“Are you talking to me?”

By

CUBE

Murat Toprak 1502756
Hakan Nizamoğlu 1502574
Yiğitalp Ertem 1502350
Saim Güveloğlu 1395078

Middle East Technical University
Computer Engineering Department
Design Project
Requirement Analysis Report
1. Introduction
   1.1. Project Title
   1.2. Project Description
      1.2.1. Purpose
      1.2.2. Tasks
      1.2.3. Scope
      1.2.4. Application Areas of our Product
   1.3. Constraints
   1.4. Team Organization
      1.4.1. Team Structure
      1.4.2. Member Roles
      1.4.3 Process Model

2. Research
   2.1 The past
   2.2 Yesterday
   2.3 Today
   2.4 The future

3. Requirements
   3.1 System Requirements
      3.1.1 Software Requirements
         3.1.1.1 Server Side Software Requirements
         3.1.1.2 Device Side Software Requirements
      3.1.2 Hardware Requirements
         3.1.2.1 Developmental Hardware Requirements
         3.1.2.2 Client Side Hardware Requirements

4. System Analysis
   4.1 Data Flow Diagrams
      4.1.1 DF Diagram
   4.2 Use Case Diagrams
      4.2.1 Caller Use Case Diagrams
      4.2.2 Callee Use Case Diagrams

5. Risk Management
1. Introduction

1.1. Project Title

Our project title is “Are You Talking To Me?”

1.2. Project Description

1.2.1. Purpose

Our Aim is to design an embedded handset which provides end-to-end secure voice over internet protocol (VoIP) communication. The main concern is to achieve the prototype of the handset. It is another issue for us to finalize the product for home or office use. This seems a design issue of the handset itself. But of course to end up with an embedded handset which is connected to a microphone and a speaker and offering telephony experience, audio transmission in technically speaking, in secure is crucial.

1.2.2. Tasks

Secure VoIP deals with specific areas which needs special attention.

- The communication part is crucial. Protocols and data transmission play key role in this project.
The security is another essential part. Cryptographic functionality should be added to the embedded handset.

The Embedded part absolutely needs extensive work.

The project will have these three main areas. There will be special subtasks according to the needs of secure voice over internet.

- Crypto functionality
- Audio compression
- Audio transmission
- Device recognition
- Embedded programming
- Embedded Linux
- Ethernet Communication

The most important task is to provide an encrypted compressed audio to the Ethernet channel. Transmission and compression will be handled by Asterisk or another VoIP application. Our first concern will be encryption, compression and transmission comes second. They will sure need some fixes. However they are not posing a problem.
1.2.3. Scope

End-to-end secure VoIP communication is still a major requirement for some applications like Asterisk, Skype and other similar VoIP communication applications. These applications are widely used in VoIP world and growing rapidly. Our embedded handset is to provide a secure, reliable telephony experience to the users of these applications. Here comes another question to take account: Is security really important for everybody? Or, for whom is security really important? The answer of the second question makes up our scope.

1.2.4. Application Areas of our Product

The prototype handset is probably won’t be ready for office or home use. It will provide secure telephony experience but in a fashionable way. The prototype is open to further industrial design development. The final product is of course can be widely used in military, government agencies and academia. Another area is small and middle enterprise offices to minimize
their cost while saving their privacy. Home use is also possible with a computer connected to internet and Skype. Big companies play a major role in telephony market since they have really huge amount of cost of communication and security is one of their main concerns. Actually this product is highly applicable for use of telephony while an internet connection and a computer is available.

1.3. Constraints

- Two end points should communicate successfully
- Encryption should be applied to the digital audio data
- A single board computer with MIC, Speaker, Ethernet, AC/DC converter by INVICTA
- Encryption should be done on this single board computer
- Embedded Linux should be used on this board
1.4. Team Organization

1.4.1. Team Structure

We think that the structure of our team will be clear in course of time. The only thing we decided is to provide a democratic environment and participation of every member.

1.4.2. Member Roles

Hakan, our contact person, has good communicative skills and vision on the project. There won’t be a work without vision. Sure.

Yiğitalp, our creative and clever boy, will bring innovative ideas to the project and we will not lack good jokes.

Saim brings discipline and gravity to the group. We should continue by the way, right?

Murat, one topic guy, can concentrate on a certain task at one time. This will help individual works and special tasks.

The workload will be equally shared among us. Above roles are just for mentioning about our group members’ profile briefly. We just have defined certain interest areas in the project for each group member. Yiğitalp deals with Security and Audio Compression, Saim is responsible of networking issues while Hakan and Murat are working on the Embedded Linux and Board.
1.4.3 Process Model

The steadiness of the requirements makes the Waterfall model the most suitable software development process. The simple and disciplined structure of the model will help us follow a concrete roadmap. Another advantage of this model is that it has discrete, easily understandable phases and marked milestones. At certain places during development, we will prepare prototypes which will ensure correctness of our progress and will help us determine the defects in the timeline.

Since there is no space for feedback in the original model, we –as many who use this model do have modified the process a bit. The results in testing may require some changes in the implementation, and it is probable that changes may be realized to be necessary during the implementation that cause some changes in the design. Since we don’t have the boards that we operate on yet, requirements may vary. Such feedbacks can be seen in the figure below.
2. Research

2.1 The past

More than 35 years ago Internet didn't exist at all. Interactive communications were only made by telephone at PSTN line cost. In Turkey, even PSTN lines for home usage are late. Data exchange was expansive (for a long distance) and no one had been thinking to video interactions (there was only television that is not interactive, as known) because of the technology and costs.
2.2 Yesterday

Few years ago we saw appearing some interesting things: PCs to large masses, new technologies to communicate like cellular phones and finally the great net: Internet; people begun to communicate with new services like email, chat, etc. and business reborned with the web allowing people buy with a "click".

2.3 Today

Today we can see a real revolution in communication world: everybody begins to use PCs and Internet for job and free time to communicate each other, to exchange data (like images, sounds, documents) and, sometimes, to talk each other using applications like Netmeeting or Internet Phone. Particularly starts to diffusing a common idea that could be the future and that can allow real-time vocal communication: VoIP.
When we compare these internet usable communication technologies with phone company activities, we see that phone companies are in some kind of dead end, so that day by day they combine internet features with their company campaigns not to lose their clients.

### 2.4 The future

We cannot know what the future is, but we can try to image it with many computers, Internet almost everywhere at high speed and people talking (audio and video) in a real time fashion. We only need to know what will be the means to do this: UMTS, VoIP (with video extension) or other? Anyway we can notice that Internet has grown very much in the last years, it is free (at least as international means) and could be the right communication media for future.
3. Requirements

3.1 System Requirements

3.1.1 Software Requirements

3.1.1.1 Server Side Software Requirements

We are planning to acquire VoIP connections with the help of Asterisk, a software implementation of a telephone private branch exchange. It is designed for Linux and it uses the GNU General Public License as a free software license. It is widely used by companies and has many features. It can be used as a PBX, gateway, call center and a public network. Asterisk will be installed in our server machine which will be provided by our department to us.

We have chosen Linux as the operating system and are going to use C/C++ since they are the programming languages we are most familiar with.

3.1.1.2 Device Side Software Requirements

We are going to use Embedded Linux on our single board computer. There will be a Linux kernel. This kernel will have capabilities to compile C/C++ codes and run them. Audio capabilities, compression and crypto functionalities will be inside the box.

We will need a design software for the board which we will be using. This program will be available in the FPGA development kit. This program helps when we program the board and develop the application on board.
3.1.2 Hardware Requirements

3.1.2.1. Developmental Hardware Requirements

We will need a computer connected to the internet and linux installed to program the board. GCC C/C++ Compiler will be one of our development tools.

3.1.2.2 Client Side Hardware Requirements

We need a single board computer which has a CPU, AC/DC Converter, MIC, Speaker, Audio Adapter or circuit that we may use as an audio adapter and embedded Linux installed on it. This FPGA board with its cables and connectors and programs will be provided by INVICTA as a FPGA Development Kit. Since INVICTA still didn’t reply our device and documentation request, we are not able to present the board which we will be using in our development process. But we found another single board computer in internet. This board is pretty well equipped for our project. The board has many features and one of them is VGA Video Port. At first we will concentrate on encrypted audio transmission in general. But we have the chance to use LCD screen connected to VGA Port. We may be screening the details of the call with the data which we have obtained from the server program while, before or after conversation.
We will need a microphone and a speaker. Microphone will be used as audio input and the speaker will be the output of the system. These two devices will be plugged into the board. We will also need an Ethernet cable. The board which we have the picture above is pretty enough for our project.

This example board is characterized as follows in its website:

- Dimensions: 141.4 x 109.5mm (expansion board), 67.7 x 47mm (CPU board)
- Working temperature: 0~70 Celsius
• Power supply: +5V
• Atmel AT91SAM9261S (ARM926EJ-S core with MMU capable of 200 MHz operation, AT91SAM9261 is compatible)
• 64Mbyte SDRAM
• 128Mbyte Nand Flash (bootable, selected through jumper, support 256MB for option)
• 4Mbyte Nor Flash (support 8MB for option)
• 4Mbyte SPI serial DataFlash (bootable, selected through jumper)
• 1Kbit EEPROM (DS2431)
• 1 2*20-pin LCD interface (STN or TFT, support resolution up to 2048 x 2048)
• Touch panel (4-channel 12-bit ADC)
• 1 VGA display port
• 1 10M/100M Ethernet interface (RJ45)
• 4 serial ports (one 9-wire RS232 serial port, one 5-wire RS232/TTL serial port, one 3-wire RS232/TTL serial port, one RS485 serial port or one 3-wire TTL serial port)
• 2 USB host and 1 USB device
• 1 SPI interface (multiplex with CAN interface)
• 4 x 4 keyboard interface
• Audio Input/Output
• RTC (battery backed)
• 1 20pin (2.0mm space) standard JTAG interface
• SD / MMC card socket
• 2 programmable LEDs
• 2 power indicators
• 4 buttons (one for Wakeup, one for Reset)
• 1 Power switch
• 1 Reset button
• 13 GPIOs
• 1 20pin standard JTAG interface

4. System Analysis

4.1. Data Flow Diagrams

4.1.1 DF Diagram

A firewall for interfaced between an internal and an external networks. The firewall includes a VoIP processor for detecting an outgoing VoIP packet sent from the internal network for changing data in a header of the VoIP packet and also changing data contents in the VoIP packet corresponding to data changed in the header to enable a bi-directional VoIP communication.

In a preferred embodiment, the VoIP processor changes a source IP address and a port number in the header of the VoIP packet and also changes the data contents in the VoIP packet corresponding to the source IP address and the port number changed in the header to enable a bi-directional VoIP communication.
Internal VoIP User Sends Out VoIP

Firewall Receives VoIP Packets & Changes Source IP Address & Source Port Number in Packet Header to

Firewall Changes Port Number of VoIP Packet to a Port Number designated by Firewall

Firewall Check! Is Packet

No

Yes

Firewall Changes End-Point Address & Port Number Embedded in VoIP Packet as New IP Address & Port

Send Out

End
4.2. Use Case Diagrams

4.2.1. Caller Use Case Diagrams

This diagram explains what a user may do using the VoIP to call someone.

Log in: The phone is no longer physically connected to the PSTN and needs to be authenticated for security issues.

Start dialing: The caller may dial the number easily, so that the server routs the number to the IP directly.

Get call: After or before the authentication, before dialing the number, caller may get a call.
4.2.2. Callee Use Case Diagrams

This diagram explains what a user may do using the VoIP when he/she get a call.

**Answer:** Callee may answer the phone so that call time starts.

**Busy:** While another call is started during the callee is doing a call, phone gives busy tone the caller. Asterisk has several methods like saving the call, leaving message and so on.

**Hung up:** User may directly end the call when he/she doesn’t want to answer.

**No answer:** Other than hung up, callee may choose not to do anything so phone rings until the caller gives up trying.

5. Risk Management
We thought that forecasting the errors may give us great power to deal and overcome with those errors. Since security constitutes our main context, risks about security might cause critical problems for our system. How we implement our security protocols and algorithms plays crucial role at this point. We may think about getting out of the standards if possible. Data transmission rate is another important part for telecommunication as usual. So, our data compression algorithms will become very important, since everyone do not have decent internet connection speed. Of course, loss of data in on-line video and sound broadcasting might be acceptable. But excess of that loss is not acceptable and again if it happens it will be a critical problem for our project.

Hardware risks are another subject. Since our project is an embedded one, we have to consider about how good or bad our hardware. Compression and encryption algorithms might make our system so slow that, decompression and decryption may not work efficient as well.

Last point is the human factor. Since our group members do not working purely on this project and have other different courses, time arranging and scheduling may become a problem. To overcome this issue, we're closely communicating each other, and make ourselves know what other members are working on etc. So far it haven't
became a problem but we guess no one can guarantee the same for the future.

6.