imposes a maximum total one-way packet delay of 150ms end-to-end for VoIP applications. Our goal is to compare two scenarios: IP Phone to IP Phone call and IP Phone to PSTN call with respect to delay, delay variation and packet loss. The approach used here is based on approach used in [9]with modifications to suit the reference network design.

3.2 Experimental Setup

3.2.1 Platform

We used OPNET Modeler Release 11.5 for the simulation running on Windows Platform on a Pentium IV 1.8 GHz PC with 1 GB RAM.

3.2.2 Network

The simulation models represent the reference VoIP network in Figure 2.1 and are based on two scenarios:

1) IP Phone to IP Phone Call
2) IP Phone to PSTN Call

The network models shown in Figure 3-1 and Figure 3-2 represent the two possible scenarios.

The network model in Figure 3-1 represents the IP Phone to IP Phone call scenario. Even though the VoIP Gateway is shown and the VoIP traffic never crosses the Ethernet switch.

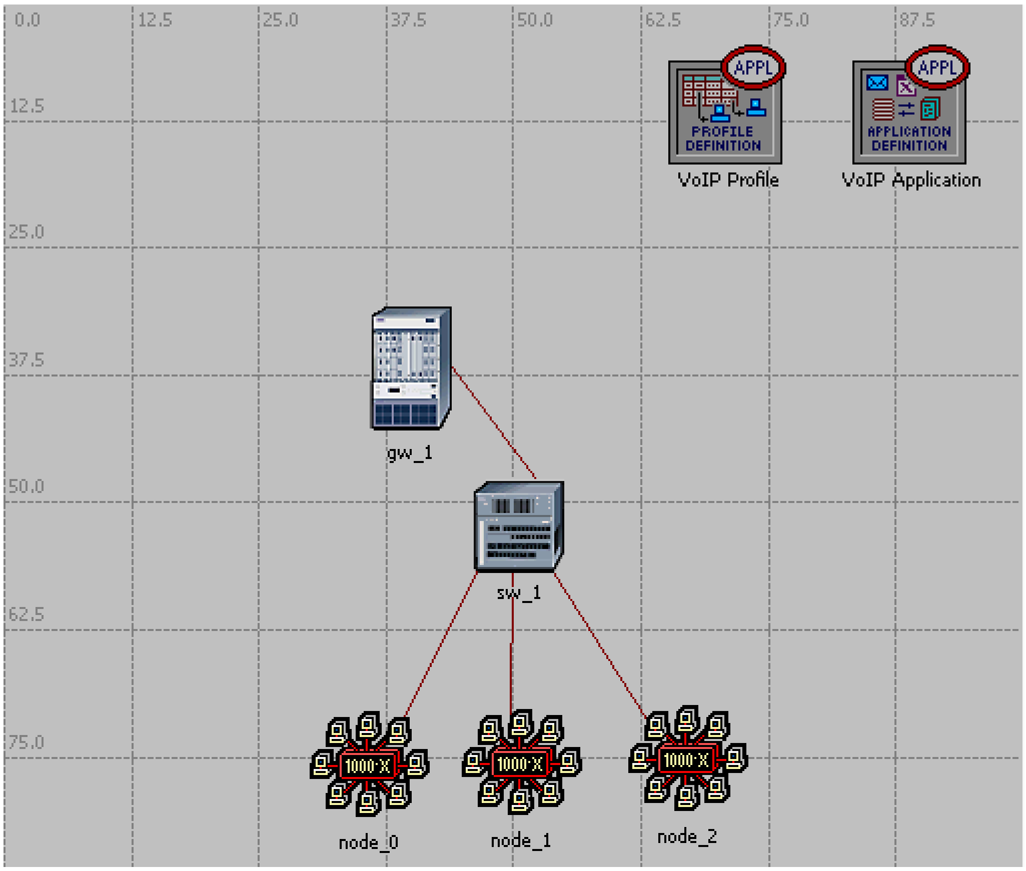

Figure 3-1 - IP Phone to IP Phone call scenario

The network model in Figure 3-2 represents the IP Phone to PSTN call scenario. All VoIP call traffic generated crosses the VoIP Gateway and the T-1 link over to the other side.

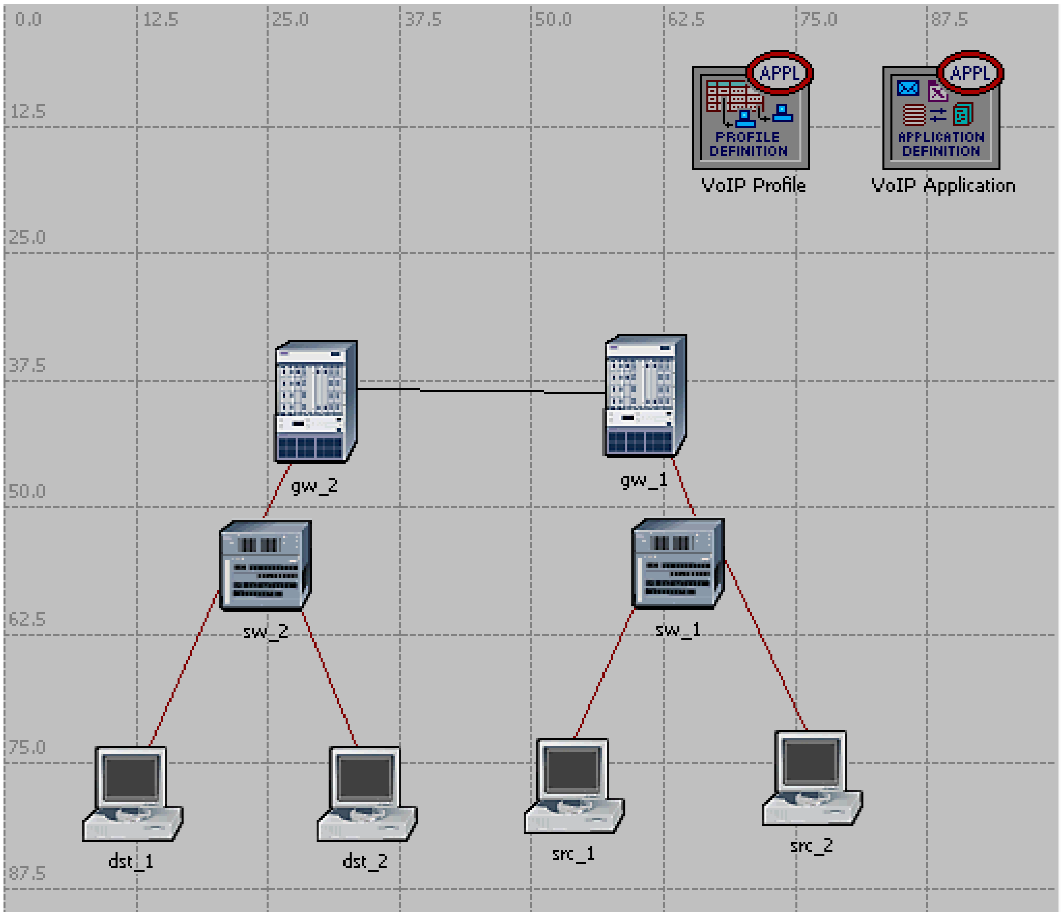

Figure 3-2 - IP Phone to PSTN call scenario

3.2.3 OPNET Configuration

OPNET includes device models for various popular network devices. However, since there was no device available in OPNET that included the required combination of interfaces, we created a new generic Layer-3 device with the required T-1 and Ethernet interfaces. VoIP network modeling is new functionality that is being developed in OPNET and hence extensive documentation for VoIP Analysis is not available. We have used the existing application traffic analysis components to model and analyze VoIP traffic.

3.2.3.1 Generating Simultaneous VoIP Calls

The Application Profile is configured to generate a new VoIP call every 20 seconds with appropriate destination preferences configured on the various IP Phone nodes until the end of simulation. Please note that these calls are concurrent and not consecutive. The weight corresponding to each destination determines how frequently that destination will be called. Appropriate setting of this weight is crucial to accurate modeling and should be configured after proper traffic analysis and baselining of the existing network. E.g. if the company is looking a replacing a PBX system with a VoIP system, appropriate call traffic measurements on the PBX system need to be taken.

3.2.3.2 Node Configuration

All the settings mentioned below were chosen as initial starting point values and can be modified to suit different scenarios to obtain different results.

Switch: A generic Layer-2 Ethernet switch with the packet service rate of 8 million packets per second was chosen as the LAN switch. All ports are configured for the same VLAN and support 100/1000 Mbps interface data rate.

VoIP Gateway/Router: A new generic device was created with T-1 and Ethernet interfaces with packet forwarding rate of 150,000 packets per seconds. Both gw_1 and gw_2 are connected with a link with T-1 data rate for Scenario in Figure 3.2.

IP Phones: In the IP-to-IP call scenario, LAN nodes are used with node_0 and node_1 as sources and node_2 as destination. In the IP-to-PSTN call scenario, workstation objects are used with appropriate VoIP Application and VoIP profile objects associated with them.

Application Configuration: VoIP Application and Profile objects were created to model the VoIP traffic. The VoIP Profile objects were then assigned to the respective nodes.

Background Traffic Configuration: No background traffic was introduced as this deployment is considered separate from the existing data network. However, background traffic can be easily incorporated in this scenario.

Signaling Traffic: Signaling traffic generated by the call processing server is ignored. The signaling traffic involving the call processing server is only generated prior to the establishment of the voice call and when the call is finished. This traffic is relatively limited and small compared to the actual voice call traffic. In general, the call processing server generates no signaling traffic throughout the duration of the VoIP call for an already established ongoing call.

3.3 Simulation Problems and Solutions

OPNET has extensive VoIP capabilities and offers flexibility in creating various types of configurations. The simulation went relatively smooth except for a problem as mention in Section 3.3.1

3.3.1 OPNET Performance Problem

Problem Description: The OPNET Simulation was taking around 30 minutes -1 Hr for the simulation time of 10-15 minutes. This was causing unnecessary delays in the execution of the project.

3.3.2 Improving the OPNET Simulation Time

In the first simulation, we were applying a lot of traffic on the network. This traffic was all discrete (simulation of each packet), which was adversely affecting the runtime.

We discovered that the solution to the problem is hybrid simulation. A hybrid simulation mixes both discrete and background (analytical) traffic. The advantage is an improvement in terms of runtime. One issue with the background traffic is that it will not give you all the statistics you would like to get for a voice application (jitter, packet loss, delay, etc). This was a problem for us as these statistics were very important for the simulation results. However, we were able to find a workaround to the problem by splitting the LAN into two parts and associating one LAN with the background traffic and the other with discrete.

3.4 Simulation Results and Discussion

The network performance characteristics for the IP Phone to IP Phone and IP Phone to PSTN scenarios are shown below. All the simulations were carried out using the G.711 codec.

3.4.1 IP Phone to IP Phone Call Scenario (G.711 Codec)

The following are the plots of performance statistics at node_1:

3.4.1.1 VoIP packet end-to-end delay vs. simulation time

We notice in Figure 3-3 that the end-to-end delay is .048 sec, which is 48 msec. This is approximately similar to our theoretical calculations in 2.3. It is notable that the above delay includes the de-jitter buffer size of 20 ms instead of 40 ms used in the calculation of 2.3. This delay value is constant as the Gigabit Ethernet network is able to handle the VoIP call traffic without any congestion.

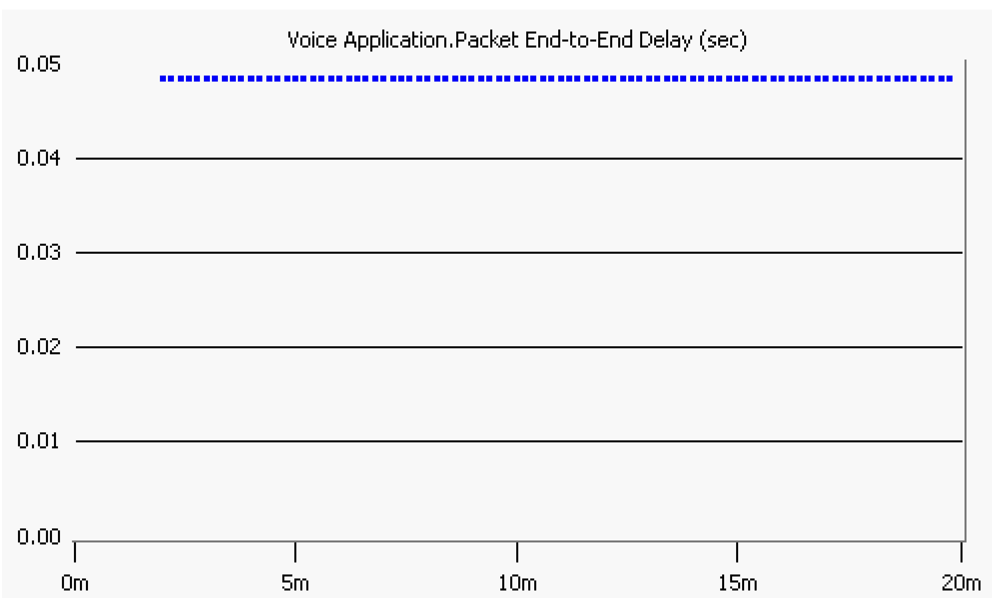

Figure 3-3 – IP Phone to IP Phone End-to-End Delay

3.4.1.2 VoIP packet delay variation (seconds) vs. simulation time

As noted in the plot in Figure 3-4, the packet jitter is negligible that points to the regular transmission of VoIP packets.

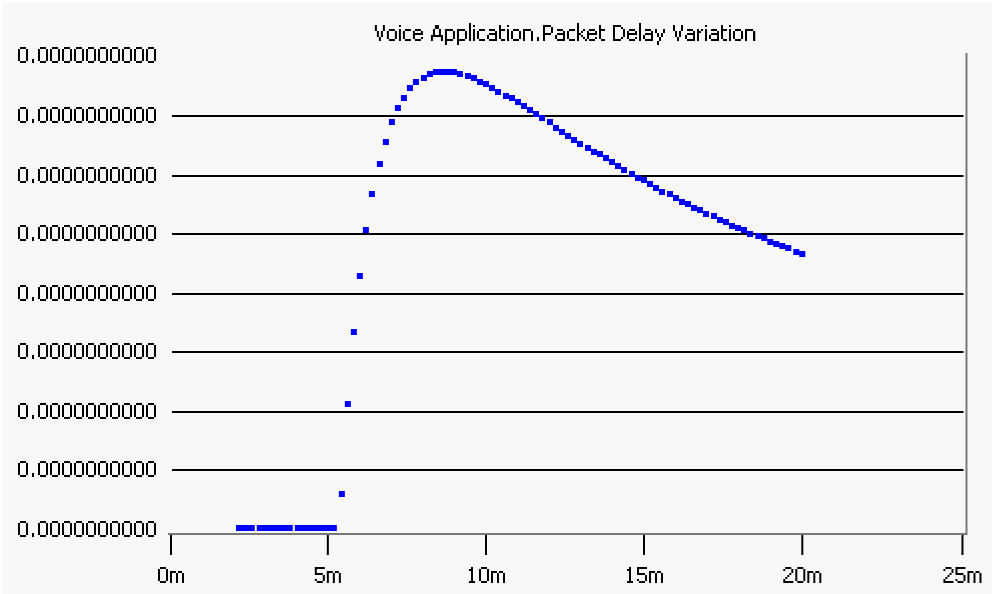

Figure 3-4 – IP Phone to IP Phone Packet Delay Variation

3.4.1.3 VoIP packet traffic sent vs. simulation time

In Figure 3-5 and Figure 3-6, we observe that the number of VoIP packets sent is the same as the number of VoIP packets received indicating no packet loss, which is expected behaviour in a high-bandwidth congestion-free reliable network.

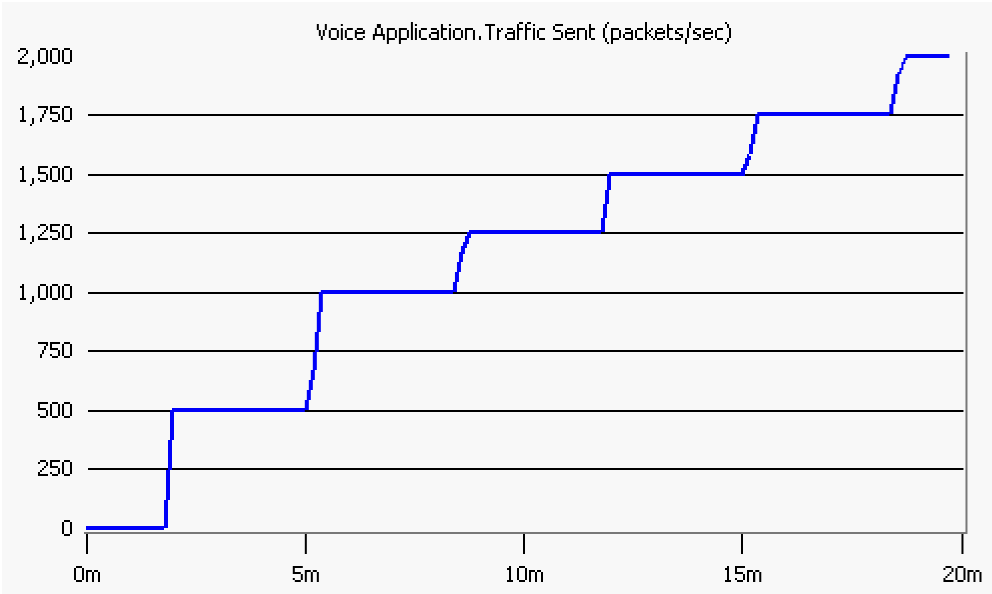

Figure 3-5 – IP Phone to IP Phone Traffic Sent

3.4.1.4 VoIP packet traffic received vs. simulation time

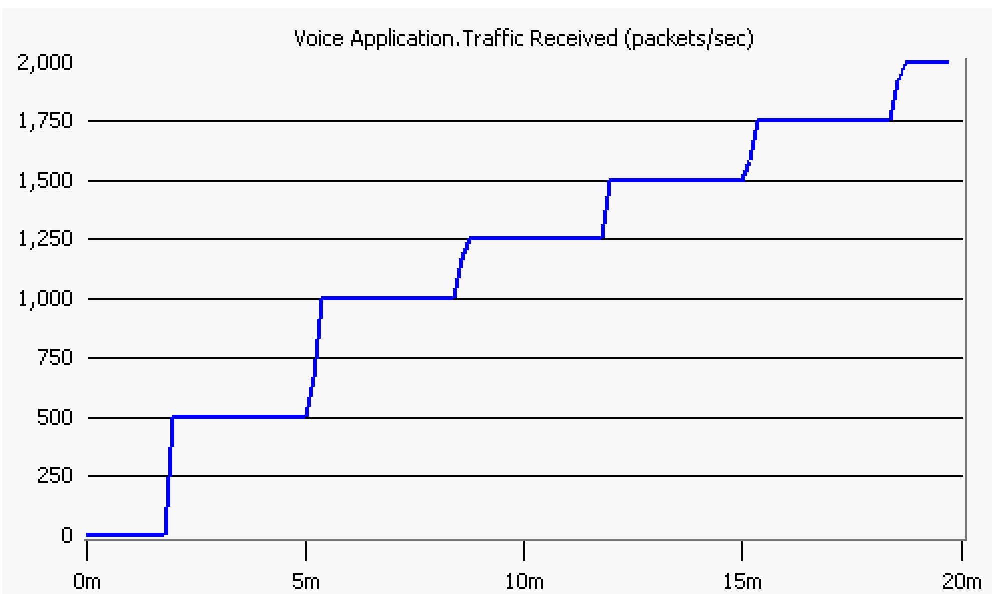

Figure 3-6 – IP Phone to IP Phone Traffic Received

3.4.2 IP Phone to PSTN Call Scenario (G.711 Codec)

The following are plots for performance statistics for src_1:

3.4.2.1 VoIP packet end-to-end delay vs. simulation time

In plot in Figure 3-7, we observe that upto the time interval of 12.5 min, the end-to-end delay value is at .05 sec, which is 50 ms. After 12.5 minutes the end-to-end delay increases considerably as the number of simultaneous calls now exceeds the capacity of the link.

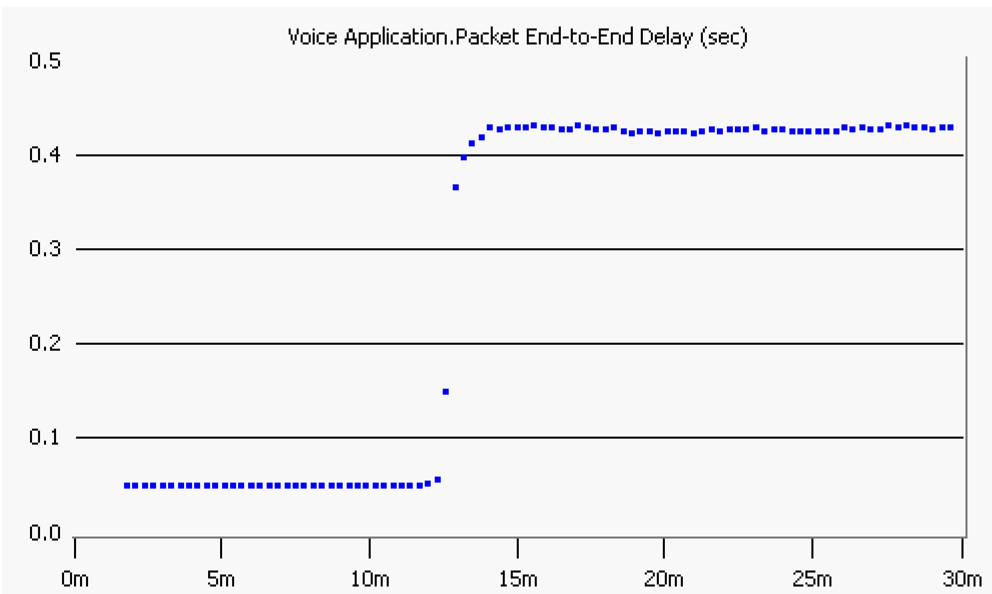

Figure 3-7 – IP Phone to PSTN End-to-End Delay

3.4.2.2 VoIP packet delay variation(seconds) vs. simulation time

As indicated in the plot in Figure 3-8, the jitter varies considerably due to the congested link and packets being dropped at the link

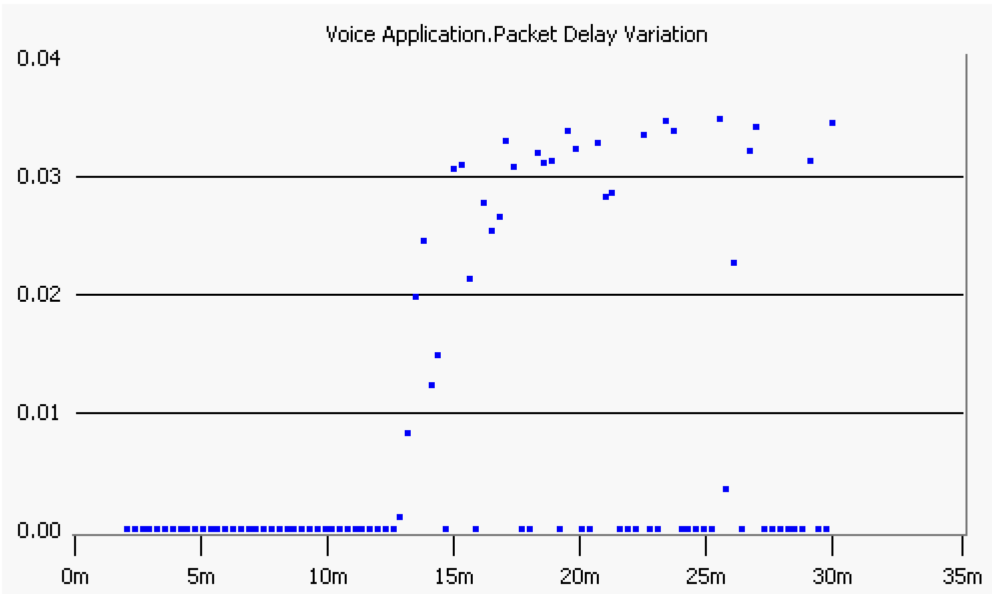

Figure 3-8 – IP Phone to PSTN Packet Delay Variation

3.4.2.3 VoIP packet traffic sent vs. simulation time

In plots in Figure 3-9 and Figure 3-10, it is observed that the number of packets being sent is similar to the number of packets received until 12.5 minutes but after the number of packets being sent keeps increasing (as the VoIP calls being generated after regular interval) but the number of received packets is decreasing as the link is congested and the packets are being dropped.

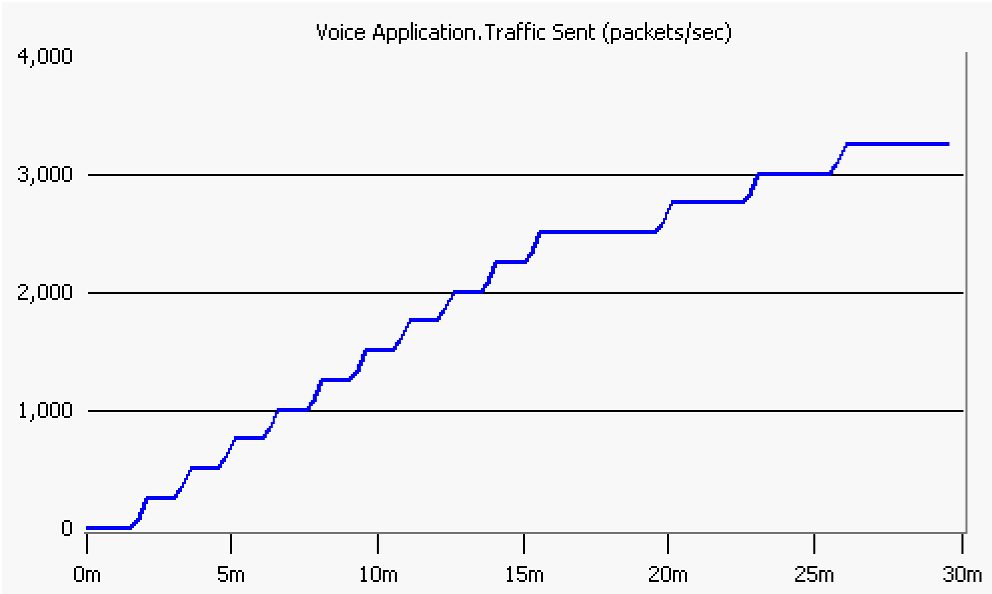

Figure 3-9 – IP Phone to PSTN Traffic Sent

3.4.2.4 VoIP packet traffic received vs. simulation time

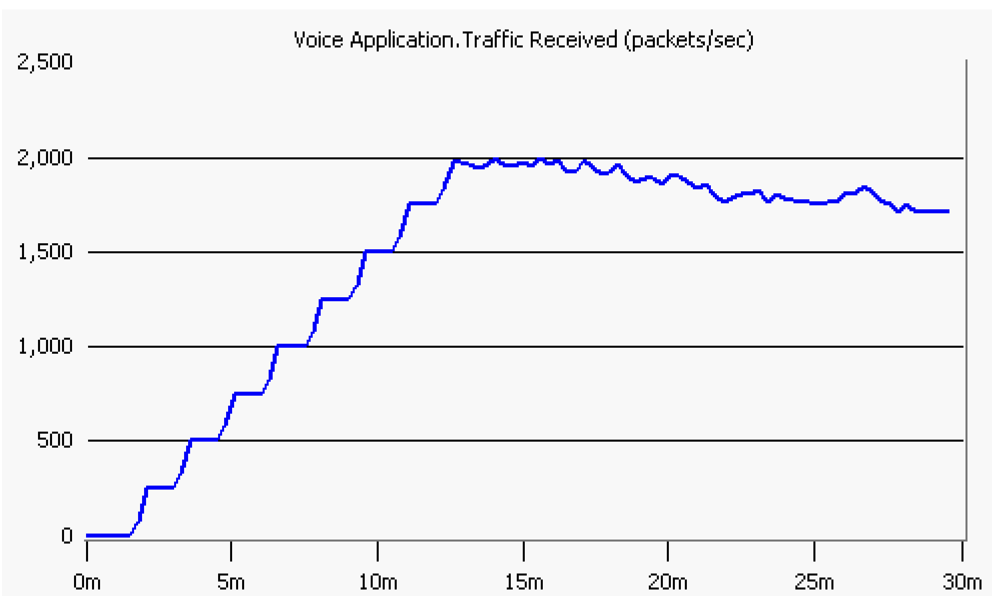

Figure 3-10 – IP Phone to PSTN Traffic Received

Chapter 4 Verification on a Production System

4.1 Experimental Setup

4.1.1 Platform

Our industry partner's network was used for verification of the calculated and simulation results and also to identify and analyze jitter and delay related issues.
We used Wandel&Goltermann DA 362 Domino FE Internetwork Analyzer for the testing of the Ethernet switch and Network Associates Sniffer WAN Analyzer for testing the VoIP Gateway T-1 side. We also used Ethereal to capture and analyze traffic on the IP side of the VoIP Gateway.

4.1.2 Network

The Network Diagram is included in Figure 2-1 and various testing configurations are included further in the report.

4.1.3 Delay, Jitter and Echo Testing

For the jitter network performance measurements, we relied on analyzing captures of VoIP packets (signaling and audio data) passing through the VoIP Gateway over period of one week and for delay and echo we had specific test setup test configurations as explained in Section 4.2

As mentioned earlier in Section 2.4, delay and echo are very closely interrelated. We performed delay testing of the network with the aim of identifying the possible causes of echo and have a deeper understanding of echo behaviour in the VoIP realm.

4.2 Delay Testing

Based on the above analysis, we decided to focus on the following areas to determine the cause of echo.

1. Delay caused by Layer-2 Switch
2. Delay caused by VoIP Gateway
3. End-to-End Voice Call Delays

4.2.1 Delay caused by Layer-2 Switch

We used the WG Domino Internetwork Analyzer for the delay testing of the Layer-2 switch. Since we were measuring the delay in very low microseconds, we had to ensure that the sending and the receiving Analyzers were synchronized to the same clock, otherwise the frame capture timestamps would not be accurate. The test bed shown in Figure 4-1 setup was used:

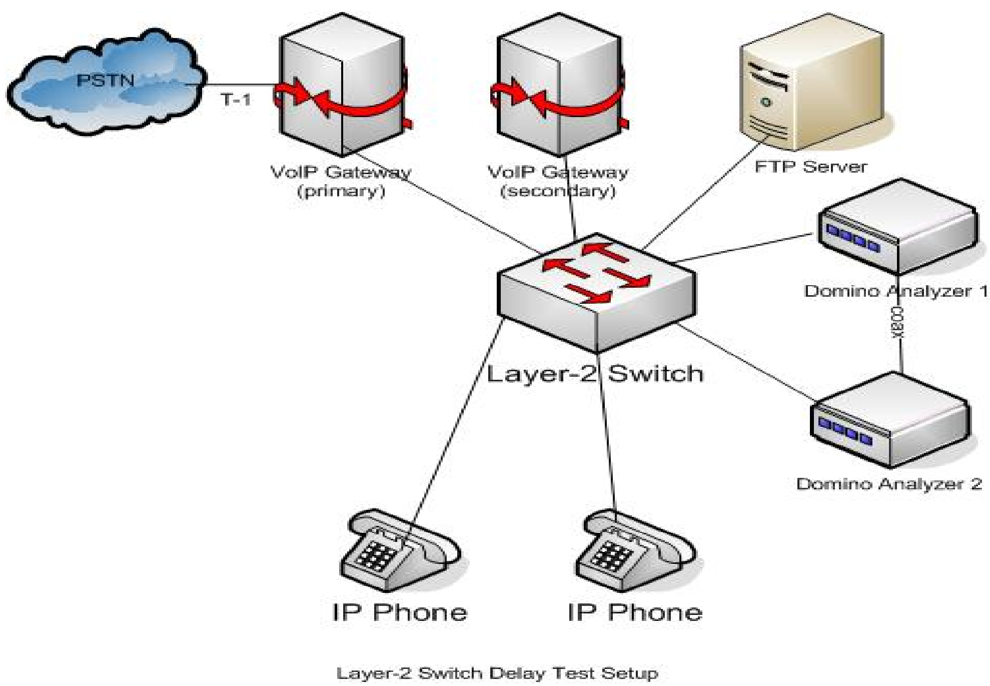

Figure 4-1 – Layer-2 Switch Delay Test Setup

4.2.2 Delay caused by VoIP Gateway

The delay testing of the VoIP Gateway presented two aspects:

a) Measuring the software processing delay incurred by packets traversing the VoIP Gateway.

b) Measuring the processing delay incurred by the packets traversing the T-1 card in the VoIP Gateway.

For Test (a) we relied on analyzing captures of VoIP packets (signaling and audio data) passing through the VoIP Gateway. The VoIP data was captured over a period of one week and then the IP Phone-to-IP Phone calls were analyzed to determine the delay mentioned in (a) above. The test bed setup shown in Figure 4-2 was used:

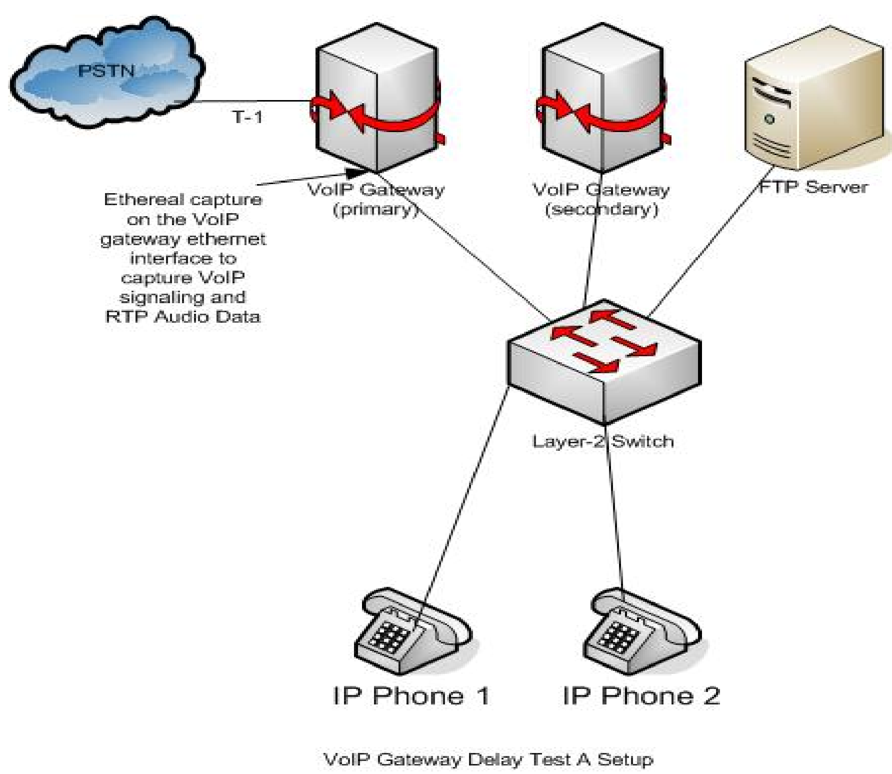

Figure 4-2 – VoIP Gateway Delay Test (a) Setup

For Test (b) we ran into a problem of capturing the T-1 frames on the PSTN side. The WG Internetwork Analyzer did not have T-1 interfaces. Hence, we acquired a T-1

Network Analyzer to solve this problem. The test bed setup shown in Figure 4-3 was used:

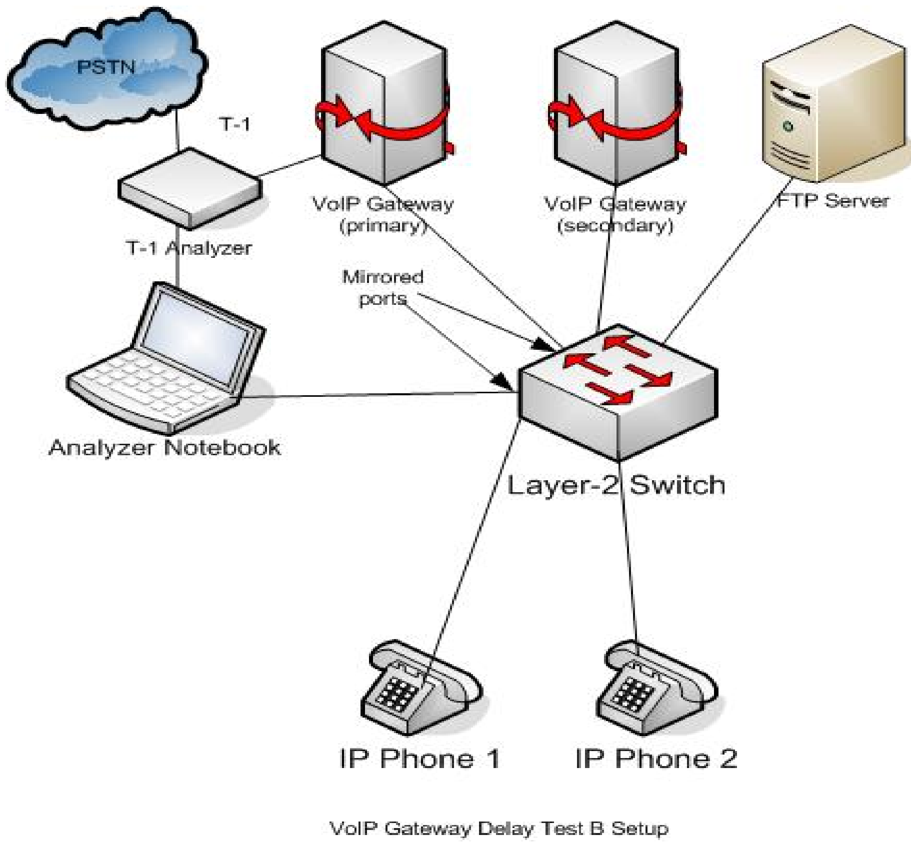

Figure 4-3 – VoIP Gateway Delay Test (b) Setup

4.2.3 IP Phone to PSTN End-to-End Voice Call Delays

For determining the Echo problem, we are concerned with the round-trip delay. Let's consider a scenario in which IP Phone 1 places a call to the analog phone as shown in Figure 4-4.

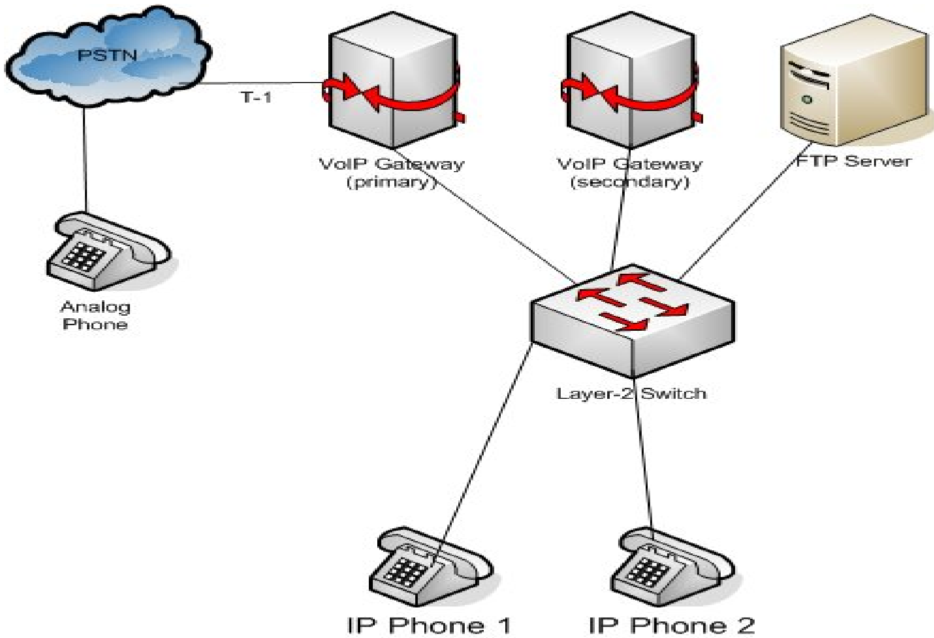

Figure 4-4 – IP Phone to PSTN End-to-End Voice Call Delay Setup

After the initial call setup signaling messages, the IP phone sends out VoIP packets the gateway every 20 ms, which means that each packet contains 20 ms of voice payload. Therefore, the IP Phone must wait to collect 20 ms of the speaker's voice before it can fill the first packet. The VoIP packet should arrive at the VoIP Gateway very fast with negligible propagation delay due the high speed Ethernet network. After this the VoIP gateway transmits the packets on the T-1 line. The return path from the analog phone to the VoIP Gateway is of the order of a few milliseconds. The return VoIP packets received at IP Phone 1 are not played out at IP Phone 1 immediately upon receipt, instead the IP Phone 1 puts incoming packets into a de-jitter buffer of at least 1 packet size (20 ms). The delay budget analysis done in Section 2.3 results in a round-trip delay of approx. 77ms in our scenario assuming no unusual buffering or queuing is occurring along the call path.

4.3 Network Performance Measurement Results and Analysis

4.3.1 Jitter Results and Analysis

The Jitter measurements were taken by analyzing the RTP streams passing through the VoIP Gateway for IP Phone to IP Phone call and IP to PSTN calls. 200 and 245 represents the extension phone number. As can be seen from the Table 4-1, the value of jitter for both IP Phone to IP Phone and IP Phone to PSTN call scenarios is very low indicating good IP network conditions.

Scenarios	Mean Jitter (ms)	Stdev	Variance	95% Confidence Interval		
IP Phone to IP Phone				Lower		Upper
245 to IP-PBX	0.320	0.114985	0.013222	0.205079		0.43505
IP-PBX to 245	0.099	0.02625	0.000689	0.072895		0.125395
200 to IP-PBX	0.314	0.112849	0.012735	0.201445		0.427143
IP-PBX to 200	0.106	0.037774	0.001427	0.068313		0.143861
IP Phone to PSTN						
245 to IP-PBX to PSTN Number	0.014	0.037164	0.001381	-0.02255		0.051773
IP-PBX to 245	0.115	0.054167	0.002934	0.061007		0.169341

Table 4-1 – Jitter Results and Analysis

It is notable here that there are 4 separate RTP streams being exchanged for the IP Phone to IP Phone scenario and 2 RTP streams for the IP Phone to PSTN scenario. It is important to take measurements for the streams separately to get accurate results. There was no packet loss observed in our measurements.

The plots in Figure 4-5, Figure 4-6, Figure 4-7, Figure 4-8, show the Cumulative Distribution Function (CDF) plots of jitter measurements for different RTP streams in the IP Phone to IP Phone. It is notable to observe that these plots are different for even the two RTP streams that are part of the same VoIP call.

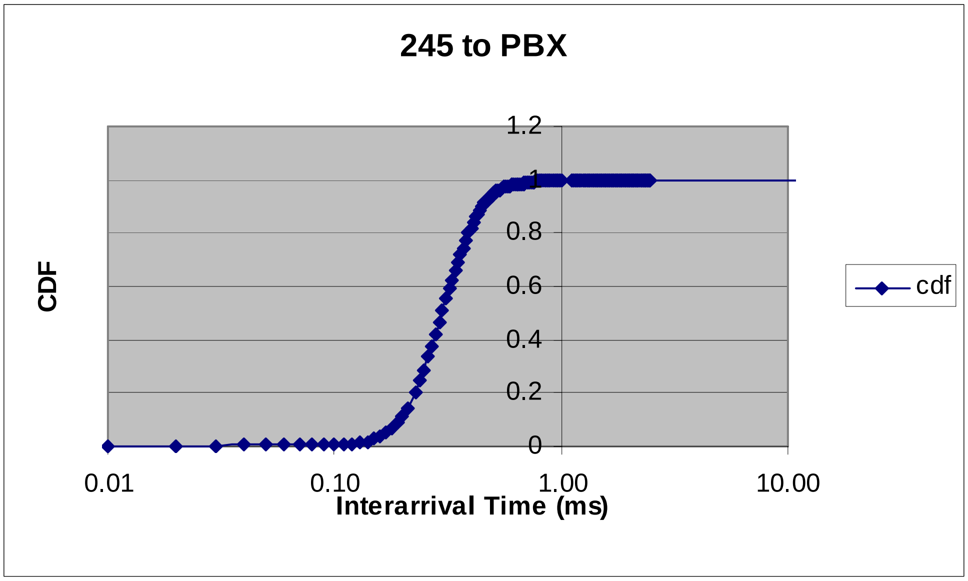

Figure 4-5 – Extension 245 to PBX Jitter CDF plot

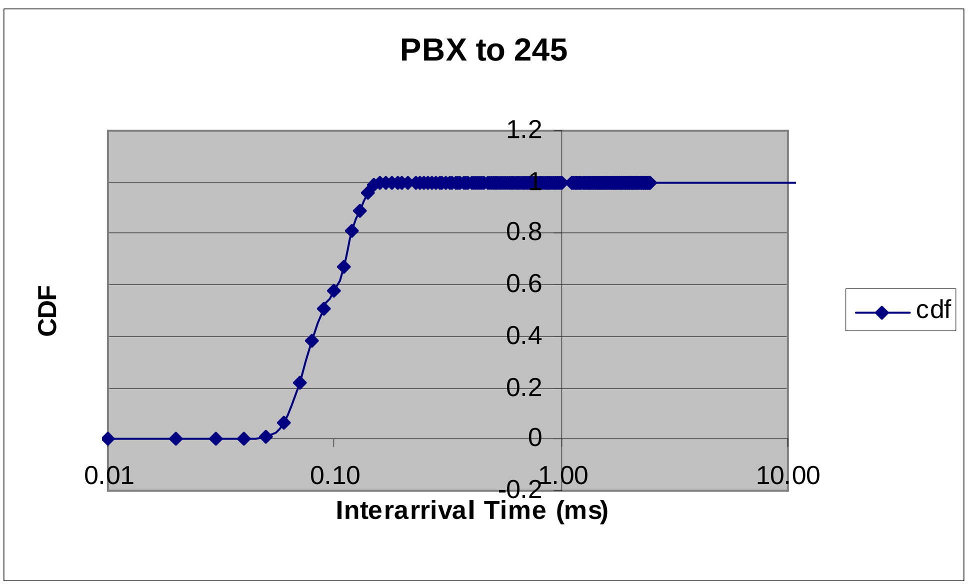

Figure 4-6 – PBX to Extension 245 Jitter CDF plot

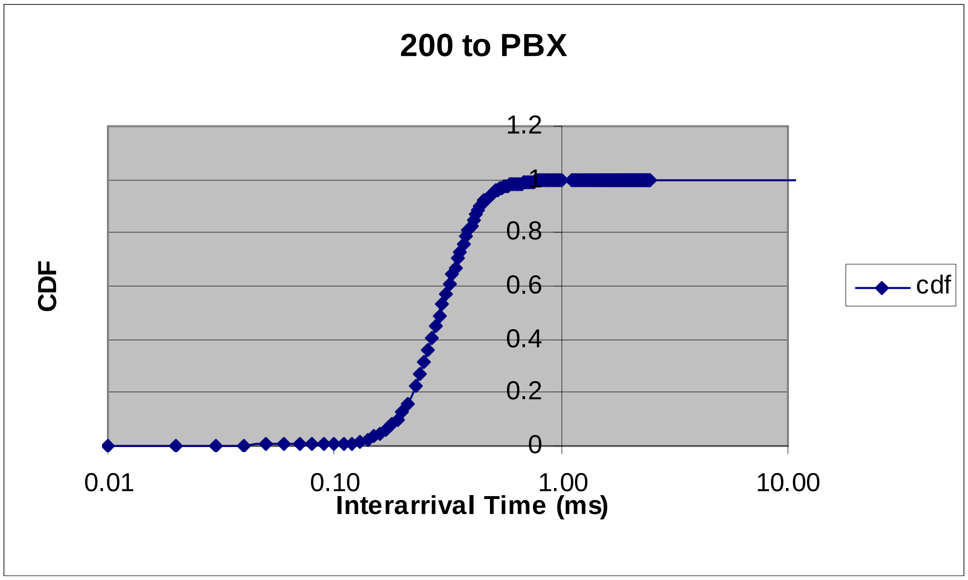

Figure 4-7 – Extension 200 to PBX Jitter CDF plot

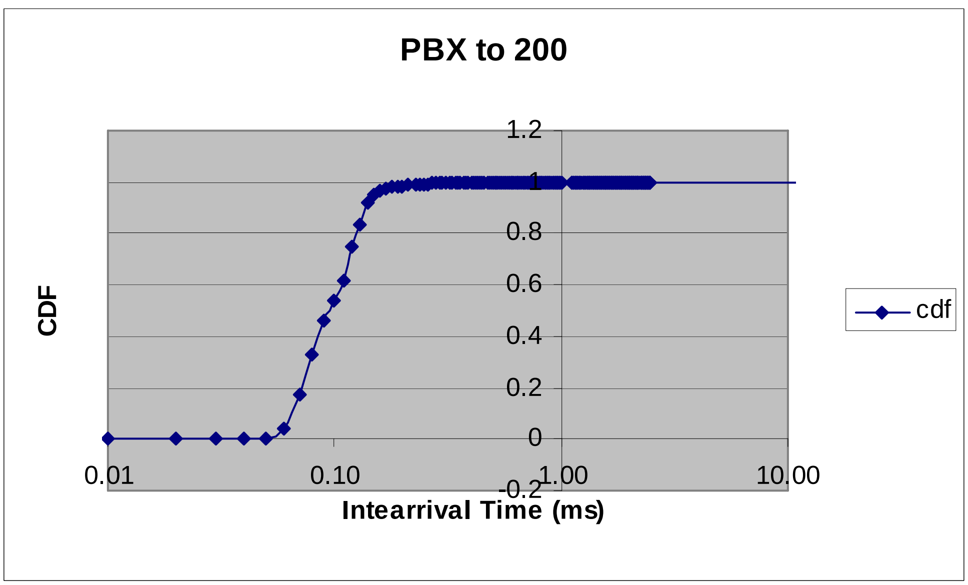

Figure 4-8 – PBX to Extension 200 Jitter CDF plot

The plots in Figure 4-9, and Figure 4-10 show the Cumulative Distribution Function (CDF) plots of jitter measurements for different RTP streams in the IP Phone to IP Phone to PSTN call scenarios.

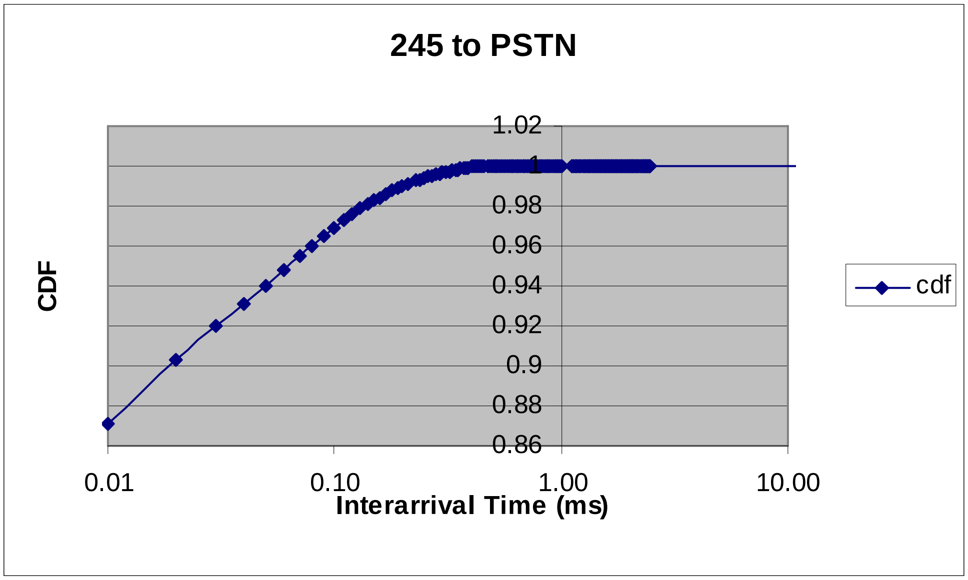

Figure 4-9 – Extension 245 to PSTN Jitter CDF plot

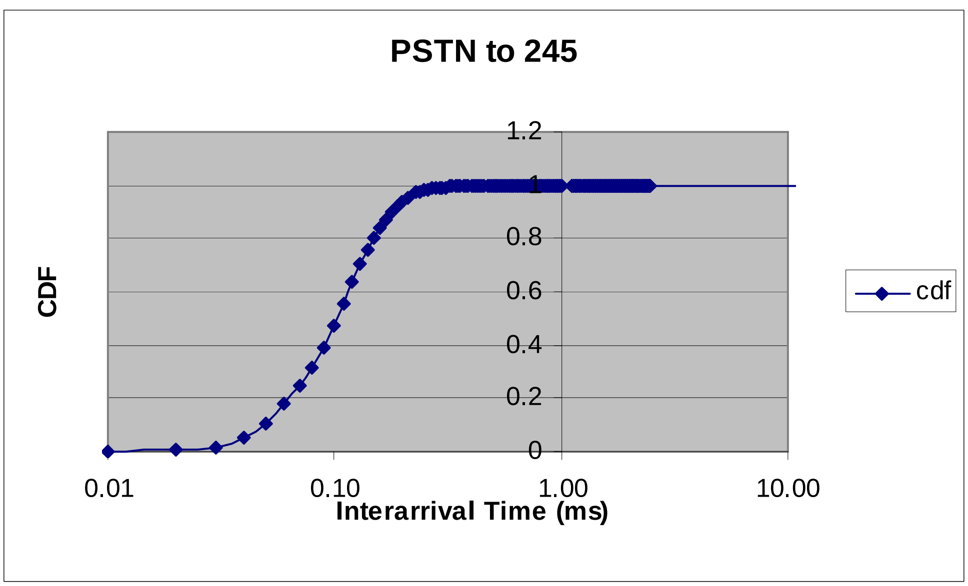

Figure 4-10 – PSTN to Extension 245 Jitter CDF plot

4.3.2 Delay Results and Analysis

4.3.2.1 Layer-2 Switch Testing Results

We observed that the Layer-2 switch is contributing an average delay of 10 microseconds approximately. Since such low delay value could not possibly contribute to the delay leading to any issues related to jitter or echo.

4.3.2.2 VoIP Gateway Testing Results

VoIP Gateway Test (a) - We observed in Test (a) that the software processing delay added by the VoIP is not significant. The VoIP packet inter-arrival time for all RTP streams corresponding to all IP Phone to IP Phone calls varies is approximately 20 ms. This is consistent with the codec processing characteristic of creating a VoIP packet every 20 ms and indicates that the VoIP Gateway is receiving a steady stream of packets from the IP Phone and the VoIP packets are incurring minimal negligible propagation delays while traveling the VoIP network.

VoIP Gateway Test (b) - We observed in Test (b) that there is a significant variation in the delay measurements for IP Phone to PSTN call. The packet inter-arrival time between the VoIP packets arriving from the IP Phone at the VoIP Gateway is approximately 20 ms but the frame inter-arrival time between frames leaving the T-1 interface varies between 10 ms – 104 ms. As an interesting observation, in the same Test (b), the inter-arrival time between VoIP packets belonging to the incoming stream from the PSTN side to the IP Phone side is approximately 20 ms as well, which is most likely due to jitter buffers and protocol translation at the VoIP Gateway as the incoming T-1 data has to be IP packetized in regular stream 20 ms chunks before it can be sent over the VoIP network to the IP Phone.

4.3.2.3 End-to-end Voice Call Delay Results

The Table 4-2is the IP to PSTN Delay Budget table based on our test observations:

Type of Delay	**Fixed Delay**	**Variable Delay**
Sender Processing Delay - Look ahead time - Encoding, compression and packetization	 0 ms 20 ms	
Network Delay - Insertion Delay - Propagation Delay - Queuing Delay (Outgoing) - Packetization/De-jitter Buffer Delay at VoIP Gateway (Incoming)	 1.1 ms .667 ms	 10-104 ms 20 ms
Receiver Processing Delay - Depacketization and decoding - De-Jitter Buffer delay	 5 ms	 40 ms
Total Delay	26.767 ms	70 -164 ms

Table 4-2 – IP to PSTN Delay Budget Measurements

Based on the above delay values, we notice that the round trip delay for an IP Phone to PSTN call varies between 96.767 ms and 190.767 ms.

For reference purposes, the Table 4-3 shows the IP Phone-to-IP Phone Delay Budget table based on test measurements.

Type of Delay	**Fixed Delay**	**Variable Delay**
Sender Processing Delay - Look ahead time - Encoding, compression and packetization	 0 ms 20 ms	
Network Delay - Insertion Delay - Propagation Delay - Queuing Delay	 .01 ms	
Receiver Processing Delay - Depacketization and decoding - De-Jitter Buffer delay	 5 ms	 40 ms
Total Delay	25.01 ms	40 ms

Table 4-3 – IP Phone-to-IP Phone Delay Budget Measurements

Based on the above delay values, we notice that the round trip delay for an IP Phone to IP Phone call is approximately 65.01 ms.

Chapter 5 Conclusion

5.1 Comparison of simulation results with the network implementation measurements

Now, we compare the simulation results with the results observed through measurements. This comparison is based calculations and measurements done earlier in Section 4.2.

Scenarios	Average Jitter (ms)	End-to-End Delay(ms)
IP Phone to IP Phone		
Simulation	0	48
Network Implementation	0.09 to .31	65.01
IP Phone to PSTN		
Simulation	9.6	50
Network Implementation	0.01 to .11	96.767 to 190.767

Table 5-1 – Jitter and End-to-End Delay Comparison

5.1.1 Average Jitter (ms)

The variation in the average jitter in the case of IP Phone to PSTN scenario is due to the fact that the simulation generated more VoIP calls that generated traffic to bottleneck the T-1 link. In our implementation scenario, we didn't have the capability to simulate simultaneous calls for this project.

5.1.2 End-to-End Delay (ms)

The variation between the End-to-End delay values is because the IP Phones have an adaptive de-jitter buffer and the simulation used a de-jitter buffer size of 20 ms whereas by observing the de-jitter buffer size on the IP Phones, we observed that the de-jitter buffer size varies between 30-50 ms and hence we used a de-jitter buffer size of 40ms in our calculations leading to the difference in values.

As noted earlier, the end-to-end delay in the case of IP Phone to PSTN Scenario network implementation is a cause of concern and investigation of reasons behind this is done in Section 2.4 and Section 4.2

The End-to-End Delay observations led us to determine that the VoIP gateway is adding significant delay resulting in the total round trip delay reaching 190.767 ms for some packets, which is most likely causing the intermittent audible echo issue. Since this unusual delay is only occurring on the outgoing stream packets, hence our suspicion is that the problem lies with the DSP chip on the T-1 DSP card. As mentioned earlier, in this case the hybrid or acoustic echo generated at the PSTN called party side will reach the IP Phone listener at the calling party side and since it is delayed by such a significant amount, it will be clearly audible. This was also consistent with intermittent echo issues as reported by some IP Phone users.

We determined that installation of Echo Cancellers in the VoIP network or installation of a better T-1 card with echo cancellation capability in the VoIP Gateway would address the issues created by excessive delay in the voice call path.

5.2 Conclusion

The testing and analysis confirmed that the normal operational characteristics of packet-based networks introduce delays in the call path that can lead to voice quality problems such as echo, which were not noticeable in the PSTN world due to lower delays. Depending on the size of the de-jitter buffers, the end-to-end delay can easily reach over 65 ms even for a network without any bottleneck or congestion. Hence, with the integration of VoIP networks with PSTN and the associated increase in end-to-end delay in VoIP call path, provisioning of high quality and high performance VoIP gateways and echo cancellers is a requirement in VoIP networks and thorough testing needs to be done before the VoIP networks are deployed to avoid problems at a later stage.

Chapter 6 Future Work

6.1 Development of a Load Balanced VoIP Solution

The work done in this project will support the development of a heartbeat based redundant load shared system with four identical VoIP gateway boxes, all running Asterisk software. This would allow for better scalability and availability by load sharing and distributing among multiple processing boxes. The plan is to integrate the Session Border Controller (SBC) functionality as discussed in Section 2.5.3 to address the NAT traversal and call control issues.

6.2 VoIP Gateway Architecture Analysis

Another possibility arising out of this project is further analysis of different types VoIP Gateway architectures with emphasis on how the different architectures handle the transcoding, protocol translation aspects of a VoIP call and what impact it has on the buffering of data that leads to high end-to-end delays as explained in Section 4.3.2

6.3 Echo Canceller Architecture Analysis

As we noted in various sections of this report especially Section 4.3, the fundamental nature of VoIP networks introduces delays. Some can be controlled and some are unavoidable. It would be very interesting to conduct a detailed analysis of architecture and analysis of different types of Echo cancellers and possibly improve their design to address the VoIP challenges.

6.4 SIP VoIP Performance Testing Application

During the project, I realized the need to test the performance of the VoIP network by generating multiple simultaneous calls and compare with the simulation performance results. I investigated into various methods to do this and discovered the SIPp performance testing tool. The primarily purpose of this tool is to test the performance of the SIP call processing engine but it can be used to transmit and receive RTP media streams, which can be used to load test the network devices and links although the RTP functionality is very limited.

It would be useful to understand the SIPp code and possibly develop some more code to expand the RTP testing capability. This would be a good alternative to some heavily priced SIP and RTP testing tools available in the market.

Chapter 7 References and Related Documents

[1] Asterisk Documentation Project at www.asteriskdocs.org.

[2] RFC3261 – Session Initiation Protocol (SIP), Rosenberg, Schulzrinne et all. June 2002.

[3] http://www.cs.columbia.edu/sip/ - A good resource for SIP.

[4] RFC 3581 - An Extension to the Session Initiation Protocol (SIP) for Symmetric Response Routing, Rosenberg, Schulzrinne et all. August 2003.

[5] RFC 3489 – STUN Simple Traversal of User Datagram Protocol (UDP) through Network Address Translators (NATs), Rosenberg, Weinberger et all. March 2003.

[6] Echo Analysis for Voice over IP, Cisco Systems. www.cisco.com

[7] NAT Traversal for Multimedia over IP, Newport Networks http://www.newport-networks.com

[8] ITU-T G.712 (11/01) – Transmission Performance Characteristics of Pulse Code Modulation (PCM) Channels. 2001. (Replaces G.712, G.713, G.714, G.715)

[9] An OPNET based Simulation Approach for Deploying VoIP. K. Salah and A. Alkhoraidly. International Journal of Network Management. 2003

[10] IP Telephony – Deploying Voice over IP Protocols, Olivier Hersnet, Jean-Pierre Petit, David Gurle. John Wiley and Sons Ltd 2005

[11] RFC 2330 – Framework for IP Performance Metrics. V. Paxson, G. Almes, J. Mahdavi, M. Mathis. May 1998.