Secure data and voice over wireless networks in disaster and emergency response

TUNG VU HOANG

KTH Information and
Communication Technology

Master of Science Thesis
Stockholm, Sweden 2005

IMIT/LCN 2005-12

Secure data and voice over wireless networks in disaster and emergency response

Vu Hoang Tung

May 16th, 2005

Masters of Science thesis performed at Ericsson Response,

Stockholm, Sweden

Examiner: Professor Gerald Q. Maguire Jr.

Academic Supervisor: Professor Gerald Q. Maguire Jr.

Industry Supervisor: Sarah Gannon, Ericsson Response

School of Information and Communication Technology (ICT)

Royal Institute of Technology (KTH), Stockholm, Sweden

Abstract

Communication is often limited in a disaster area and other emergency situations where no infrastructure exists or existing infrastructure has been destroyed. This makes its difficult for relief workers in the field to communicate with one another and with their home head office. Ericsson Response has developed a Wireless LAN in Disaster and Emergency Response (WIDER) solution. WIDER is based on broadband Wireless LAN internetworking to satellite and GSM networks. The WIDER solution has identified ways for organizations to share their communication infrastructure, and information in a secure and cost effective manner during an emergency response operation. Data over WIDER needs to be secured to prevent from unauthorized access to sensitive information of relief organizations. VoIP calls should be protected against eavesdropping. The thesis investigated how to enhance security solution in WIDER and implement a secure VoIP client. Measurements of the performance of WIDER and the total delay of VoIP over satellite were used to estimate the capability of WIDER before deployment in the field.

Keywords: **Disaster Response, WIDER, SIP, SRTP, MIKEY, VoWLAN, Satellite**

Abstract in Swedish

Kommunikation är ofta begränsad i katastrofområden och andra nödsituationer där infrastruktur saknas eller har blivit förstörd. Det gör det svårt för fältarbetande personal att kommunicera, både med varandra och centraliserade kontor. Ericsson Response har utvecklat en lösning kallad "Wireless LAN in Disaster and Emergency Response" (WIDER). WIDER använder trådlöst LAN och är en bredbandsbaserad internetteknik mot satelit- och GSM-nätverk. WIDER har identifierat lösningar för organisationer att dela deras infrastruktur för kommunikation och information på ett säkert och kostnatseffektivt sätt vid nödsituationer. Informationen som skickas via WIDER behöver bli skyddad för att förhindra oaktorisierad tillgång till känslig information. VoIP förhindrar obehöriga att avlyssna trafiken. Examensarbetet har undersökt den utökade säkerhetslösningen för WIDER och har implementerat en säker VoIP-klient. Mätningar av prestanda hos WIDER och den fördröjning som sker med VoIP över satelitlänk användes för att estimera WIDERs kapacitet innan systemet används i fält.

Nyckelord: **Disaster Response, WIDER, SIP, SRTP, MIKEY, VoWLAN, Satellite**

Abstract in Vietnamese

Các hệ thống thông tin thường bị giới hạn ở các nơi xảy ra thảm họa khi mà cơ sở hạ tầng bị phá hủy hoặc không tồn tại trước đấy. Điều này tạo ra khó khăn cho các tổ chức cứu trợ trong việc liên lạc nội vùng thảm họa hoặc liên lạc với trụ sở chính. Ericsson Response đã phát triển giải pháp không giây trong các trường hợp xảy ra thảm họa gọi là Wireless LAN in Disaster and Emergency Response (WIDER). WIDER được xây dựng dựa trên mạng LAN không dây băng thông rộng kết nối với mạng GSM và vệ tinh. Giải pháp WIDER giúp các tổ chức chia sẻ hạ tầng thông tin, và tin tức một cách an toàn và hiêu quả trong hoạt động cứu trợ thảm họa. Thông tin truyền trên WIDER cần đảm bảo bảo mật chống truy nhập bất hợp pháp tới thông tin quan trọng của các tổ chức cứu trợ. Các cuộc gọi VoIP cũng cần phải đảm bảo không bị nghe lén. Luận văn này nghiên cứu làm thế nào để tăng cường an ninh cho giải pháp WIDER và phát triển phần mềm bảo mật cho VoIP. Đo đạc hiêu suất của WIDER và tổng thời gian trễ của VoIP qua vệ tinh được sử dụng để ước tính khả năng đáp ứng của WIDER trước khi ứng dụng trên thực tế.

Từ khóa: **Disaster Response, WIDER, SIP, SRTP, MIKEY, VoWLAN, Satellite**

Acknowledgements

This report is a result of a Master's thesis project at Ericsson Response. I would like to express my sincere gratitude to people at Ericsson:

- Sarah Gannon, for her continuous support and help during the project.
- Rene Francis, for his helpful advising in project management and soft skills.
- Dag Nielsen, for his supporting and sponsoring the project.
- Jonas Sjöberg and Fridrik Lindholm, for their technical guidance and programming advices.

The thesis project would not been successfully without helping of my supervisor and examiner, Professor Gerald Q. Maguire Jr. I would like to thank for all his supporting, his quick feedback and his inestimable comments.

During the project, my parents and my girlfriend encourage me; give me the confidence and lovely moment. I would like to thank for their deep love and support, for their patience listening and giving encouragements and advices.

Finally, I would like to thank my friends, Carlos Loarca, Ha Nguyen, Quan Duong who helped me in various testing scenarios as well as in technical discussion.

Contents

List of Figures

List of Tables

Acronyms

AES	Advanced Encryption Standard
AH	Authentication Header
ALG	Application Level Gateway
AMR	Adaptive Multi-Rate
DHCP	Dynamic Host Configuration Protocol
DoS	Denial of Service
DTLS	Datagram Transport Layer Security
ESP	Encapsulating Security Payload
GDOI	Group Domain Of Interpretations
GEO	Geostationary Earth Orbit
ICE	Interactive Connectivity Establishment
IKE	Internet Key Exchange
ISAKMP	Internet Security Association and Key management Protocol
IV	Initialisation Vector
MAC	Message Authentication Code
MAC	Media Access Control
MIDCOM	Middlebox Communication
MIKEY	Multimedia Internet Keying
MKI	Master Key Identifier
MOS	Mean Opinion Score
NAT	Network Address Translation
PEAP	Protected Extensible Authentication Protocol
PoC	Push to Talk over Cellular
PSTN	Publish Switched Telephone Network
QoS	Quality of Service
RTCP	Real-time Control Protocol
SA	Security Associations
SBC	Session Border Controller
SDP	Session Description Protocol
SigComp	Signal Compression
SIP	Session Initiation Protocol
SRTP	Secure Real-time Protocol
SSRC	Synchronization Source Identifier
STUN	Simple Traversal of UDP Through NAT
TLS	Transaction Layer Security
UA	User Agent
UDVM	Universal Decompressor Virtual Machine
UPnP	Universal Plug and Play
VLAN	Virtual Local Area Network
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
WIDER	Wireless LAN in Disaster and Emergency Response
WLAN	Wireless Local Area Network

This page intentionally left blank

1 Introduction

This thesis investigated and developed a secure VoIP solution for a Wireless Local Area Network (WLAN) in Disaster and Emergency Response (WIDER). First we describe the WIDER project and the WIDER infrastructure. Then we show the challenges it presents and propose a solution. Later chapters will examine this solution in detail.

1.1 WIDER solution and its infrastructure

Today, natural and man-made disasters have increased both in their severity and scope. Responders, both from local governments and international organizations, need to share information to facilitate rapid recovery. Practice has shown that communications are often limited because the previous communication infrastructure was destroyed or is in useless due to congestion. As a temporary solution, many relief organizations rely on their own communication systems at a disaster site. This is both expensive and inefficient, the later doubly so because most recovery operations require local coordination between relief organizations locally. Ericsson Response's WIDER addresses this issue by creating an efficient, reliable, and highly available shared infrastructure for relief organizations working at disaster sites. Figure 1 illustrates the WIDER network architecture for disaster and emergency communications.

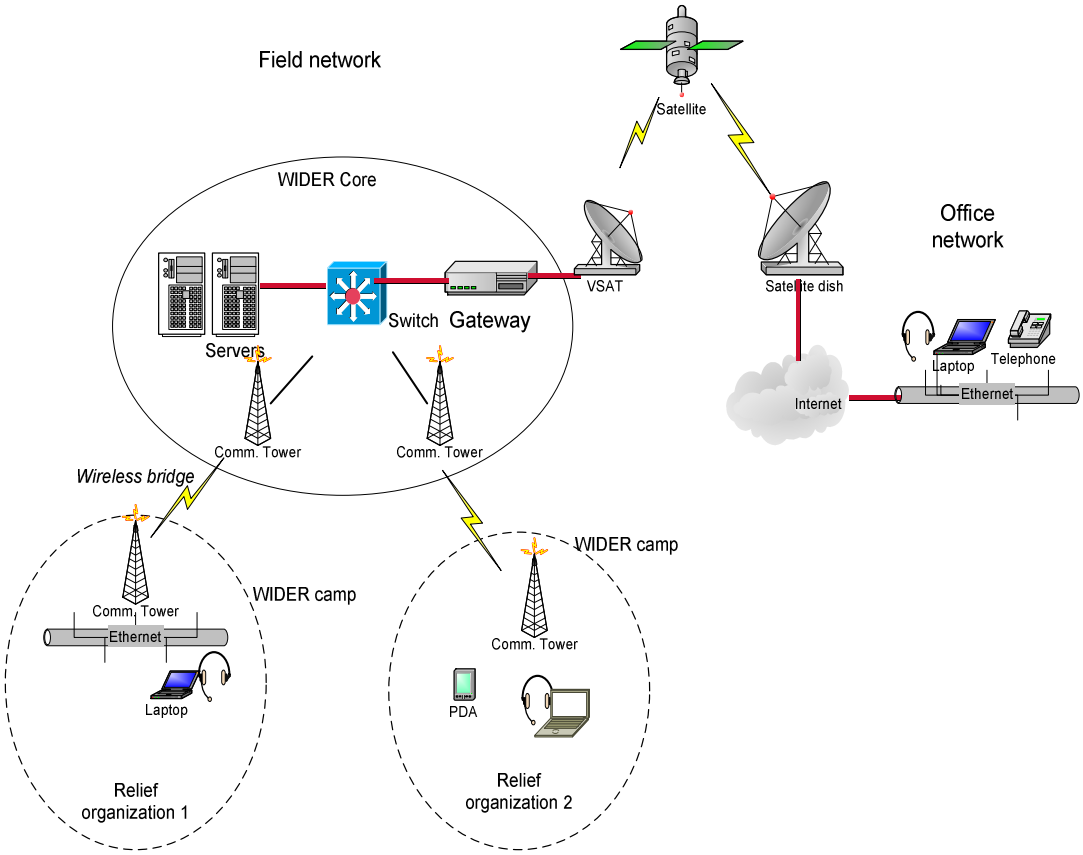


Figure 1. WIDER architecture for disaster and emergency communications

.

A WIDER pilot has been run in Geneva between the UN World Food Program (WFP), International Red Cross/Red Crescent (IRFC), and the UN High Commissioner for Refugees (UNHCR) since July 2004.

The tsunami on December 26, 2004 is representative of most natural disasters that happens in the world. At least 295,000 people died in the disaster, with 1 500 000 displaced and over 500 000 homeless (source: IFRC[1], February 2, 2005). The WIDER system has shipped to Indonesia. It was installed in the Humanitarian Information Centre that distributes the information internally and among 160 humanitarian agencies. WIDER services provided updated daily reports from the UN office for the Coordination for Humanitarian Affairs (OCHA) and the Red Cross/Red Crescent.

[1] www.ifrc.org

1.2 Problem statement

The WIDER infrastructure offers several advantages: ease of use, high availability, flexibility, mobility, and cost efficiency over multiple uncoordinated networks. However, security for voice and data had not been concerned. Authentication in WIDER system is based on certificates using Extensible Authentication Protocol with Transport Layer Security (EAP-TLS). Certificates are difficult to deliver and install in such a disaster area. While connected to Internet, WIDER needs a firewall to protect network itself rather than leaving this to the satellite operator. There is also no method for secure voice communications both within the local area and between the local area and remote locations, such as the organizations' homes. Today sniffer voice software, such as Network Associate's (NAI) SnifferVoice[1] can easily capture and play back whole VoIP sessions. To avoid these problems, we will propose and examine a solution that enhances secure data and voice communications in WIDER.

In the case of remote voice calls via satellite, delay is the most serious impairment, which must be overcome. The total delay is large because of the propagation delay of the satellite connection. The latency and performance of voice over satellite links needs to be examined so that the solution will be in practical. Thus we are to measure the performance of WIDER, in particular the delay of packetised voice over a satellite link.

1.3 Proposed solution

An improved security solution in WIDER should guarantee compatibility with the earlier WIDER solution. Two key features that need to be addressed are the authentication system and a firewall to protect WIDER's internal network. We propose using Tunnel TLS that supports user credentials (user/password) to reduce the complication of distributing certificates. Two solutions are Protected Extensible Authentication Protocol (PEAP) and Extensible Authentication Protocol using Tunneled Transport Layer Security (EAP-TTLS) that can be used parallel with certificates based on EAP-TLS. (Section 3.1.1).

WIDER provides both data and voice services. Hence, the firewall should have an ability to allow voice traffic in two-way communications by understanding VoIP

[1] www.sniffer.com

sessions. Choosing a firewall that has Application Level Gateway (ALG) (Section 3.6.4) will addresses this issue.

Our solution to securing VoIP was to develop a secure VoIP client that will be used in WIDER. We implemented VoIP client based on Session Initiation Protocol (SIP) [RFC3261]. We have investigated open source Windows-based SIP softphones since most relief workers use Windows Operating System for their work. Unfortunately, we could not find any Windows-based open source softphone.

SleIPner is a test client for Push to Talk over Cellular (PoC) that was developed by Node Test and Test tools design Unit, IP Multimedia Subsystem department, Ericsson AB[1]. This client supports signal compression (Sigcomp) [RFC3320] for SIP Signaling and uses Adaptive Multi-Rate (AMR) [3GPP TS 26.090] CODEC in media layer. The AMR CODEC is more efficient than other CODEC since it automatically operates at different bit rates depending on signaling conditions. The AMR CODEC is a mandatory part of the 3GPP standard [3GPP TS26.101]. Sigcomp greatly reduces the SIP signaling for a call setup, especially over satellite links. [J. Christoffersson] shows that a call setup delay between the terminal and the first SIP proxy is reduced by more than 66% and the system capacity increased approximately 17%. This client is thus highly suitable to deploy over the WIDER infrastructure to provide VoIP/PoC service.

Securing data between WIDER and a headoffice could be achieved by using IPsec [RFC2401] to create a tunnel from the gateway between the field and the network of the head office. For VoIP, we provide secure voice by implementing a VoIP client that has built-in security functions. This solution uses Secure RTP (SRTP) to secure the media layer. The signaling layer should utilize a TLS-enabled SIP server. Key management in secure VoIP client can be archived using MIKEY [RFC3830] or SDP Descriptions [draft-sdp-descriptions]. IPsec security is discussed in section 3.3.2, while the details of the secure voice client is presented in sections 4.2.

Measurement of the performance and QoS demonstrates the practicality of using VoIP over WLAN and satellite networks. This is discussed in detail in section 5.1 and 5.2.

[1] www.ericsson.com

2 Background and Related work

This chapter introduces the underlying concepts of a Wireless Local Area Network (WLAN) and Voice over WLAN (VoWLAN); with a focus on their use in Emergency and Disaster Response situations. Related work in voice over wireless networks is described in the next following section.

2.1 Wireless LAN

2.1.1 IEEE 802.11 Wireless LAN standard

The IEEE[1] 802.11 WLAN standards are a group of specifications developed by IEEE to provide Media Access Control (MAC) and Physical (PHY) layer functionality for wireless connectivity of fixed or portable terminals within a local area [IEEE 802.11]. Figure 2 illustrates the IEEE 802.11 protocol model in the context of IETF[2]'s TCP/IP stack

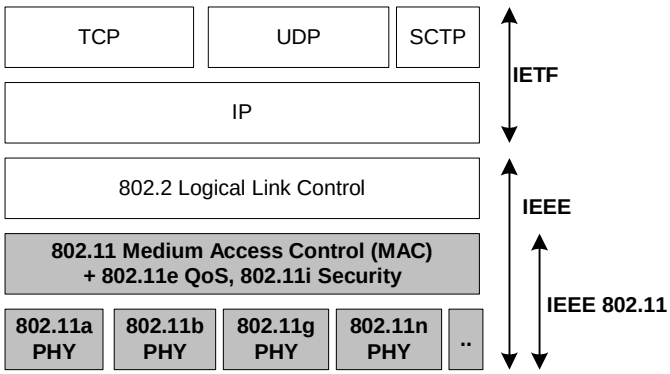


Figure 2. IEEE 802.11 protocol model underlying the TCP/IP stack

The 802.11 architectures can be divided into infrastructure and ad hoc architectures. In ad hoc mode, a mobile station works independently and communicates directly with others when in signal range. In infrastructure mode, each mobile station will connect to an Access Point, which acts as a Base Station that connects between

[1] IEEE: Institute of Electrical and Electronic Engineers
[2] IETF: Internet Engineering Task Force

mobile stations and another wired or wireless network. In comparison with infrastructure mode, ad hoc mode has some advantages: lower cost, rapid set up, and better performance. However, because of limitations in coverage of a single cell and difficulty of managing a multihop ad hoc network, practical solutions that use ad hoc mode are not deployed widely this time. Figure 3 illustrates these two 802.11 architectures.

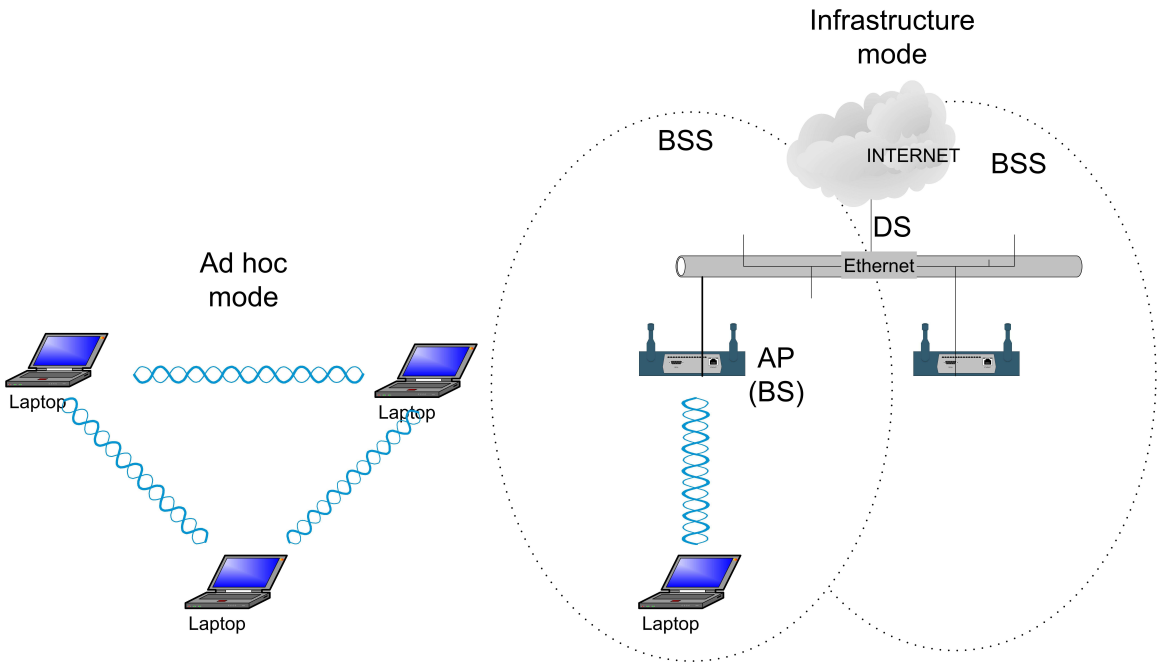


Figure 3. The two 802.11 architectures

Table 1 lists some of the 802.11 WLAN standards and proposed standards. We can see that some are concerned with specific media while others are applicable to multiple types of media.

802.11a	Operates in 5Ghz band, data rates up to 54 Mbps
802.11b	Operates in 2.4 Ghz band, data rates up to 11 Mpbs
802.11e	Enhances 802.11 MAC to improve QoS for real-time services
802.11f	Inter-Access Point Protocol; increases compatibility between Access Point devices from multiple vendors.
80.2.11g	Operates in 2.4 Ghz, data rate up to 54 Mbps, compatible with 802.11b devices.
802.11h	Enhances to provide network management and control extensions for spectrum and transmit power management in the 5 GHz band
802.11i	Enhances the security and authentication mechanisms
802.11n	Proposed standard, data rates up to 540 Mbps

Table 1. 802.11 WLAN proposal standard

While WLANs provide a wireless solution for the local area, IEEE has set up another standard track that provides Metropolitan Area Network (MAN) broadband connectivity, one of these is IEEE 802.16 WiMax standard[IEEE 802.16]. WiMax is an air interface for fixed broadband wireless access systems, and can transmit signals in a fixed direction up to 30kms. WiMax is an alternative to 802.11 to provide long distance wireless communication that is especially helpful in disaster and emergency recovery. For example, the wireless connectivity between the core network and relief organizations in disaster area could utilize WiMax (e.g. Wireless bridge between the WIDER core and WIDER camp as shown in Figure 1).

2.2 WLAN in Disaster Response

Communications systems are essential for relief workers in controlling and managing of disasters and emergency recovery. A rapidly deployed network is a mandatory requirement during an emergency response operation. Table 2 lists differences between traditional telecommunication networks and rapidly deployed networks for use in emergency and disaster settings [S.F. Midkiff].

Traditional networks	Rapidly-Deployed networks
• Wireline technology, i.e., optical fiber, coaxial cable, or DSL. • A lengthy planning process to increase the likelihood of a high quality network deployment. • Operation is robust due to use of wireline technologies and careful advance planning • Security may be available through limited physical access, encrypted links, or security gateways • System designs and deployments are usually highly sensitive to cost per user.	• Wireline technologies are unlikely to exist or function and cannot be quickly deployed. Hence, wireless technologies are typically the only viable option. • Planning must be "on-the-fly". There is little opportunity to do traditional site planning for these wireless systems. • Sub-optimal deployment and a frequently changing environment can reduce reliability and increase costs. • Wireless links increase the potential for eavesdropping. Key management is difficult in rapid deployment due to a lacking of knowing who will participate in a given operation. • While total system cost is still important, the cost per user is less important.

Table 2. Features of traditional and rapidly deployed networks

WLAN is an ideal solution for deploying a local IT[1] infrastructure at disaster or emergency site. WLAN based on 802.11 standards operates in license and license-free frequencies. In the case of 802.11, 802.11b and 802.11g, the 2.4Ghz band is available worldwide; making it easy to operate in different locations and in all countries. More over, WLAN can integrate different networks (such as, Ethernet, UMTS, and satellite) and many devices or sensors. However, license-free frequency bands are limited; therefore, the total bandwidth that can be used is limited and often not adequate for applications requiring large bandwidth such as videoconferences. For

[1] IT: Information Technology

example, Microsoft's Netmeeting[1] requires average 550 Kbps for a two-way communication with full-duplex audio and medium window/high-quality video[2]. Our measurement in later chapters has shown that even with data rate capacity 11Mbps, one access point can only handle limited number of VoIP calls. Currently, there are some other projects (e.g., MESA project [MESAproject]) investigating the use of licensed frequencies to provide broadband wireless access networks for public safety and disaster response. In the regulatory context, in Europe, there is no final decision to allocate bandwidth for broadband public-safety applications. While in North America, the FCC[3] allocated 50MHz of spectrum at 4900 – 4950 MHz for broadband services in support of public safety on February 14, 2002.

In the meantime, many companies and organizations are examining WLAN solutions for public safety and emergency response. Some WLAN solutions are based on infrastructure mode and some ad hoc mode. Table 3 is a short list of companies and solutions specifically relevant for disaster and emergency recovery scenarios.

Company	Architectures	Products and Services
Rajant (www.rajant.net)	Wearable WLAN; Ad hoc and infrastructure mode	Data, voice and video, network sensors.
Mesh networks (www.meshnetworks.com)	WLAN; Ad hoc mode	Built-in GIS, telemetry, voice and video monitor
Network Anatomy (www.networkanatomy.net)	WLAN Infrastructure	Voice, video and data, GIS
308systems (www.308systems.com)	Ad hoc mobile systems	GPS, cellular telephone and video camera.

Table 3. WLAN solutions for disaster and emergency recovery

[1] http://www.microsoft.com/windows/netmeeting/
[2] http://www.microsoft.com/windows/NetMeeting/Corp/reskit/Chapter7/default.asp
[3] FCC: Federal Communications Commission

2.3 Voice over IP and SIP

Voice over Internet Protocol (VoIP) is considered the next revolution in telecommunications and computer networks. In short, VoIP digitizes voice streams, then packetizes them for transmission over conventional IP networks. It not only reduces the costs of long distance calls, but also creates a convergence between data and telephony networks.

WLAN can provide limited mobility for VoIP users. Users using a WLAN equipped PDA[1] or WiFi[2] phone can walk around and make a call. In places such as a disaster area, where neither PSTN nor mobile networks are available, VoIP is an ideal solution to provide relief workers with voice communications. In United State, there is a project called "Voice Disaster Recovery" [Internet2VoIP] that implements the national system for VoIP over Internet2 purposely to replace the existing PSTN system in case of a regional or national crisis.

There are many standards for VoIP: SIP, H323, and MGCP. SIP [RFC3261] and MGCP [RFC2275] are developed within the IETF, while H323 [ITU-T H323] is a standard from ITU-T. H323 is a suite of many protocols for dealing with setting up media connections for real time services, interactive video conferencing, and audio applications. Media Gateway Control Protocol (MGCP) provides a control and signal standard for communication between gateways. SIP defines procedures for setting up, modifying and tearing down multimedia sessions. SIP is based on client-server protocols and follows the HTTP style of message exchange. Figure 4 shows a simplified VoIP call using SIP.

[1] PDA: Personal Digital Assistants
[2] WiFi: Wireless Fidelity, a set of compatibility standards for WLAN based on the IEEE 802.11

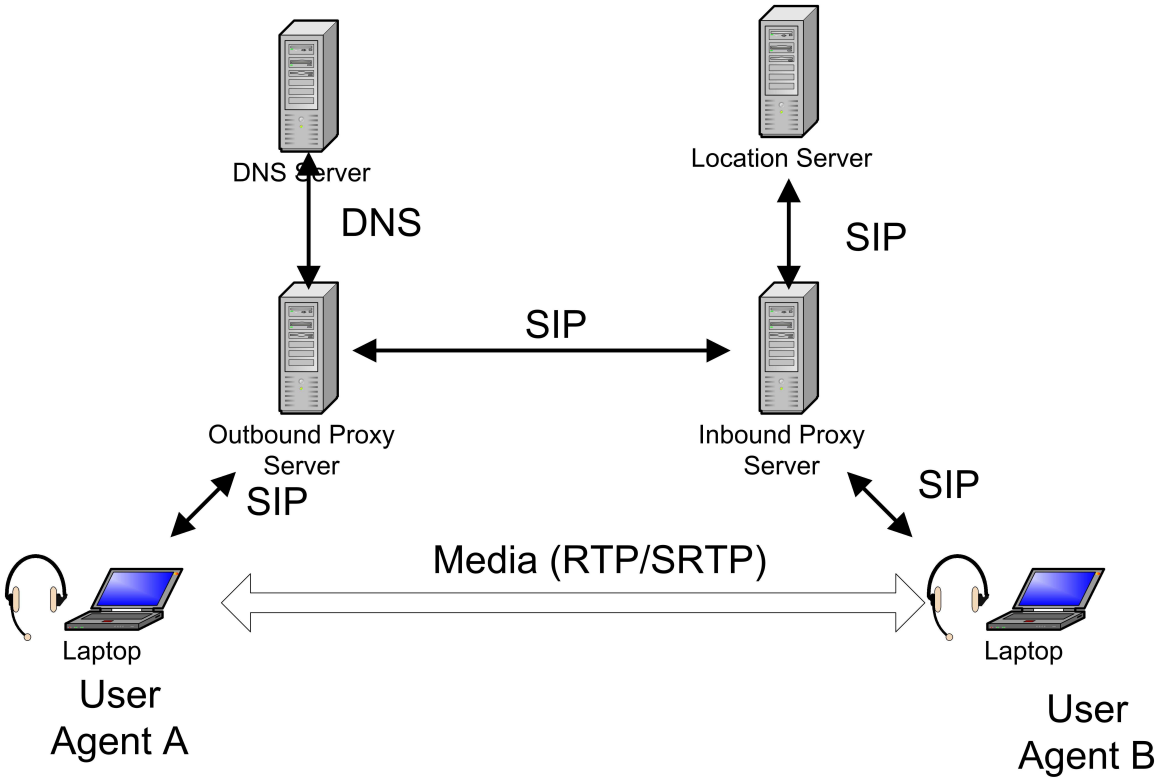


Figure 4. SIP trapezoid

SIP has proven that it will be the dominant VoIP in future since it is simple, scalable and easy-to-implement. 3GPP releases 5 and 6 choose SIP as the signaling protocol for setting up multimedia sessions. SIP is considered as a session protocol for the convergence of wired and wireless networks. Therefore, this thesis investigates VoIP using SIP. From now on, when we mention VoIP, we assume it is implemented by using SIP.

2.3.1 SIP architecture

The SIP logical entities include user agent (UA), proxy server, redirect server, back-to-back user agent, and registrar server. A user agent is a SIP client that can be hardphone or softphone; both must handle SIP signaling and (de)code data packets to voice and vice versa. The user agent that initiates a call is called a User Agent Client (UAC) while the user agent that answers the call is called a User Agent Server (UAS). A proxy server is a network host that relays requests and responses between a UAC and a UAS. The registrar server and redirect server are responsible for registration and redirecting UAs requests respectively. The registrar server, redirect server and proxy server can be integrated into one server. For example, Opensource SIP SER (SIP

Express Router)[1] has all SIP functionality in one serve while other SIP Vovida SIP[2] servers are separately.

SIP is text-based protocol with two kinds of messages: Requests and Responses. Figure 5 shows the structure of a SIP message that consists of a start line, several headers, and optional message body.

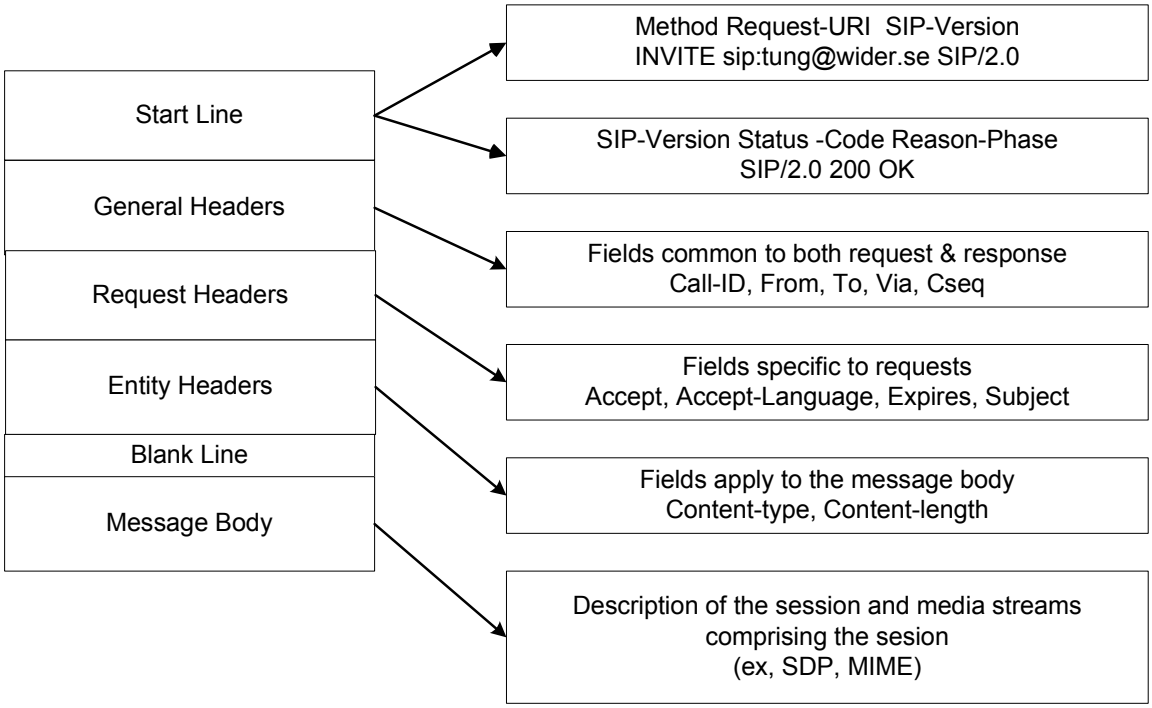


Figure 5. The structure of SIP messages

SIP request messages are INVITE, REGISTER, BYE, ACK, CANCEL, OPTIONS, REFER, SUBSCRIBE, MESSAGE, PRACK, UPDATE etc. An example of a SIP INVITE message is:

```
INVITE sip:tung@wider.se SIP/2.0
Via: SIP/2.0/UDP 147.214.160.103:5061;branch=z9hG4bK43.51f4.2
To: tung <sip:tung@wider.se>
From: tung <sip:tung@wider.se>;tag=22e6844-7260.2
Call-ID: df2-35e8-145d2@147.214.160.103
CSeq: 29281 INVITE
Max-Forwards: 70
User-Agent: PoC-client/OMA1.0 SleIPner/1.08
Contact: tung <sip:tung@147.214.160.103:5061>
Content-Length: 263
Content-Type: application/sdp
```

[1] www.iptel.org
[2] www.vovida.org

```
Supported: timer
Allow: INVITE, ACK, BYE, UPDATE
Proxy-Authorization: Digest username="tung",realm="wider.se",
nonce="586543a30df26ff25609145dh5322de", uri="sip:tung@wider.se",
response="f9121e721f2e4528d14531f0cda0a830",algorithm=MD5,
cnonce="145d5609",
opaque="95311d5adc3ec12efcdd6a7289aad271",qop=auth,nc=00000001


v=0
o=tung 88728872 0 IN IP4 147.214.160.103
s=SIP session
c=IP4 IN 147.214.160.103
t=0 0
m=audio 4904 RTP/AVP 109
a=rtpmap:109 AMR/8000/1
a=ptime:100
a=maxptime:400
a=fmtp:109 mode-set=1; octet-align=1
a=key-mgmt:default encryption and authentication
```

A SIP response message answers the request with a response code. Table 4 lists the SIP response codes with a short description.

Class	Description	Action
1xx	Informational	Indicates status of call prior to completion. If first informational or provisional response
2xx	Success	Request has succeeded. If for an INVITE, ACK should be sent; otherwise, stop retransmissions of request
3xx	Redirection	Server has returned possible locations. The client should retry request at another server
4xx	Client error	The request has failed due to an error by the client. The client may retry the request if reformulated according to response
5xx	Server failure	The request has failed due to an error by the server. The request may be retried at another server
6xx	Global failure	The request has failed. The request should not be tried again at this or other servers

Table 4. SIP response code

SIP is not limited to setting up VoIP calls; SIP can also be used to setting up multimedia conferences, instant messaging and presence service, along with text and general messaging services. In the following sections, we will investigate SIP features for registration, setting up and tearing down call, presence and instant messaging, and SIP conferencing.

2.3.2 SIP registration

SIP registration builds upon a location service and provides mobility. The location service maintains by the Registrar who acts as the front end bind UAs' location (normally an IP address) with a user URI based on the receipt REGISTER messages. The SIP Proxy consults the Registrar in order to route SIP messages between UAs. SIP provides both user mobility and device mobility. User mobility enables SIP devices that use the same identifier or SIP URI in different endpoints while remaining reachable by one or many of these devices at the same time. Device mobility means that user can have one identifier in many different locations, i.e. they do not care about their IP address, but are transparently reachable via a single application-layer identifier (the URI).

UAs send REGISTER message to add, remove, and query bindings. Figure 6 shows the REGISTER message

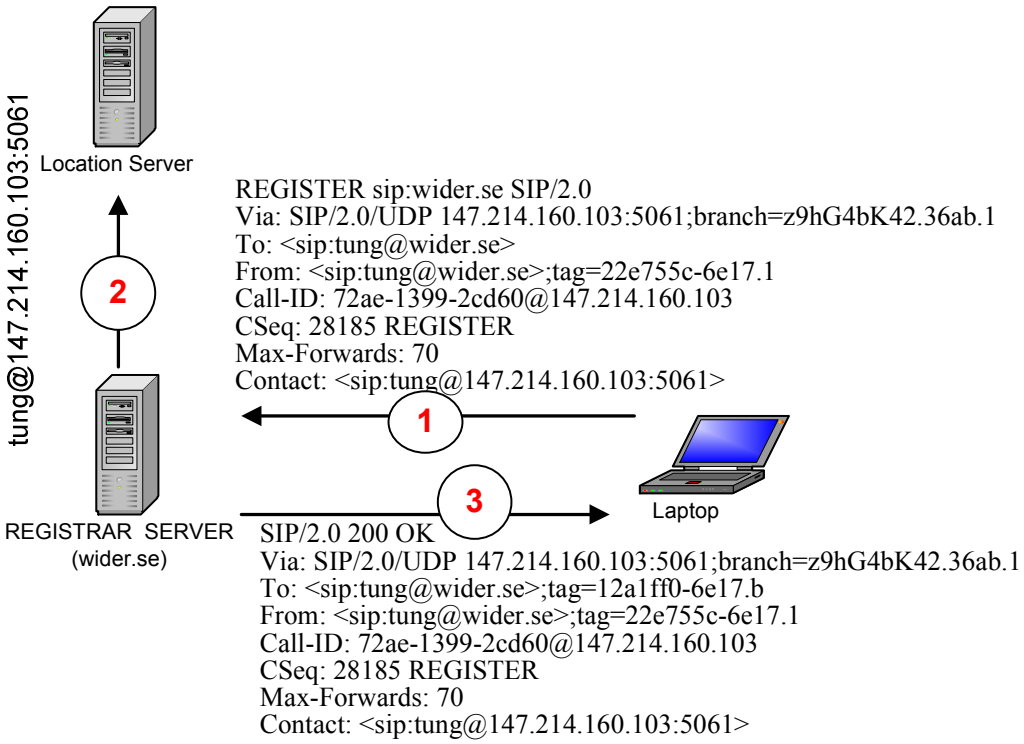


Figure 6. SIP register operations

2.3.3 SIP session establishment

SIP enables and automates the steps needed to set up a multimedia session by solving the rendezvous problem, i.e. routing a request to setup a session without requiring user to know the location of the targeted user. A SIP Session involves the exchange of media information using Session Description Protocol (SDP)[RFC2327]. A SIP-enabled VoIP session usually starts by a UAC/UAS sending/receiving an INVITE message and ends by sending/receiving a BYE message. The INVITE message includes the session description in a message body and even can send INVITE during the session (which called re-INVITE) to change the session state. Figure 7 shows a typical SIP session establishment between two end-points. An ACK used to confirm session establishment and can only be used with an INVITE. The BYE message terminates the session while a CANCEL message cancels a pending INVITE. SIP Response messages (180 Trying, 186 Ringing, and 200 OK) are generated by a UAS or a SIP server to reply to a request generated by a UAC.

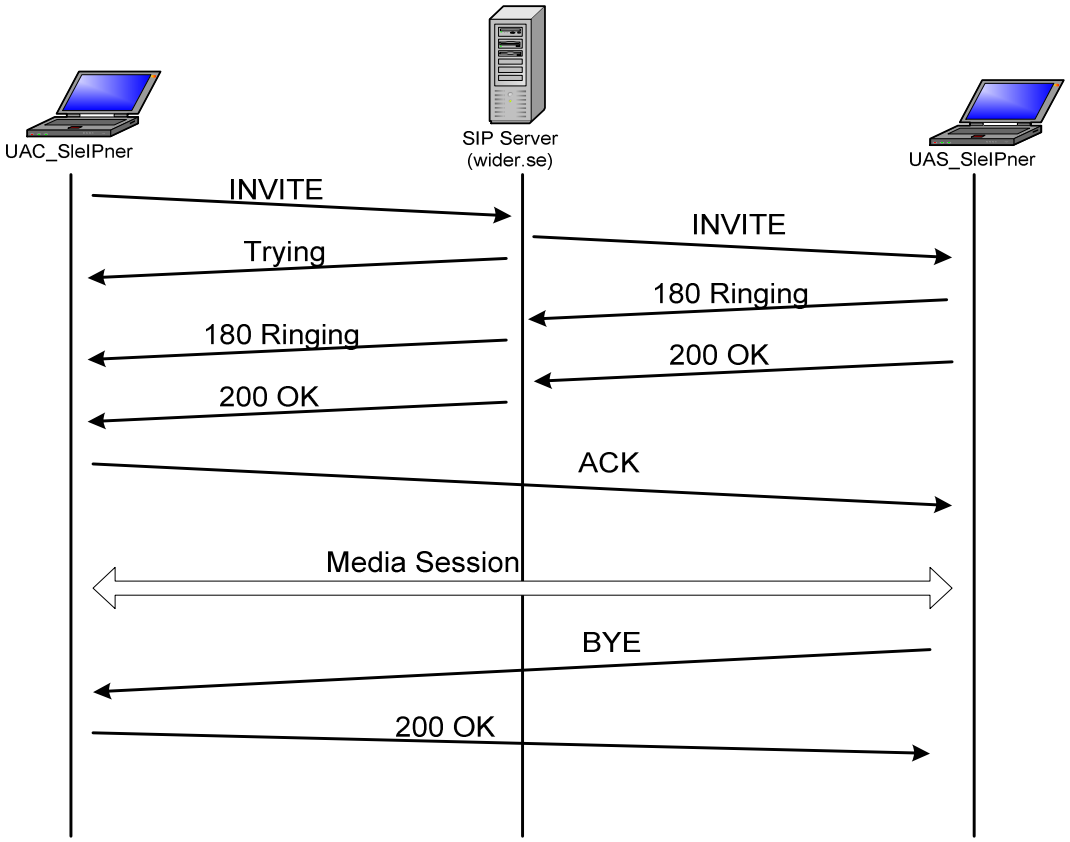


Figure 7. A SIP session

2.3.4 SIP Presence and Instant Messaging

SIP has been extended new methods to support Instant Messaging and Presence (which is called SIMPLE [RFC3856]). The IETF SIMPLE working group proposed these new methods in SIMPLE:

SUBSCRIBE method: This method is used to request status or presence updates from the presence server. The address (URI) of the user is included in the request

NOTIFY method: Once a subscription is authorized a NOTIFY method is generated. This method is used to deliver that information to the requestor or presence watcher

MESSAGE method Message methods uses to send instant messages. The message is stored in the body of MESSAGE. The request IM URI is used: im:user@network.com in compared with SIP request-uri: sip:user@network.com

SIMPLE entities include a Presence User Agent (PUA), Presence Agent (PA) and Watcher. The PUA manipulates the presence information for a presentity. Each presentity can have one or more PUAs. Each PA is a user agent that has unique SIP URI. The PA responsible for sending NOTIFY messages and receiving SUBSCRIBE requests uses a NOTIFY to send this information to the PUA. PA normally located at the Proxy/Registrar or along with a PUA at the presentity. The watcher is a subscriber that sends SUBSCRIBE messages and receives notifications as the state of the presentity changes. It eventually terminates its subscription when it is no longer interested in receiving notifications. Figure 8 shows the interaction between different SIMPLE components

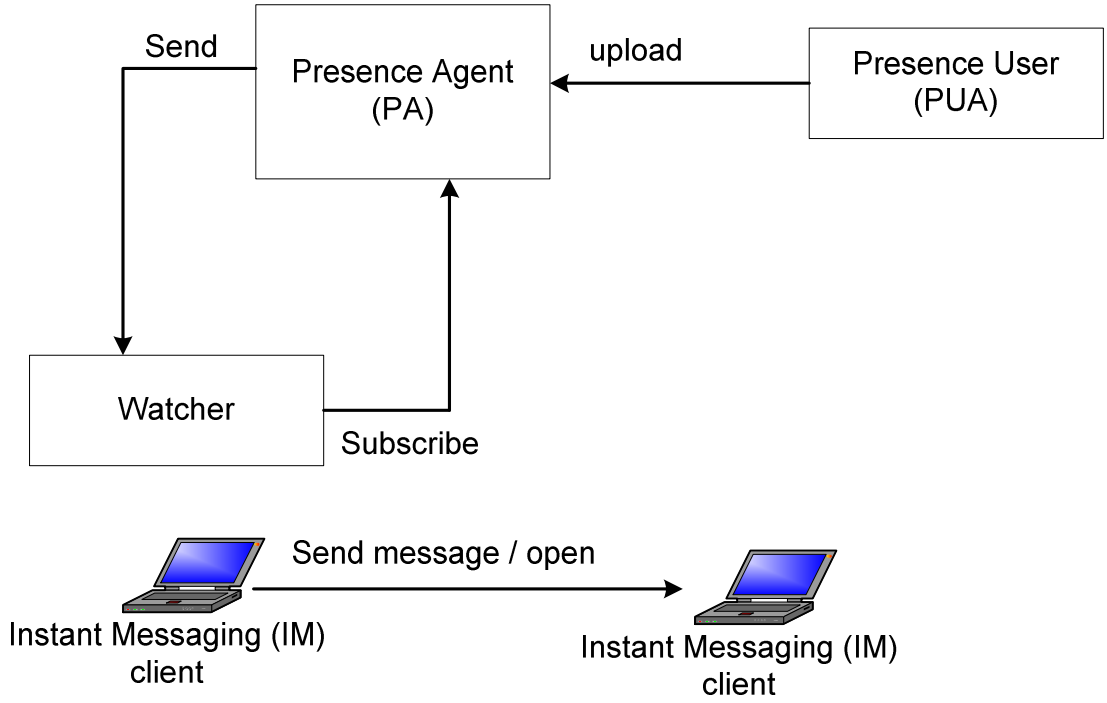


Figure 8. Interaction of SIMPLE components

2.3.5 SIP conference

A SIP-based conference can set up following three models: centralized conference, full-mesh conference, and end-point forwarding. In a centralized conference, a conference server (CS) establishes a point-to-point signaling connection with participants. If a UA initiates a conference, this conference is called dial-in mode; while if a CS initiates conference, it is called dial-out mode. In both modes, the CS is responsible for forwarding and mixing streams from/to UA participants. Nodes can not invite new members. This kind of conference is usually an open conference where each node joins the conference by sending a request to the conference server. An end-

point forwarding conference is similar to a centralized conference server, but each end-point acts as a conference servers. Each can do dial-out or dial-in mode; as well as support open or closed conferences. Usually the end-point initiating the conference has the rights to authorize and authenticate participants. In a full-mesh conference, all nodes are pair wise connected by calls, hence there is and no need for forwarding streams. Any member can invite new members but not all other members need accept them.

Several SIP new methods are created (depending on the conference models). For supporting a full-mesh conference, [J. Lennox] uses ten abstract messages: four initial messages, JOINT, CONNECT, LEAVE, and UPDATE, and the responses JOIN Ok, JOIN Ack, JOIN Reject, CONNECT Ok, CONNECT Ack, and CONNECT Reject. [Miladinovic] introduces a new SIP method CONF for optimizing signaling traffic in a centralized conference. Participants a conference have the following states: active, invited, or join. Miladinovic defines a new status value for participants, the "chair" that delivers information instead of conference server. However, these new SIP methods make interoperating difficult. An IETF design team has worked on standard SIP-based conferencing with basic SIP method support [draft-conf-framework]. In this design, initiation of a conference or adding participants to a conference by occurs by an INVITE or REFER; leaving a conference occurs using a BYE, and expelling a participant from a conference is done using a REFER(method="BYE"). Conference control provides state change notifications by SUBSCRIBE/NOTIFY, while conference and media policy control uses framework of SIMPLE context. Figure 9 shows call flow example when a user Tung joins a SIP-based conference.

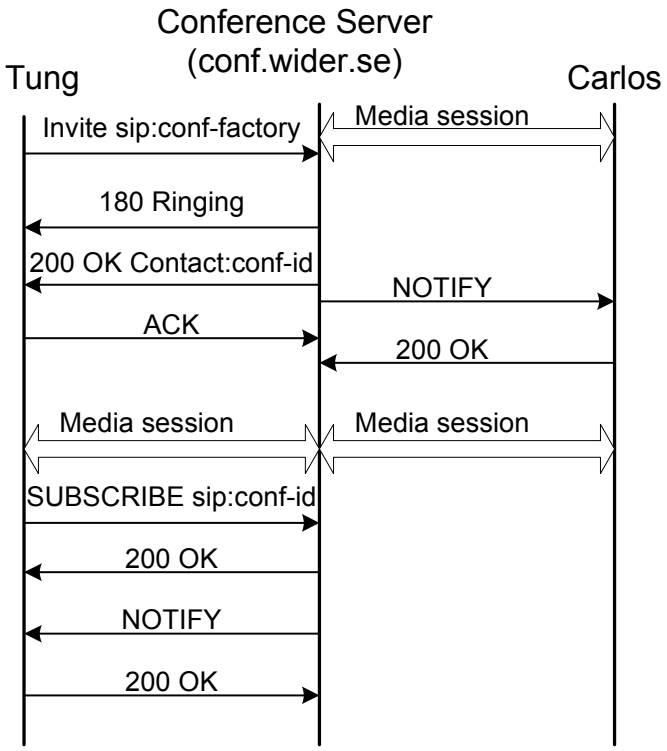


Figure 9. A call flow of a user joining a SIP conference

2.3.6 SIP location-based service

SIP can be customized to support location-based services. [R. Shacham] describes SIP location-based architectures that consist of four components: source for location information, messaging for user profile, configurable end-devices, and a device controller (DC). The source for location information includes both stationary and mobile source. Stationary location source are fixed hosts that identify users entering a particular location and publish this information. Mobile location sources are user devices that know their current location, for example, by GPS[1]. Then location state information is sent to a SIP server by SIP PUBLISH method [RFC3903] whose message body includes location objects with civil or geodesic information [draft-GEOPRIV]. User profile state can be updated using the SIP event framework (e.g. SIP NOTIFY and SIP SUBSCRIBE) [RFC3265]. A device controller (DC) automatically updates location information following configuration profile. The DC acts as a Service Location Protocol (SLP) User Agent that sends a Service Request to the Directory Agent to ask for devices whose locations are at a give place which it serves. Service Location Protocol (SLP) with location-based queries which includes common attributes of communication devices, such as vendor, supported media, and

[1] GPS: Global Positioning System

location parameters provides a mechanism that pushes service update events to subscribers [S. Berger]. Figure 10 describes the architecture of a location-based service mobility framework.

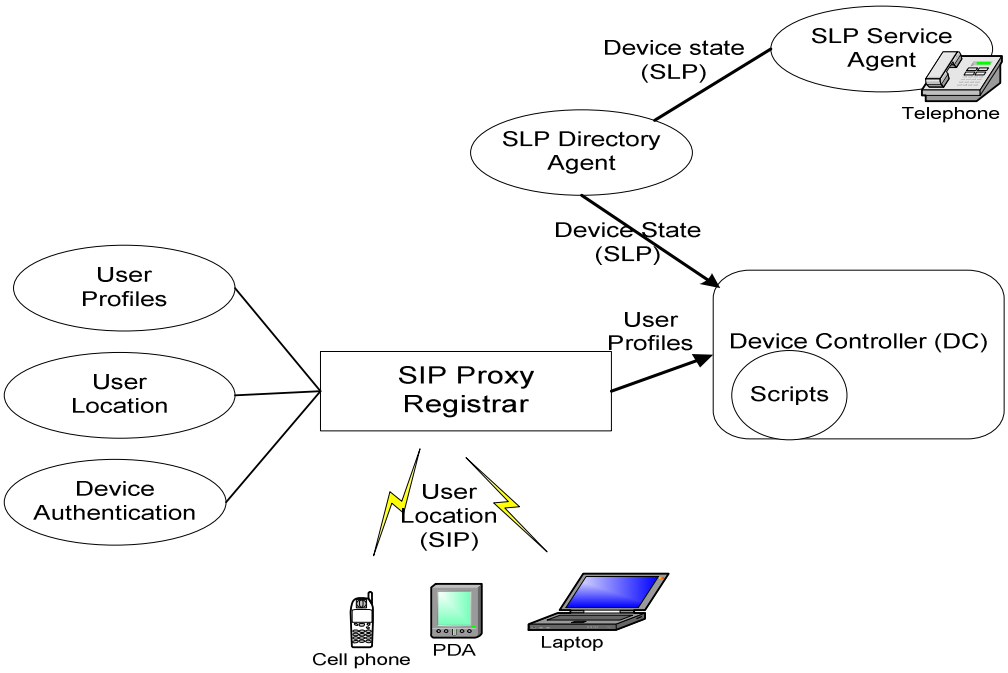


Figure 10. Architecture of a location-based service

2.3.7 Sigcomp

SIP is text-based protocol using the ISO 10646[1]-character set UTF-8[2] encoding [RFC3629]. This makes SIP easy to troubleshoot, fosters rapid development of applications, and interoperability between devices, applications, call controllers, and gateways. Embedded in SIP message body is the Session Description Protocol (SDP) [RFC2327] that is used for describing streaming media sessions, and session announcement, session initiation.

SDP is also text-based and has dynamic size. SIP messages containing SDP range from 200 to 1500 bytes. These protocols follow an offer-answer model [RFC3264]. Thus a sender sends a request and waits until a response is received, this will take a number of round trip times resulting in a delay depending on the calling environment. In a wireless environment, especially GSM/GPRS/UMTS or satellite, bandwidth is scarce and delay is high, optimized messages could reduce the delay for setting up

[1] ISO 10646: The standard for Universal Character Set encoding that map hundreds of thousands of abstract characters; each identified by an unambiguous name, to integers, called numeric code points.
[2] UTF-8: 8-bit Unicode Transformation Format that is a lossless, variable-length character encoding for Unicode

calls and make more efficient use of bandwidth resources. To address this issue, Signal Compression (Sigcomp) [RFC3320] was developed by the IETF. Sigcomp is a layer between the application layer and transport layer that compresses ASCII-based messages on the sender side and decompress on the receiver side. The core of the decompression of a Sigcomp message is the Universal Decompressor Virtual Machine (UDVM). The UDVM executes decompression when receiving messages by loading them into decompression memory together with decompressor code and a dictionary. These steps are necessary for the needs of Sigcomp in both low-end terminals (e.g. mobile phones) and powerful devices (e.g. SIP servers). Figure 11 illustrates Sigcomp decompression operation.

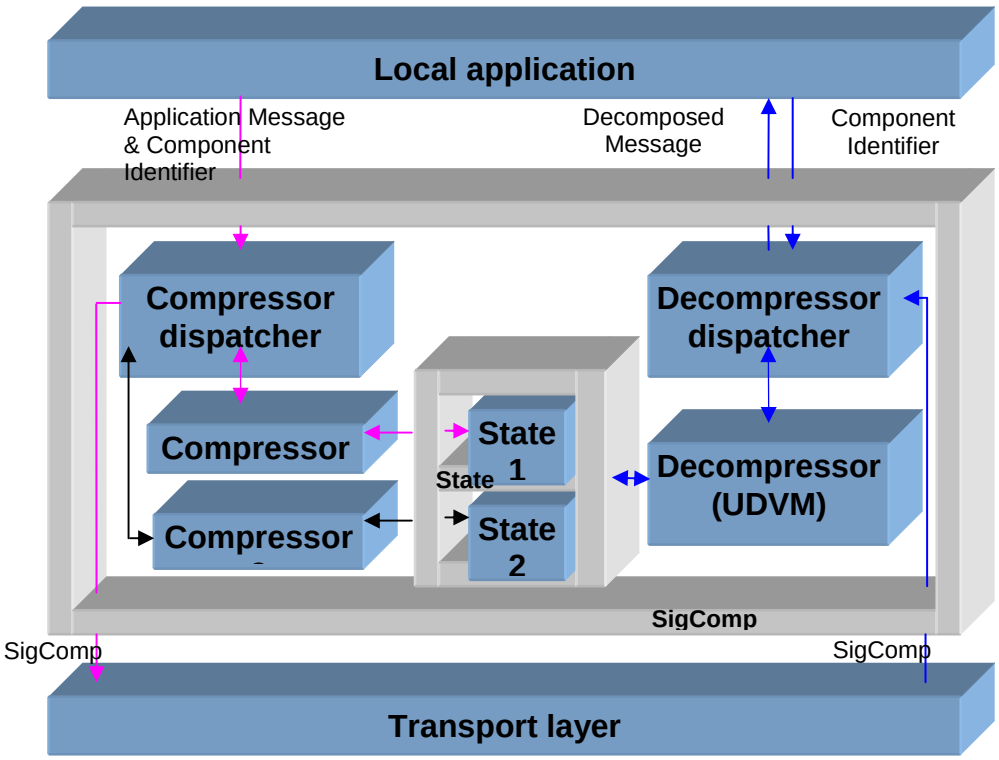


Figure 11. Sigcomp decompression operation

In the compressor side, messages are sent to compressor. Compressor creates reference dictionary. Compression is implemented by using a compression algorithm such as LZSS[1] for the string that exists in the reference library or in the already processed part of the message and replaces them with references. Then new information that will be used for compressing subsequent messages is stored in the temporary storage repositories.

[1] LZSS: A dictionary-based compression method that replaces a re-occurring sequence with a pointer to its earlier occurrence, developed in 1982 by Storer and Szymanski

There are three dictionaries used for Sigcomp: a static dictionary, a dynamic dictionary and a mixed compression. Static dictionary is specific to SIP and SDP, it contains well-known SIP phrases and keywords. In dynamic compression, new messages are compressed once a sent message is acknowledged. This means the SIP dictionary together with sent acknowledged messages are used as a dictionary for compression. In mixed compression, received messages are used to compress the messages to be sent; a static dictionary together with both sent acknowledged messages and received messages are used as the dictionary for compression.

2.4 Voice over wireless networks

2.4.1 VoIP over WLAN

In the context of this thesis, VoIP over WLAN (VoWLAN) is a VoIP call over a 802.11 WLAN. The call can be between users within a single WLAN, users in a WLAN and a wired LAN, or a user in a WLAN via a suitable gateway to a PSTN(including a cellular user). Users can use an 802.11 softphone, i.e., a general-purpose computer executing the VoIP client as software, or a hardphone to make a call. A softphone uses software installed on a PC, PDA or smart mobile phone. A hardphone is a IP phone, usually using an embed Digital Signal Processor (DSP) to reduce power or lower delay.

There are some specific issues that VoWLAN needs to address. These are spectrum congestion and interference while transmitting over a wireless link, large delays due to handoff, QoS, and WLAN security mechanisms (e.g., 802.1x, web-based authentication).

[E. Dimitriou] indicates that an 802.11b access point can support only a limited number of VoIP flows (much less than the theoretical 11 Mbps). Round-trip delay, jitter, and packet loss increase as the number of flow increases. Although WLAN allows the user to be mobility, the quality of the voice deteriorates as the distance between the user and access point increases. More precisely, [T. J. Patel] concludes that 802.11b can support 14-18 simultaneous VOIP sessions using a G723.1 CODEC and 8-10 VoIP session using a G711 CODEC. [M. Coupechoux] concludes that with a fixed distance to an access point, evaluating quality of voice calls by the E-model

and with a network delay of 20 ms, when using an 802.11b access point, G711 supports 5 simultaneous calls, GSM-EFR up to 12 calls, and G723.1 up to 18 calls.

When the VoWLAN user moves around, handoff between access points will occur. This will contribute to delay and may result in loss of communications. [T. Kanter] and [J-O. Vatn] show that in the case of handoff without authentication, the delay of handoff intra-domain between access points is acceptable (around 150 to 200ms). However, when authentication mechanisms and/or inter-domain handoff exist, then the delay is more than the acceptable VoIP delay (i.e. greater than 400ms). We have measured handoff delay when 802.1x is enabled, and the resulting in a delay was between 500-800ms [V. Tung].

[K. J. Khan] measured handoff delay in StockholmOpen.net using Mobile IPv4 and web-based authentication; the resulting delay was up to 1000ms.

Previous research has shown that VoIP over WLAN has a large delay. In practice, positive results were found by the ITU[1] when they deployed a VoWLAN solution for the provision of rural communication in Bhutan [T. C. Tobgyl].

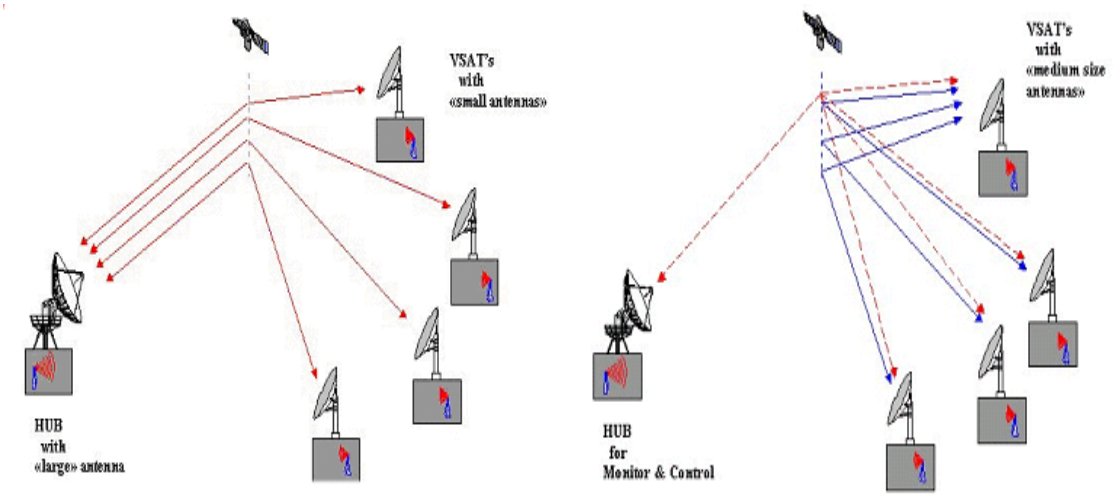
2.4.2 VoIP over Satellite

In isolated locations where there is no existing telecom or data communication infrastructure, satellite solutions provide an ideal solution for international communication. Such links can support both packet and circuit switch services. Satellite can be deployed quickly and deliver consistent QoS regardless of the user's location. However, satellite communication can be expensive and the cost is based the available bandwidth provided. Hence packet service has a marked advantage over traditional circuit switch service over a satellite link. The fact that packet services do not permanently reserve 64kbps or other fixed bandwidth means that the available bandwidth is used more efficiently and bandwidth can be shared with other data services of lower priority. VoIP over satellite takes advantage of this by using low bandwidth CODECS that provides the same QoS as a normal circuit-switched telephone call, but require less bandwidth.

There are two satellite network topologies: Mesh networks and Star networks. The star topology places an earth station at the center of the network. It is often requires two satellite hops for a call between two remote earth stations. An interactive voice

[1] ITU: International Telecommunications Union

call in this case is unacceptable because of the excessive delay. A mesh network allows remote earth stations to communicate directly with each other via the satellite, therefore, reducing the delay. Mesh networks are required for deploying VoIP and real time services although it may increase the cost for buying routing devices and/or radio frequency terminals at each earth station. Figure 12 illustrates communication in star topology and mesh topology.



a. Star topology b. Mesh topology

Figure 12. Satellite topology

There are many key factors that affect the quality of VoIP: propagation delay, jitter, and packet loss. Propagation delay is the time for transmission of a signal between an Earth Station and a satellite. All Geostationary Earth Orbit (GEO) satellites have an altitude of 36000 km, hence the propagation delay over the radio link up and back is always approximately 240ms. For Medium Earth Orbit (MEO) satellites and Low Earth Orbit (LEO) satellites, the propagation delay is approximately 100ms and 10 ms respectively. LEO satellites have lowest delay but only cover a small footprint and most use satellite to satellite links. When comparing with ITU's recommendation that the delay for a call should be below 150 ms and unacceptable if it is above 400ms, it seems that a VoIP call over a GEO satellite link is likely to be unacceptable. However, by using low bandwidth CODECS (G729, G723.1, AMR), echo cancellation and silence suppression, [T. Nguyen] concludes that satellite links provide a robust medium for transporting VoIP traffic, tolerating BERs as high as 10^{-5}. [J. Janssen] gives a model to calculate the delay budget. They have showed that the

delay budget depends on type of CODEC, the packet loss ratio, and echo loss[1] value, enabling VoIP with PSTN-quality voice over LEO satellites.

[1] Echo loss is the attenuation between the voice coder input and the voice coder output

3 WLAN and VoIP security

This chapter analyzes WLAN and VoIP from security perspective. First we examine WLAN authentication mechanisms and the security offered by WEP. Second we examine encryption and integrity in WLAN. Following this, we investigate SIP-based VoIP security solutions for heterogeneous networks. We then explore methods to secure SIP-signaling and to secure the media data. Finally, we investigate methods to provide VoIP traversal over firewall/NAT-enabled networks.

3.1 WLAN Security

WLAN is frequently considered more susceptible to attacks than wired network since it does not require a physical connection. Security risks in WLAN are: insertion attack, interception and monitoring wireless traffic, misconfiguration, and Denial of Service (DoS) attack. An insertion attack means placing an unauthorized device in the wireless network without going through the authentication process or by cracking the authentication process. An insertion attack could be man-in-the-middle attack where the attacker setups a rogue access point to capture sensitive information when users attempt to login to their services. Because access points broadcast data to all nodes within range, an attacker could capture packets and decode data or even inject packets into a connection based on data collected previously. Misconfiguration usually happens by carelessly using the default (manufacture) password or a small sized WEP key. A DoS attack in WLAN is as easy as simply using an RF[1] generator in WLAN band (usually 2.4 Ghz) or by flooding bogus packets into a access point.
Solutions to provide WLAN security include using an authentication system and data encryption. The next section goes to details of each of these methods.

3.1.1 Authentication in WLAN

Authentication is the process of verifying the identity and legitimacy of a person based on who they claim to be. There are several solutions to provide WLAN authentication: MAC filtering, WEP authentication, captive portal, and Port-based authentication. The first two solutions could only be used in a home or a very small

[1] RF: Radio Frequency

wireless network since they are easy to crack or do not scale. After giving some details of this, we will examine the captive portal solution and the port-based authentication system that will be use in WIDER system.

MAC filtering

APs can check the MAC address of the stations that associate to them. The AP can reject packets from unauthorized stations based on their MAC addresses. Using MAC filtering requires pre-configuring MAC address of all that are allowed to this network. MAC filter is easy to crack by using a wireless sniffer to capture MAC address of authorized interface and then spoofing MAC address with one of these (for example, using the a-Mac Address Change[1] tool).

WEP authentication

WEP authentication has two modes of operation: Open authentication and Shared key authentication. In Open Authentication mode, the AP accepts associations from all stations thus stations can connect via any available AP(s). In Shared key authentication, all interfaces share a single key which is used to authenticate into the network. Shared key mode uses a simple version of a challenge response protocols that need not involve a key exchange. Figure 13 illustrates WEP authentication in pre-shared key mode.

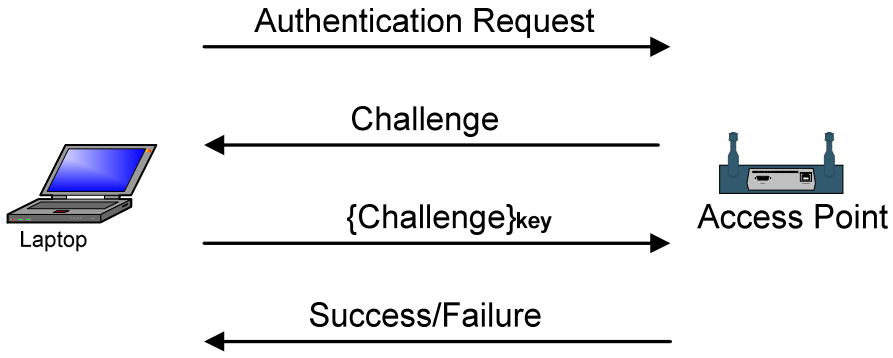


Figure 13. WEP authentication in pre-shared key

WEP uses an RC4 stream cipher to encrypt the stream of packets. Packets are encrypted using encryption key as follows

Encryption key = IV+ WEP key

[1] www.paqtool.com

Initialization Vector (IV) is a random value that prevents to use the same session key for multiple packets.

The data is encrypted as follows:

Encrypted stream = Packet XOR RC4 (Encryption key)

RC4 is a variable key-size stream cipher with byte-orientated operations invented by Ron Rivest[1]. The algorithm is based on the used of a pseudo random permutation.

Keystream in WEP could be reproduced if knowing data (packets) and encrypted data as follows:

To encode: *Encrypted stream = Packet XOR keystream*

To decode: *Packet = Encrypted stream XOR keystream*

Because of the XOR:

Packet XOR Encrypted stream = (Encrypted stream) XOR (Encrypted stream)

XOR (keystream)

Packet XOR Encrypted stream = keystream

WEP authentication is insecure. First, of all users share the same key, *anyone of them* could by accident leak the password to an attacker. Second, the IV is too small value (as 24bits means only 16 776 216 different keystreams for a WEP key), hence it is likely that the keystream will be reused. We only need a wireless sniffer to capture the challenge (in plain text) and the response (IV is sent in clear text; thus we know when a keystream is reused). We don't need to know the key, we could send the authentication request and encrypt it using earlier keystream to send the response and then get access to the network. Finally, WEP authentication is not mutual authentication. The AP only authenticates users while users do not authenticate the AP; this could be vulnerable in man-in-the-middle attack.

Captive Portal

Captive Portal is the WLAN authentication solution that is usually used in hotspots and campus networks. Captive portal forces clients to a website for authentication. If authentication is successful, clients are permitted to access network (Internet), otherwise, traffic is block at the captive portal controlled gateway. Captive portal

[1] http://theory.lcs.mit.edu/~rivest/

requires that the infrastructure includes: DHCP Server, Firewall controlled by an Authentication Server, and log in via a web page over SSL [SSLv3]. Figure 14 is captive portal model based on a subnet that implements StockholmOpen[1] network and IT-University[2], Stockholm, Sweden.

A WLAN user broadcasts a DHCP request asking for an IP address. A DHCP relay in the subnet forwards these requests to the central DHCP server. If the MAC address is not found in the database, the DHCP issues a temporary private IP address; otherwise, it relays the request to the chosen ISP which then assigns a public IP address to the user. If users are not in DHCP database, when the users make an HTTP request, they are redirected to the a website that asks for users to choose an ISP. After the user chooses an ISP, the DHCP server registers this user in the MAC database, and leases the IP address. The user send another broadcast DHCP request again, this time the DHCP will relay directly the request to the chosen ISP who will assign a public IP.

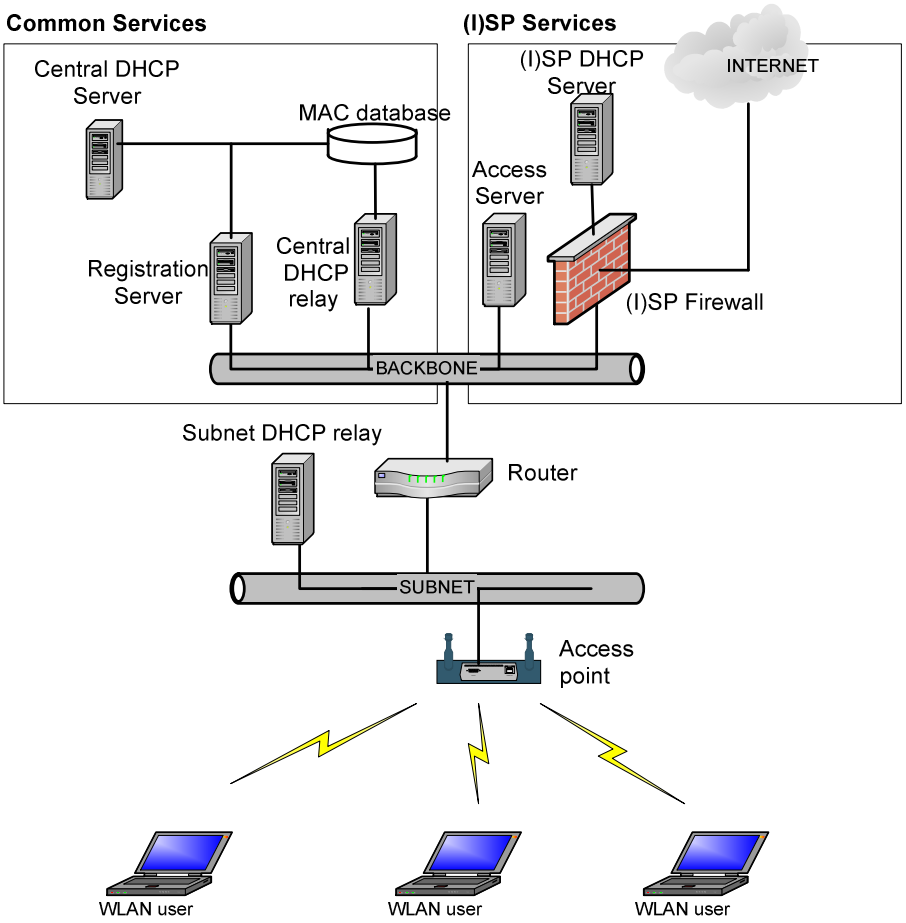


Figure 14. Captive portal based on opening holes through a firewall from a subnet

[1] www.stockholmopen.net
[2] www.it.kth.se

The key is that when a user makes an HTTP request it will be re-direct to the ISP's registration page (if the user is not yet registered). If the user enters a valid user and correct password, the access server will open a hole in the firewall to allow the user to access the global Internet.

The captive portal approach leaves data encryption to the application layer. It does not do anything special to provide a secure radio link. It is vulnerable to a passive attack where the attacker intercepts and monitors sensitive data. The captive portal's disadvantage is that it does not support mobility. A solution could be use firewalled mobile IP or VPN tunneling. However, captive portal's advantage is that it does not require any additional software to be installed on the client side. This feature makes it very popular for public wireless network access.

Port-based authentication

Port-based authentication, using IEEE 802.1x [802.1x] is a robust authentication method that enables both authentication and key management. IEEE 802.1x utilizes the Extensible Authentication Protocol (EAP) framework that supports a variety of authentication methods, including certificate-based authentication, smartcards, one-time passwords etc.

The IEEE 802.1x framework defines three entities involved in Port-based authentication:

- Supplicant: User or client's interface that wants to be authenticated
- Authenticator: Controls physical access to network based on the authentication status of the client. By default it closes ports (block traffic) and only allows EAP requests pass through until the supplicant is authenticated. Authenticators usually are access points and 802.1x-enabled switches.
- Authentication server: Provides authentication, authorization, and accounting (AAA). Although not defined in standard, authentication servers are usually RADIUS [RFC2865].

.

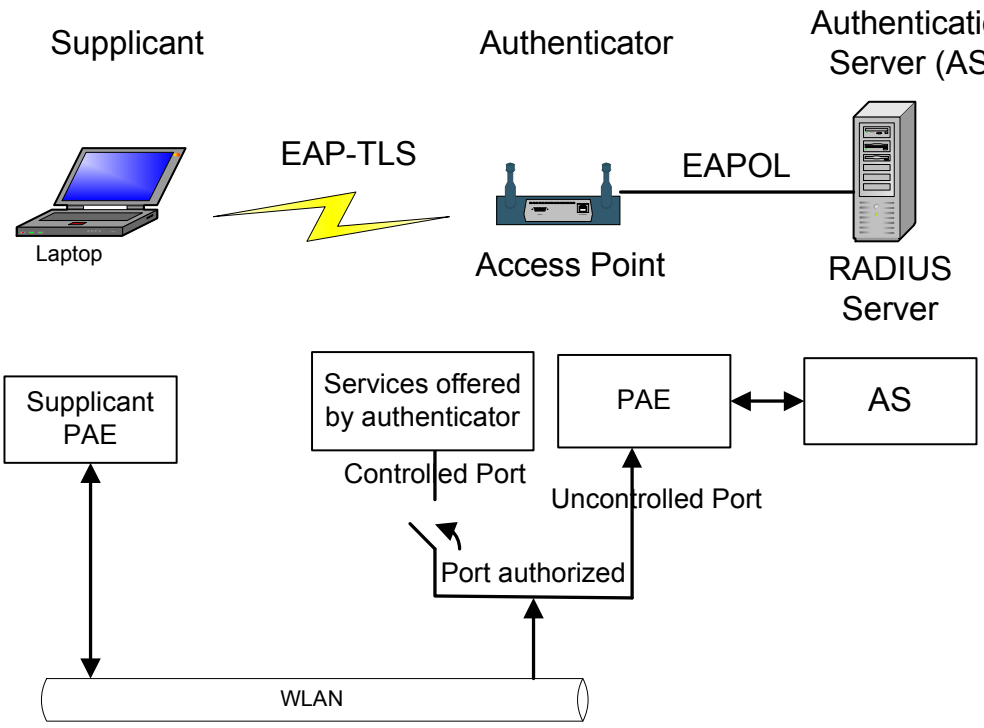


Figure 15. Physical and logic entities of 802.1x

Figure 15 shows the physical and logical entities of 802.1x. The Port Access Entity (PAE) is responsible for requests/responses during the authentication process. If the supplicant is authorized, then the authenticator opens the controlled port to offer connectivity to the network. The EAP framework allows mutual authentication and supports several different types of authentication: EAP-TLS, EAP-TTLS, EAP-MD5, LEAP.

After evaluating the advantages and disadvantages of different EAP authentication methods, we have implemented EAP-TLS and EAP_TTLS over WIDER. Then next chapter goes through details of the implementing them. Table 5 compares features of these different EAP authentication methods.

	EAP-MD5	LEAP	EAP-TLS	EAP-TTLS	PEAP
Server Authentication	None	Password Hash	Public key (Certificate)	Public key (Certificate)	Public key (Certificate)
Supplicant Authentication	Password Hash	Password Hash	Public key (Certificate or Smart card)	CHAP, PAP, MS-CHAP, EAP	Any EAP or public key
Dynamic Key Delivery	No	Yes	Yes	Yes	Yes
Security Risks	Identity exposed, Dictionary attack, MitM[1] attack, Session hijack	Identity exposed, Dictionary attack	Identity exposed	MitM attack	MitM attack

Table 5. Comparison of EAP authentication methods

- **EAP-TLS**

EAP-TLS is based on X.509 certificates [RFC2459] to handle authentication. It requires validating both client and server certificates to validate. EAP-TLS provides strong mutual authentication. It also generates dynamic WEP keys after the authentication exchange. Figure 16 shows the process of authentication exchange using EAP-TLS over WIDER.

- **EAP-TTLS**

EAP-TTLS is actually an extension of EAP-TLS. EAP-TTLS uses a certificate to authenticate servers but on the user side, it allows another authentication protocol **inside** an encrypted TLS tunnel. Supplicant can then use a challenge-response user/password or token-based authentication or certificates etc.

[1] MitM: Man in the Middle attack, an attack where the attacker is able to read, and possible modify at will, messages between two parties without letting either party know that they have been attacked

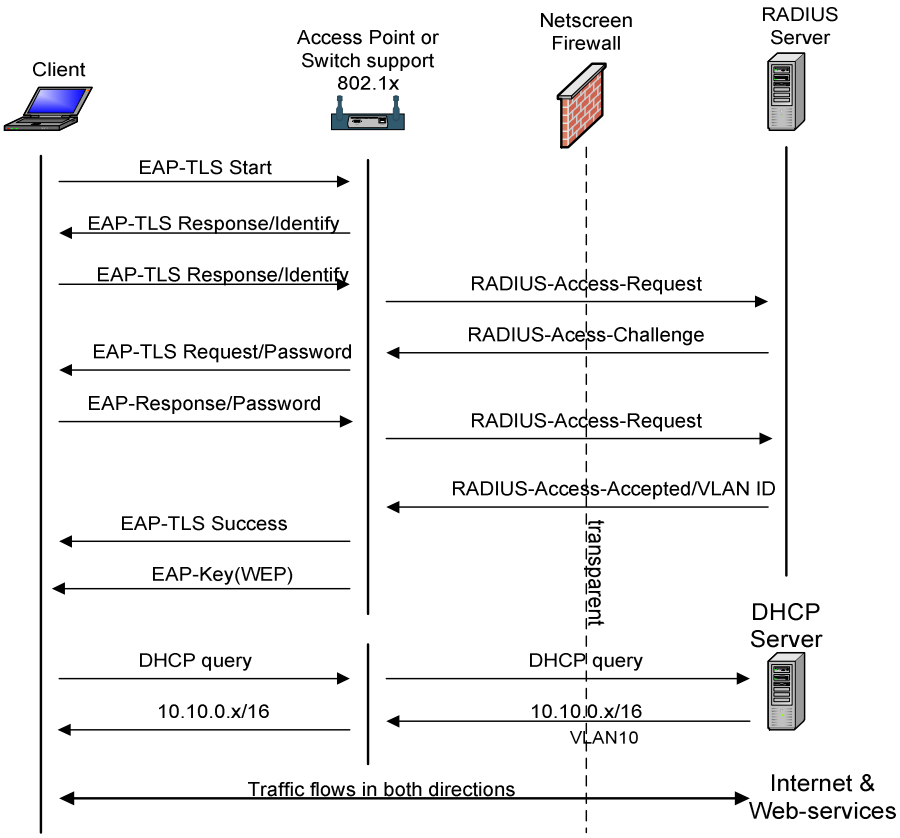


Figure 16. EAP-TLS authentication exchange over WIDER

3.1.2 Encryption and integrity in WLAN

- **Wired Equivalent Privacy (WEP)**

WEP is weak *both* in authentication *and* in encryption and integrity. Section 3.1.1 introduces how WEP is encrypted. Here we list some weakness of WEP on encryption and integrity.

- WEP does not have any mechanism for key management and the key size is small (only 40 bits).
- The Integrity Check Value (ICV) is based on CRC-32 so that it can be re-computed after the packet is modified.
- Encryption using weak key [S. Fluhrer] could disclose shared secret.

In our implementation, we use Port-based authentication that issues dynamic WEP. Moreover, the length of the WEP key is 108 bits thus increasing its security.

- **WiFi Protected Access (WPA) and 802.1i**

Realizing the weakness of WEP, the Wifi Alliance[1] teamed up with the IEEE 802.11 working group to introduce new WLAN security standards: WPAv1 and 802.1i. WPAv1 is a subset of the 802.11i security framework that helps to quickly deploy a secure WLAN solution in the market before the 802.11i standards are approved. Authentication in WPA and 802.11i is based on Port-based authentication (see section 3.1.1). Data encryption and integrity are based on the Temporal Key Integrity Protocol (TKIP) and Counter Mode with CBC-MAC Protocol (CCMP).

TKIP process begins with a 128-bit "temporal key" shared among supplicant and authenticator. TKIP combines this temporal key with the supplicant's MAC address and adds a relatively large 16-octet initialization vector to produce a key that will encrypt data. The method ensures that each supplicant uses a different key stream to encrypt data. TKIP still uses RC4 to perform encryption, but it is different as TKIP changes temporal keys every 10 000 packets. This later fact reduces the risk of exposing the key.

CCMP uses Advanced Encryption Standard (AES) in Counter Mode with Cipher Block Chaining Message Authentication Code (CBC-MAC). Counter Mode is used for data privacy and a CBC-MAC is used for data integrity and authentication. The Message Authentication Code (MAC) provides the same functionality as Message Integrity Code (MIC), used with TKIP.

3.2 SIP Security

Voice over IP is believed to be easier to eavesdrop than traditional circuit switched telephone networks. Since voice packets transmitted over public IP infrastructures can be sniffed, recorded, and reconstructed providing a complete record of a voice communications session. A VoIP call usually includes two parts: Signaling and Media. A secure voice calls requires both to be secured.

SIP signaling security is divided in two parts: end-to-end security and hop-by-hop security. Figure 17 illustrates the security segments within SIP.

[1] www.wi-fi.org

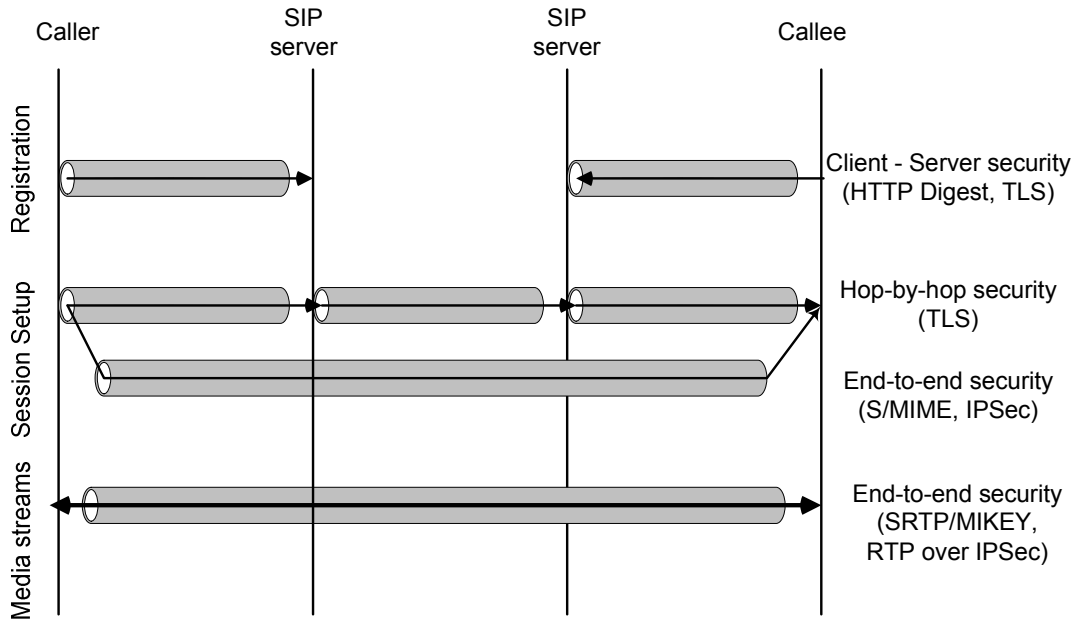


Figure 17. SIP security segments

3.2.1 SIP digest authentication

SIP's digest authentication is based on the simple challenge-response paradigm from HTTP's digest authentication [RFC2617]. The SIP digest authentication is usually only one way where the SIP proxy authenticates the SIP client, but not the reverse. SIP's digest authentication provides message authentication and replay protection but not integrity or confidentiality. There are four header fields that use for proxy and UA authentication: WWW-Authentication, Authorization, Proxy-Authentication and Proxy-Authorization. Proxy-Authentication and Proxy-Authorization are used when a proxy demands authentication before forwarding a message. The WWW-Authentication header is used when authenticating to the server that will deliver a service. When a Proxy receives a request for a protected domain that is not authenticated, it responds with a 401 (Unauthorized) or 407 (Proxy Authentication Required) which contains the WWW-Authenticate header. The mandatory fields for WWW-Authenticate header are realm and nonce, while optional fields are domain, opaque, stale, algorithm, qop-options, and auth-param. The client needs to respond to this challenge by using the Authorization header containing credential information of the client. The mandatory fields of Authorization headers are: username, realm, nonce, digest-uri, response; while optional fields are: algorithms, cnonce, opaque, message-qop, nonce-count, and auth-param. Bellowing is an example of UA registering with a SIP proxy.

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP 147.214.160.103:5061;branch=z9hG4bK41.682c.0
To: <sip:tung@wider.se>;tag=129e870-705c.16
From: <sip:tung@wider.se>;tag=2389114-705c.0
Call-ID: 72ae-a7c-2cd60@147.214.160.103
CSeq: 28765 REGISTER
Max-Forwards: 70
Server: Ericsson-SIP-Core-Reference-Server/CXC1328365R1A010
 WWW-Authenticate: Digest
realm="wider.se",nonce="3ce60b9f3e5b03ab682c2583h50d1ca",
algorithm=MD5,opaque="d069d2b65984c25d872137ec2e583035",qop="auth",st
ale=false

REGISTER sip:wider.se SIP/2.0
Via: SIP/2.0/UDP 147.214.160.103:5061;branch=z9hG4bK42.682c.1
To: <sip:tung@wider.se>
From: <sip:tung@wider.se>;tag=22e71c4-705c.1
Call-ID: 72ae-a7c-2cd60@147.214.160.103
CSeq: 28766 REGISTER
Max-Forwards: 70
User-Agent: PoC-client/OMA1.0 SleIPner/1.08
Contact:
<sip:tung@147.214.160.103:5061>;q=1.0;expires=3600;description="Avail
able";+g.poc.talkburst
Authorization: Digest username="tung",realm="wider.se",
nonce="3ce60b9f3e5b03ab682c2583h50d1ca" ,uri="sip:wider.se",
response="207fd504d0a4ee3b351e89423fbaf09c",algorithm=MD5,cnonce="258
3682c",opaque="d069d2b65984c25d872137ec2e583035",qop=auth,nc=00000001
```

3.2.2 S/MIME in SIP

SIP messages can carry Secure MIME (S/MIME) [RFC2633] bodies to provide public key distribution, authentication, integrity protection and confidentiality of SIP signaling data. There are two types of S/MIME bodies for SIP: multi-part/signed that are used to sign messages without encryption and application/pkcs7-mime that first signed and then encrypted SIP message bodies. S/MIME requires using certificates or private keys. In multipart/signed MIME type, the user certificate can be forwarded to the recipient and embedded into the pkcs7-mime or pkcs7-signature. In application/pkcs7-mime, it is required to know the recipient's public key. This often achieved by getting the public key from a public directory. Below is an example of the SIP MESSAGE method that signed message by S/MIME.

```
Contact: <sip:alice@a.example.com:5070>
```

```
Max-Forwards: 70
Content-Type: multipart/signed;boundary=65b6563f5e8ef632;\
              micalg=sha1;protocol=application/pkcs7-signature
User-Agent: SIPimp.org/0.2.2 (curses)
Content-Length: 1653

--65b6563f5e8ef632
Content-Type: text/plain
Content-Transfer-Encoding: binary

Hi
--65b6563f5e8ef632
Content-Type: application/pkcs7-signature;name=smime.p7s
Content-Disposition: attachment;handling=required;filename=smime.p7s
Content-Transfer-Encoding: binary

******************
* BINARY BLOB 1   *
******************
--65b6563f5e8ef632--
```

3.2.3 SIP over TLS

TLS [RFC2246] can be used to protect SIP signaling messages against loss of integrity, confidentiality, replay protection as well as integrated key-management with mutual authentication and secure key distribution. TLS provides hop-by-hop security between UAs and proxies and between proxies and requires a public key infrastructure for handling certificates. [RFC3261] requires the use of TLS for SIP Proxy, SIP registrars and SIP redirect server. Because TLS provides a secure and reliable transport layer, SIP signaling over TLS can not run over UDP but must run over TCP. In SIP messages, a secure SIP URI is defined with additional “s”, e. g. **sips:user@example.com**. Below is an example of secure SIP MESSAGE method sent over TLS with a URI having SIPS and having a VIA header indicating TLS is used.

```
MESSAGE sips:kumiko@example.net SIP/2.0
To: <sips:kumiko@example.net>
From: <sips:fluffy@example.com>;tag=03de46e1
Via: SIP/2.0/TLS 127.0.0.1:5071;
     branch=z9hG4bK-d87543-58c826887160f95f-1--d87543-;rport
Call-ID: 0dc68373623af98a@Y2ouY2lzY28uc2lwaXQubmV0
CSeq: 1 MESSAGE
```

```
Contact: <sips:fluffy@127.0.0.1:5071;transport=TLS>
Max-Forwards: 70
Content-Transfer-Encoding: binary
Content-Type: text/plain
Date: Sat, 19 Feb 2005 00:48:07 GMT
User-Agent: SIPimp.org/0.2.5 (curses)
Identity: qKUEWvgss+F0pQHJCyarb8IMbDh1d1gi1Aq51ty61bO+ug5ZQzo31xn
          MAFHUe0tzNVoyOfmGUY2dIEWJ2iZlGI5EW3RF5hGN9f0y39iCRqGEAE
          B4UG5ocU4RzgXfK3Durle/66rkyCaLPJQ/pzgA+qW/nQytSuzewhDrD
          FRrCBQ=
Content-Length: 6

Hello!
```

3.2.4 SIP over IPsec

IPsec [RFC2401] provides security at the network layer. IPsec can be used for hop-by-hop or end-to-end security that provides authentication, integrity and confidentiality. The IPsec implementation is independent of SIP since it operates at network layer. However, IPsec is usually used for setting up long-lived connections between SIP proxies or between SIP proxies and UAs. IPsec has two protocols that provide security services: Authentication Header (AH) is responsible for authentication and Encapsulating Security Payload (ESP) that supports both encryption and authentication. To setup a secure connection, IPsec creates a Security Association (SA) that contains the secret key, algorithms, IP address, IPsec protocol (AH, ESP) and Security Parameter Index (SPI). Note that each SA is one-way relationship, to set up bi-directional security, both sender and receiver need to initiate SAs with each other. There are two modes of operations in IPsec: transportation mode and tunnel mode. In transportation mode, IPsec inserts an AH or ESP header and applies their security mechanism on certain parts of the original IP packets. In tunnel mode, IPsec encapsulates the whole IP packet into a new packet with a new IP header, and AH or ESP to provide security mechanisms. Transport mode normally operates to secure direct communication between two endpoints while tunnel mode is often used between two intermediate gateways. In WIDER, IPsec is applied in tunnel mode to connect the field network with the office network or to connect dial-up users in order to provide both secure data and voice.

In IPsec, key management can occur automatically or manually. Automatic key management is used for large environments that need a large number of SAs. Internet

Key Exchange (IKE) is used by default for this case. IKE has two phases: Phase 1 negotiates SA (i.e. ISAKMP SA or IKE SA) and prove each party's identity. Phase 2 negotiates an IPsec SA (i.e. ESP or AH) so that a secure connection is established.

IPsec sessions can be set up by SIP UAs. [J. Orrblad] introduces a novel way to set up IPsec session using a SA embedded in a MIKEY [RFC3830] message within a SIP MIME payload. His approach adds two IPsec SAs and two IPsec policies in MIKEY messages and defines a MIME content-type application/mikey.

3.3 Media Security

VoIP and real-time applications use the Real-time Transport Protocol (RTP) [RFC3550] as a transport protocol for packet voice and other real-time data. In conjunction with RTP, the Real-time Transport Control Protocol (RTCP) (also defined in RFC3550) is used to provide feedback regarding the quality as seen from the receiver and sender. The media layer (conveyed in RTP packets) often travels in one way and without protection, hence it is considered easy to be eavesdropped, injected of forged content or modified of packets to degrade voice quality. However, due to restrict real-time requirements, these packets are sensitive to delay and jitter and are therefore transmitted over UDP, thus are only some security mechanisms are suitable for protecting the media layer. These include: SRTP (Secure RTP), IPsec, RTP over DTLS.

3.3.1 Secure RTP

Secure RTP [RFC3711] is an extension of the RTP profile that provides confidentiality, message authentication, and replay protection to both RTP and RTCP. SRTP adds two fields: a Master Key Identifier (MKI) and a MAC (Message Authentication Code). MKI is optional field that identifies the master key from which the session keys were derived. The optional MAC is a cryptographic checksum computed over the header and payload of the RTP packet. It protects the packet against un-authorized modification. Secure RTCP is constructed as the same way as SRTP but MAC is mandatory field. The reason is to protect against an attacker modifying packets to teardown a session, for example, by sending a BYE in an RTCP packet. Figure 18 shows the layout of both the Secure RTP packet and the Secure RTCP packet.

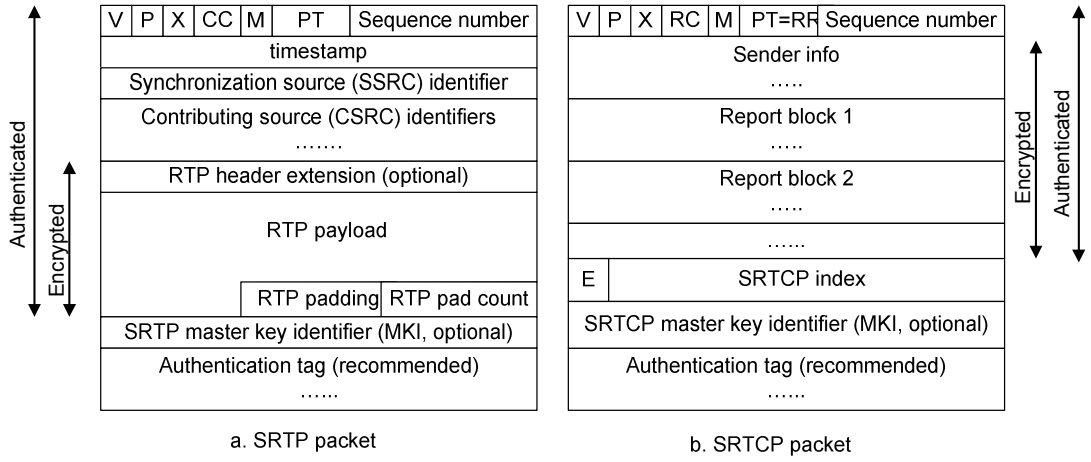


Figure 18. SRTP and SRTCP packet

SRTP encrypts the payload of an RTP packet, the default encryption algorithms is Advanced Encryption Standard in Counter-mode (AES-CTR). In the case of UMTS, AES-f8 mode is used. Figure 19 illustrates the encryption of an RTP/RTCP payload with AES-CTR mode.

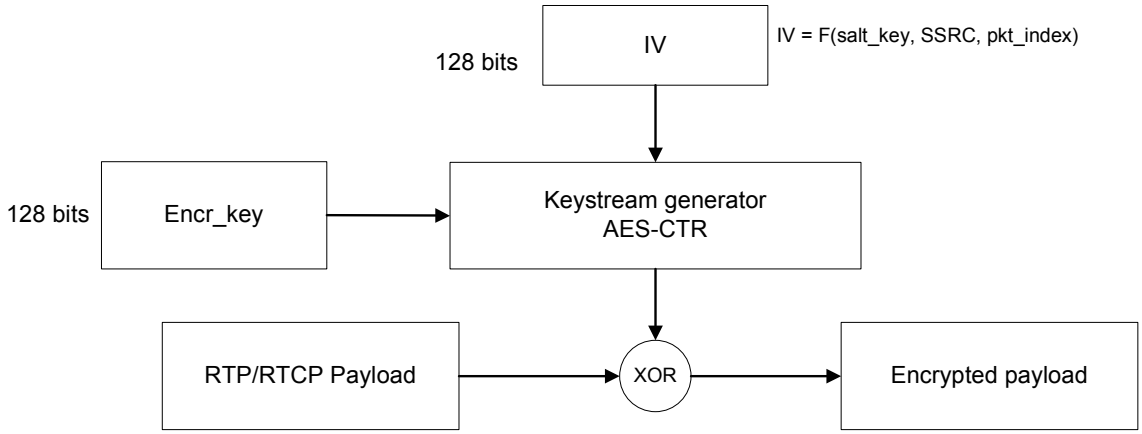


Figure 19. Encryption of RTP/RTCP payload with AES-CTR mode

SRTP use of AES-CTR has several advantages: the keystream can be pre-computed even before sender's payload arrives (thus reducing computation delay and can be used in lower processing equipment). In addition, the stream cipher encrypts the payload without needing of any additional padding unlike block encryption. AES in counter mode acts as a *keystream generator* producing a pseudo-random keystream of arbitrary length that is applied in a bit-wise fashion to the RTP/RTCP payload by means of a logical XOR function, thus working as a classical stream cipher. AES itself is a block cipher with a block size of 128 bits and a key size of 128, 192, or 256 bits. In order to work as a pseudo-random generator AES is loaded at the start of each

RTP/RTCP packet with a distinct initialisation vector (IV) that is derived by hashing a 112 bit salt_key, the synchronisation source identifier (SSRC) of the media stream, and the packet index. Encrypting this IV results in an output of 128 pseudo-random bits. Next the IV is incremented by one and again encrypted, thus generating the next 128 bits of the keystream. By incrementing the IV by increments of one as many keystream blocks can be generated as are required to encrypt the whole RTP/RTPC payload. Any remaining bits from the last keystream block are simply discarded.

SRTP uses HMAC-SHA1 as the default message authentication algorithm. The authentication tag (MAC) has an 80-bits length as the result of HMAC-SHA1 (160bits auth_key, selective header + payload RTP/RTCP).

SRTP [RFC3711] does **not** define mechanism for exchange key management. RFC3711 only defines how to derive the session key from master key and IV (Initiator Vector). Figure 20 show how session key and authentication key derived. MIKEY [RFC3830] and [draf-sdp-descriptions] describe the method used to exchange a master key, a master salt, and a host identity that can be used with SRTP. The master key has 128, 192, or 256 bits and is AES encryption key. The IV is created by a pseudo-random function based upon: a 128 bit master_salt, a 1 byte label, and a session key number. Labels from 0x00 to 0x05 create session keys (e.g. encryption key, authentication key, salt key) for both SRTP and SRTCP.

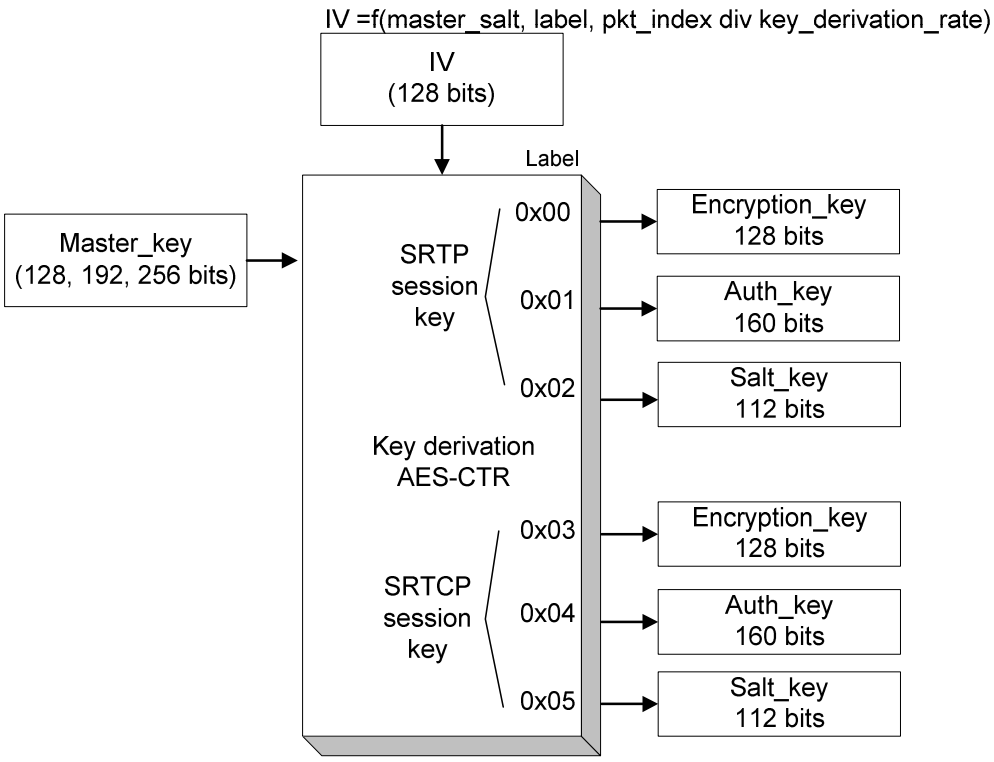


Figure 20. Session key and authentication key derive

Because of VoIP regulations in some countries not allow encrypted VoIP, SRTP is still not widely deployed. There are only some IP Phone products from SNOM (www.snom.com), Cisco (www.cisco.com) and Minisip SIP Softphone (www.minisip.org) that supports SRTP.

[I. Abad] has measured an implementation of secure media in VoIP calls using SRTP that was shown to increase delay 70-80ms on a 700 Mhz Pentium III laptop . I have also implemented SRTP/MIKEY and measured the delay between RTP and SRTP. These results are presented in the following chapters.

3.3.2 IPsec

As mentioned before IPsec can be used for securing both signalling and media. Section 3.1.4 gaves an overview of IPsec. When applying IPsec to the media layer, the most significant effect on QoS of VoIP is the reduction in effective bandwidth due to the larger header (AH, ESP and a new IP header for tunnelling mode) and increasing delay by processing encryption/decryption of packets. Note that this increased packet overhead also impacts the transmission delay, internal router internal, and queueing delays thus affecting jitter and overall packet delay.

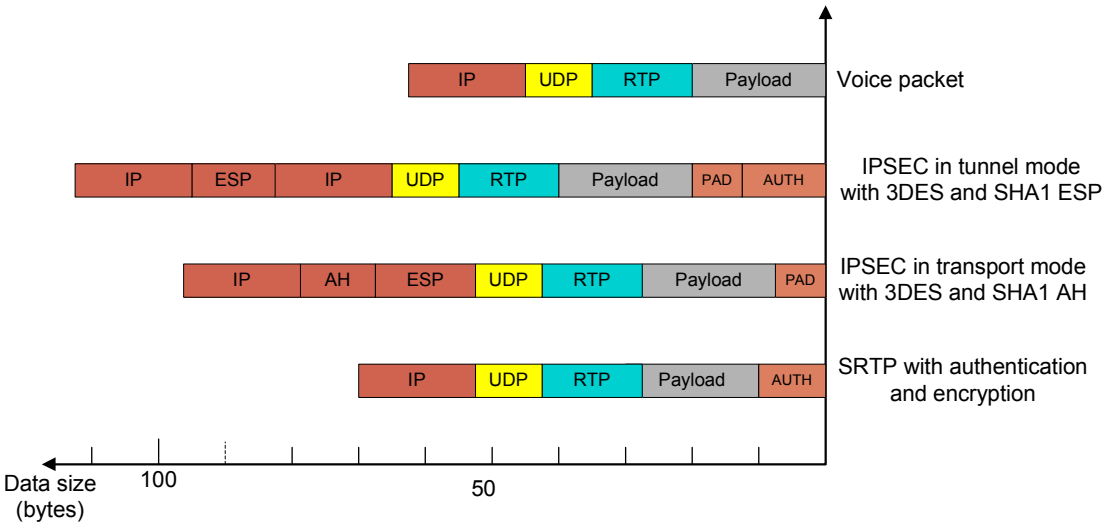


Figure 21. Comparison SRTP, IPsec and voice packet overhead

Figure 21 compares the difference in packet sizes with a normal voice packet, a SRTP packet and a IPsec voice packet. Depending on the mode of operation in IPsec, the voice packet dramatically increases to around 100 bytes while a normal voice packet is only 60 bytes (assuming payload of 20 bytes). In comparison with a normal voice packet, SRTP adds 10-12 bytes for the MKI and authentication tag.

Computational delay is the delay needed for encryption/decryption of the packet voice. Table 4 compares the computational delay between IPsec and SRTP for secure media.

	Sender side	Receiver side
IPsec	delay = Enc(UDP header||RTP header||RTP payload) + GenMAC(ESP header||UDP header||RTP header||RTP payload SRTP computation)	delay = Dec(UDP header||RTP header||RTP payload) + VerMAC(ESP header||UDP header||RTP header||RTP payload)
SRTP	delay = Enc(RTP payload) + GenMAC(RTP header || RTP payload)	delay = Dec(RTP payload) + VerMAC(RTP header || RTP payload)

Table 6. Comparison of computational delay IPsec and SRTP

3.3.3 DTLS

Datagram Transport Layer Security (DTLS) was created to solve the issue of missing mechanisms for UDP transport layer security. DTLS mimics TLS [RFC2246] but changes some points to make it suitable for unreliable transport:

- Like TLS, all DTLS data is carried in records. But DTLS requires DTLS records to fit within a single datagram to avoid fragmentation.

- TLS uses implicit record sequence numbers (RSNs) for replay protection. RSNs in DTLS must be explicitly specified since records can be lost or be arrived out of order.

- DTLS uses CBC mode or AES-CTR. DTLS does not allow using an RC4 cipher stream because random access is difficult for a keystream.

- Stateless cookies in DTLS prevent a Denial of Service (DoS) attack.

DTLS is still a work in progress [draft-rescorla-dtls], but it has been implemented on OpenSSL[www.openSSL.org] and the popular SIP Stack Resiprocate (www.resiprocate.org). DTLS has advantages over IPsec because it is easy to implement from an application point of view. In compare with SRTP, DTLS is independent of other protocols as it does not rely on RTP or other protocols for key exchange.

3.4 Key management

Key management provides a mechanism to negotiate keys and security associations between communicating parties. Depending on the communication environment, there are many key exchanges standards: MIKEY, ISAKMP, GDOI, etc. In a SIP/VoIP environment, MIKEY is recommend because of its design goals which facilitate real-time multimedia applications.

3.4.1 MIKEY (Multi-media Internet Keying)

MIKEY is the key management protocol for multimedia communications. MIKEY was purposely designed for use in a wireless environment to minimize the number of round trips, hence minimize delay, consume low bandwidth, require low computational cost and have a small memory footprint. MIKEY is an independent key exchange protocol that can be embedded in SIP and H323. The scenarios for key

exchange in MIKEY can be peer-to-peer, one-to-many, and small-size interactive group.

MIKEY introduces some new concepts for setting up a crypto session. The first is Traffic Encryption Key (TEK) that used as a master key for creating session keys (encryption keys, authentication keys, salt keys). Second is TEK Generation Key (TGK) that is used as a kernel to generate the TEK. Note that one TGK can generate multiple TEKs to serve in multiple media sessions. To have the same Crypto Session, each side maintains a Crypto Session Bundle (CSB) that has common TGKs and security parameters. To agree upon a single TGK, the two parties authenticate each other by sending credentials that are encrypted with a shared-secret, public-key, or Diffie-Hellman. After the same TGK is established, each side generates a TEK using a derivation function and then uses this TEK to encrypt and authenticate the session. Figure 22 illustrates the MIKEY key exchange and how it applies to SRTP.

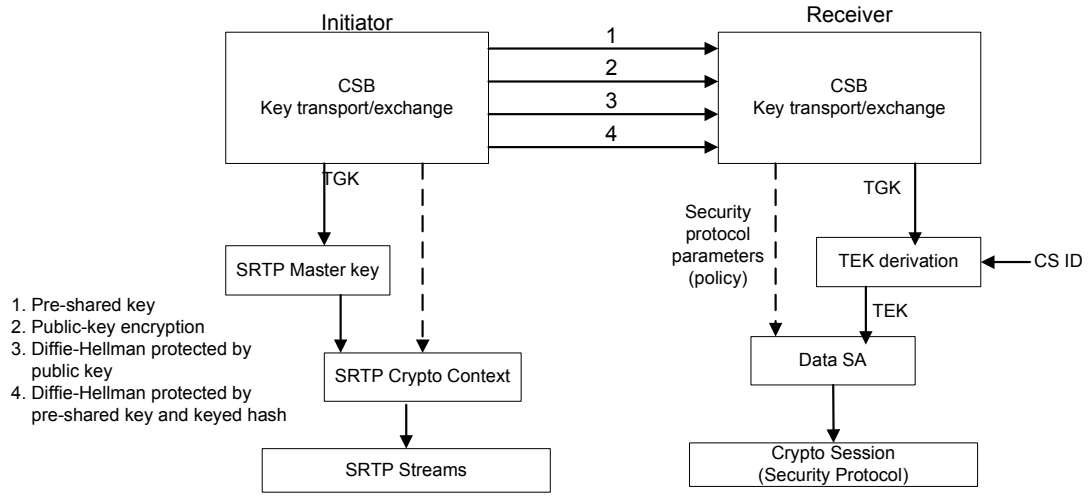


Figure 22. MIKEY operation

RFC 3870 does not specify the transport protocol that carries the MIKEY message. In implementation, MIKEY is embedded in the body of a SIP message with a new attribute a=key-mgmt: mikey [draft-kmgmt-ext] in the SIP INVITE method.

Even with only one round trip time to set up the key exchange, MIKEY still adds additional delay, especially when using public keys and Diffie-Hellman. The effect of key management during call establishment was exmined in [J. Billien]. Applying SIP security, particular when using Diffie Hellman key exchange, J.Billien et al. shows that an additional approximately 80 ms is required for the calling delay and answer

delay for call establishment. This delay will be much smaller if pre-shared key is used instead.

3.5 VoIP Security solution

A complete Secure VoIP solution requires secure SIP signaling, secure media and a secure query of a DNS server. Figure 23 is a solution that demonstrates a security-enhanced SIP trapezoid.

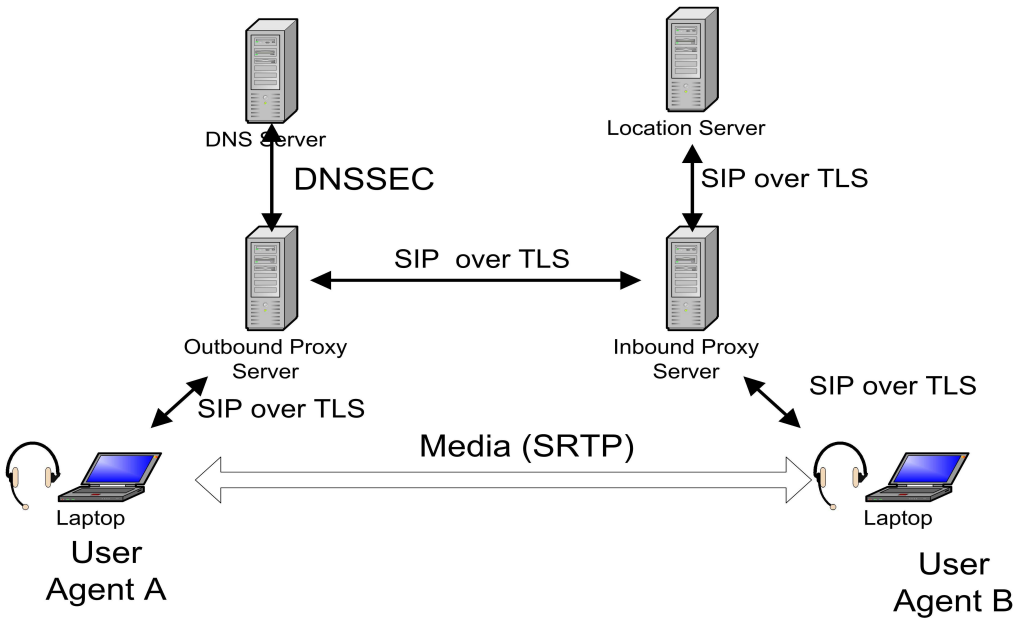


Figure 23. Secure SIP trapezoid

3.6 Firewall/NAT

Firewall and NAT challenges current implementions of VoIP over data networks. Firewall strictly controls incoming and outgoing traffic to protect the network from unauthorized access. Typically, firewalls only allow outgoing traffic that originates from a trust zone and allows incoming traffic (from un-trusted zones) only if the session is initiated from a computer in the trusted zone. VoIP, on other hand, needs to support two way communications thus the initiator can be in either a trusted or untrusted zone. In addition, VoIP separates the signaling and media, use dynamic port numbers for the media over datagram; however, a firewall will normally block all incoming datagrams other than those to some well-known ports to prevent from Denial of Service attacks (DoS). Opening ports for VoIP traffic means opening a pinhole during the network is insecure but may be acceptable for a short time, for example, during a session. An desirable solution to this problem must allow secure

two-way communications without changing the firewall rules or reducing firewall's security level.

Today, NAT (Network Address Translation) [RFC1631] is considered the most obstacle to the deployment of VoIP in data networks. NAT was created to ease the shortage of IPv4 address. To do so, NAT devices interconnect between private and public networks. The NAT translates private IP address and port number into a public address (globally routable) when traffic traverses from the private to public networks. The problem NAT causes in two-way VoIP calls occurs because we need to know the IP addresses of both end-points for both transmitting media and signaling. However, because the endpoint behind a NAT has private non-routable IP address, the connection will fail. Depending on the type of NAT, there are various solutions to this problem. There are 4 types of NAT defined in [RFC3489]:

Full Cone: A full cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port. Furthermore, any external host can send a packet to the internal host, by sending a packet to the mapped external address.

Restricted Cone: A restricted cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port. But unlike a full cone NAT, an external host (with IP address X) can send a packet to an internal host only if the internal host had previously sent a packet to IP address X.

Port Restricted Cone: A port restricted cone NAT is like a restricted cone NAT, but the restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host has previously sent a packet to IP address X and port P.

Symmetric: A symmetric NAT is one where all requests from the same internal IP address and port, to a specific destination IP address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping will be used. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

The following are some current proposals to address the NAT and Firewall problem:

- Universal Plug and Play (UPnP)
- Simple Traversal of UDP Through NAT devices (STUN)
- RTP Relay (TURN)
- Application Layer Gateway (ALG)
- Middlebox Communication (MIDCOM)
- Session Border Controller (SBC)
- Interactive Connectivity Establishment (ICE)
- Tunneling techniques

3.6.1 UPnP

UPnP [UPnP 1.0] was created by Microsoft to allow network devices to discover and configure other network components, including UPnP-enabled NATs and Firewalls. A UPnP-enabled client queries the NAT via the UPnP protocol to map a particular public IP address:Port to the client's IP address:Port. The client can than modify the SIP/SDP message use this new mapping IP:Port to set up two-way communications. Today some products support UPnP such as: Zultys's IP phone (www.zultystechnologies.com), SNOM's 105 IP phone (www.snom.com), and Hitachi's WirelessIP5000 (www.hitachi.com),

3.6.2 STUN

Simple Traversal of UDP Through NATs (STUN) [RFC3489] is aclient-server architecture that can *discovers* public IP:Port mapping and also determines the NAT-type. A STUN client sends a request with several parameters: RESPONSE-ADDRESS, Change IP, and Change Port. A STUN server responded with the mapped IP: Port in RESPONSE-ADDRESS. Depending on the flags Change IP and Change Port, STUN will answer with different IP and Port value. The STUN client can determine after receiving enough responses from STUN server if the client is on a public Internet, behind a firewall that blocks UDP, and what type of NAT it is behind (if any).

Many current clients and SIP phones support STUN, such as: eyeP's Media (www.eyepmedia.com), XTEN's X-lite (www.xten.com), SNOM's IP phone (www.snom.com), Cisco's ATA (www.cisco.com), HotSIP (www.hotsip.com),

SIPURA's SPA-2000 ATA (www.sipura.com), and Leadtek's BVA Series (www.leadtek.com).

3.6.3 TURN

One of STUN's flaws is that it does not work for symmetric NATs. IETF has proposed Traversal Using Relay NAT (TURN) [draft-turn] as alternative solution to solve the media traversal problem for symmetric NATs. TURN relies on a server that is inserted in the media and signalling path. This TURN server is located either in the customers' DMZ[1] or in the Service Provider's network. A TURN-enabled SIP client sends an exploratory packet to the TURN server, which responds with the public IP address and port (used by the NAT) to be used for *this* session. This information is used in the SIP call establishment messages and for subsequent media streams. The advantage of this approach is that there is no change in the destination address seen by the NAT and, thus, symmetric NAT can be used. Few softphones and hardphones support TURN, such as: eyeP's Media (www.eyepmedia.com), SNOM's IP phone (www.snom.com)

3.6.4 Application Layer Gateway (ALG)

If Application Layer Gateway (ALG) software is embedded in the Firewall/NAT device, then it can understand the relationship between signaling messages and the media streams. It can dynamic control the NAT/Firewall by opening ports and re-writing SIP/SDP message or map between a private IP address:Port to a public IP address:Port.

A number of products implement ALGs for SIP, such as: Juniper's Netscreen204 (www.juniper.com), Intertex's IX66 (www.intertex.se), and Ingate's Firewall 1600 (www.ingate.com).

3.6.5 Middlebox Communication (MIDCOM)

Middbox Communication (MIDCOM) [RFC3303] utilizes a device that is outside the Firewall/NAT to control the Firewall/NAT for VoIP. MIDCOM performs the same role as ALG (i.e. parses VoIP traffic and instructs the firewall/NAT to open/close

[1] DMZ: Demilitarized Zone, in computer networking, usually meaning a subnet that sits between trusted internal network (private LAN) and unstrusted external network (public Internet).

ports via MIDCOM protocol). The IETF MIDCOM Working Group is in the process to standardizing the MIDCOM protocol.

3.6.6 Session Border Controller (SBC)

A Session Border Controller (SBC) is an all-in-one VoIP solution that alleviates configuring the NAT/Firewall from client's side. A SBC is a dedicated appliance that provides the following services: Firewall/NAT traversal, Call Admission Control, Service Level Agreement monitoring, and protocol interworking. A SBC typically includes a Signaling Proxy that acts as a high performance Back-to-Back User Agent (B2BUA) and a Media Proxy that acts as a transit point for RTP/RTCP media stream between UAs.

Some SBC products on the market are Newport Networks' 1406 (www.newport-networks.com), Netrake's nCite (www.netrake.com), Data connection's DC-SBC (www.dataconnection.com).

3.6.7 Interactive Connectivity Establishment (ICE)

Interactive Connectivity Establishment (ICE) [draft-ice] is a methodology for NAT Traversal. ICE makes use of STUN, TURN, RSIP, or MIDCOM. ICE lists all supported NAT traversal protocols in preference order and use them to see if a host or port are reachable. In short, ICE defines 8-steps to set up two-way communication: Allocation, Prioritization, Invitation, Allocation, Verification, Affirmation, Verification, and Communication. An ICE-enabled caller first allocates its resources and collects server information (STUN server address, TURN, MIDCOM). After order according to preference, ICE sends a SIP INVITE message. The SIP INVITE will hopefully reach the callee by some mechanisms (TURN, SBC etc). Then the callee first does the same Allocation step. Thus the callee can verify how this caller can be reached. The callee sends an affirmation listing addresses it allocated in the 200 OK message. Now the caller checks which addresses are callee reachable in priority order and sends ACK via that path. Finally communication can commence.

ICE is currently a work in process; however, there are some applications supporting ICE. These include softphone and IP Phone: M2's Softphone (www.megapin.com), SNOM's 360 (www.snom.com), and XTEN's eYeBeam (www.xten.com)

3.6.8 Tunneling Techniques

VPNs can also solve NAT/Firewall problem, especially for corporate networks where the SIP server is located in the Intranet. Another tunneling solution is to use HTTP's port (80) to carry voice and signaling traffic. Skype[1] detects the type of network and changes its method to transfer media from UDP to TCP (even using port 80) if users are behind a port-restricted NAT and UDP-restricted firewall [S. A. Baset]. Both caller and callee in this case can send and receive voice traffic to and from another Skype user (the later having a public IP address). This model can take advantage of distributed RTP Proxies (i.e. the third Skype user) who acts as a media proxy.

Table 6 compares the advantages and disadvantages of above NAT/Firewall traversal solutions.

	Advantage	Disadvantage
UPnP	-Works in 4 types of NAT -No change of infrastructure - Lower delay, P2P - Suitable for residential use	-Needs a router/firewall support UPnP - Don't work with cascaded NAT - Opens a pinhole in firewall/router - Not suitable for corporate use
STUN	- Gathers Information about the type of NAT - Lower delay, P2P - No change configure on Firewall.	- Doesn't work for symmetric NAT - Needs a STUN server and client - Doesn't work if both clients are behind NATs
ALG	-No configuration on client side - Doesn't create a security hole - Lower delay, P2P	- Not many firewall support this - Requires a high performance statefull firewall
TURN	- Supports all types of NAT - No change of infrastructure	- Large delay due to RTP Relay - Not many available TURN clients or servers
SBC	- Support all types of NAT -No configuration on client side	- Expensive - Large delay due to RTP Proxy

Table 7. Comparison of various firewall/NAT solutions

[1] www.skype.com

4 Method and Implementation

This chapter describes secure WIDER solution, the architecture of our VoIP/PoC client, and the design and implementation of a secure VoIP system.

4.1 Secure WIDER network

The WIDER network is the field network that provides shared local communications and Internet access in a disaster and emergency response area. The WIDER network is divided into two parts: WIDER core and WIDER camp. WIDER core connects to several relief organization head offices via satellite links and connects to WIDER camps via point-to-multipoint wireless links. The WIDER service provides shared communication including: Instant Messaging, VoIP, Voice Conferencing, Bulletin Board, Web services, FTP, and Proxy. Nodes at the WIDER camp access WIDER services and the Internet via Access Points or Switches that connects to the point-to-multipoint wireless links. Because of the sensitive information to be communicated regarding transport and logistics, the WIDER network needs to be well secured.

Such a secure WIDER network includes secures three parts of wireless link:

1. *Securing access network from nodes to/from the Access Point*
2. *Securing the point-to-multipoint wireless link*
3. *Securing the satellite link.*

Previously WIDER has used IEEE 802.1x EAP-TLS authentication and WEP encryption in the access network. The local wireless links use WEP encryption. There was only a NAT box and no firewall protecting WIDER internal network. The assumption was to leave security functions to the satellite operator for the connection to the satellite. Figure 24 shows this prior WIDER solution.

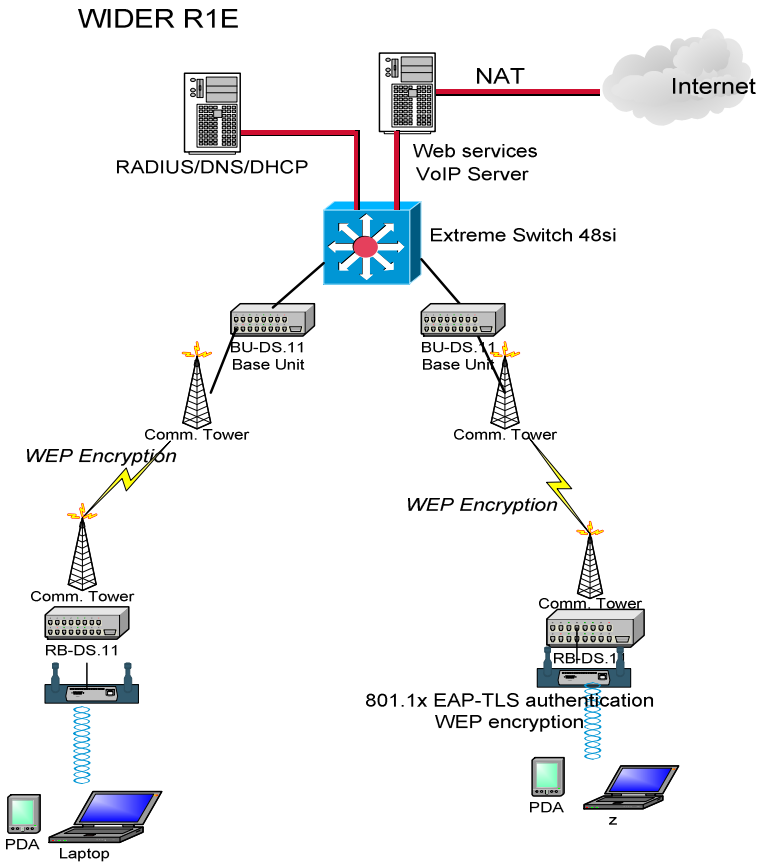


Figure 24. WIDER solution version 1E

We have worked on re-design WIDER to be more secure, more robust and, and to better support mobility. This including:

- Adding hardware firewall to protect WIDER while connected to a satellite link.
- Configure it to enable SIP/VoIP traversal over NAT/Firewall.
- Separate VLAN traffic for voice and data to enable QoS based on a VLAN, thus providing specific QoS by type of traffic (voice or data).
- Configuring a VPN to support dial-up VPN and an Office2Camp VPN.
- Enable EAP-TLS and EAP-TTLS to support both mobility and easy of login with username/password.

Figure 25 show this enhanced WIDER solution.

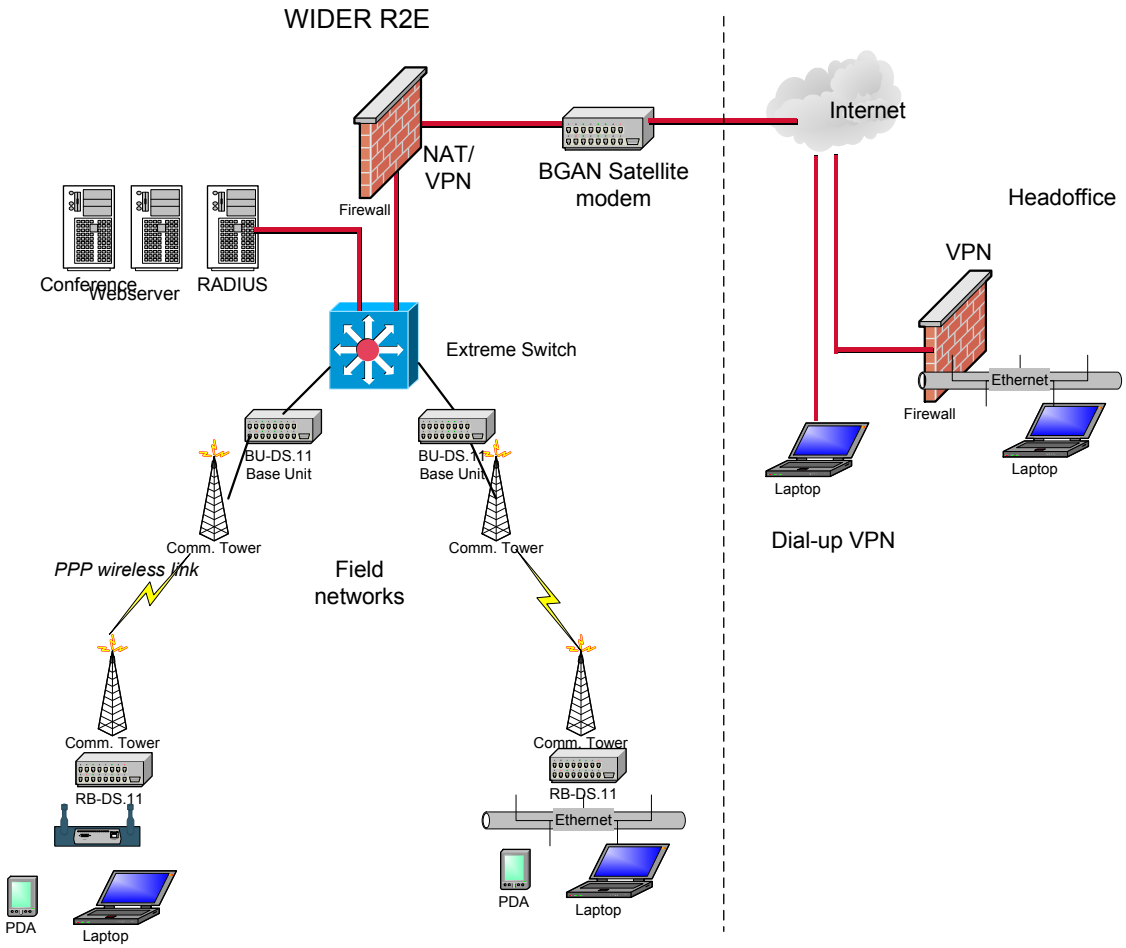


Figure 25. Current WIDER solution

We use Juniper's Netscreen204 firewall that has a built-in ALG. As described in section 3.5.4, the ALG parses each SIP message to learn which ports will be used for media transmission; it then creates a pinhole for this SIP session. The following information is needed by the SIP ALG to create such a pinhole.

– Destination IP: The parser extracts the destination IP address from the c= field in the media or session level.

– Destination port: The parser extracts the destination port number for RTP traffic from the m= field and calculates the destination port number for RTCP as: *RTP port number + one.*

– Lifetime: This value indicates the length of time (in seconds), during which a pinhole is open to allow a packet through. A packet may go through the pinhole during its lifetime. When the lifetime expires, the SIP ALG removes the pinhole.

After a packet goes through the pinhole within the lifetime period, the SIP ALG removes the pinhole for the direction from which the packet came.

Figure 26 illustrates a call setup between two SIP clients and SIP ALG creates pinholes to allow RTP and RTCP traffic.

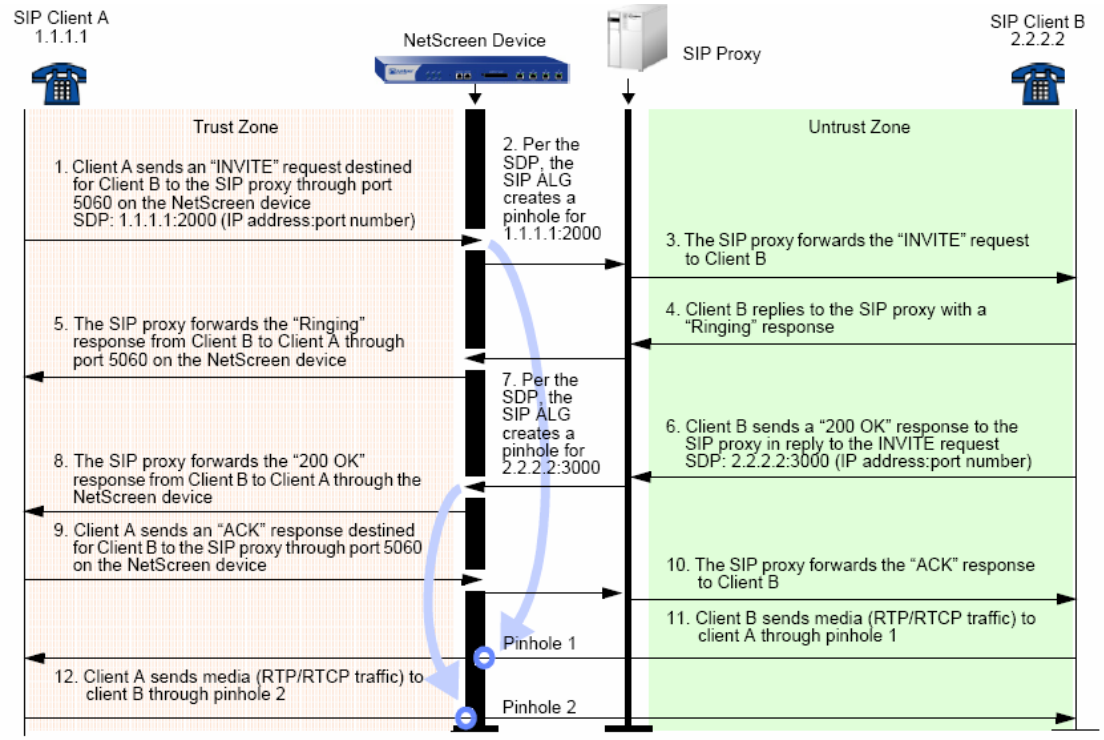


Figure 26. Call setup established via Netscreen Firewall

We make an assumption that VoWiFi equipment will probably be used in WIDER in the near future, so a separation between data and voice is necessary (as recommend by NIST[1] [NIST 800-58]). VLAN is a good solution since we can keep a high security level and increase QoS priority for voice traffic. We consider each relief organization will use one VLAN. Specifically VLAN uses only for voice via a VoWIFI phone. The RADIUS server controls authentication and authorization. The user account for this VLAN is saved in database (MySQL) and a flatfile (text file) at the RADIUS server. Users log in to the WIDER system via port-based authentication 802.1x in EAP-TLS and EAP-TTLS. After successful authentication, the user receives a VLAN IP address issued from a DHCP server. Because of central authentication, users in one relief organization can utilize the same VLAN even while moving around different camps.

We implemented self-signed certificates for port-based authentication. Each user needs to install a copy of the server certificate and user certificate in case using 802.1x EAP-TLS and download a server certificate when using 802.1x EAP-TTLS.

[1] NIST: National Institute of Standards and Technology

Microsoft's Windows XP, Windows 2000, Windows Mobile 2003 (*default*) supports EAP-TLS. To enable EAP-TTLS, users can install free, open source SecureW2 client (www.securew2.com). Most of latest VoWIFI phones also support 802.1x.

VPNs are used by people in the home office to access the field network. This helps the home office latest update from the field. For example, a home office user can remotely participate in conference or access a database of missing people. We configured a dial-up VPN using IPsec mode ESP, using shared IKE ID. Site-to-site VPNs are not yet implemented due to lacking of firewall equipment to test.

Currently local and point-to-multipoint wireless links are encrypted using WEP. As described in section 3.1.4, WEP is insecure. In the future, wireless equipment should be updated to support WPA or 802.1i.

Beyond set up and configuring secure WIDER solution, I have developed and implemented a Secure VoIP client. Thus the WIDER solution includes a complete secure voice and data network.

4.2 Secure VoIP client

4.2.1 Platform

Relief workers in a disaster area need a simple, fastest and reliable VoIP that can help them effectively communication. We searched for an open source client that could make it easy for end-user. However, there was not an open source VoIP client running on the Microsoft's Windows operating systems that most end-users would require.

Therefore, we decided to select the SleIPner VoIP/PoC test client. SleIPner is a test client from Ericsson's PU IMS department. It runs on Windows and is a fully functioned as VoIP client. Moreover, it includes a Push-To-Talk feature that can alleviate the issue of delay while communicating over long distances such as satellite link.

The Sleipner architecture bases on a SIP stack called eSIP. The eSIP stack is a cross-platform library that developed by Ericsson to support applications using the SIP/IP core. Figure 27 illustrates the eSIP stack architecture.

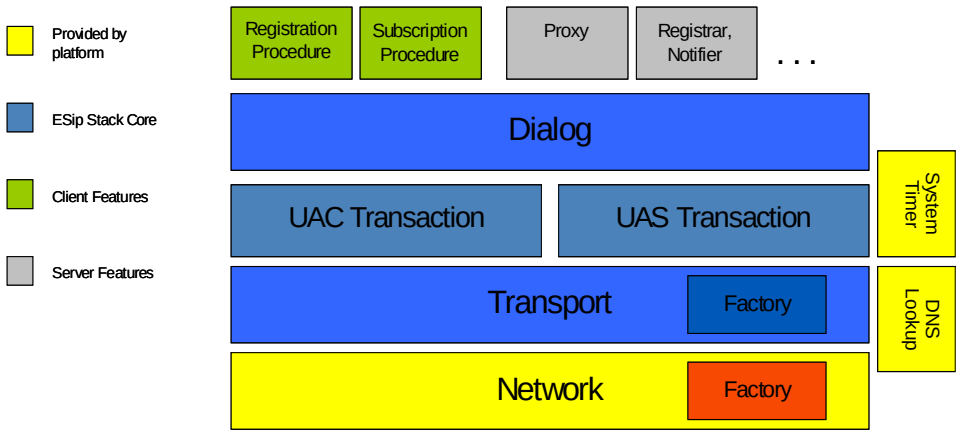


Figure 27. ESip architect

eSIP Stack is divided into four layers as defined in RFC 3261: network layer, transport layer, transaction layer, and dialog layer. Between each layer, an abstract interface is implemented so that different implementation can be added to the stack. For example, Sigcomp could be added between the Network and Transport layers.

Sleipner architecture also consists of four layers: Graphical user interface components (GUI), control layer, proxy layer, and eSIP stack wrapper layer. Figure 28 shows the high level architecture of SleIPner client

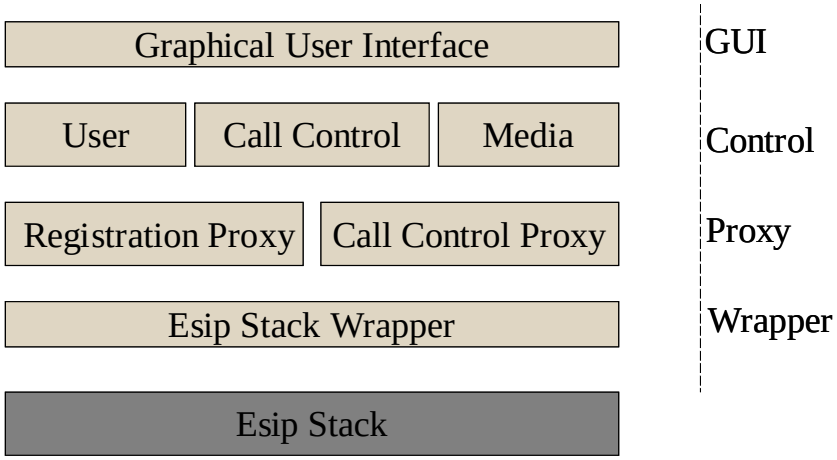


Figure 28. High level architect of SleIPner client

The GUI receives and sends message to control graphical output layer over the Windows Message Queue. The control layer handles both a VoIP controller and a PoC controller. These controllers use their own media resources for communication and activate their media through the media manager class. Figure 29 shows the design of a module controller for a SleIPner client.

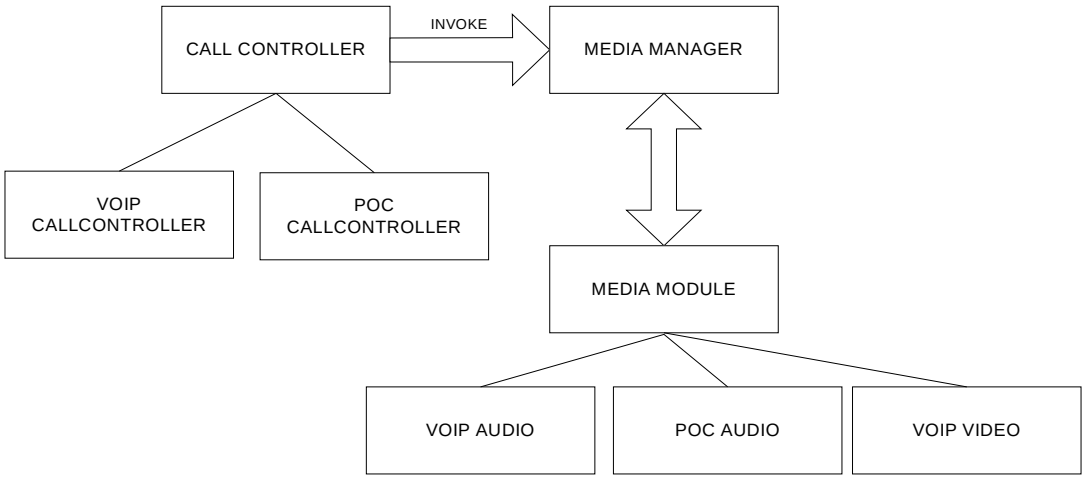


Figure 29. Media controller architect

The eSIP stack wrapper provides a wrapper function for the eSIP stack. The proxy layer provides the dynamic handling of different service types, by forwarding messages to the correct Call Control function.

4.2.2 Secure VoIP design

As mentioned in the previous chapter, secure VoIP includes secure signaling, secure media, and a mechanism for key exchange between two peers. After researching available on the market SIP servers, we have seen that most SIP servers do not support TLS. This makes it difficult to implement secure signaling. Our VoIP implementation would be vulnerable if we used SDP Security Description to carry key authentication in a SIP message without protected signaling. For that reason, the implementation was changed to use Secure RTP using MIKEY in order to secure the VoIP client. In addition, I designed four security options so that the end-user could customize their security level.

The lowest level is no security for VoIP. This makes it possible to call any other SIP-based VoIP client. The second option has built-in security function, where the master key, the salt, and the MKI are pre-defined and generated by a strong random number. Information about using built-in security functions is carried by SDP message with the new attribute: a=encryption:default. The advantage of this feature is that it provides security between SleIPner clients. Thus the SleIPner user does not need to know share-secret or exchange certificate before making a secure call. The disadvantage is that it is not highly secure since the master key is fixed, and it cannot communicate with other client. The third and fourth options use MIKEY and SRTP

with a shared-secret and a public key. It provides end-to-end VoIP security if both peers have a shared-secret or certificates. One disadvantage of a shared-secret is that if you have different shared-secret with different peers, then the end-user will not know who will call them with a correct shared-secret key. A warning box is displayed that tells the callee about the caller's identification. The callee then can know who is calling in order to use the appropriate shared-secret key. Public-.key will not have same problem as a shared-secret. However, in public key, PKI is required to certify that the embedded certificate in SIP is correct. The implementation simplifies this by having the client automatically generate a certificate when it executes for the first time. A certificate embedded in a SIP message may result in a large message (i.e. more than 1440 bytes) that causes SIP over UDP fragmentation. Figure 30 illustrates the security options in SleIPner that users can select.

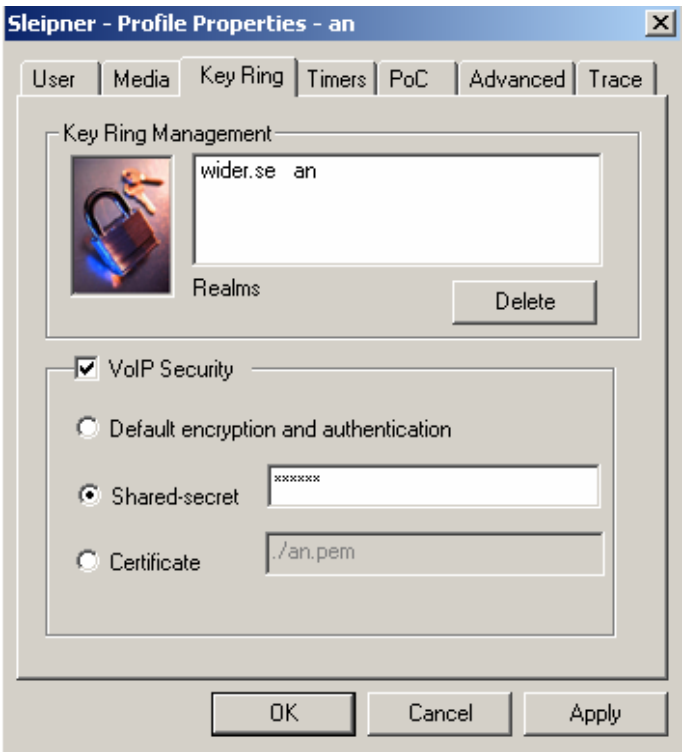


Figure 30. Security option in Secure SleIPner

The Ericsson Research's SRTP- MIKEY library has been modified to integrate with SleIPner. Figure 31 shows the security classes of SleIPner. Two classes were added to eSIP stack: SrtpPacket and CryptoContext. SrtpPacket is derived from RtpPacket class purposely encrypt RTP packets and add an authentication tag. CryptoContext is the class that provides a basic crypto context to the SrtpPacket class to encrypt packets. VoIpAudioModule is the class inherited from

CMediaModule that handles VoIP media sessions. VoIPsec class is responsible all security functionality. It adds a MIKEY message to SDP when the VoIPAudioModule generates SDP. The VoIPsec class is also used to encrypt and decrypt the RTP streams in the VoIPMedia class. CMediaManager queries the VoIPsec class in the SDP Negotiation class to authorize the MIKEY message and send back a MIKEY response.

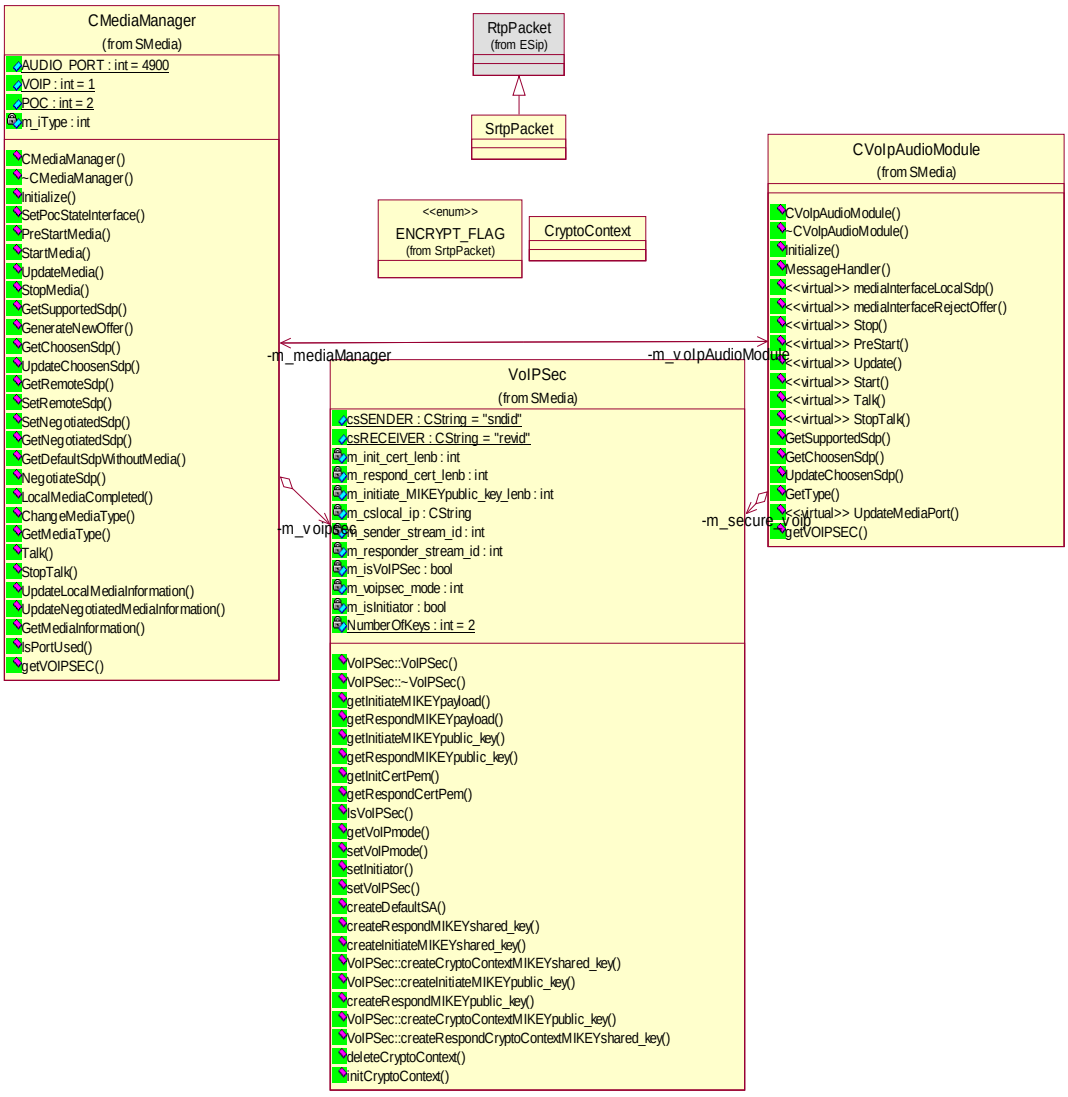


Figure 31. Secure SIeIPner class design

4.2.3 SIP and MIKEY

MIKEY is carried in a SIP message following the guideline in the draft "Key management extensions for Session Description Protocol (SDP) and Real-time Streaming Protocol (RTSP)" [draft-kmgmt-ext]. A caller packages a MIKEY message in the INVITE message with attribute: a= keymgmt: mikey. The MIKEY message can be part of a session where SRTP protects the all streams or simply at the media level where SRTP protects a single media session. To prevent downgrading attacks by SDP negotiation, we do not allow the SleIPner client to generate a media offer with both security and non-security. End-users have to explicitly select a non-secure option if they want to participate in non-secure calls. Re-keying is not yet supported in our implementation; however, it should be included in any re-INVITE messages.

4.2.4 Interaction with non-secure VoIP client

If a secure VoIP client originates a call to a non-secure VoIP client, then the non-secure VoIP client will not understand the attribute a= keymgmt: mikey . As defined in SDP [RFC3237], a non-secure client will ignore the unknown attribute; and will send back 200 OK message without the keymgmt attribute and an optional warning code 306 "Attribute not understood". Since there is no standard for handling this event, and for compatiblity with receiving 200 OK messages, we propose the call should continue with the caller sending an ACK message.

If non-secure client is the caller, then a secure client after parsing the SDP will know that the call is un-secure. We use a warning box to allow the secure client to select a non-secure option. Thus the secure client can reject or accept the call. If the callee chooses reject, the non-secure client will not know why the call is rejected. Therefore, we issue a CANCEL message with the 305 Warning: "Incompatible media format". If the secure client accepts the call, then he knows that the call is not secure and must evaluate his own risk.

4.2.5 Implementation issues

The MIKEY message is encoded in Base64[1] and sent in the SIP INVITE. The MIKEY message is an attribute in the SDP body. If the MIKEY message has a line feed return (in Linux "\n", Windows CRLF), then the SDP parse engine will

[1] Base64: A data encoding scheme whereby binary encoded data is converted to printable ASCII characters.

consider MIKEY payload only upto the line feed and skip the rest of MIKEY body. In my implementation, I have seen that in some cases MIKEY includes "\n" or CRLF . For example, certificate stores in PKCS#8 with file structure:

"—BEGIN ENCRYPTED PRIVATE KEY---" CRLF

<encrypted_key> CRLF

"---END ENCRYPTED PRIVATE KEY"

Because of this, the authentication will fail. The program recognizes it as an incorrect shared-secret or certificates in MIKEY message. I have made a temporary solution by removing line feed(s) before encoding the text into base64.

4.2.6 Further in Secure VoIP

Secure VoIP is still not widely deployment. One of the main reasons is that although SRTP is mature enough; there are many different ways of handling key management to make interoperability difficult. MIKEY seems to be the right choice to provide end-to-end security, but still needs time to build a market.

Lack of demand for Secure VoIP and concerns for lawful intercept (VoIP to be considered under wiretap laws[1]) are additional reasons limiting Secure VoIP deployment.

From the end-user's point of view, difficulties configuration and NAT/Firewall traversal make not only Secure VoIP but even non-secure SIP-based VoIP grow slower than Skype's propriety internet phone.

I propose a Secure VoIP solution that reduces the computational resources needed to allow VoIP terminals to use machine with callee limited resources. If we assume authorized users are trusted users, callee can block caller from the block list that it could send to its SIP proxy. Additionally, UAS can accept or reject the calls from UAC manually. Secure VoIP protects signaling and media. If signaling between two user agents is protected by TLS, then SIP server could generate a Security Association (SA) and send both user agents the same session key, session salt, etc via the SDP Security description method [draft-sdp-descriptions]. If SIP signaling is not protected, then the SIP server and client could use a shared secret key (that used for clients to REGISTER). MIKEY could be used to set up a session key between user agents and the SIP server. The SIP server then can send in encrypted form the same session key and salt to both clients so that both clients

[1] http://itmanagement.earthweb.com/erp/article.php/3390671

can initiate a secure call. This method reduces the computational resources needed for public key calculation and alleviates the configuration on ther client side. However, this method requires both client and server support.

In the future, Host Identity Protocol (HIP)[1] could replace the IP address of the UA in SIP. Then SIP signaling would exchange Host Identities (HI)/Host Identity Tags (HIT) between the two user agents. Following this they can provide end-to-end secure communications using by HIP/SRTP.

[1] http://www.ietf.org/html.charters/hip-charter.html

5 Testing, Measurement and Evaluation

This chapter describes and evaluates our measurements. The secure client purposely uses for relief workers in disaster areas to communicate within their camp or to their home office. Our testing includes measuring the performance of VoIP over WIDER, including the VoIP delay over satellite networks.

5.1 VoIP Performance in WIDER

The goal of this measurement is to evaluate the capacity of the WIDER solution in terms of the number of simultaneous VoIP. Due to the limited equipment available, we only tested the performance and QoS when using a single access point. Performance testing of a wireless bridge should be implemented in the future.

Two laptops connect to the WIDER core via 1 Cisco's AP1100 access point. Figure 32 shows the testbed.

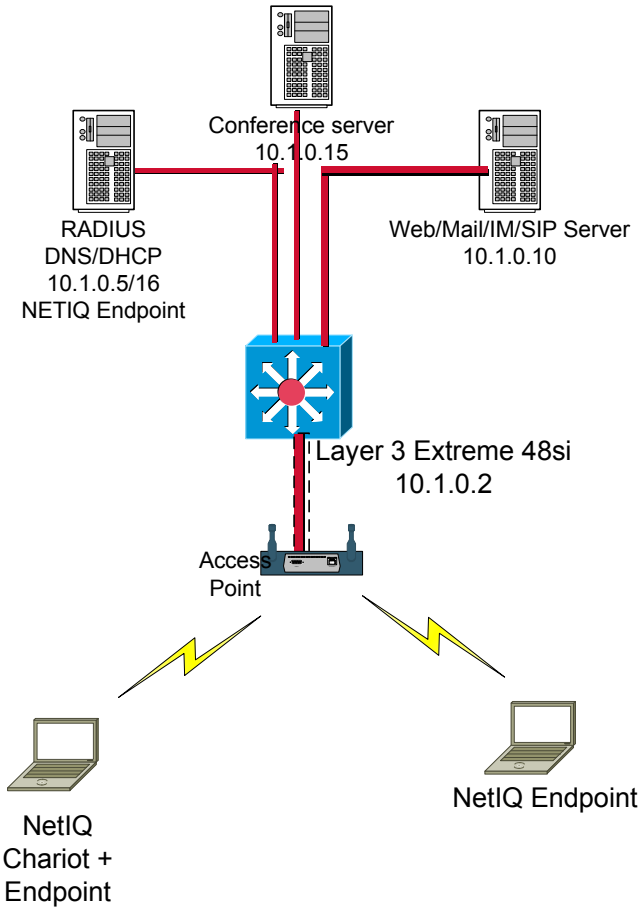


Figure 32. WIDER testbed for performance measurements

The two laptops are follows configured :

Compaq Presario 2500	HP NC8000
- Pentium 4 2.4 Ghz - 512 Mb Memory - Windows XP Operating System - Proxim Gold WLAN card -NetIQ Chariot Assessor and NetIQ endpoint	- Pentium 4 Mobile 1.6 Ghz - 512 Mb Memory - Windows 2000 Operation System - DLink DWL-650 WLAN card - NetIQ Endpoint

Table 8. Laptop configuration

The software used to measure performance was NetIQ's Chariot Assessor (www.netiq.com). NetIQ's Chariot Assessor emulates VoIP traffic, collects key call quality measurements, and analyzes the results. NetIQ contains two components: a VoIP NetIQ Assessor and NetIQ endpoints. The VoIP Assessor is a server that creates the Test Case (TC). Each TC is then compiled into a script and

sent to the NetIQ endpoints. The NetIQ endpoints use these scripts to generate VoIP flows as the real application could do. The TCs specify the types of CODEC types, source IP address, destination IP address, number of concurrent calls, jitter buffer size, call duration, and total measurement time, etc. The VoIP Assessor receives data from each NetIQ endpoint periodically and finally calculates the Call Quality based on a Mean Opinion Score (MOS). [ITU-T P800]. MOS standard as defined by ITU recommendation P800 describes how human would scores the audio. The listener grades the audio as they hear it due different aspects of delay or datagram loss. MOS ranges from 1 to 5, where MOS of 5 is excellent and 1 is unacceptably bad. Table 9 summarizes the relation between the MOS and user satisfaction [ITU G.107].

Mean Opinion Score (lower limit)	**User Satisfaction**
4.34	Very satisfied
4.03	Satisfied
3.60	Some users dissatisfied
3.10	Many users dissatisfied
2.58	Nearly all users dissatisfied

Table 9. MOS and user satisfaction

The test emulates simultaneous voice sessions between two laptops. The scenario is:

- Two computers simulate from 15 to 20 concurrent calls, each using a G723.1-ACELP CODEC
- Each VoIP call lasts random for a duration of from 1 to 2 minutes.
- Total test time: 7-14 hours
- Jitter buffer: 50 ms, no silence suppression, no service quality.
- Delay between datagrams: 30 ms

The two laptops are 25m distant from the access point. The access point uses channel 1 and has a transmitter power of 100mW. Proxim wireless cards have a wireless tool that can log signal level, noise level and, signal to noise ratio. Table 10 is the average signal parameters during the tests.

Signal level (dB)	Noise level (dB)	Signal to Noise Ratio (dB)
-59, 5	-91,5	32

Table 10. Signal level at receiver side

During the testing, we increase the total concurrent calls from 10 to 20 calls using the CODEC G723.1-ACELP. This CODEC produces has data at the rate 5.3 kbps. However, the actual required bandwith for two-way communication is higher than it seems to be. Table 11 lists bandwidth as calculated by NetIQ Assesment. [NetIQ]

CODEC	Data Rate	Datagram size	Packetization Delay	Combined Bandwith (2 flows)	Default jitter buffer	Theoretical Maximum MOS
G.711u	64kbps	20ms	1ms	174.4kbps	40ms	4.40
G.711a	64 kbps	20ms	1ms	174.4kbps	40ms	4.40
G.729	8kbps	20ms	25ms	62.4kbps	40ms	4.22
G723.1 MMMLQ	6.3kbps	30ms	67.5ms	43.73kbps	60ms	3.87
G723.1 ACELP	5.3kbps	30ms	67.5ms	41.60kbps	60ms	3.69

Table 11. Practical bandwidth and MOS for CODECS

Table 11 is grade of these QoS value.

Measurement	**Good**	**Acceptable**	**Poor**
MOS	Above 4.03	4.03 to 3.60	Below 3.60
Delay (ms)	below 150	150 to 400	above 400
Jitter (ms)	below 40	40 to 60	above 60
Lost Data (%)	below 0.50	0.50 to 1.00	above 1.00

Table 12. Grade of QoS value

Figure 33 is the result of a number of total call summaries.

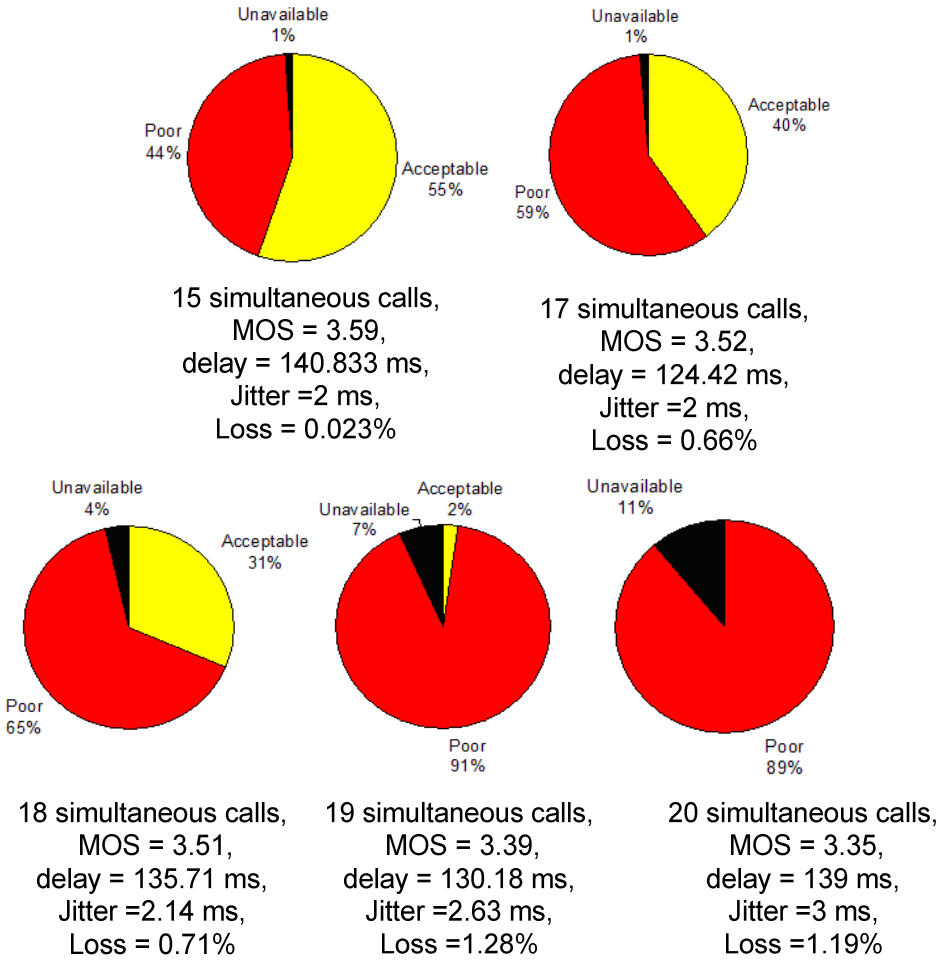
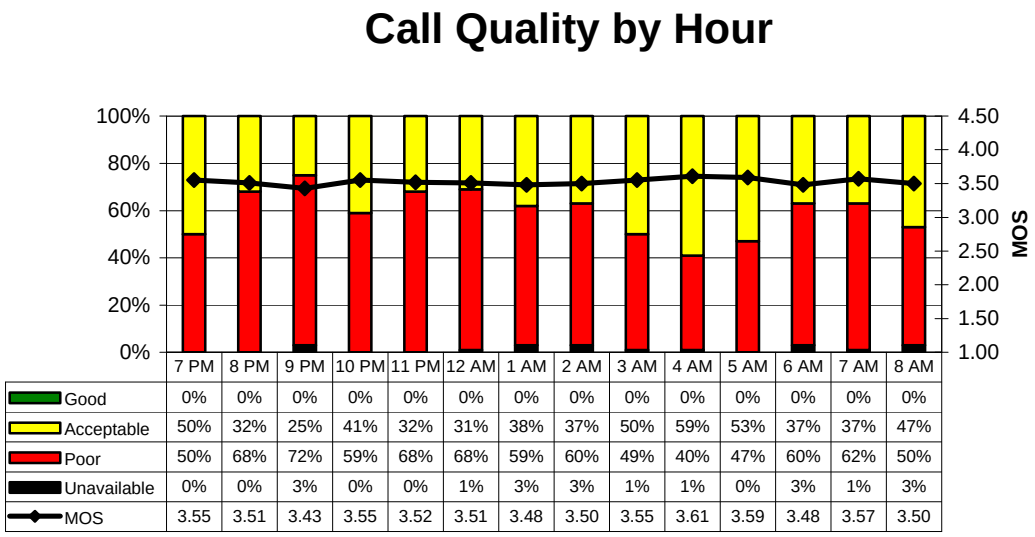


Figure 33. Call quality summary

The measurement results showed that even in ideal environments without competing data traffic, each access point only able to handle up to 18 concurrent calls with a G723.1-ACELP CODEC. Users perceived VoIP QoS goes down quickly, as shown in the case of 19 concurrent calls only 2% of these calls are acceptable. Unavailable means the calls failed or could not be connected. In the cases of 19 and 20 simultaneous calls, there are 7% and 11% unavailable respectively, meaning almost one in 10 calls are failed respectively. Other CODECs (G711, G729 etc) require more bandwidth, hence should support a lower number of concurrent calls. We could test using the AMR CODEC since NetIQ does not support this CODEC. The AMR CODEC has a variable bit rate with 8 narrowband codec modes: 4.75kbps, 5.5kbps, 5.9 kbps, 6.7 kbps, 7.4 kbps, 7.95 kbps, 10.2 kbps, and 12.2 kbps. We expect that the QoS of AMR is QoS better than G723.1-ACELP because it automatically changes its data rate depending on the available bandwidth

Additionally, we also measure call quality, delay, jitter, and packet loss *per hour* to estimate the stability of system.

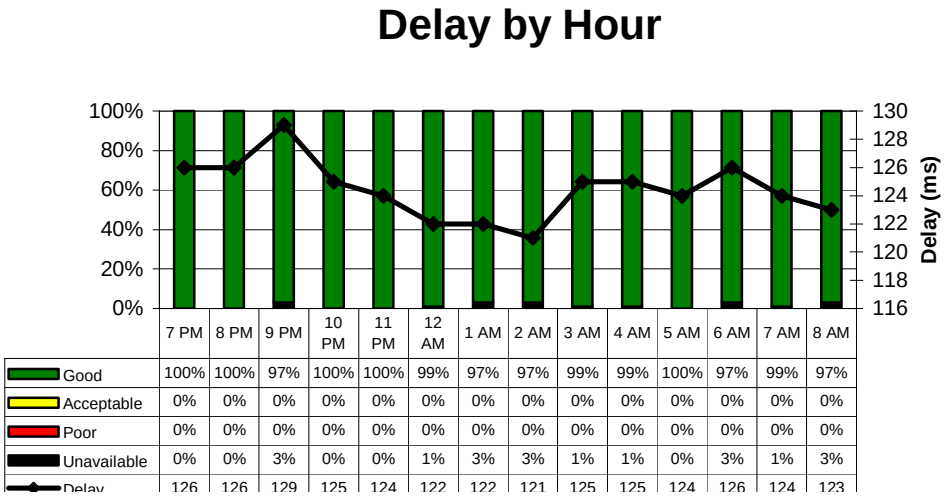
Figure 34 shows the average MOS per hour (over 13 hours) for 17 concurrent calls. The bar graph evaluates each hour's MOS values according to the MOS result ranges defined for the assessment and shows the number of Good, Acceptable, Poor, and Unavailable calls for each hour



	7 PM	8 PM	9 PM	10 PM	11 PM	12 AM	1 AM	2 AM	3 AM	4 AM	5 AM	6 AM	7 AM	8 AM
Good	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
Acceptable	50%	32%	25%	41%	32%	31%	38%	37%	50%	59%	53%	37%	37%	47%
Poor	50%	68%	72%	59%	68%	68%	59%	60%	49%	40%	47%	60%	62%	50%
Unavailable	0%	0%	3%	0%	0%	1%	3%	3%	1%	1%	0%	3%	1%	3%
MOS	3.55	3.51	3.43	3.55	3.52	3.51	3.48	3.50	3.55	3.61	3.59	3.48	3.57	3.50

Figure 34. Call quality by hour

Figure 35, 36, and 37 show result the average delay, jitter, and lost packet per hour (for the same 13 hours) for 17 concurrent calls. The bar graph evaluates each hour's delay values according to the delay ranges defined for the assessment and shows the number of Good, Acceptable, Poor, and Unavailable calls for each hour.



	7 PM	8 PM	9 PM	10 PM	11 PM	12 AM	1 AM	2 AM	3 AM	4 AM	5 AM	6 AM	7 AM	8 AM
Good	100%	100%	97%	100%	100%	99%	97%	97%	99%	99%	100%	97%	99%	97%
Acceptable	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
Poor	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
Unavailable	0%	0%	3%	0%	0%	1%	3%	3%	1%	1%	0%	3%	1%	3%
Delay	126	126	129	125	124	122	122	121	125	125	124	126	124	123

Figure 35. Delay by hour

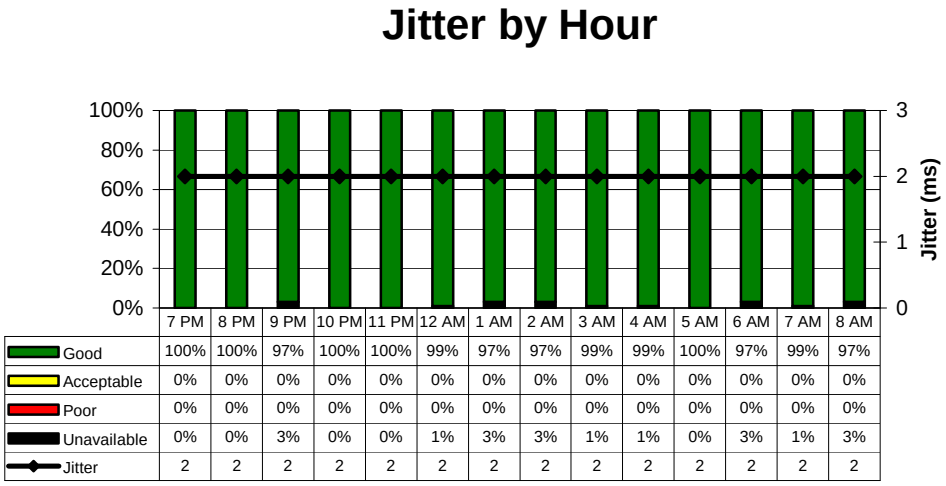


Figure 36. Jitter by hour

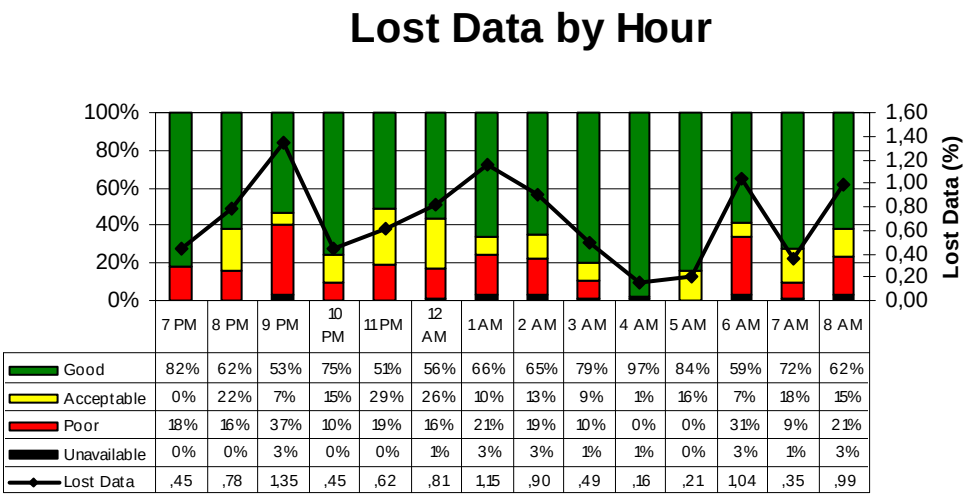


Figure 37. Lost data by hour

We go analyze in detail the changing voice quality per hour. From the Figure 36, we conclude that jitter does not have much of an effect on voice calls in one access point. Probably the jitter buffer of 50ms is enough to handle packet interval time. This effect on QoS for long distance transmission is measured for VoIP over satellite calls. Figure 35 is the delay with averages from 120ms to 130ms. Packet loss varies over time with value from 0.16% to 1.35%. Packet loss varies because of the IEEE 802.11 CSMA/CA medium access scheme. A collision occurs when two laptops attempt to transmit packets simultaneously. All of the above quantity values are within the good or acceptable ranges. That means the choice of CODEC is an important contribution to the observed QoS. Figure 38 shows the NetIQ calculation of factors affecting call quality.

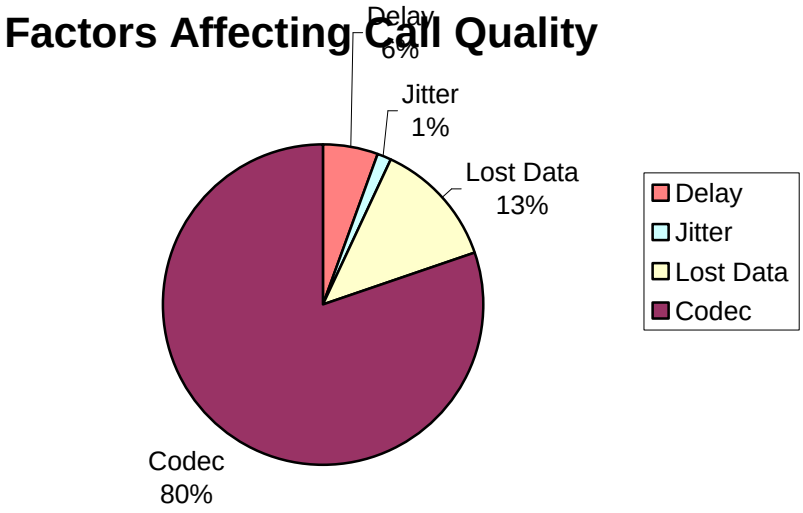


Figure 38. Factors affecting call quality

I was surprised with these measurement results. The 802.11b access point should carry 11Mbps/44.6kbps = 246 concurrent G723.1-ACELP calls. What prevents us from achieving this capacity for this access point?

If we calculate the overhead of voice packet, the IP/UDP/RTP header is 40 bytes. Its transmission time is 40*(8/11) Mbps = 29 µs. G723-ACELP payload is 20-24 bytes and its transmission time is 24*(8/11) Mbps = 17 µs. The MAC header overhead is 34 bytes requiring 34*(8/11) = 25 µs. However, the 802.11 MAC/PHY round trip transmission is more than 800 µs due to the physical preamble, the MAC backoff time, the MAC ACK, and the intertranmission times of both the packet and the acknowledgement.

The 802.11b standard provides two modes of MAC operation: mandatory Distributed Coordination Function (DCF) and an optional Point Coordination Function (PCF) mode. However, most commercial access points support DCF, as PCF is not always interoperable and does not effectively allocate bandwidth. DCF, on the other hand, is very *ineffective* in handling voice traffic. The DCF protocol is based on CSMA/CA, where stations must determine that the medium is idle before transmitting. The DCF mode specifies two types of Inter Frame Spacing (IFS), including the Distributed IFS (DIFS) and Short IFS (SIFS). Every station that needs to send a packet first senses the channel for at least duration of DIFS (50 ms). If the medium is determined to be free for duration of a DIFS, then the station transmits the packet. Otherwise, it enters the backoff phase in which it

chooses a random backoff timer uniformly from a collection of values known as the Contention Window (CW). The standard specifies a CW from 32 to 1024 Time Slots (TSs), with TS = 20 µs. After a backoff time has been chosen, then the station continues to monitor the medium until it observes an idle period equal to a DIFS. Then it decreases the backoff timer after every idle timeslot. If the medium becomes busy during the countdown, then stations suspends the decrement operation until the channel is idle (a period of DIFS). When the backoff timer reaches zero, the station transmits the packets. After transmission, the sender station expects to receive an ACK within the SIFS period (10 µs). If an ACK is not received within this period, then the packet is assumed to be lost. CW then doubles its duration until it reaches its maximum value. If a successful transmission occurs, CW is reset to its minimum value. The sender station may attempt to retransmit the packet up to a maximum number of times. Table 13 lists the constant parameters in 802.11a, 802.11b, and 802.11g.

Parameter	**802.11b**	**802.11a**	**802.11g**
SLOT	20µs	9µs	9µs
SIFS	10µs	16µs	10µs
DIFS (SIFS + 2xSLOT)	50µs	34µs	28µs
Physical Layer Header length	192µs	20µs	20µ
Min. mandatory data	1Mbps	6Mbps	6Mbps
ACK packet size	14 Bytes	14 Bytes	14 Bytes
CW (min-max)	31 – 1023	15-1023	15-1023
Signal extension	N/A	N/A	6µs
MAC header	34 bytes	34 bytes	34 bytes

Table 13. Constant parameter in access point

Note that the PHY header takes time because it is transmitted at 1 Mbps. The ACK frame is transmitted at the basic rate of 2Mbps regardless of the data rate and takes 14*(8/2) Mbps = 56 µs. The ACK package its PHY header so that it takes a total of 56 + 248 = 248 µs. Now we can calculate the possible support VoIP stream by an 802.11b access point. We define N as the maximum number of sessions that can be supported. Supposed that two way communication requires 2*N streams. T_{avg} is average time between two consecutive packets in the WLAN.

For simplicity, we ignore the collision and increases in backoff time. F_{pkt} is number of packets sent by one VoIP stream per second. We have:

$1/T_{avg} = 2N * F_{pkt}$ with F_{pkt} = Codec_rate / (payload * 8)

The packet transmission overhead is:

$T_{OH} = T_{Payload} + T_{RTP} + T_{UDP} + T_{IP} + T_{MAC.}$

= (Payload + RTP + UDP + IP + MAC header)*8/dataRate

Due to the CSMA/CA scheme, the additional duration needed at the sender's side:

$T_{send} = T_{DIFS} + T_{averageCW} + T_{PHY}$

If we ignore collisions, then the average Contention Window (CW) is

Average CW = (CWmin – 1)/2

$T_{averageCW}$ = slotTime * (CWmin -1)/2

A successful transmission with an ACK has the following overhead:

$T_{answer} = T_{SIFS} + T_{ACK}$

So, the total transmission time is:

$T = T_{OH} + T_{send} + T_{answer} = T_{avg} = 1/(2N * F_{pkt})$

Then:

$N= 1/(2 * (T_{OH} + T_{send} + T_{answer}) * F_{pkt})$

With 802.11b and G723.1 codecs, datarate = 11 Mbps, and payload = 20 byte, we find that:

$F_{pkt} = 5,3.10^3/(20*8) = 33$

T_{answer} = 10 + 248 =258 µs

T_{send} = 50 + 20*(31-1)/2 + 192 = 542 µs

$T_{OH} = (20 + 8 + 12 + 20 + 34)*8 /11.10^6 = 68$ µs

So:

N = 18,9.

The result is vary similar to our measurement.

5.2 Delay measurements of VoIP over a satellite link

A Regional BGAN (RBGAN) satellite modem is used to connect with the WIDER core network. RBGAN is ultra-portable satellite equipment from Inmarsat[1]. Inmarsat has an operational regional broadband access system via Thuraya satellite[2] based on

[1] www.inmarsat.com
[2] www.thuraya.com

ETSI[1] GMR-1 release 2 [ETSI TS 101 367-3-1]. RBGAN protocols are based on GSM/GPRS that provides data rates up to 144 kbps. RGBAN connects to a geo-stationary satellite using the L-band.

In our testbed, we connect the VPN via satellite to measure the delay of the VoIP call and Secure VoIP call over a satellite link. Because our R-BGAN link does not support a public IP address, it is not possible for the VPN to directly serve the WIDER system (i.e. RBGAN needs an update service to get a public IP address). We construct a VPN connection via Ericsson network. Two laptops with an installed SleIPner secure client, a dial-up VPN client is installed in the computer that connects to the satellite modem. Figure 39 shows our satellite testbed.

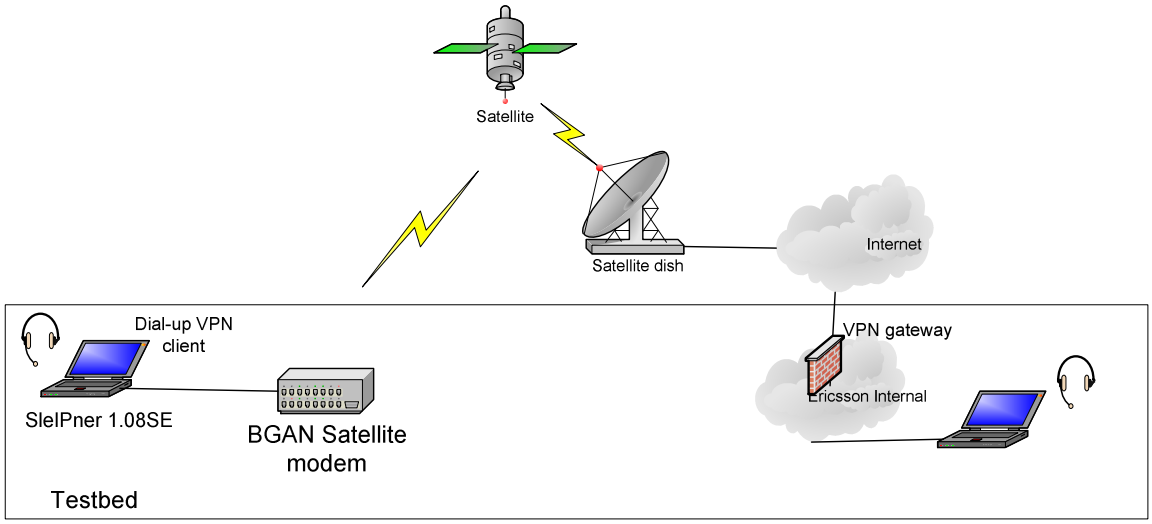


Figure 39. Voice over satellite testbed

The two laptops have following configuration:

HP NC8000	HP NC6000
- Pentium 4 Mobile 1.6 Ghz - 512 Mb Memory - Windows 2000 Operation System - SleIPner_receiver 1.08 SE - OS Non-Proxy Atomic Syn 2.5 - Ericsson SIP Server	- Pentium 4 Mobile 1.8 Ghz - 512 Mb Memory - Windows 2000 Operating System - SleIPner_sender 1.08 SE - OS Non-Proxy Atomic Syns 2.5 - RADCOM Netsafe VPN client

Table 14. Laptop configuration for Voice over satellite testing

[1] ETSI: European Telecommunication Standards Institute

Each laptop installs the OS Non-Proxy Atomic Syn to synchronize the time with time server (nist.time.gov). RADCOM Netsafe VPN client is a dial-up VPN client that connects to the company site. VPN client uses IPsec ESP mode.

SleIPner client was modified so that it logged time of each RTP packet with a resolution of 1µs. In this test, both sides concurrently send RTP/SRTP packets. A consistent test can be done by repetitively sending a wave file. To reduce the log file processing, I only utilize a log in one direction of sending or receiving. I defined the SleIPner_sender logged voice packets so that it sends the time when each RTP packet is prepared to be encrypted to a SRTP packet . On the receiver side, the SleIPner_receiver logs the time after receiving SRTP packets and decrypting them into RTP packets. For non-secure calls, the log time on both sides indicates the time before sending and after receiving RTP packets.

To reduce the actually "RINGING" time and waits for a callee to pick up the call, I modified SleIPner to support auto-answer mode so that it immediately sends a SIP message 200 OK upon receiving the first INVITE.

Before the test, we measure the R-BGAN bandwidth capacity. According to the R-BGAN specification, it supports maximum of 144 kbps in shared channel. We have tested this bandwidth by connecting to the ZDNet[1] (www.zdnet.com) and CNET[2] (www.cnet.com) bandwidth meter test service. After 10 samples, we saw that the available bandwidth is around 54-76 kbps. This is much lower than the maximum R-BGAN capacity. However, this bandwidth is enough for our test as it has only a single two-way voice call.

We have done some trials with non-secure calls and secure calls. Figure 40-47 are the results that we measured with normal non-secure call and secure calls.

Type of calls:	**Un-secure (RTP), sender terminate**
Sender call duration:	53. 927 s (12:58:04.521 –12:58:58.448)
Receiver call duration:	53.977 s (12:58:04.985 - 12:58:04.962)
Average delay time:	731 ms

Table 15. Summary of the first non-secure VoIP over satellite call

[1] http://reviews-zdnet.com.com/Bandwidth_meter/7004-7254_16-0.html
[2] http://reviews.cnet.com/7004-7254_7-0.html

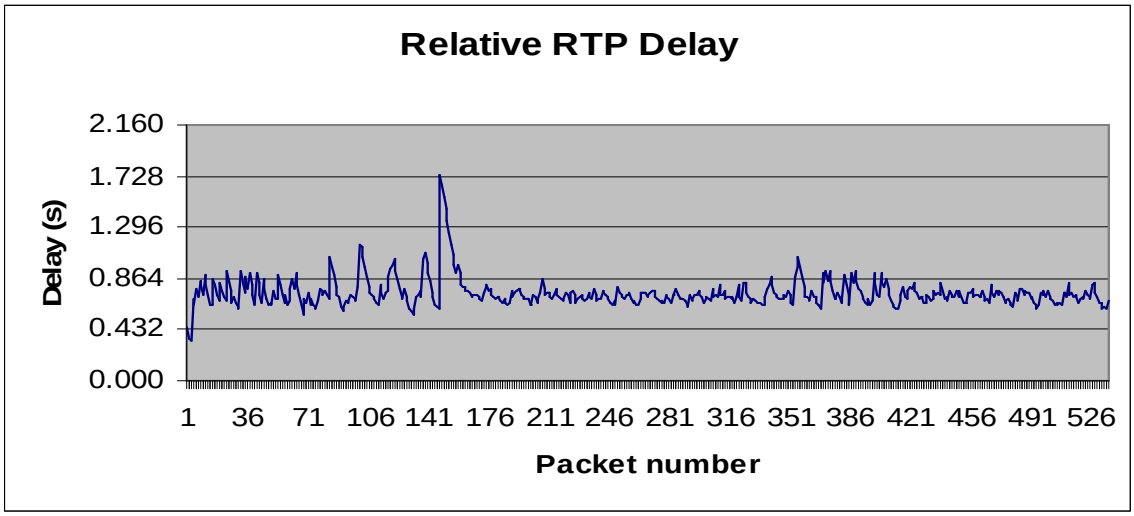


Figure 40. Relative RTP delay of the non-secure call

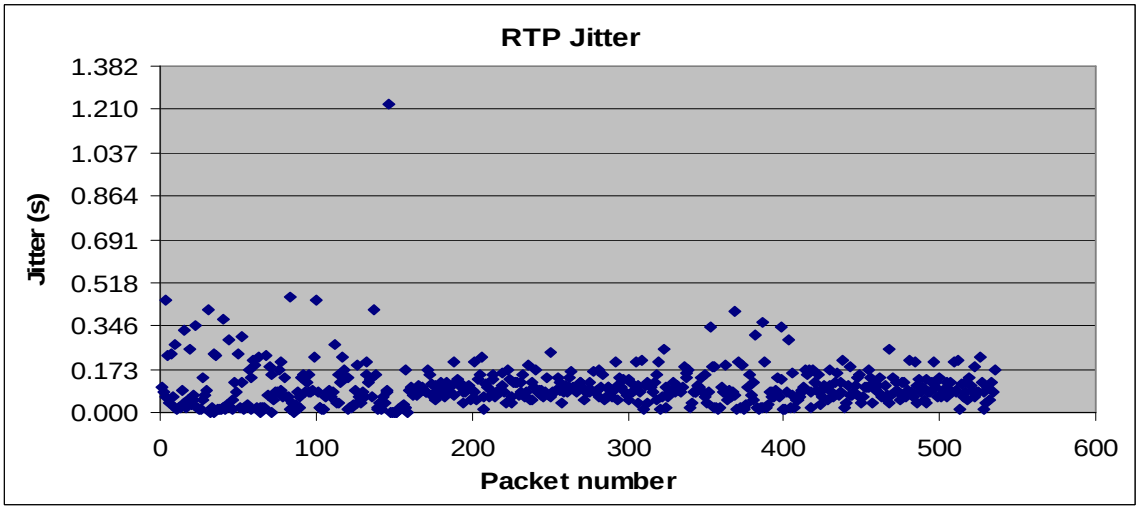


Figure 41. Interarrival jitter of the first non-secure call

Type of calls:	**Un-secure (RTP), receiver terminate**
Sender call duration:	58. 364 s (13:09:02.437 –13:10:00.801)
Receiver call duration:	57.202 s (13:09:02.771 - 13:09:59.973)
Average delay time:	739 ms

Table 16. Summary of the second non-secure VoIP over satellite call

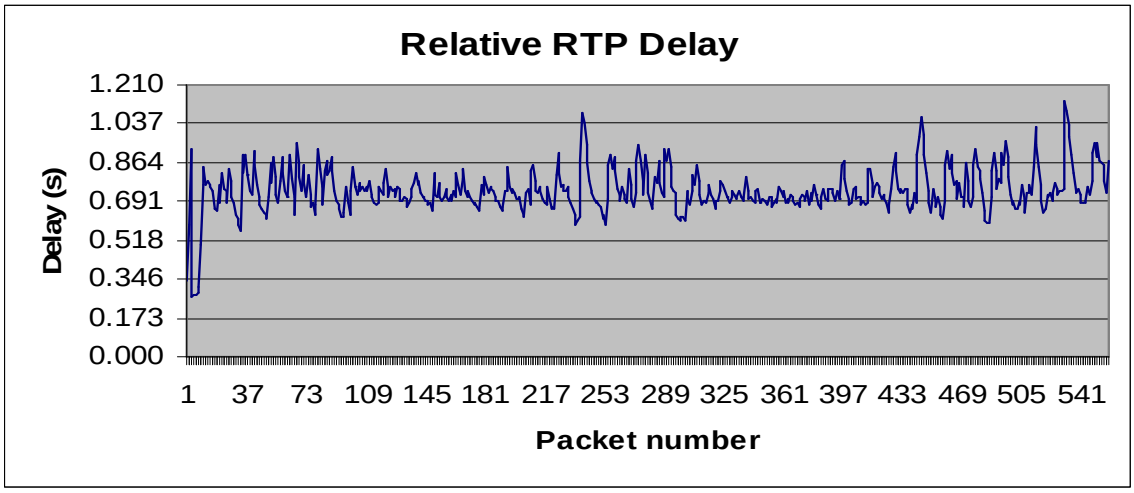


Figure 42. Relative RTP delay of the second non-secure call

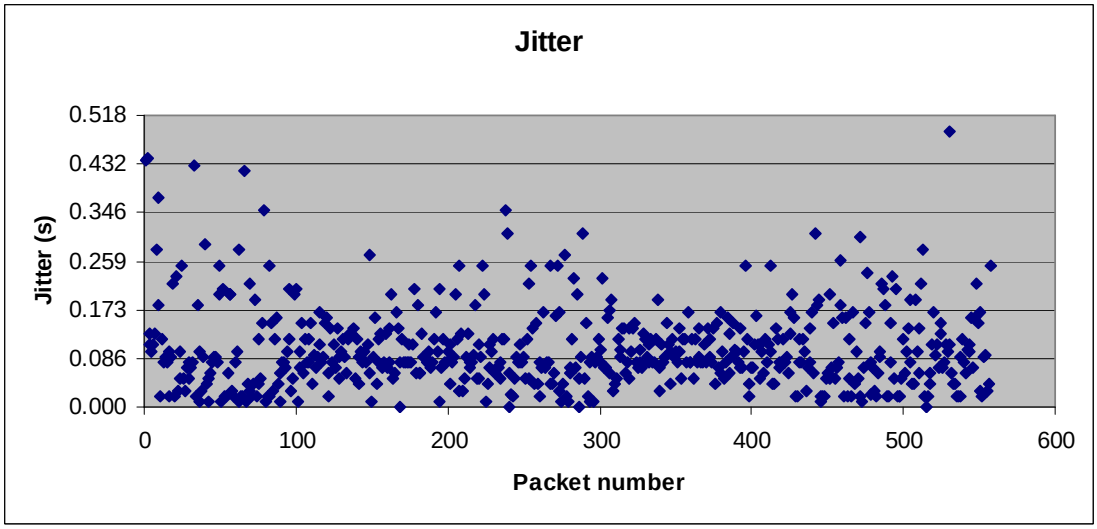


Figure 43. Interarrival jitter of the second non-secure call

Type of calls:	**Secure call (SRTP/MIKEY), receiver terminate**
Sender call duration:	309.845 s (13:01:41.373 – 13:06:51.218)
Receiver call duration:	308.824 s (13:01:41.707 – 13:06:50.531)
Average delay time:	762 ms

Table 17. Summary of the first secure VoIP over satellite call

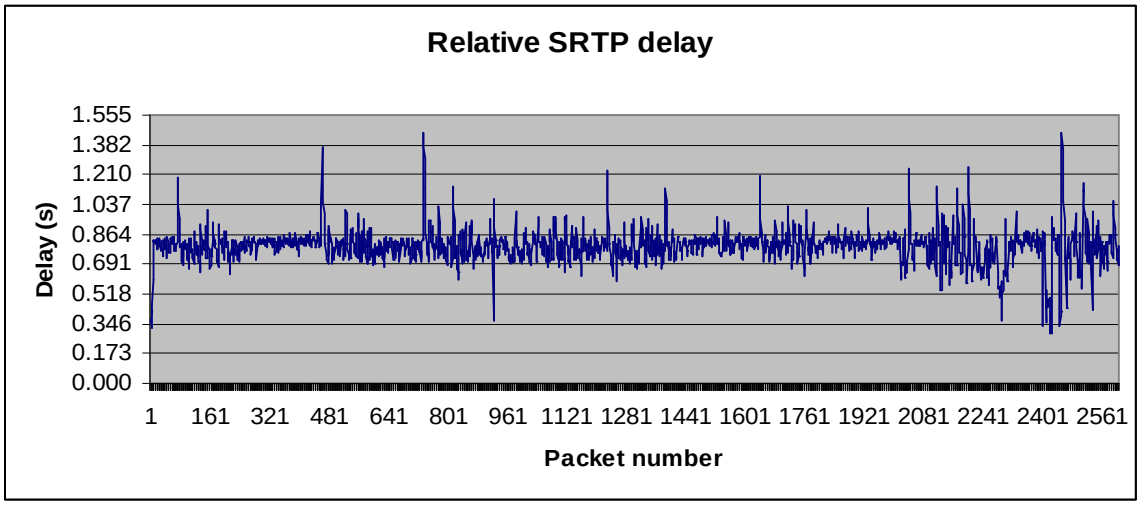


Figure 44. Relative SRTP delay of the first secure call

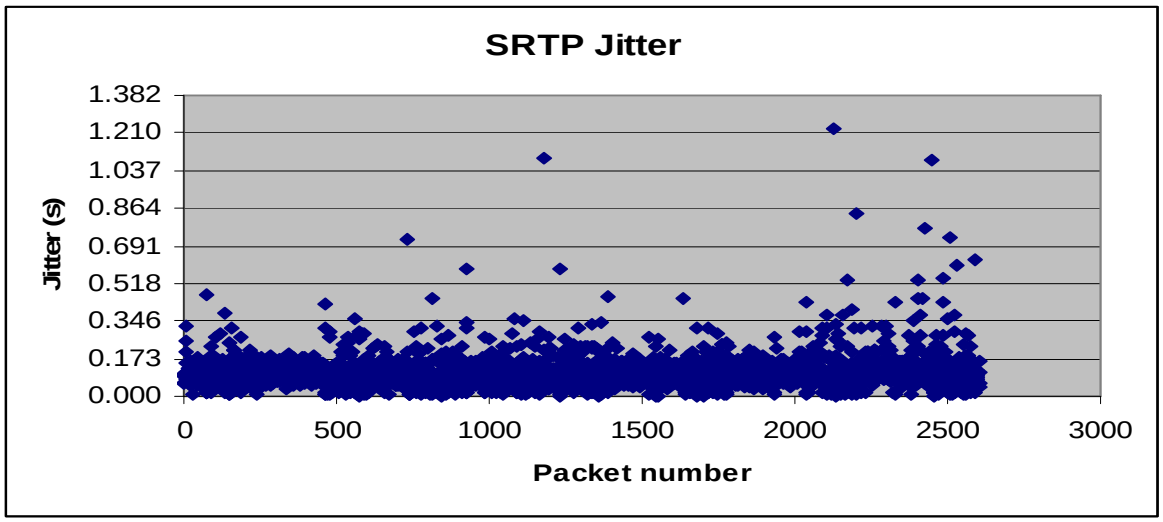


Figure 45. Interarrival jitter of the first secure call

Type of calls:	**Secure call (SRTP/MIKEY), receiver terminate**
Sender call duration:	288.555 s (12:49:43.510 –12:54:32.065)
Receiver call duration:	289.016 s (12:49:43.884 - 12:54:32.900)
Average delay time:	793 ms

Table 18. Summary of the second secure VoIP over satellite call

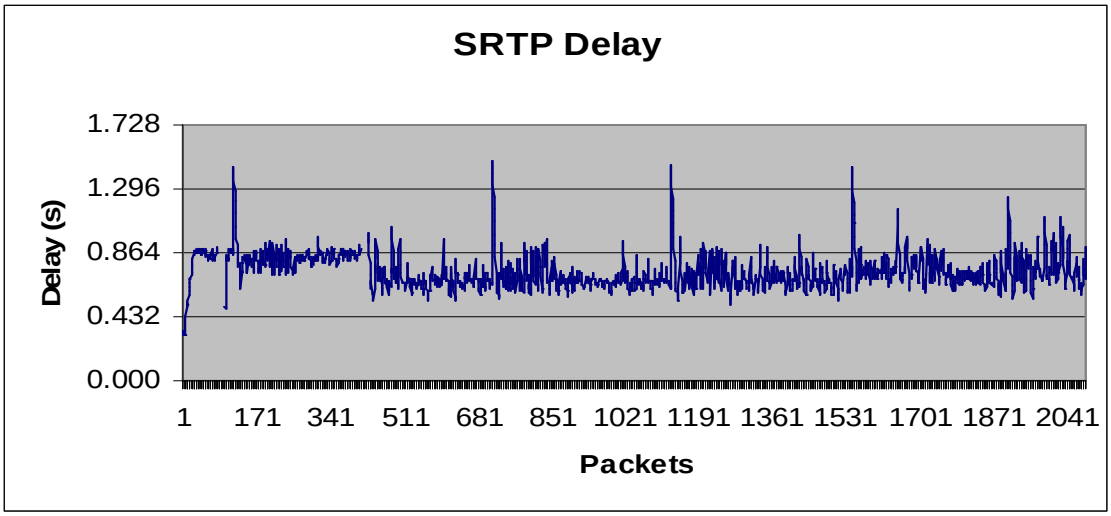


Figure 46. Relative SRTP delay of the second secure call

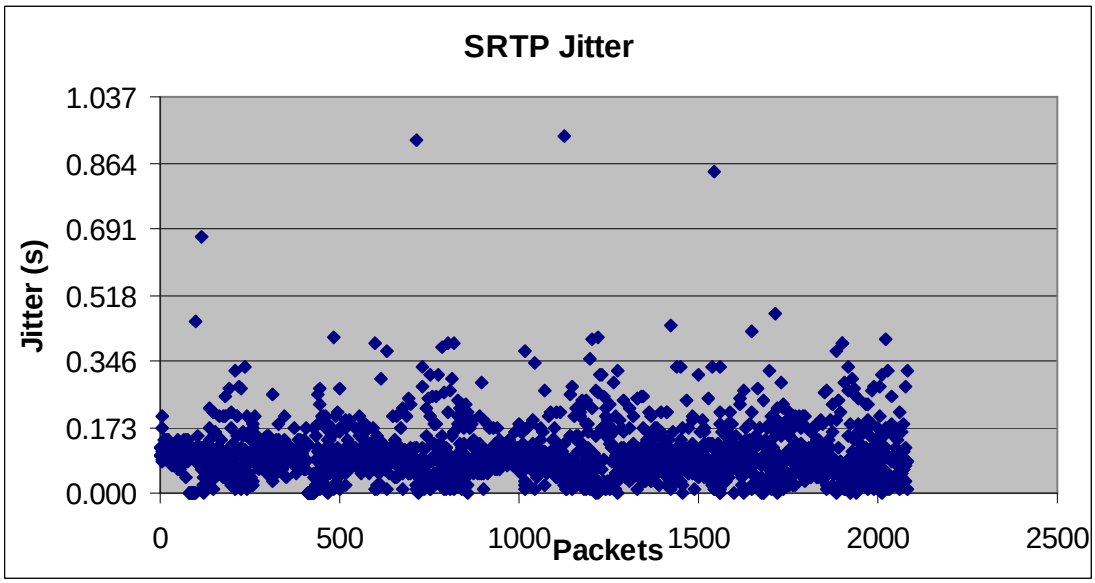


Figure 47. Interarrival jitter of the second secure call

The delay of VoIP over VPN over a satellite is quite high. Three major factors contribute to this total delay: propagation delay, IPsec encryption delay, and SRTP encryption delay. The propagation delay plays a key role in the total delay. An uplink to the GEO satellite and downlink from GEO satellite totals around 500 ms (see section 2.5). SRTP adds 20-70 ms of additional delay compared with RTP. We calculate the delay based on packets' arrival, regardless of the sequence number and timestamp. Average delay varies from 720 ms to 740ms with a non

secure call, and from 760 ms to 800ms for a secure call. Figure 40, 42, 44, and 46 show our measurement of relative delay.

Jitter is calculated continuously as each data packet is received from source SSRC according to the fomula:

$$J = J + (| D(i-1, i) – J) /16$$

Where D(i, j) is the jitter between two RTP packets: i, j. D(I,j) can be calculated:

$$D(i, j) = (Rj –Ri) – (Sj- Si) = (Rj – Sj) – (Ri – Si)$$

Where: Si, Sj : RTP timestamp from packets i, j respectively

Ri, Rj: Time of arrival in RTP timestamp units for packet i, j respectively

We do not calculate to the jitter value for complete calls but figure 41, 43, 45 and 47 show inter-arrival jitter value. We saw that jitter value changes frequently over satellite.

We count the packets lost by calculating number of the packets sent and the number of packets received. During a VoIP session, after one side terminates the call, the other side still sends packets until it receives a BYE message. By comparing the sequence number, we can exclude the packets sent after one side terminates the session. Our results show that the ratio of packet lost is around 1,5 to 3%.

In addition to quantitative measurements, we have tried some subjective measurements. The clients that we tested were: SleIPner, Xten's X-Lite, Microsoft's Windows messenger. Our first impression is that when we have conversations with VoIP over a satellite link, the quality is pretty good with SleIPner (using AMR CODEC) and Xten's X-Lite (using the GSM CODEC). The delay does not seem really relevant to our conversation. When we counted from 1 to 10, we recognize the effect of delay because the number heard on the softphone is lower than the number that speaks. Microsoft's Windows messenger, on other hand, has very bad quality and high delay, it takes several seconds to deliver the voice. The reason is that Windows Messenger uses G711 CODEC that requires 174 kbps bandwidth for two-way communication (see table 10).

6 Conclusions

Designing a solution that provides both secure data and voice is a challenge. In my thesis, I have improved upon and implemented a WIDER solution that can support security in voice and data, allow user mobility, and ease to use the system.

From my point of view, the security perspective should be recognized on both sides: end-user application and network infrastructure. The network infrastructure should provide basic security with authentication, authorization, encryption, integrity, and confidentiality. Authentication refers to the process of verifying the user's identity. Our implementation of authentication provides both mobility and ease of logging in by setting up IEEE 802.1x Port-based authentication using EAP-TLS and EAP-TTLS. EAP-TLS has the advantage that it supports mobility and automatically logins via the system, but has a disadvantage of distributed certificates, especially in disaster area. EAP-TTLS alleviates this issue by using a TLS tunnel that can be done using user/password (MD5, PAP over EAP).

Separating data and voice in the same infrastructure is necessary for enhancing both security and quality of service. A VLAN was chosen as the solution to split data and voice. VLAN Voice always has higher priority than VLAN data. We configure a VLAN for each relief organization; however, this separation is transparent from the user's view point. VLAN purposely reduces the security risk in a collision domain, not for separating communication between relief organizations.

A firewall is an effective tool that protects a trusted network from the outside. The firewall has been configured to support specific policies and to enable NAT so as accessible to make the internal WIDER network to the outside world. Using an ALG in the firewall allows two-way VoIP calls, but still maintains the same security level. Setting up a VPN allows dial-up user or headquarter user to remotely access the internal WIDER network.

VoIP today is not secure since packet voice is transmitted without any encryption or authentication. This can be solved by implementing a flexible Secure VoIP client. Hence we have integrated SRTP/MIKEY in a SleIPner client. Beyond the shared key

and public key using key management (MIKEY), a strong random master key and salt are generated to support the built-in Secure VoIP client.

Thorough measurement of performance and QoS is important before practical deployment. Our measurement includes determining VoIP capacity in each access node and the QoS when transmitting voice calls over a satellite link. We conclude that the capacity of VoIP over WLAN is limited due to the physical layer design of IEEE 802.11, in particular its CSMA/CA scheme. Delay is the most significant factor with regard to QoS, especially when sending a voice call over a VPN over a satellite link. Our measurements have shown that the delay is large; however, subjective tests give us the impression that this not really damage conversation. Secure VoIP has added greater delay than non-secure call. This is due to the process of encryption packets and decryption them; however, again the effect does not damage the conversation.

7 Future work

7.1 Further improvements of the WIDER solution

WEP encryption is currently used via the wireless bridge and access point. However, WEP is not a secure protocol and will soon be obsolete. Updating equipment to support WPA and 802.1i in the access point and wireless bridge is ***necessary***.

WiMax could be an alternative solution to provides broadband wireless access. This solution should be investigated for future deployments, especially when WiMAX products are available on the market.

Ericsson Response currently deploys a MiniGSM solution and a WIDER solution separately. This MiniGSM and WIDER solution can be integrated to support communication between GSM and VoIP. My suggestion is to use a GSM gateway to WIDER side that connects to the miniGSM via wireless interface. This solution provides mobility via two communications systems. With this integration, EAP-SIM can be an alternative method to login to WIDER network. The VoIP server used in WIDER only allows SIP over UDP. It needs to be updated to support TCP and TLS.

Security should be end-to-end at the application layer. Secure FTP, secure email, secure instant messaging and conferencing have to be implemented in WIDER. Video over WIDER and GIS information could be additional services for WIDER.

WIDER is based on infrastructure model. This model has the disadvantage that a failure of the WIDER core causes a complete WIDER network failure. The WIDER core and WIDER camp are connected by a wireless bridge that requires Line of Sign (LoS). This is not suitable in the case of a disaster area because of has the complicated geographic structure (mountain, forest, etc). Mesh networks or ad hoc networks could be an alternative WIDER solution. In ad hoc mode, each WIDER entity has the ability to operate independently as well as to automatically connect to other WIDER entities while they are within range. Security and routing are concerns in ad hoc mode. Each WIDER entity considers a trusted entity that has an agent to synchronize its authentication database, application service database etc.

7.2 Improving VoIP client

SleIPner client needs to be improved in both its user interface and stability. Auto configuration and a friendly user interface makes it easy to use. There is still a small memory leak in the Secure SleIPner client. This needs to be fixed in the future. The SleIPner client already supports Instant Messaging and Push-to-talk. In the future, Secure Instant messaging and Push-to-talk need to be implemented.

Secure SleIPner client uses AES-CTR for encryption and HMAC-SHA1 for authentication of the RTP stream. Secure RTCP is also required in next stage of development. AES-f8 encryption could be an option for the next Secure SleIPner client release. Certificates used in the SleIPner client are self generated; it should possible to import certificates from other sources. WIDER is a small and dynamic wireless network, hence PKI is not really necessary. However, some end-users uses certificates that were issued for login with EAP-TLS. These certificates should be reused by the Secure SleIPner client in order to reduce the complexity of distributing and installing certificates.

NAT traversal is still an issue for SleIPner client. A STUN/TURN/ICE client needs to be integrated into SleIPner. Currently AMR is only the CODEC that SleIPner supports; however it should be possible to communicate with other popular VoIP clients: Xten's X-Lite, Microsoft's Windows Messenger. SleIPner should be modified to allow plug-in CODECs. Speex, G723.1, G711, GSM and EVRC needs to be added.

While we have measured the performance for an access point and QoS over a satellite link, thus the performance of the whole WIDER system should be evaluated, especially the wireless bridge and to/from satellite links.

8 Reference

[3GPP TS 26.101]	Mandatory Speech codec speech processing functions; AMR Speech codec frame structure (Release 4), 3GPP TS 26.101 v4.2.0, September 2004.
[3GPP TS26.090]	3GPP TS 26.090, "Adaptive Multi-Rate (AMR) speech transcoding", version 4.0.0 (2001-03), 3rd Generation Partnership Project (3GPP).
[draft-conf-framework]	J. Rosenberg, “A Framework for Conferencing with the Session Initiation Protocol” , draft-ietf-sipping-conferencing- framework-5, Working in progress, May 2005
[draft-conf-package]	J. Rosenberg, “Session Initiation Protocol (SIP) Event Package for Conference State”, draft-ietf-sipping-conference-package-10, Working in progress, March 2005.
[draft-GEOPRIV]	J. Peterson, “A Presence-based GEOPRIV Location Object Format”, Internet Draft, Internet Engineering Task Force, January 2004. Work In Progress
[draft-ice]	J. Rosenberg, “Interactive Connectivity Establishment (ICE): A Methodology for NAT Traversal for multimedia Session Establishment Protocols”, draft-ietf-mmusic-ice-00, Working in progress, October 2003.
[draft-kmgmt-ext]	J. Arkko, “Key management Extensions for SDP and RTSP”, draft-ietf-mmusic-kmgmt-ext-12, Working in progress, November 2004
[draft-kmgmt-ext]	J. Arkko, “Key management extensions for Session Description Protocol (SDP) and Real Time Streaming Protocol (RTSP), draft-ietf-mmusic-kmgmt-ext-12, Working in progress, November 2004.
[draft-sdescriptions]	M.Baugher, “Session Description Protocol Security Descriptions for Media Streams”, draft-ietf-mmusic-sdescriptions-07.txt, July 2004. Working in progress
[draft-turn]	J. Rosenberg, “Traversal Using Relay NAT (TURN)” draft-rosenberg-midcom-turn-02, Working in progress, October 2003.
[E. Dimitriou]	Eleftherios Dimitriou, “Internet telephony over WLANS”, whitepaper, www.globalsoundip.com, accessed in November 2004
[ETSI TS 101 367-3-1]	GMR-1 03.001 , “GEO-Mobile Radio Interface Specification; Part 3: Network specifications, sub-part 1: Network Functions”, February 2005.
[I. Abad]	I. Abad, “Secure Mobile VoIP”, Master Thesis, KTH, Stockholm, Sweden, June 2003

[I. Miladinovic] I. Miladinovic , J. Stadler , “Multiparty conference Signalling using the Session Initiation Protocol (SIP)”, International Network Conference 2002, Plymouth, UK

[IEEE 802.11] IEEE, “802.11 Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications”, June 1997

[IEEE 802.16] IEEE, “802.16 Part 16: Air Interface for Fixed Broadband Wireless Access Systems”, October 2004

[IEEE 802.1x] IEEE standard for local and metropolitan area networks, “Port-based access network control”, Approved October 2001.

[Internet2 VoIP] VoIP Disaster Recovery Working Group, http://voip.internet2.edu/dr/ , Accessed on 18 April 2005.

[ITU P.800] ITU-T, “Methods for subjective determination of transmission quality”, Amended at Helsinki, 1993; revised in Geneva 1996

[ITU-T G.107] ITU-T recommendation G.107, “The E-Model, a computational model for use in transmission planning”, December 1998

[ITU-T H323] ITU-T Recommendation H.323v.4 "Packet-based multimedia communications systems", November 2000

[J. Bilien] J. Bilien, “Key Agreement for secure Voice over IP”, Master Thesis, KTH, Stockholm, Sweden, December 2003

[J. Bilien] J. Bilien, E. Eliasson, and J-O Vatn, “Call establishment delay for secure VoIP”, WiOpt'04, Cambridge UK, March 2004

[J. Christoffersson] J. Christoffersson, et al., “Reducing call setup delays using SIP/SDP compression”, Proceedings of RVK 02 – Radioetenskap och Kommunikation 02, Stockholm, June 2002

[J. Janssen] Jan Janssen, et al., “Delay bounds for Voice over IP Calls Transported over Satellite Access Networks”, ACM Mobile Networks and Applications, Volume 7, Issue 1, page 79-89, January 2002

[J. Kullemwall] Jonas Kullemwall, “Study of security aspects for Session Initiation Protocol”, Master thesis, Linkoping University, 2002

[J. Lennox] Jonathan Lennox and Henning Schulzrinne, “A protocol for reliable decentralized conferencing” ,International Workshop on Network and Operating System Support for Digital Audio and video, June 2003

[J. Orrblad] J. Orrblad, Alternatives to MIKEY/SRTP to secure VoIP, Master Thesis, KTH, Stockholm, March 2005

[J. Vatn] Jon-Olov Vatn, “An experimental study of IEEE 802.11b handover performance and its affect on voice traffic”, Technical Report TRITA-IMIT-TSLAB R 03:01, Telecommunication Systems Laboratory, Department of Microelectronics and Information

Technology, KTH, Royal Institute of Technology, Stockholm, Sweden, July 2003.

[K. J. Khan] Khurram Jahangir Khan, ”Voice over Wireless LAN and analysis of MiniSIP as an 802.11 phone”, Technical report, Royal Institute of Technology (KTH), June 2004.

[M. Coupechoux] M. Coupechoux, et al., “Voice over IEEE 802.11b capacity”, 16th ITC Specialist Seminar on Performance Evaluation of Wireless and Mobile Networks, August 31 – September 02, 2004, Antwerp, Belgium.

[N. Modadugu] N. Modadugu, et al., "The Design and Implementation of datagram TLS", Proceedings of ISOC NDSS 2004, February 2004

[NIST 800-58] D. Richard Kuhn, Thomas J. Walsh, Steffen Fries, “Security Considerations for Voice Over IP Systems”, Recommendations of the National Insititute of Standars and Technology, February 2005.

[Project MESA] “Project MESA:Making Progress Toward an International PPDR Standard”, Accessed in December 2004, www.projectmesa.org/whitepaper/MESA_whitepaper.pdf.

[R. Shacham] R. Shacham et al, “An architecture for location-based service mobility using the SIP event model”, Mobisys Workshop on Context Awareness, 2004

[RFC1631] K. Egvang, “The IP Network Address Translator”, rfc 1631, http://www.faqs.org/rfcs/rfc1631.html

[RFC2246] T. Dierks, “The TLS Protocol Version 1.0”, rfc 2246, http://www.faqs.org/rfc/rfc2246.html

[RFC2275] M. Arango, “Media Gateway Control Protocol”, rfc 2275, http://www.faqs.org/rfcs/rfc2705.html

[RFC2327] M. Handley, “SDP: Session Description Protocol”, rfc 2327, http://www.faqs.org/rfc/rfc2327.html

[RFC2401] S. Kent, “Security Architecture for the Internet Protocol”, rfc 2401, http://www.faqs.org/rfcs/rfc2401

[RFC2459] R. Housley, “Internet X.509 Public Key Infrastructure Cerficate and CRL Profile”, rfc 2459, www.ietf.org/rfc/rfc2459.txt

[RFC2617] J. Franks et al, "HTTP authentication:Basic and Digest Access Authentication", rfc 2617, http://www.faqs.org/rfc/rfc2617.html

[RFC2633] B. Ramsdell, “S/MIME Version 3 Message Specification”, rfc 2633, http://www.faqs.org/rfcs/rfc2633.html

[RFC2865] C. Rigney, “Remote authentication dial in user service (RADIUS)”, rfc2865, www.faqs.org/rfcs/rfc2865.html

[RFC3261] J. Rosenberg, “SIP: Session Initiation Protocol”, rfc 3261, http://www.faqs.org/rfcs/rfc3261.html

[RFC3264] J. Rosenberg, “An Offer/Answer Model with Session Description Protocol (SDP)”, rfc 2364, http://www.faqs.org/rfc/rfc3264.html

[RFC3265] A. B. Roach, “Session Initiation Protocol (SIP)-Specific Event Notification”, rfc3265

[RFC3303] P. Srisuresh, “Middlebox communication architecture and framework”, rfc 3303, http://www.faqs.org/rfcs/rfc3303.html

[RFC3320] R. Price, “Signaling Compression (SigComp)”, rfc 3320, http://www.faqs.org/rfcs/rfc3320.html

[RFC3629] F. Yergeau, “UTF-8, a transformation format of ISO 10646”, rfc 3629, http://www.faqs.org/rfcs/rfc3629.html

[RFC3711] M. Baugher et al, “rfc 3711: The secure real-time transport protocol (SRTP), http://www.faqs.org/rfcs/rfc3711.html

[RFC3830] J. Arkko et al, “MIKEY: Multimedia Internet KEYing”, rfc 3830 http://www.faqs.org/rfcs/rfc3830.html

[RFC3849] J. Rosenberg et al, “STUN- Simple Traversal of UDP through Network Address Translators (NATs)”, rfc 3489, http://www.faqs.org/rfcs/rfc3849.html

[RFC3856] J. Rosenberg, “Presence Event Package for the Session Initiation Protocol (SIP)”, rfc 3856, http://www.ietf.org/rfc/rfc3856.txt

[RFC3903] A. Niemi, “Session Initiation Protocol (SIP) Extension for Event State Publication”, rfc3903, http://www.faqs.org/rfcs/rfc3903.html

[S. A. Baset] S. A. Baset, et al, “An Analysis of the Skype Peert-to-Peer Internet Telephony Protocol”, Technical report, Computer Science Department, Columbia University, 2004.

[S. Berger] Stefan Berger, Henning Schulzrinne, Stylianos Sidiroglou, Xiaotao Wu, "Ubiquitous Computing Using SIP", ACM NOSSDAV 2003, Jun. 2003.

[S. F. Midkiff] Scott F. Midkiff and Charles W.Bostian, “Rapidly deployable broadband wireless communications for emergency management”, First IEEE Workshop on Disaster Recover Networks (DIREN '02), June 24, 2002, New York

[S. Fluhrer] S. Fluhrer, et al., “Weaknesses in the key scheduling algorithm of RC4”, Eighth Annual Workshop on Selected Areas in Cryptography, August 2001.

[S. Pearsall] S. Pearlsall, “Doing a VoIP Assessment with Vivinet Assessor”, whitepaper, www.netiq.com

[SSLv3] A. Frier, P. Karlton, and P. Kocher, "The SSL 3.0 Protocol", Netscape Communications Corp., Nov 18, 1996

[T. C. Tobgyl] Tensin C. Tobgyl, ”Use of VoWLAN (Voice over WLAN) for the

provision of rural communications", ITU report, 2002

[T. Nguyen] Thuan Nguyen, et al., "Voice over IP Service and Performance in Satellite Networks", IEEE communications, March 2001

[T.J.Patel] T.J.Patel, et al., "Capacity Estimation of VoIP channels on wireless networks", Technical report, University of Texas at Austin, March 26th 2003

[Theo Kanter] Theo Kanter, et al., "802.11b handoff measurement", Technical report, Royal Institute of Technology, 2001.

[UPnP 1.0] UPnP Forum, "UPnP Device Architecture 1.0", Version 1.0.1, 2 December 2003.

[V. Tung] Vu Hoang Tung, et al., "Handoff delay measurement and fast hand off using graph neighbors", Technical report, KTH Royal Institute of Technology, Stockholm, May 2004.

IMIT/LCN 2005-12

www.kth.se