

VoIP Implementation Using Asterisk PBX

Mohammad Masudur Rahman¹, Nafish Sarwar Islam²

^{1,2} (Department of Business Administration, East West University, Bangladesh)

Abstract : *The ability to transmit and process voice over Internet protocol (VoIP) networks has important implications for technology users especially by the using Asterisk PBX. Many companies nowadays are rushing to bring different VoIP products to market with a wide variety of features. This article will focus on the introduction of VoIP and its implementation by the use of Asterisk PBX. Firstly, the paper presents the project objective with some introductory theory about VOIP. Secondly, the project includes report on the viability of utilizing the Asterisk PBX as a foundation for conducting research performance studies for VoIP. Finally, the project is showing on live experimental studies of SIP voice traffic. The article experimentally studied the performance of voice calls initiated using SIPp simulator for testing SIP protocol performance and found much more stability and accuracy using Asterisk PBX. The purpose is to suggest those VoIP technology attributes that best meet users' needs. Asterisk, the open source PBX of choice is used to show that this is maturing fast and ready for main stream VoIP implementation.*

Keywords: VoIP, SIP, PBX, GSM, ISDN, TCP, RTP.

I. INTRODUCTION

The development of Asterisk based on Voice Exchange and its applications needs to take into account the growing complexity with the existing PBX (private branch exchange) networks. A promising solution is the voice exchange based on Asterisk, which works on VoIP (Voice over Internet Protocols) and provides a more promising and flexible solution (1). VoIP technology was originated in 1995. At that time hobbyists began to realize the potential of sending voice data packets over the Internet rather than communicating through standard telephone systems. The proposal is to use the internet as a telephone network with some additional capabilities. Instead of communicating over a circuit switched network, this application allows communication between two parties over the packet switched Internet and it was in 1995, when the first internet soft phone appeared (2).

This concept allowed PC users to avoid long distance charges, hence great for international calling and a promising solution to the corporate world. In fact one of the main areas of its applications is the call centres, where the customer is able to communicate with an operator via the internet. Recently there have been proposals in various countries to allow internet telephony, which is indeed a welcome step. This means that people can use VoIP services to connect the traditional telephone. In this case call travels across the IP network, but terminates at the local phone network, whose owner than charges a fee that is typically the price of a local call.

The development of Asterisk Based Voice Exchange which works on VoIP takes into consideration the various complexities associated with a conventional Private Branch Exchange (PBX). A conventional circuit switching based PBX is not only expensive but limited in terms of functionality as well (1). Asterisk is a complete phone system in software. It can replace large and expensive phone systems powering thousands of extensions, or it can help users save money on long international call usages. Because being implemented in software, it's extremely versatile, easy to customize as and easy to extend Unrestricted VoIP, which is the basic building block of Asterisk based voice exchange makes it possible for people to use their phones either fixed or mobile to call others in the country and internationally for a fraction of the prices they pay now.

All that one needs to make a PC to PC call is a VoIP based software. Beside this, economies also play an important role in popularizing this new concept (3). A normal telephonic call requires 64 kbps, whereas the same can be handled an as low as 6-8kbps using codecs such as GSM. Hence the bandwidth requirements will be considerably reduces and one can stuff up, more calls in the same given bandwidth, providing a clear cut advantage. In future people will stop paying for voice calls just like for each e-mail they sent or each site course a service provider would charge providing internet services, but ever beyond that point wouldn't carry any future where everyone will always be convergence several commentators about will finally become a reality (1).

Objectives and Scope

As stated in the article's scope of work, the primary focus of this article will consider developing and implementing VoIP using Asterisk PBX. In realizing this, the following objectives shall be met:

- Understand the important and benefit of VoIP as the medium of communication.
- To fully utilize Asterisk private branch exchange and other open source software for the article.
- Proposing several suggestions to enhance the current VoIP implementation.
- Develop a medium scale network that implements VoIP using Asterisk PBX.

This article will include:

- Research on the current development of VoIP by the industry.
- Development of an exemplary model server using the Asterisk PBX.
- Comparing conventional telephone architecture with VoIP.
- Suggesting how the VoIP using Asterisk can further be improved.
- Measuring the service quality performance of VoIP using SIPp as a performance testing tool.

II. LITERATURE REVIEW

A large portion of literature on Asterisk reveals that it is different for many reasons, the most important being it's all software approach. Instead of switching analogue lines in hardware, it routes and manipulates Voice over Internet Protocols (VoIP) Packets in software. The backbone of the system generally becomes an IP enabled network, and phones can be hooked into that. However it also supports old analogue phones using gateway devices (4). Asterisk provides more than what one would expect from a conventional PBX. Users get a variety of features such as paging, (which may be from one-to-one or many to-one, depending on the usage requirements), Interactive voice responses (IVR), Conferencing, Voicemail, Music on hold to name a few.

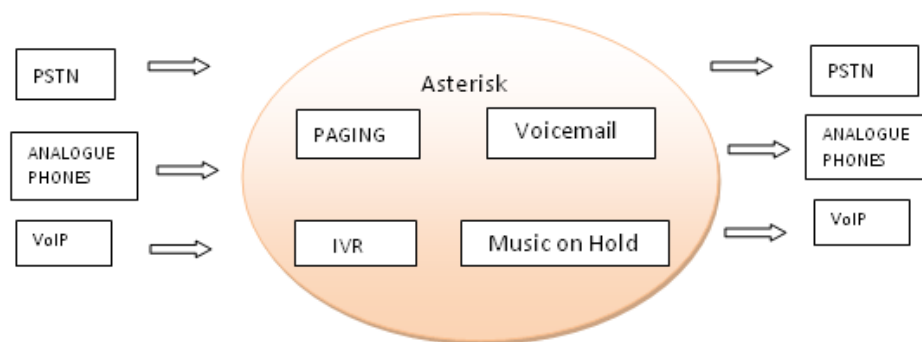


Fig 1: Overview of ASTERISK based System

On top of that users can get interfaces to the operating system and programming languages for the extreme in power, optional web based administration interfaces, configuration in SQL databases or flat files, detailed call logging into a database and many more features (3). Hence to summarize it up, it can be said that that with Asterisk user can:

- Provide basic service to Analogue and Digital phones.
- Develop a call routing logic in order to choose a least expensive way to route a particular call.
- Route Incoming and Outgoing voice calls over standard voice lines or the internet (5).
- Provide voicemail and teleconferencing services
- Develop complex or simple interactive menus
- Operate small or large queues for call centers
- Announcing the estimated hold time to the callers
- Call other programs on the system.

Asterisk's Architecture

Asterisk is carefully designed flexibility. Specific APIs are defined for maximum around a central PBX core system (1). This advance internal interconnection of the PBX from the specific protocols, codecs hardware interfaces from the telephony application. This allows Asterisk to use any suitable hardware and technology available now or in the future to perform its essential functions, connecting hardware applications.

PBX Switching

The essence of Asterisk is of course a PBE connecting calls together between automated tasks. The Switching transparently connects callers arriving on various hardware and software interfaces (6).

Application Launcher

Application Launcher launches applications which perform services for users, such as voicemail, file playback, and directory listing (2).

Codec Translator

It uses codec modules for encoding and decoding of various formats used in the telephony industry. A number of codecs are available to ensemble diverse needs and arrive at the best balance between audio quality and bandwidth (3).

Loadable Module API'S

Four APIs are defined for loadable modules, facilitating hardware and protocol abstraction. Using this loadable module system, the Asterisk core does not have to agonize about details of how a caller is connecting, what codecs are in use, etc.

Using these APIs, Asterisk achieves an absolute abstraction between its core functions as a PBX server system and the varied technologies existing (or in development) in the telephony arena. The application API provides for supple use of application modules to perform any function flexibly on demand, and allows for open development of new applications to suit unique needs and situations (2). In addition, loading all applications as modules allows for a bendable system, giving administrators the ability to design the best suited path for callers on the PBX system and modify call paths to suit changing communication needs.

Channel API

The channel API handles the type of connection a caller is arriving on, be it a VoIP connection, ISDN, PRI, Robbed bit signaling, or several other technology (3). Dynamic modules are loaded to hold the lower layer details of these connections.

Application API

The application API allows for various task modules to be run performing various functions like Conferencing, Paging and Directory Listing. Voicemail, In-line data transmission, and any other task which a PBX system may execute now or in the future are handled by these separate modules (3).

Codec Translator API

It loads codec modules to support various audio encoding and decoding formats such as GSM, Mu-Law, A-law, and even MP3.

File Format API

This handles the reading and writing of various file formats for the storage of data in the file system (5). Using these APIs Asterisk attains a complete abstraction between its core functions as a PBX server system and the diverged technologies existing (or in development) in the telephony arena. The modular form is what allows Asterisk to seamlessly incorporate both currently implemented telephony switching hardware and the growing Packet Voice technologies emerging today (6). The ability to load codec modules allows Asterisk to sustain both the tremendously compact codecs necessary for Packet Voice over slow connections such as a telephone modem while still providing high audio quality over less constricted connections. The application API provides for bendable use of application modules to perform any function flexibly on demand, and allows for open development of new applications to suit unique needs and situations. In addition, loading all applications as modules allows for a flexible system, permitting the administrator to design the finest suited path for callers on the PBX system and modify call paths to suit the changing (6).

III. PROJECT DESIGN

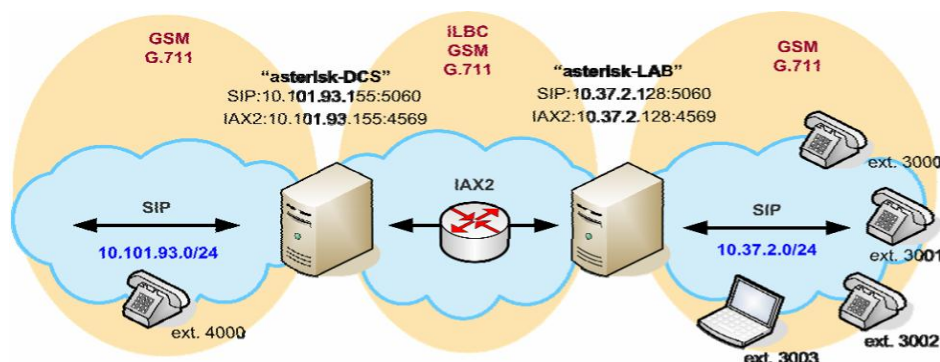


Figure 2: Physical Design

The project basically consists of 2 different networks with an Asterisk server placed in each network. Both servers labeled as asterisk-DCS and asterisk-LAB each. Extension 4000 is registered with Asterisk-DCS server using SIP.

Phones with extension 3000, 3001, 3002 and 3003 are registered to the Asterisk-LAB server using SIP also. However RTP is the actual protocol to transfer real-time voice data between clients and server in collaboration with SIP which is used for signaling between clients and server. For the connection between two Asterisk servers, IAX2 is used as a trunk to transfer both signaling and real-time voice data.

GSM, G71 1 and iLBC are codec types that are given permission (by administrator) to be transmitted within the particular protocol (SIP/IAX2) defined.

IV. EXPERIMENT AND MEASUREMENT

Server Stress Load Test

- Experimental Setup

This experiment focuses on comparing the VoIP servers load performance between two different servers of unequal architectures. The two boxes were loaded with Fedora 7 (F7) and Asterisk version 1.4.x. The first one is AMD Sempron 1.5GHz with 512 MB RAM and the other one is more powerful with Pentium D Dual Core 2.8GHz which had 1.5GB RAM.

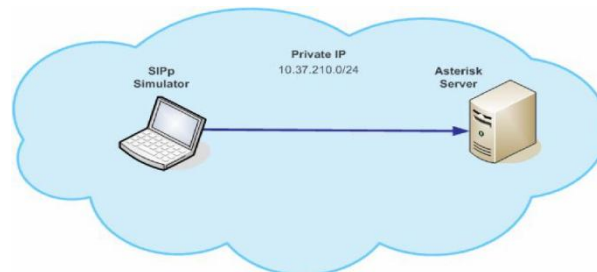


Figure 3: Server Stress Test configuration

The SIPp simulator was used to generate voice calls of short length using a specific codec for the Asterisk server located on the same network through SIP signaling protocol and RTP for real time voice transmission. For controlled testing, studies with G711 and GSM codec performances were undertaken by establishing some amount of predefined concurrent calls between SIPp simulator and Asterisk servers. The concurrent calls were terminated after 10 seconds call duration and the CPU load percentage were recorded. This process was repeated for a number of concurrent calls until the CPU load reached 100% or the significant call failure occurred.

In order to increase the concurrent calls, system file descriptor has to be modified from the default setting. The following steps have to be referred:

Open file /etc/sysctl.conf

add line *fs.file-max = 200000*

To increase user file descriptor, following lines have to be referred:

Open file /etc/security/limit.conf

add the following lines: *asterisk soft no file 200000 asterisk hard no file 200000 root soft no file 200000 root hard no file 200000*

Asterisk server configuration files are as follows:

extensions.conf

```
[general]
static=yes
writeprotect=yes
;
[stress-test]
;
for codec translation
exten => 1111,1,Answer()
exten => 1111,2,Echo()
exten => 1111,3,Hangup()
;
;
codec processing(G711 a sound sample)
exten => 3333,1,Answer()
exten => 3333,2,Playback(alaw-demo-instruct)
exten => 3333,3,Hangup()
;
;
codec processing(GSM sound sample)
exten => 4444,1,Answer()
exten => 4444,2,Playback(gsm-demo-instruct)
exten => 4444,3,Hangup()
```

Sip.conf

```
[general]
context=default
realm=asterisk-A
bindport=5060
bindaddr=0.0.0.0
language=en
canreinvite=no
dtmfmode=rfc2833
compactheaders = yes
registertimeout=20
registerattempts=10

[sipp]
type=friend
context=stress-test
host=dynamic
user=sipp
canreinvite=no
disallow=all
allow=gsm
allow=alaw
```

Network stress load test

- Experimental Set up

This test were undertaken to measure the capability of existing network to handle VoIP traffic. Figure 2 display the set up of the second experiment that was to conduct network stress load. Asterisk servers were placed in two different sub networks and connected through a router and IAX2 trunk. All signaling and real-

time voice transferred using SIP and RTP protocol. The experiment was conducted during peak load on the network at about 11:00 AM in the morning time. For comparison purpose the article took a second reading in the night time around 10:00 PM to see what is the effect of network load on the VoIP simulation are.

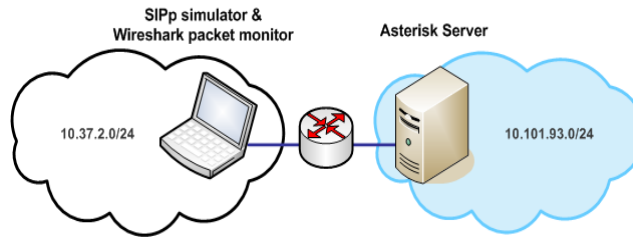


Figure 4: Network Stress Test Configuration

The project had to use G711 codec as project SIPp server was not properly configured for GSM codec. Though this gives less than optimum performance but was good indication what might expect in GSM implementation? GSM codec makes much smaller bandwidth consumption in comparison to G711 codec. That is why GSM implementation will be more efficient.

V. RESULTS AND DISCUSSIONS

The project is analyzing the server stress load test result.

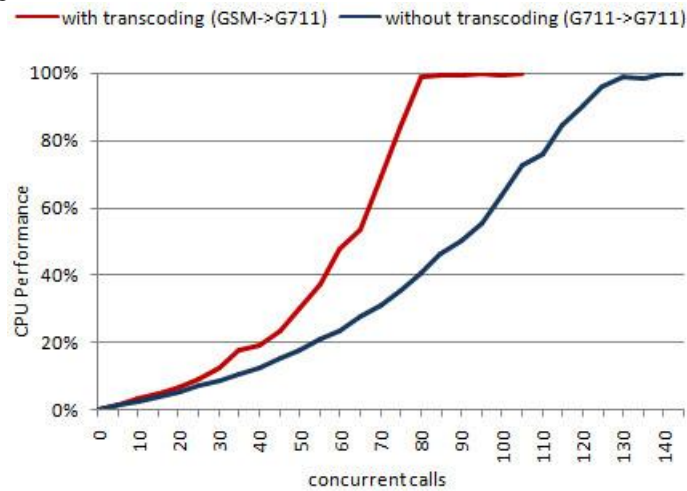


Figure 5: AMD Sempron 1.5GHz Performance

Based on Figure 5 refer to the results revealed from the first machine, AMD Sempron 1.5GHz. It shows that the optimal CPU load (60-70%) for concurrent calls (One way calls / 2) are 35 calls for codec transcoding and 53 calls for without codec transcoding.

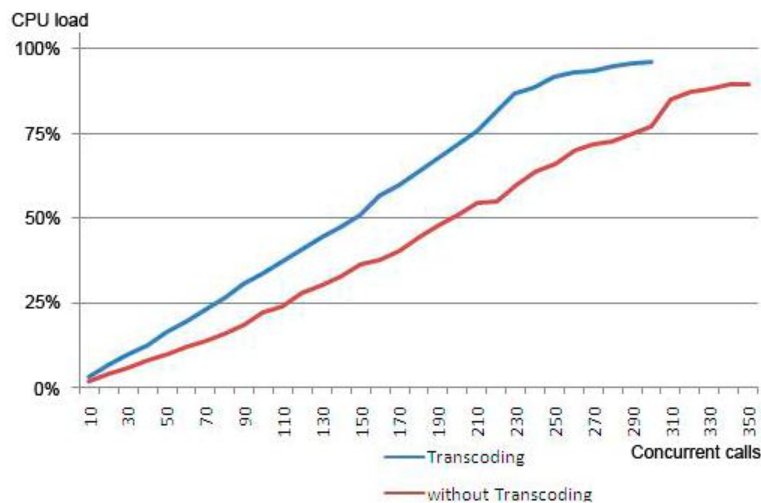


Figure 6: Pentium D 2.8GHz Performance

Figure 6 displays the results of the second machine, Pentium D 2.8GHz. It shows that the optimal CPU load for concurrent calls (One way call / 2) are 100 calls for codec transcoding and 130 calls for without codec transcoding. The reason for these high percentages is because of the processor itself using the latest technology, Centrino Duo processor.

As the Asterisk refused to send GSM codec voice to SIPp simulator, it is unable to purely measure the GSM codec performance through the server stress load. In the below project is analyzing the Network stress load test result.

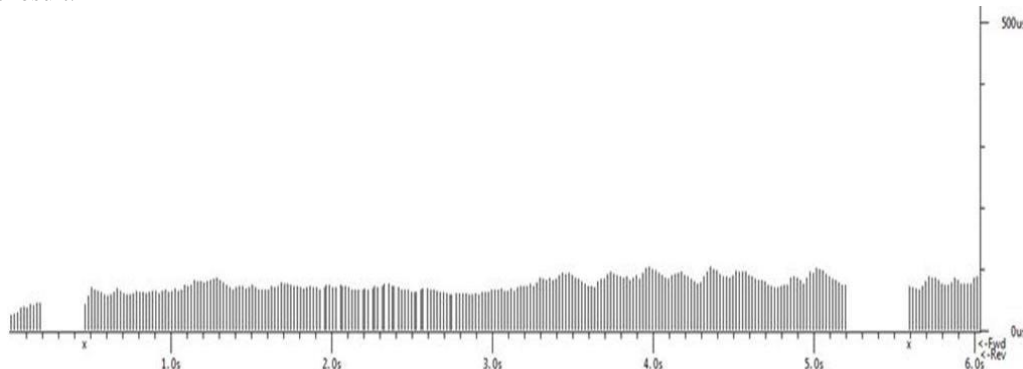


Figure 7: G711 worst case scenario

Figure 7 shows a worst case scenario during network stress test using G711 codec. The test is conducted using 50 concurrent calls with 10 seconds in duration. There is no big issue in packet jitter and delay since the overall delay is less than 0.22ms. However, it shows that the packet lost happened during 1st second and 6th second are about 6.7% packets lost. This might happened at the router buffer where some packets are dropped because of the processing capability of the router itself cannot handle such a large traffic. In such amount of packet lost, it gives a big significant in voice quality for a long conversation.

As the Asterisk refused to send GSM codec voice to SIPp simulator, it is unable to purely measure the GSM codec performance through the network stress load test.

Solution for this worst case scenario, we can use a different codec than did not consume more bandwidth such as GSM as a short term solution. However it is suggested to upgrade the router up to VoIP standard for long term benefit.

VI. CONCLUSION

Essentiality of the communication in the global world is the core area of concern, since people increasingly becoming relay on Internet. This is so with the voice over internet protocols (VOIP) which has now become the most useful technology to communicate for long distance calling. VoIP is a fast growing technology in IP network, which requires real time support as it is time sensitive application. VoIP in IP network is designed for data communication, but to achieve reliable, high-quality voice over the IP network is an engineering challenge. For designing a good quality VoIP implementation using Asterisk PBX system includes choosing the best codec and applying perfect technique.

This article can go further to have more details study with more codec available to find out the effect of such compression on the CPU load. As there is a new version of Asterisk with IPv6 support available, this project can be continued to tests on IPv6 and on the same hardware then have used for current tests on IPv4.

REFERENCES

- [1] Davidson, J., Peters, J., Gracely, B. 2000, Voice Over IP Fundamentals, CISCO Press, Indiannapolis, IN, pp 56-120.
- [2] Ghita, B.V., Furnell, S.M., Lines, B.M., Le-Foll, D., Ifeachor, E.C. (2001), "Network quality of service monitoring for IP telephony", Internet Research: Electronic Networking Applications and Policy, Vol. 11 No.1, pp, 26-34.
- [3] Thorne, D.J. (2001), "VoIP-the access dimension", British Telecom Technical Journal, Vol. 19 No.2, pp.33-43.
- [4] Camp, Ken. 2002. IP Telephony Demystified. McGraw-Hill Professional, pp, 222-254.
- [5] Bourreau, M., Dogan, P. (2004), "Service-based vs. facility-based competition in local access networks", Information Economics and Policy, Vol. 16 No.2, pp.287-306.
- [6] Qadeer, Mohammed A & Ale Imran, 2008, "Asterisk Voice Exchange-An alternative to conventional EPBX" in Proc. IEEE ICCEE 2008, pp 20-22.
- [7] Jim Van Meggelen, Leif Madsen, Jared Smith 2007, "Asterisk the Future of Telephony" Second Edition, Harvard Journals, pp 99-100.
- [8] Andre du Toit, 2001, "Private PBX Networks: Cost Effective Communication Solutions" in Proc. IEEE 3rd AFRICON Conference, pp 5-10.
- [9] David, P.A., A. Waterman, and S. Arora, 2003 The Free/Libre Open Source Software Survey for 2003, Stanford University: Stanford, CA, pp 89-200.
- [10] Danielle, Dunne. 2001, What is VoIP?. Darwin publications. pp 30-34.

- [11] Derek, Slater. 2001, Voice over IP, CIO publications, pp 30-45.
- [12] M. Arango, A. Dugan, I. Elliott, C. Huitema, and S. Pickett. Media Gateway Control Protocol (MGCP), October 1999. RFC 2705.
- [13] R. Bhat and R. Gupta. JAIN protocol APIs. *IEEE Communications Magazine*, 38(8):100–106, January 2000.
- [14] Collins. *Carrier Grade Voice over IP*. McGraw-Hill, 2001.
- [15] T. Eyers and H. Schulzrinne. Predicting Internet telephony call setup delay. In *IPTel 2000*, April 2000.
- [16] S. S. Gokhale and J. Lu. Signaling performance of SIP based VoIP. In *International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS'04)*, pages 190–196, San Jose, California, July 2004.
- [17] A.Kist and R. Harris. SIP signalling delay in 3GPP. In *Sixth International Symposium on Communications Interworking of IFIP - Interworking 2002*, Fremantle WA, October 2002.
- [18] International Engineering Consortium (IEC). Voice over Internet Protocol. <http://www.iec.org/online/tutorials/>
- [19] VoIP History: <http://www.voipreview.org/news.details.aspx?nid=51>
- [20] Extended version of this paper <http://pubs.doc.ic.ac.uk/AsteriskCallManagementPolicy/AsteriskCallManagementPolicyextended.pdf>
- [21] National Institute of Standards and Technology (NIST). IP Telephony Project. <http://dns.antd.nist.gov/proj/iptel/>
- [22] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol, July 2002. RFC 3261.
- [23] Asterisk the Open Source PBX. <http://www.asterisk.org/>
- [24] SIPp Community. <http://sipp.sourceforge.net>