Implementation and Analysis of VoIP application

Saw Mya Aye, Thinn Thu Naing
University of Computer Studies, Yangon
sawmyaaye09@gmail.com,ucsy21@most.gov.mm

Abstract

Voice over Internet Protocol (VoIP) is one of the quickest developing Internet services and steadily replaces traditional telephony. VoIP application provides a means of transmitting voice communication over an IP based network and transmit voice conversations over an internal or external data network using IP packets (digital form) without reducing its functional, and reliability. This system illustrates how VoIP server uses SIP protocol and IAX2 protocols in VoIP applications. Moreover, this system applies VoIP application in small enterprise like UCSY (Hlawgar) campus and it analyzes call information between different sites. In this system, Asterisk open source server is used to implement VoIP application and an analysis system of that VoIP application is implemented by using Java.

Keywords: *VoIP, Asterisk, SIP, H.323, IAX2*

1. Introduction

The VoIP Implementation was selected because of its potential to be commercialized in the real world. The aim of this paper is to setup VoIP server using Asterisk open source. The focus will be on Session Initiation Protocol (SIP) and IAX2 protocol during transmitting voice to other party on the network. To setup a server that provides VoIP using Asterisk we must determine the advantages of using VoIP, check the differences of Asterisk over the type of VoIP server. Early VoIP service relied on advertising sponsorship to subsidize costs, rather than by charging customers for calls. The gradual introduction of broad band Ethernet service allowed for greater all clarity and reduced latency.

Voice over Internet Protocol (VoIP) is a form of communication that allows to make phone calls over a broadband internet connection instead of typical analog telephone lines. Basic VoIP access usually allows to call others who are also receiving calls over the internet. Interconnected VoIP services also allow to make and receive calls to and from traditional landline numbers, usually for a service fee. Some VoIP services require a computer or a dedicated VoIP phone, while others allow to use the landline phone to place VoIP calls through a special adapter, see Figure 1. VoIP converts voice into data and sends the voice packets over the network [6].

Voice over Internet Protocol (VoIP) requires handsets, softphones, gateways, gatekeepers, Conference Bridge, IP PBX, H.323, SIP, and MGCP/Megaco.

The results are the following:

1. Setup server that provides VoIP using Asterisk.
2. Determine the advantages of using VoIP.
3. Analyze the transaction between distributed sites via VoIP.

Figure 1. VoIP Connection

2. Asterisk

Asterisk is an open source converged telecommunications platform, designed to allow different types of IP telephony hardware, middleware, and software to interface with each other consistently. Like any PBX, it allows a number of attached telephones to make calls to one another, and to connect to other telephone services including Public Switched Telephone Network (PSTN). It provides multiple layers, managing both TDM and packet voice at lower layers while offering a highly flexible platform for PBX and telephony applications such as IVR. Asterisk can bridge and translate different types of VoIP protocols like SIP, MGCP, and H.323. At the same time it can provide a full-featured server platform for predictive dialing, custom IVR, remote and central office PBX, and conferencing [8].

Officially, Asterisk is an Open Source hybrid TDM and packet voice PBX and IVR platform with ACD functionality. Unofficially, Asterisk is quite possibly the most powerful, flexible, and extensible piece of integrated telecommunications software available. Its name comes from the asterisk symbol, *, which in UNIX (including Linux) and DOS environments represents a wildcard, matching any filename. Similarly, Asterisk the PBX is designed to interface any piece of telephony hardware or software with any telephony application, seamlessly and consistently [5]. Asterisk server based on Linux. Asterisk is any call, any time, from anywhere to anywhere else.

Typical features supported by Asterisk include [1]:

- Automated Attendant
- Call Detail Records
- Call Forward on Busy
- Call Forward on No Answer
- Call Monitoring
- Call Recording
- Call Transfer
- Call Pickup
- Call Waiting
- Call groups
- Call Blocking
- Calling ID
- Conference Bridging
- Do Not Disturb (DND)
- Music on Hold
- Voice Mail

3. Types of protocol

There are many types of protocol used in Voice over Internet protocol (VoIP) implementation such as Session Initiation protocol (SIP), H.323 Protocol and IAX2 Protocol as the following:

3.1. Session Initiation Protocol

The Session Initiation Protocol (SIP) is an application-layer control (signaling) protocol for creating, modifying and terminating sessions with one or more participants. These sessions include Internet multimedia conferences, Internet telephone calls and multimedia distribution. Members in a session can communicate via multicast or via a mesh of unicast relations, or a combination of these. SIP invitations used to create sessions carry session descriptions which allow participants to agree on a set of compatible media types. SIP supports user mobility by proxying and redirecting requests to the user's current location. Users can register their current location. SIP is not tied to any particular conference control protocol. SIP is designed to be independent of the lower-layer transport protocol and can be extended with additional capabilities [2]. Text based, Model similar to HTTP: uses client-server model.

SIP supports five facets of establishing and terminating multimedia communications [4]:

User location: determination of the end system to be used for communication.

User availability: determination of the willingness of the called party to engage in communications.

User capabilities: determination of the media and media parameters to be used.

Session setup: “ringing”, establishment of session parameters at both called and calling party.

Session management: including transfer and termination of sessions, modifying session parameters, and invoking services [4].

3.1.1. SIP Entities. A SIP network is composed of four types of logical SIP entities. Each entity has specific functions and participates in SIP communication as a client (initiates requests), as a server (responds to requests), or as both. One "physical device" can have the functionality of more than one logical SIP entity.

Following are four types of logical SIP entities:

User Agent (UA) — the endpoint component, which can be represented by a hardware or software device implementing SIP (e.g., an IP phone), and consists of two main components [3]:

- **User Agent Client (UAC)** — a client application that initiates the SIP request.
- **User Agent Server (UAS)** — a server application that contacts the user when a SIP request is received, then returns a response on behalf of the user. The response accepts, rejects or redirects the request [10].

SIP Proxy Server — a proxy server forwards the SIP messages to multiple proxy servers, creating a search tree, in order for the SIP messages to reach their destination. There are two different operating modes for these servers: stateless (the server forgets all the information once the request is sent) and stateful (the server save previous routing information and is able to use it for improving the message transfer) [3].

SIP Redirect Server — a redirect server is a server that accepts a SIP request, maps the address into zero or more new addresses and returns these addresses to the client [10].

SIP Location Server — a location server is used by a SIP redirect or proxy server to get a user’s location information [10].

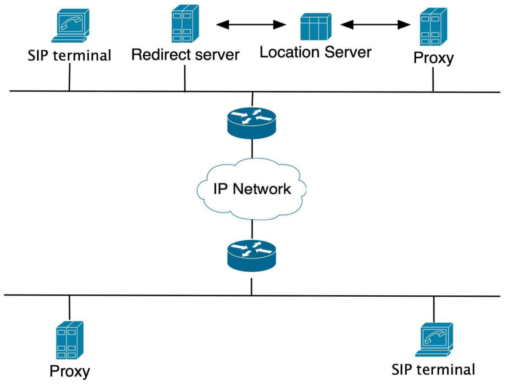


Figure 2. Session Initiation Protocol Architecture

3.2. H.323 Protocol

H.323 is the first international multimedia communications protocol standard. It was published by the ITU Telecommunication Standardization Sector. H.323 allows the convergence of voice, video, and data on packet networks. It features World Wide Web and Internet integration, together with PSTN interfacing. Furthermore, it provides diverse applications such as wholesale transit of voice, prepaid calling card services, enterprise voice and video services. Remote users can perform a video call and simultaneously edit a document in real time over the Internet. H.323 goes beyond, allowing phone or phone services customization, user location, call transfer, or other tasks taking advantage of the HTTP interface between the client/server on the network [12].

A H.323 system comprises of the following entities: Terminals, Gatekeepers, Gateways, Multipoint Controllers, Multipoint Processors and Multipoint Control Units.

Terminals — provide the audio/video/data communications capability in point-to-point or multipoint conferences, as well as handling the H.323 signaling issues on behalf of the user [9].

Gatekeepers — provide admission control and address translation services [9].

Gateways — are needed to provide interworking with terminals using other signaling protocols, such as PSTN terminals, ISDN terminals, SIP terminals, etc [9].

Multipoint Controllers, Multipoint Processors and Multipoint Control Units — provide support for multipoint conferences [9].

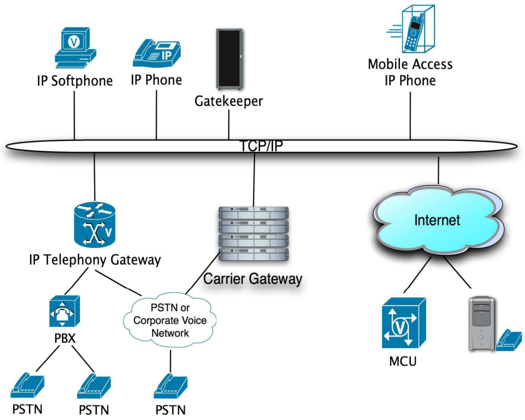


Figure 3. H.323 Architecture

3.3. IAX2 Protocol

Asterisk (the open source PBX server) is rapidly gaining in popularity as a powerful alternative to expensive PBX systems. The IAX2 (Inter Asterisk Exchange version 2) protocol is the native language of Asterisk [13]. The IAX2 protocol is used to send audio and control information from one physical site to another [11]. The main strength of IAX2 when compared to competing protocols such as RTP/SIP/H.323 is its friendliness to NAT (Network Address Translation) and firewalls. IAX2 uses only a single UDP port 4569 to carry both media and control messages. The IAX2 protocol is usually used for server-to-server communication. The IAX2 Call Analyzer can extract all IAX call details, analyze the bandwidth usage, analyze the interarrival delay, analyze the jitter experienced, and plot all IAX2 events and the call start time [13].

3.4. Comparison of H.323, SIP, and IAX2

H.323 is inherently complex, has overheads and is thus inefficient for VoIP and H.323 lacks the extensibility required of the signaling protocol for VoIP. As SIP has been designed by keeping the Internet in mind, it avoids both the complexity and extensibility pitfalls. H.323 is still limited when performing loop detection in complex multi-domain searches. It can be done by storing messages but this technique is not very scalable. On the other hand, SIP uses a loop detection method by checking the history of the message in the header fields, which can be done in a stateless manner [7]. The IAX2 protocol provides control and transmission of streaming media over IP networks. It is used by the Asterisk VoIP PBX as an alternative to SIP and H.323 to connect to other devices that support IAX2 [12]. IAX2 is primarily used for passing calls between Asterisk servers [5].

There are reasons for using SIP and IAX2 protocols. Full implementation of H.323 requires a lot of overhead. SIP is a much more streamlined protocol, developed specifically for IP telephony. Smaller and more efficient than H.323, SIP takes advantage of existing protocols to handle certain parts of the process. IAX2 is an extremely efficient protocol, the fact the SIP is now the dominant protocol of IP-based telecommunication and is supported by virtually everybody should give it more visibility in Asterisk [8].

4. Implementation

The implementation of the system design should set up the Asterisk server where there are four main things to be done:

1. The operating system that compatible with Asterisk (i.e. Linux) in this paper.
2. Download and install all the packages needed as platform for the Asterisk to run.
3. Install the Asterisk and configure it.
4. Install softphone in clients and try to connect to the Asterisk server.

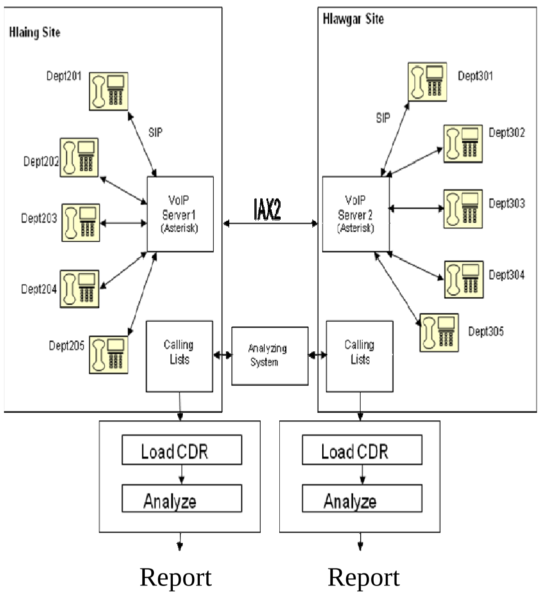


Figure 4. System Design

4.1. System Design

Telecommunication is essential to communicate among the people all over the world. Therefore, there are thousands of ways for communication system. Among these communication ways, VoIP is widely employed by carriers.

In this paper the system is designed to use the VoIP servers using Asterisk open source. It is hoped that the users can use the VoIP servers to connect the different sites. This design includes two Asterisk server and four clients with SJphone (softphone) connected to the server. In this system, the extension phones in Hlaing can be connected the VoIP server 1 and the extension phones in Hlawgar can be connected the VoIP server 2 using session initiation protocol (SIP). Moreover, the VoIP server 1 in Hlaing can be connected the VoIP server 2 in Hlawgar via IAX2. In addition, this system shows the calling lists from each site and analyzes calling lists between different sites.

4.2. Steps to Implement System Design

The following steps should be done before implementation system design.

Linux Operating System has been chosen in this paper as the main platform for the Asterisk server. The Asterisk open source software has been downloaded and installed in the server. Then, two clients have been chosen as the clients to the Asterisk server. For the clients SJphone (softphone) is used because this software supports the Session Initiation Protocol (SIP).

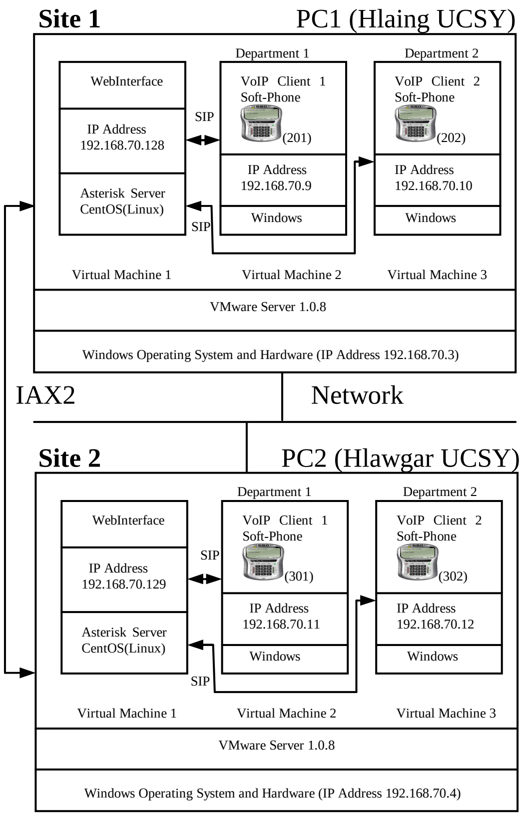


Figure 5. System Architecture and Implementation

Using a softphone application on a PC that is IP-connected to the asterisk server, users may make or receive VoIP phone calls to any phone in the world free of charge. The configurations for the clients are made in the server itself such as the clients Id, and password. This will ensure that the server can be used by a specific client that has been registered in the server. The configuration will also choose what type of protocol that can be used to establish connection between the clients and the Asterisk server. The connection between clients and the server are satisfied using the SIP protocol.

In this system, the extension phone in Hlawgar Campus can be called the other extension phone in Hlawgar Campus by using Session Initiation Protocol (SIP). Otherwise, the extension phone in Hlaing Campus can be called the extension phone in Hlawgar Campus by using IAX2 Protocol. If the user from Hlawgar Campus who uses extension "301" wants to call Hlaing Campus extension "201", then he/she has to call "11201". In this system calling processes over the extension phone "201" in site 1. In this way, the users can call the desired extension phone in same site or other site using IAX2 Protocol over internet without using landline phone line, see Figure 5.

5. Results

This paper provides all the results obtained according to implementation which is "Voice over IP" (VoIP). The open source software (Asterisk) is used because it is more secure due to this VoIP system uses server configuration itself. To transmit data from client to the server or other client, there are many types of protocol that can be used such as SIP, H.323, and IAX2. This paper used the SIP protocol and IAX2 protocol. The server will reject the connection with clients when the clients use different protocol. This design includes two Asterisks server and four clients installed with SJphone (softphone) connected to the server. The analyzer allows user to analyze the call lists of each Asterisk server and produce the summarize reports.

Date : 2010-08-10

Source	Destination	Total Calls	Total Duration
202	201	2	20
201	12302	7	31
201	202	4	11
201	12301	5	17
301	11202	3	19
202	12301	3	30
301	11201	8	42
302	301	1	14
301	302	2	6
Total		35	604

Figure 6. Analysis Table

At the beginning of the analysis system the user first needs to select the Call Detail Reports (CDRs) to analyze. All called records are saved in CDRs file. User can explore the details of CDRs. Each record has eight fields: Number, Call Date, Channel, Source, Caller id, Destination, Disposition, and Duration, see Figure 8.

This system allows the user to view analysis reports in four ways. User can analyze the Call Detail Reports by Source, Destination, Date, and Date/Time. The users will choose source, destination, date/time, or date they want to analyze, the system will show all sources and destinations which are called by source. Moreover, the total number of calls and total duration of calls are also shown. Using this system the number of calls can be estimated. User can know the status of calls for the selected date easily. It also lets us know how many outgoing and incoming calls occur during a period, see Figure 6.

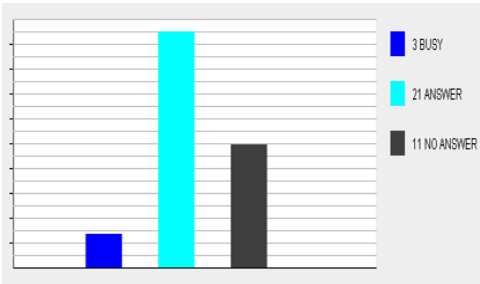
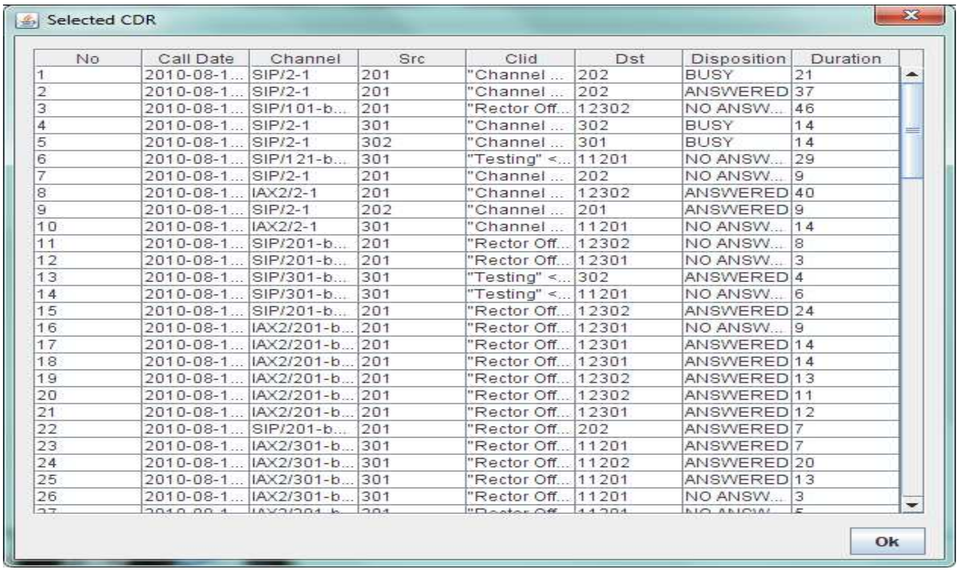


Figure 7. Analysis Graph

And the system will report the user summarized by Busy, Answer, and No Answer. This system tells us how many times it shows BUSY, ANSWER, or NO ANSWER signal. It shows which section denotes maximum calls, see Figure 7.

6. Conclusion

From the implementation of the proposed system, the results of the performance can be seen. The Voice over Internet Protocol can be cost less than the normal telephone line. The open source VoIP server (Asterisk) also has some advantages such as it is controlled by the server itself. The Asterisk server ensures the security as it will establish connection for the clients that have been assigned in the server. The Asterisk server can be used for many types of protocol such as Session Initiation Protocol (SIP), H.323, and also IAX2, where in this paper only focused in SIP and IAX2 protocol. The use of VoIP as telephony can save costs. Since voice and data traffic can be integrated, the necessary infrastructure to provide both services is reduced. This system displays the number of users' calling, users' call time and traffic time between distributed sites. By using this system, the users not only analyze the transaction between distributed sites but also know the information from one place to another punctually.



Selected CDR

No	Call Date	Channel	Src	Clid	Dst	Disposition	Duration
1	2010-08-1...	SIP/2-1	201	"Channel ...	202	BUSY	21
2	2010-08-1...	SIP/2-1	201	"Channel ...	202	ANSWERED	37
3	2010-08-1...	SIP/101-b...	201	"Rector Off...	12302	NO ANSW...	46
4	2010-08-1...	SIP/2-1	301	"Channel ...	302	BUSY	14
5	2010-08-1...	SIP/2-1	302	"Channel ...	301	BUSY	14
6	2010-08-1...	SIP/121-b...	301	"Testing" <...	11201	NO ANSW...	29
7	2010-08-1...	SIP/2-1	201	"Channel ...	202	NO ANSW...	9
8	2010-08-1...	IAX2/2-1	201	"Channel ...	12302	ANSWERED	40
9	2010-08-1...	SIP/2-1	202	"Channel ...	201	ANSWERED	9
10	2010-08-1...	IAX2/2-1	301	"Channel ...	11201	NO ANSW...	14
11	2010-08-1...	SIP/201-b...	201	"Rector Off...	12302	NO ANSW...	8
12	2010-08-1...	SIP/201-b...	201	"Rector Off...	12301	NO ANSW...	3
13	2010-08-1...	SIP/301-b...	301	"Testing" <...	302	ANSWERED	4
14	2010-08-1...	SIP/301-b...	301	"Testing" <...	11201	NO ANSW...	6
15	2010-08-1...	SIP/201-b...	201	"Rector Off...	12302	ANSWERED	24
16	2010-08-1...	IAX2/201-b...	201	"Rector Off...	12301	NO ANSW...	9
17	2010-08-1...	IAX2/201-b...	201	"Rector Off...	12301	ANSWERED	14
18	2010-08-1...	IAX2/201-b...	201	"Rector Off...	12301	ANSWERED	14
19	2010-08-1...	IAX2/201-b...	201	"Rector Off...	12302	ANSWERED	13
20	2010-08-1...	IAX2/201-b...	201	"Rector Off...	12302	ANSWERED	11
21	2010-08-1...	IAX2/201-b...	201	"Rector Off...	12301	ANSWERED	12
22	2010-08-1...	SIP/201-b...	201	"Rector Off...	202	ANSWERED	7
23	2010-08-1...	IAX2/301-b...	301	"Rector Off...	11201	ANSWERED	7
24	2010-08-1...	IAX2/301-b...	301	"Rector Off...	11202	ANSWERED	20
25	2010-08-1...	IAX2/301-b...	301	"Rector Off...	11201	ANSWERED	13
26	2010-08-1...	IAX2/301-b...	301	"Rector Off...	11201	NO ANSW...	3

Ok

Figure 8. CDRs (Call Detail Reports)

7. References

[1] Barrie Dempster, and Kerry Garrison, "TrixBox Made Easy", Packt Publishing Ltd., Birmingham, B27 6PA, UK, ISBN 1-904811-93-0, September 2006.

[2] Fredrik Thernelius, "SIP, NAT, and Firewalls", Department of Teleinformatics, May 2000.

[3] IXIA, "Session Initiation Protocol (SIP) Technology", 26601 W. Agoura Rd, Calabasas, CA 91302, USA, Copyright © Ixia, 2004.

[4] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "SIP: Session Initiation Protocol", IETF Internet Draft <draft-ietf-sip-rfc2543bis-09.ps>, February 27, 2002.

[5] Mark Spencer, Christopher Rhodes and Mack Allison (The Asterisk Documentation Team), "The Asterisk Handbook", March 2003.

[6] MATTHEW DESANTIS, US-CERT, "Understanding Voice over Internet Protocol (VoIP)", Produced 2006, updated 2008.

[7] Miguel Rodríguez Álvarez and Jaime Gómez Gil, "Distributed Architecture for Voice Integration in Data Networks", University of Valladolid, SPAIN, December 2006.

[8] Nasser K. Manesh, "Asterisk: A Non-Technical Overview", Millenigence, Inc., June 2004, Revised December 2004.

[9] Pavlos Papageorgiou, "A Comparison of H.323 vs SIP", University of Maryland at College Park, June 4, 2001.

[10] Sangheon Pack, Eun Kyoung Paik, and Yanghee Choi, "Design of SIP Transformation Server for Efficient Media Negotiation", Multimedia & Communication Laboratory, Seoul National University, Korea, May 2001.

[11] Steve Rodgers, "The Asterisk PBX as a Linked Repeater Controller", October 2005.

[12] Vasco N. G. J. Soares1, 3, Paulo A. C. S. Neves1, 3, and Joel J. P. C. Rodrigues2, 3. "Past, Present and Future of IP Telephony", 1 Superior School of Technology, Polytechnic Institute of Castelo Branco, C. Branco, Portugal, 2Department of Informatics, University of Beira Interior, Covilhã, Portugal, 3Institute of Telecommunications, Networks and Multimedia Group, Portugal, 978-0-7695-3190-8/08 $25.00 © 2008 IEEE, June 2008.

[13] Vivek Rajagopalan (Unleash Networks), "IAX2 Call Analyzer for Unsniff", Unleash Networks Pvt Ltd., February 2006.