SUMMER INTERNSHIP

On

VOIP (Voice Over Internet Protocol)

At

Institute for Systems Studies & Analyses (ISSA)

Project Submitted by :-

Akash Gupta
Mohit
Sourav Sharma
Vijay Bisht

Project Submitted to :-

Praveen Kumar Singh, Sc-E
ISSA, DRDO

Date: 09 - 08 - 2021

DECLARATION

We hereby declare that the Project Report entitled "Voice over IP (VoIP)" is an authentic record of the work carried out for summer internship from Institute for System Studies & Analyses (ISSA) in DRDO New Delhi, during the 6th semester for the award of degree of B.Tech. (Electronics & communication) from NSUT East Campus (formerly Ambedkar Institute of Advanced Communication Technologies and Research (AIACTR), New Delhi, under the guidance of Mr. Praveen Kumar Singh.

(Signature of students) **Date: 09-08-21**

Akash Gupta

Mohit

Sourav Sharma

Vijay Bisht

Acknowledgement

Firstly, we wish to express our deep sense of gratitude to

Praveen Kumar Singh, Sc-E *, our guide, for his consistent encouragement, incalculable guidance and cooperation to carry out this project, and for giving us an opportunity to work on this project and providing us with a great environment to carry our work in ease.*

We would also like to thank our friends, for providing insightful comments and helping enhance the quality of thesis, and family, for being understanding and supportive throughout. Their help cannot be penned with words.

Organization Background of ISSA-DRDO

DRDO is the R&D wing of Ministry of Defence, Govt of India, with a vision to empower India with cutting-edge defence technologies and a mission to achieve self-reliance in critical defence technologies and systems, while equipping our armed forces with state-of-the-art weapon systems and equipment in accordance with requirements laid down by the three Services

ISSA is involved in System Analysis, Modeling & Simulation for various defence applications pertaining to employment/deployment, tactics & force potential evaluation, tactical/strategic & mission planning etc. It develops wargaming software for all the three Services.

ISSA mission is to conduct system study and develop high quality integrated software for system analysis & decision support in application areas of Sensors & Weapons, Electronic Combat, Land & Naval Combat, Air-to-Air Combat and Air Defence for effective use by DRDO and Services for Design, Mission Planning, Tactics development and Training.

Training Objective

To develop understanding of the protocols, layers & technique used in establishing VOIP communication such as TCP, IP, SIP, OSI Model, Transport layer, application layer etc. and to establish the voice and data communication between systems using VOIP technology via open source code based application software.

Abbreviations

VoIP :- Voice over internet protocol
OSI Model :- Open System Interconnection Model
TCP/IP Model :- Transmission Control Protocol / Internet Protocol Model
UDP :- User Datagram Protocol
HTTP :- Hyper Text Transfer Protocol
HTTPs :- Hypertext Transfer Protocol Secure
FTP :- File Transfer Protocol
SMTP :- Simple Mail transfer Protocol
DNS :- Domain Name System
DHCP :- Dynamic Host Configuration Protocol
TFTP :- Trivial File Transfer Protocol
SIP :- Session Initiation Protocol
SIP SIMPLE :- The Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions
IP-PBX :- Internet Protocol Private Branch Exchange
RTP/RTCP :- Run Time Transport Control Protocol
SRTP :- Secure Real Time Transfer Protocol
ZRTP :-Z and Real Time Transfer Protocol
NAT :- Network Address Translation
TURN :- Traversal Using Relays around NAT
ICE :- Interactive Connectivity Establishment
CELT :-Centre For Engineering Learning And Technology
IM :- Instant Messaging
IPv4 :- Internet Protocol Version 4
IPv6 :-Internet Protocol Version 6
Speex :-An audio compression codec specifically tuned for the reproduction of human speech
Opus :- A lossy audio coding format developed by the Xiph.Org Foundation
TLS :- Transport layer Security

GSM :- Global System for Mobile Communications
GPL :- General Public License
GUI :- Graphical User Interface
Nacl :- Networking and Cryptography library
MPEG4 :-Moving Picture Expert Group

Contents

1. Scope 9

2. Introduction 10

3.Working of VoIP Software 11

i. Features Required in a VoIP Software 11

ii. Open Source Software & Comparison 12

A. Teamspeak 14

B. Mumble 16

C. LinPhone 17

D. QTox 18

iii. Source Code Links 19

4. Comparison of some voip softwares 20

5. Selection of final software to be used for further development 21

6. Working on Mumble software 22

A. Connecting mumble to SuperUser server 28

B. Connecting to private 31

7. Conclusion 34

8. References 35

1. Scope

The scope of this document is to cover the basic protocols and methodolgies used in the VOIP communication, basic working principle, comparison of various available open source software. It also provides an insight to one of the open source software whose source code was tweaked for the UI portion to suit our requirement.

2. Introduction

One of the biggest inventions of the 21st century is VoIP. It’s a wonderful technology because it’s the most effective way of delivering voice between people and organizations today.

One of the main reasons for which businesses are moving over to VoIP technology is its cost. VoIP is a great way to cut down communication costs and allows people to share instant messages, make voice calls, and video calls for free.

Our aim was to set up the mumble over a public network using server and over LAN network with slight change in UI of VoIP application using Qt. To set up the communication between two or more systems we started with various open source software available online. We studied the OSI model, TCP model and various internet protocols used in them.

From the available open source software, we selected Mumble as it was easy to configure. We were successfully able to set up communication between two or more devices over the server.

	Point of Difference	VoIP	Landline
1	Setup Cost	Low	High
2	Maintenance Cost	Low	High
3	Multimedia	Ability to transmit video, or any type of multimedia	Only voice can be transmitted
4	Scalability	Easy to scale as business grows	Requires additional investment to scale
5	Usability	Cannot be used without internet connection	Can be used without internet connection
6	Pricing	Allows free VoIP to VoIP calls and international calls charged at nominal rate	No free calling allowed

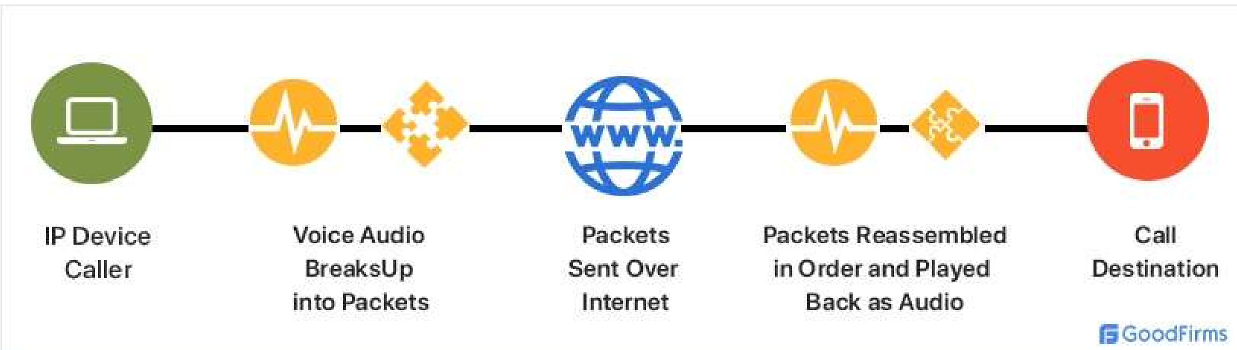
Voice over Internet Protocol (VoIP) is a category of hardware and software helping people conduct telephone-like calls using the internet. VoIP software enables you to turn a computer into a communication platform.

VoIP software allows you to create a real-time communication channel allowing you to make voice and video calls globally using an internet connection. This system becomes highly essential for businesses required to make international calls or fax documents frequently.

3.Working of VoIP Software

VoIP is a revolutionary technology that converts your voice into a digital signal and transmits it on the internet using TCP/IP (Ethernet) Protocol.

It helps you to make and receive calls all over the world using a computer, VoIP phone, mobile phones, data-driven devices, or even traditional telephones connected to the VoIP adapter known as Analog Telephone Adapter.



i. Features Required in a VoIP Software

Besides enabling phone calls through the internet, a VoIP system should include the below-mentioned features, which can help businesses extensively in improving their communication processes.

- Call Forwarding
- Conferencing
- Mobility
- Auto Attendants
- Voicemail to Email Transcription
- Call Rejection.
- Multiple Device Support
- Integrations
- Scalability

ii. Open Source Software & Comparison

Top VOIP softwares available are given as :-

- **Jitsi** - Innovative open source voice and video conferencing
- **Mumble** - Voice chat application for groups
- **Ekiga** - VoIP and video conferencing application for GNOME
- **Linphone** - Voice over IP softphone, SIP client and service
- **qTox** - Chat, voice, video, and file transfer IM client
- **Empathy** - Instant messaging and voice over IP client
- **Asterisk** - Complete PBX system
- **FreeSWITCH** - Telephony platform for voice and chat driven products
- **GNU Gatekeeper** - Feature-rich project that implements an H.323 gatekeeper.

- **Kamailio** **-** Build large platforms for VoIP and real time communications
- **Discord** - All-in-one voice and text chat for gamers
- **TeamSpeak** - A mature and popular gaming tool
- **Skype** - Make free phone calls via the Internet worldwide
- **Mumble** - Mumble is a voip primarily designed for use by gamers

A: Teamspeak

Teamspeak :-TeamSpeak (TS) is a proprietary VoIP application for audio communication between users on a chat channel, much like a telephone conference call.

Users typically use headphones with a microphone. The client software connects to a TeamSpeak server of the user's choice, from which the user may join chat channels.

The target audience for TeamSpeak is gamers, who can use the software to communicate with other players on the same team of a multiplayer video game. Communicating by voice gives a competitive advantage by enabling players to keep their hands on the controls.

■ Server

The TeamSpeak server runs as a dedicated server on Linux, macOS, Microsoft Windows and FreeBSD and uses a client based user interface or a command-line interface to control server administration and configuration.

TeamSpeak clients are available for Linux, macOS, Windows, Android, and iOS.

- The TeamSpeak 3 server can be used at no cost for up to 32 slots (simultaneous users).
- For non-commercial use, non-profit licenses were available, until September 2018, that allowed users to use the server with up to 512 slots.
- With it, server admins can choose to split up the slots into multiple virtual server instances (up to 2).

- The new client introduces a number of brand new additions to their services, including a modern global chat feature, a fully responsive user interface, free voice servers and many upgraded audio functions.
- TeamSpeak 3 introduced the use of *unique IDs*, maintained in the program as identities, that are randomly generated at the time of a client's initial setup.
- An identity contains a nickname, which can be changed at any time, the Unique ID and an identity name, which is not visible to other users on the server. The unique id is used by the server to grant permissions to the user.
- Unique IDs replaced the need for a user to register with the server to keep their user group, be it a channel group or a server group.

Audio Quality

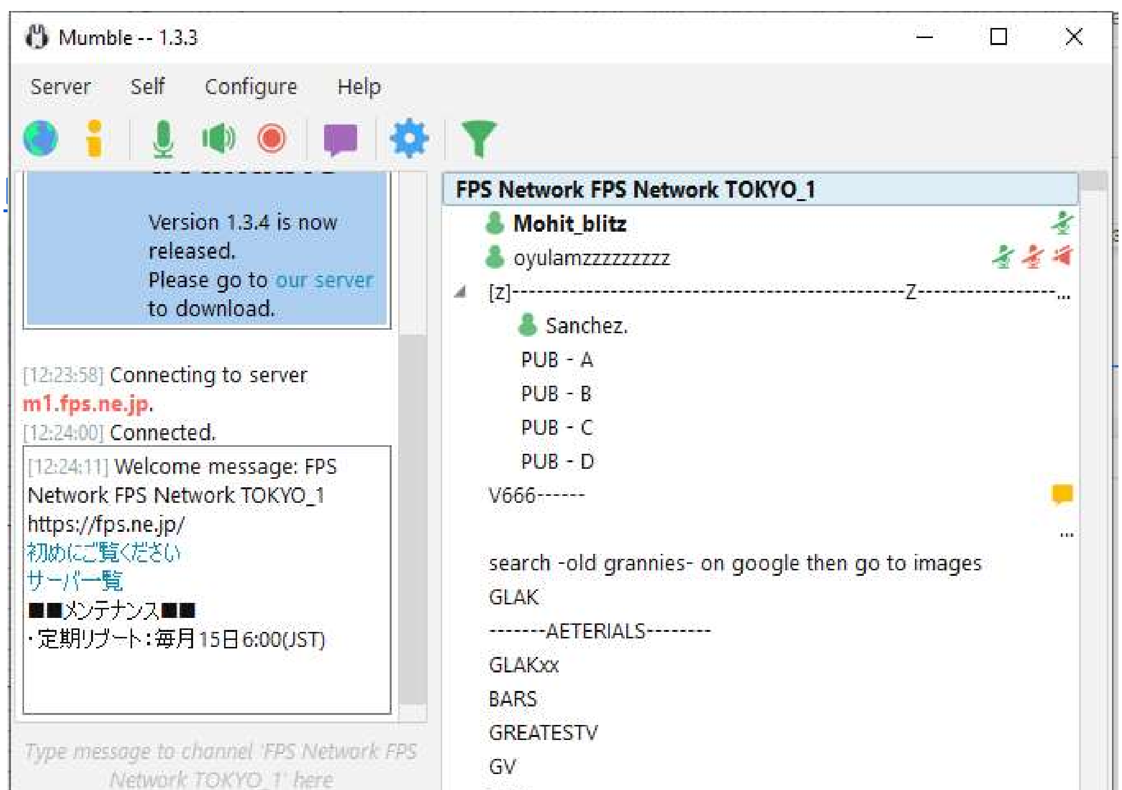
It is possible to boost the microphone volume by applying software-based gain albeit at the loss of audio quality. Low THD hardware and dual-microphone noise suppression help clean up the noise floor to allow software-base gain with less of an impact to audio quality.

- **Language of the software in which it is written** :- C++
- **Protocol use in voice transmission** :- User Datagram Protocol
- **For file transfer, server query(raw) , server query(ssh), webQuery(http & https), Tsdns**:- Transmission control protocol
- **Constrained Energy Lapped Transform (CELT) :-**

It is an open, lossy audio compression format and a free software codec with especially low algorithmic delay for use in low-latencyaudio communication.

B: Mumble

- It's a low latency software.
- It generally has transmission delay of 60ms at a transmitting rate of 144Kb/s.
- Follows SIP and UDP protocol.
- End to end encryption.
- Windows/ Mac/ Linux compatible.
- Best software for gaming as it consumes less data and provides efficient communication.
- Unlike some heavy softwares, Mumble needs to be downloaded first.
- Faster



C: LinPhone :

- Linphone is a free voiceover IP softphone, SIP client and service. It may be used for audio and video direct calls and calls through any VoIPsoftswitch or IP-PBX.
- Possibility to exchange instant messages.
- Initially developed for Linux but now supports many additional
- Platforms including Microsoft Windows, macOS, and mobile phones running
- Windows phone , iOS or Android.
- It supports ZRTP for end-to-end encrypted voice and video communication.

The Linphone client provides access to following functionalities :-

- Multi-account work
- Registration on any SIP-service and line status management
- Contact list with status of other users
- Conference call initiation
- Combination of message history and call details
- DTMF signals sending (SIP INFO / RFC 2833)
- File sharing
- Additional plugins

D. QTOX

- qTox is an open source chat, voice, video, and file transfer IM client using the encrypted peer-to-peer Tox protocol. It follows the Tox design guidelines. Tox is a peer-to-peer instant-messaging and video-calling protocol that offers end-to-end encryption.
- Tox supports two transport protocols: UDP and TCP.
- Uses IPv4 and IPv6 protocols.

Features:-

- One to one chat with friends
- Group chats
- File transfers, with previewing of images
- Audio calls, including group calls
- Video calls
- ToxMe and Tox URI support
- Translations in over 30 languages
- Faux offline messages - your friend receives them when they come online if you are online at the same time.
- History - qTox can save your sent and received messages.
- Screenshots
- Emoticons - replace smileys (e.g. :-)) with corresponding graphical emoticons. There’s support for custom emoji packs.
- Auto-updates on Windows and packages on Linux

iii. Source Code Links:-

Teamspeak -https://github.com/VerHext/TS3Client

Mumble - https://github.com/mumble-voip/mumble

Qtox - https://github.com/qTox/qTox

Linphone -https://github.com/BelledonneCommunications/linphone-android

4. Comparison of some VOIP softwares

Program	Operating system	License	Cost	Protocols	Codec	Encryption
TeamSpeak	Linux, Windows, macOS, Android, iOS	Proprietary	Free	Unknown	CELT, Speex Opus	Yes
Mumble	Linux, macOS, iOS, Windows, Android	New BSD license	Free	ICE	CELT, Speex, Opus	TLS and OCB-AES128
Linphone	Linux, Windows, macOS, Android, iPhone, BlackBerry	GPL-3.0-or-later	Free	SIP	Speex, Opus, G711, GSM, G.722, VP8 (WebM), H263, MPEG4, Theora and H264 (plugin)	TLS, SRTP, ZRTP
q**Tox**	Linux, macOS, Windows, Android, FreeBSD	GPL-3.0-or-later		Tox, VP8	Opus,	NaCl

5. Selection of final software to be used for further development

Software selected - Mumble

Reason of Selection -

- It Consumes less data comparing to other. It works good below or at bandwidth rate of 128 Kbits/s. Also the size is only 22.7 MB
- Source code is easy to understand in comparison to other software's source code.
- C++ language is used in source code so its execution is fast.

6. Working on Mumble Software

In the configuration file of mumble, it was required to do some changes like setting up ip address of the server, for selecting the username of the server, also changing the default password, opus codec settings, URL of the website, if connecting to the public server, ACL settings etc.

- In the below given screenshot , databases of the murmur server are called using SQlite database driver. We can change the database driver if required by providing details of that driver.

murmur.ini

```
; Murmur configuration file.

; Murmur defaults to using SQLite with its default rollback journal.
; In some situations, using SQLite's write-ahead log (WAL) can be
; advantageous.
; If you encounter slowdowns when moving between channels and similar
; operations, enabling the SQLite write-ahead log might help.
;
; To use SQLite's write-ahead log, set sqlite_wal to one of the following
; values:
;
; 0 - Use SQLite's default rollback journal.
; 1 - Use write-ahead log with synchronous=NORMAL.
;     If Murmur crashes, the database will be in a consistent state, but
;     the most recent changes might be lost if the operating system did
;     not write them to disk yet. This option can improve Murmur's
;     interactivity on busy servers, or servers with slow storage.
; 2 - Use write-ahead log with synchronous=FULL.
;     All database writes are synchronized to disk when they are made.
;     If Murmur crashes, the database will be include all completed writes.
;sqlite_wal=0

; If you wish to use something other than SQLite, you'll need to set the nam
; of the database above, and also uncomment the below.
; Sticking with SQLite is strongly recommended, as it's the most well tested
; and by far the fastest solution.
;
;dbDriver=QMYSQL
;dbUsername=
;dbPassword=
;dbHost=
;dbPort=
;dbPrefix=murmur_
;dbOpts=

```

In the below given snap :-

- Whenever we connect to the murmur server every time in mumble application, we get a welcome message . To edit that welcome message, we just need to modify “welcometext” named starting code line shown in below snapshot.

- Also, there is always a port number to connect to the TCP and UDP data transmission protocols sockets. We can change port no. to connect to different servers.

- We can also set server password to secure it from new joiners to the server.

```

; The below will be used as defaults for new configured servers.
; If you're just running one server (the default), it's easier to
; configure it here than through D-Bus or Ice.
;
; Welcome message sent to clients when they connect.
; If the welcome message is set to an empty string,
; no welcome message will be sent to clients.
welcometext="<br />Welcome to this server running <b>Murmur</b>.<br />Enjoy your stay!<br />"

; The welcometext can also be read from an external file which might be useful
; if you want to specify a rather lengthy text. If a value for welcometext is
; set, the welcometextfile will not be read.
;welcometextfile=

; Port to bind TCP and UDP sockets to.
port=64738

; Specific IP or hostname to bind to.
; If this is left blank (default), Murmur will bind to all available addresses.
;host=

; Password to join server.
serverpassword=

; Maximum bandwidth (in bits per second) clients are allowed
; to send speech at.
bandwidth=558000

; Murmur and Mumble are usually pretty good about cleaning up hung clients, but
; occasionally one will get stuck on the server. The timeout setting will cause
; a periodic check of all clients who haven't communicated with the server in
; this many seconds - causing zombie clients to be disconnected.
```

In the below given Snap :-

- We changed allowed clients to the server in the given picture of code
- Limiting the users to the server is a must.
- Every server runs with a defined capacity. We can change this setting according to the requirement.

```
; Note that this has no effect on idle clients or people who are AFK. It will
; only affect people who are already disconnected, and just haven't told the
; server.
;timeout=30

; Maximum number of concurrent clients allowed.
users=50

; Where users sets a blanket limit on the number of clients per virtual server,
; usersperchannel sets a limit on the number per channel. The default is 0, for
; no limit.
;usersperchannel=0

; Per-user rate limiting
;
; These two settings allow to configure the per-user rate limiter for some
; command messages sent from the client to the server. The messageburst setting
; specifies an amount of messages which are allowed in short bursts. The
; messagelimit setting specifies the number of messages per second allowed over
; a longer period. If a user hits the rate limit, his packages are then ignored
; for some time. Both of these settings have a minimum of 1 as setting either to
; 0 could render the server unusable.
messageburst=5
messagelimit=1

; Respond to UDP ping packets.
;
; Setting to true exposes the current user count, the maximum user count, and
; the server's maximum bandwidth per client to unauthenticated users. In the
; Mumble client, this information is shown in the Connect dialog.
allowping=true

; Amount of users with Opus support needed to force Opus usage, in percent.
; 0 = Always enable Opus, 100 = enable Opus if it's supported by all clients.
;opusthreshold=100
```

In the below given picture :-

- Administrator server / Public LAN based server can be setup. Username of the server as we want to show to the client
- Password of the server for the client, but if it is public then no password should be set
- For creating a public server , a URL is must which defines further hostname

```
; Murmur retains the per-server log entries in an internal database which
; allows it to be accessed over D-Bus/ICE.
; How many days should such entries be kept?
; Set to 0 to keep forever, or -1 to disable logging to the DB.
;logdays=31

; To enable public server registration, the serverpassword must be blank, and
; this must all be filled out.
; The password here is used to create a registry for the server name; subsequent
; updates will need the same password. Don't lose your password.
; The URL is your own website, and only set the registerHostname for static IP
; addresses.
; Location is typically the country of typical users of the server, in
; two-letter TLD style (ISO 3166-1 alpha-2 country code)
;
; If you only wish to give your "Root" channel a custom name, then only
; uncomment the 'registerName' parameter.
;
;registerName=Mumble Server
;registerPassword=secret
;registerUrl=http://www.mumble.info/
;registerHostname=
;registerLocation=

; If this option is enabled, the server will announce its presence via the
; bonjour service discovery protocol. To change the name announced by bonjour
; adjust the registerName variable.
; See http://developer.apple.com/networking/bonjour/index.html for more information
; about bonjour.
;bonjour=True
```

In the given below screensnap :-

- We can add SSL certificates of the server that is used for security and encryption purposes. For adding a SSL certificate , we need to have sslCert and sslKey.
- If we don't have SSL certificate then , Murmur will automatically generate this.

```
; If you have a proper SSL certificate, you can provide the filenames here.
; Otherwise, Murmur will create its own certificate automatically.
;sslCert=
;sslKey=

; If the keyfile specified above is encrypted with a passphrase, you can enter
; it in this setting. It must be plaintext, so you may wish to adjust the
; permissions on your murmur.ini file accordingly.
;sslPassPhrase=

; If your certificate is signed by an authority that uses a sub-signed or
; "intermediate" certificate, you probably need to bundle it with your
; certificate in order to get Murmur to accept it. You can either concatenate
; the two certificates into one file, or you can put it in a file by itself and
; put the path to that PEM-file in sslCA.
;sslCA=

; The sslDHParams option allows you to specify a PEM-encoded file with
; Diffie-Hellman parameters, which will be used as the default Diffie-
; Hellman parameters for all virtual servers.
;
; Instead of pointing sslDHParams to a file, you can also use the option
; to specify a named set of Diffie-Hellman parameters for Murmur to use.
; Murmur comes bundled with the Diffie-Hellman parameters from RFC 7919.
; These parameters are available by using the following names:
;
; @ffdhe2048, @ffdhe3072, @ffdhe4096, @ffdhe6144, @ffdhe8192
;
; By default, Murmur uses @ffdhe2048.
;sslDHParams=@ffdhe2048

; The sslCiphers option chooses the cipher suites to make available for use
; in SSL/TLS. This option is server-wide, and cannot be set on a
; per-virtual-server basis.
```

In this below given snap :-

- If a client tries to connect to the server with wrong password many times then murmur will autoban the client's IP address . After that client cannot connect to any server for autoban time frame defined below .

- Owner of the server can modify settings for the autoban time and allowed attempts to connect to the server.

```
; If a client attempts autobanAttempts connections in autobanTimeframe seconds,
; they will be banned for autobanTime seconds. This is a global ban, from all
; virtual servers on the Murmur process. It will not show up in any of the
; ban-lists on the server, and they can't be removed without restarting the
; Murmur process - just let them expire. A single, properly functioning client
; should not trip these bans.
;
; To disable, set autobanAttempts or autobanTimeframe to 0. Commenting these
; settings out will cause Murmur to use the defaults:
;
; To avoid autobanning successful connection attempts from the same IP address,
; set autobanSuccessfulConnections=False.
;
;autobanAttempts=10
;autobanTimeframe=120
;autobanTime=300
;autobanSuccessfulConnections=True

; Enables logging of group changes. This means that every time a group in a
; channel changes, the server will log all groups and their members from before
; the change and after the change. Deault is false. This option was introduced
; with Murmur 1.4.0.
;
;loggroupchanges=false

; Enables logging of ACL changes. This means that every time the ACL in a
; channel changes, the server will log all ACLs from before the change and
; after the change. Default is false. This option was introduced with Murmur
; 1.4.0.
;
;logaclchanges=false
```

A. Connecting mumble to SuperUser server :-

After starting up the mumble application, we need to connect it to a server. In server , there are two types of categories that are -

1. Public server
2. Private Server

Let's just start working on mumble by connecting it to an admin or public server. A public server is that server to which anyone around the globe can connect easily depending upon its accessibility.

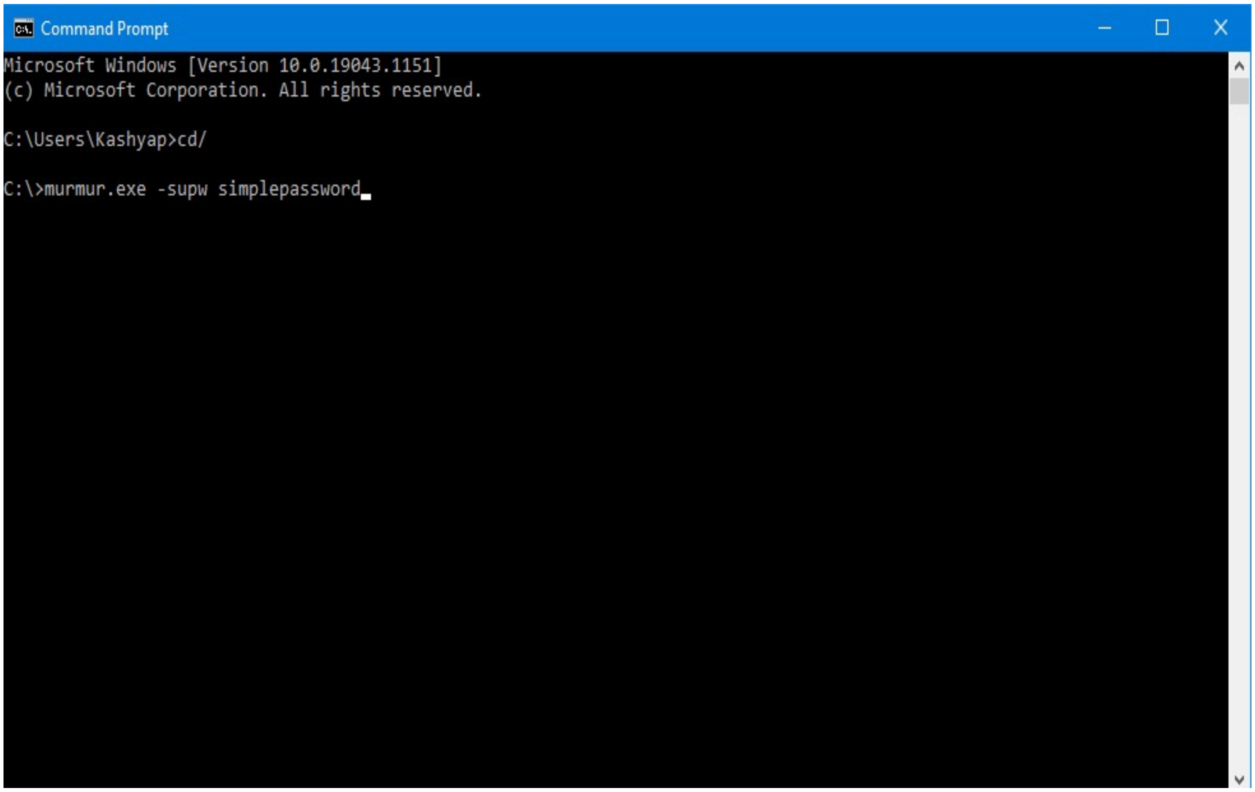
- Connecting to public servers is easy because they are officially available on mumble applications according to their locations.

- But superuser is an admin server that can be used to change the settings of users and other permissions like ACL, groups , audio settings etc

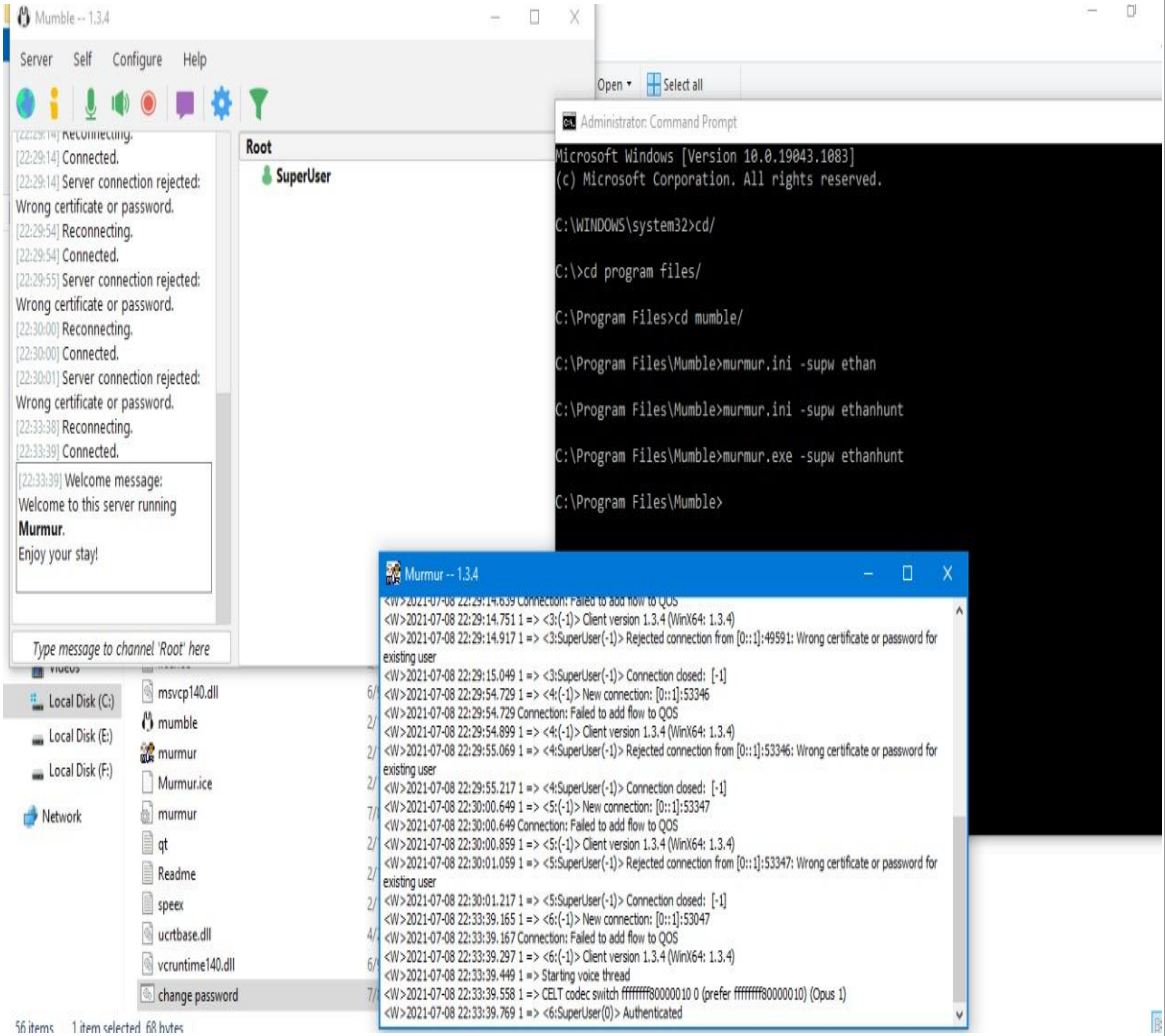
- To successfully connect mumble to admin server or superuser , it is required to change the default server password from the configuration file whose extension is .INI , it is located in the installation folder of mumble.

- The server default password can also be changed by an alternate method, that is by using command prompt.

By typing the following command given in screenshot :-



A screenshot is given below showing that superuser server is connected to mumble



B. Connecting to private server :-

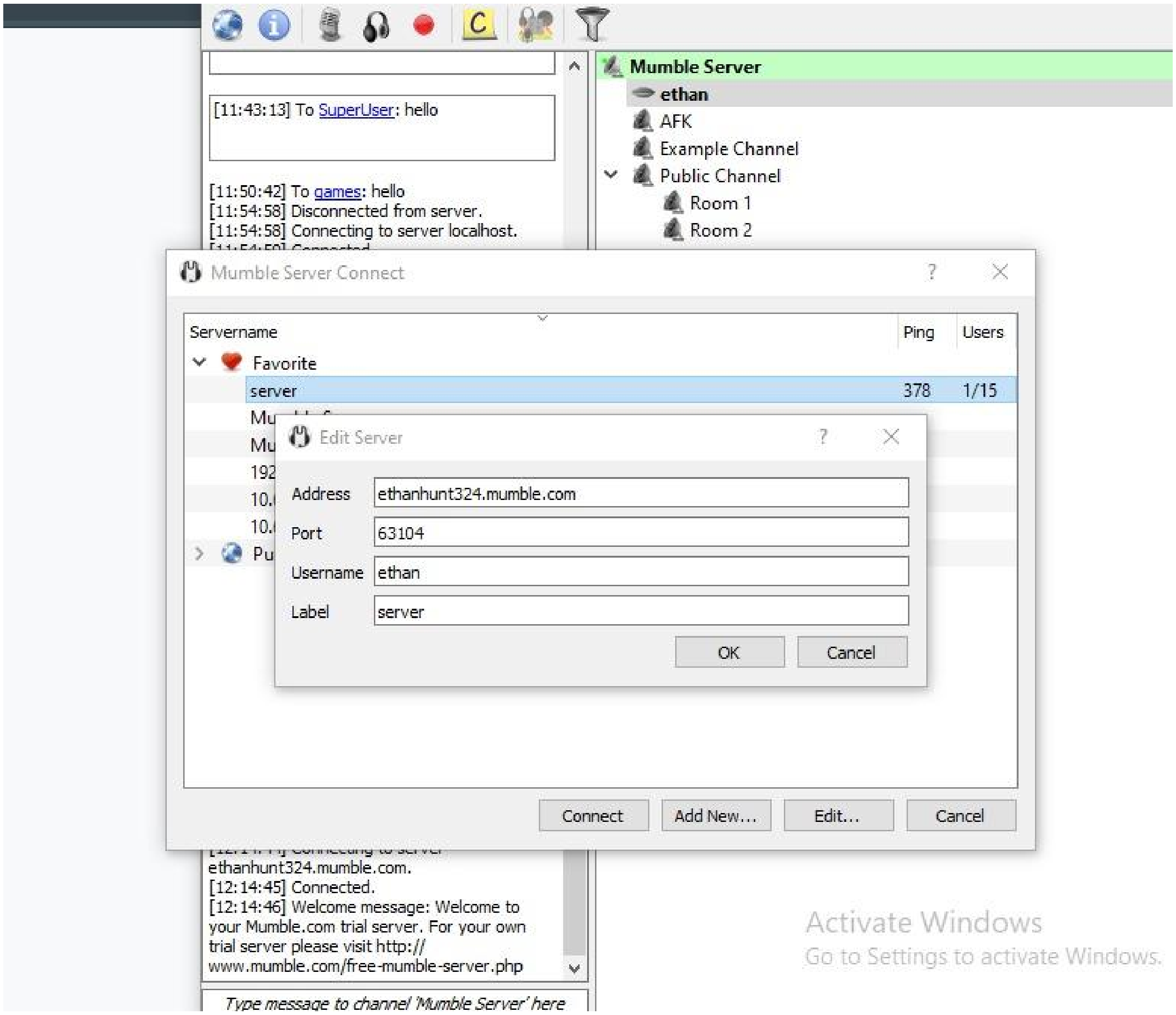
- A private server is the secured server that can be used for professional purposes. A private server can be rented by paying demonstrated charges. We rented a server on a trial basis to try connecting our mumble application. A screenshot is given below:

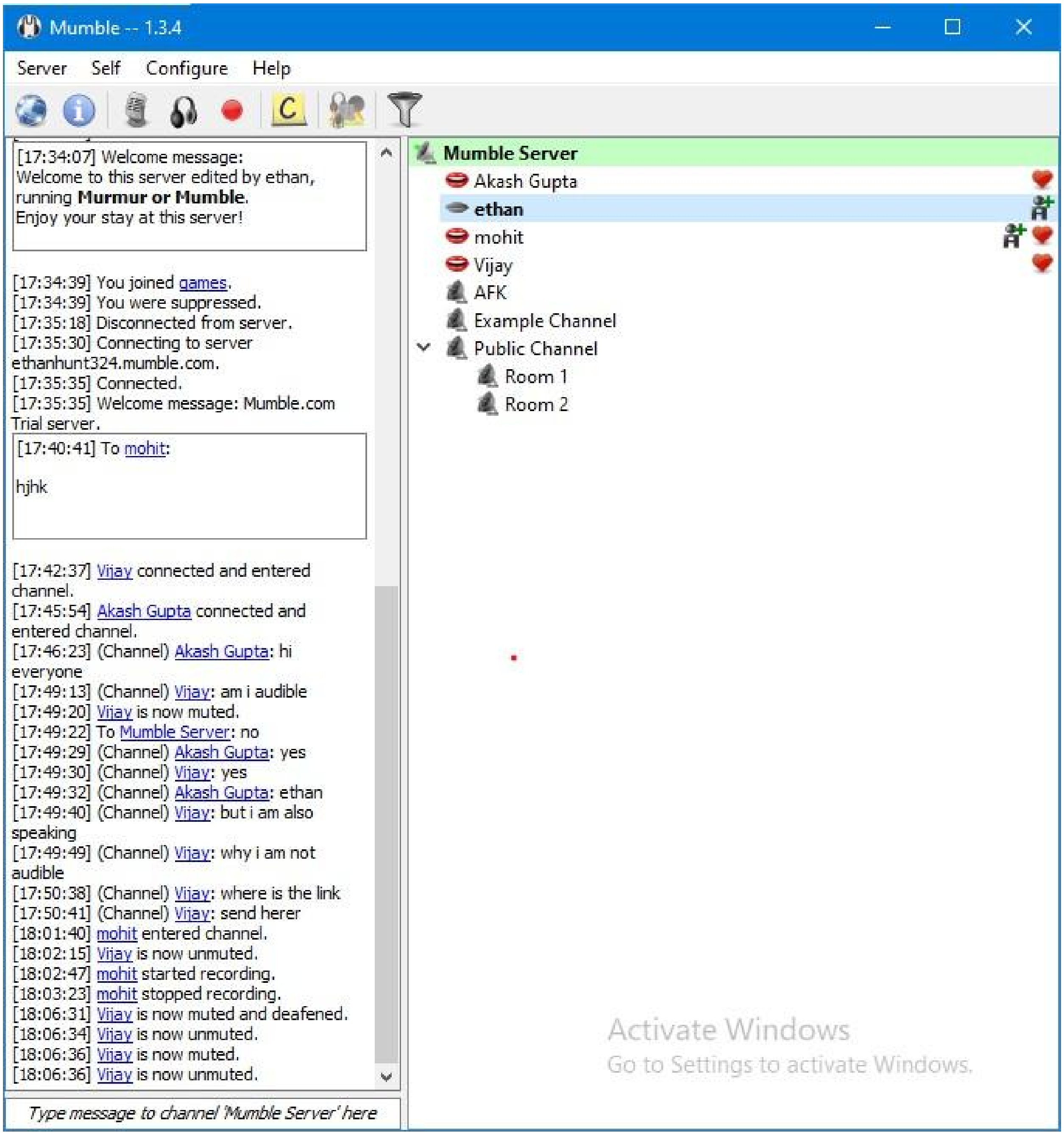
- In the given snap the address and port is entered according to the designated server.

- This private server consists of some channels that can be created and deleted instantly.

- Also, these channels allow us to confidentially send our messages and voice transmission to all the members that are connected within that channel. Therefore, no other member who is not present in that channel but connected to the server can’t listen us.

- To successfully connect mumble to admin server or SuperUser


[11:43:13] To SuperUser: hello
[11:50:42] To games: hello
[11:54:58] Disconnected from server.
[11:54:58] Connecting to server localhost.
Mumble Server
ethan
AFK
Example Channel
Public Channel
Room 1
Room 2
Mumble Server Connect
Servername
Ping
Users
Favorite
server
378
1/15
Edit Server
Address
ethanhunt324.mumble.com
Port
63104
Username
ethan
Label
server
OK
Cancel
Connect
Add New...
Edit...
Cancel
ethanhunt324.mumble.com.
[12:14:45] Connected.
[12:14:46] Welcome message: Welcome to your Mumble.com trial server. For your own trial server please visit http://www.mumble.com/free-mumble-server.php
Activate Windows
Go to Settings to activate Windows.
Type message to channel 'Mumble Server' here


Mumble -- 1.3.4
Server Self Configure Help
[17:34:07] Welcome message:
Welcome to this server edited by ethan,
running **Murmur or Mumble**.
Enjoy your stay at this server!
[17:34:39] You joined games.
[17:34:39] You were suppressed.
[17:35:18] Disconnected from server.
[17:35:30] Connecting to server ethanhunt324.mumble.com.
[17:35:35] Connected.
[17:35:35] Welcome message: Mumble.com Trial server.
[17:40:41] To mohit:
hjhk
[17:42:37] Vijay connected and entered channel.
[17:45:54] Akash Gupta connected and entered channel.
[17:46:23] (Channel) Akash Gupta: hi everyone
[17:49:13] (Channel) Vijay: am i audible
[17:49:20] Vijay is now muted.
[17:49:22] To Mumble Server: no
[17:49:29] (Channel) Akash Gupta: yes
[17:49:30] (Channel) Vijay: yes
[17:49:32] (Channel) Akash Gupta: ethan
[17:49:40] (Channel) Vijay: but i am also speaking
[17:49:49] (Channel) Vijay: why i am not audible
[17:50:38] (Channel) Vijay: where is the link
[17:50:41] (Channel) Vijay: send herer
[18:01:40] mohit entered channel.
[18:02:15] Vijay is now unmuted.
[18:02:47] mohit started recording.
[18:03:23] mohit stopped recording.
[18:06:31] Vijay is now muted and deafened.
[18:06:34] Vijay is now unmuted.
[18:06:36] Vijay is now muted.
[18:06:36] Vijay is now unmuted.
Type message to channel 'Mumble Server' here
Mumble Server
Akash Gupta
ethan
mohit
Vijay
AFK
Example Channel
Public Channel
Room 1
Room 2
Activate Windows
Go to Settings to activate Windows.

7. Conclusion

Throughout the tenure of internship, we explored various VoIP softwares and compared features provided by them. We chose mumble because of it's low latency, faster, high transmission rate, Windows/ Mac/ Linux compatible, E2EE and is the best software for gaming and for normal communication. We were able to set up the communication between two systems using VoIP.

We downloaded the mumble software for client and murmur software for server. We connected on a public server using a common address and port number. Over the public network, we were able to share voice, text and images over the public server. We also set up the SuperUser account with a password. We set up the mumble server using bonjour to remove problems related to LAN connection. We tried to connect on LAN by setting up static IP in all the systems. We also downloaded the Qt app and tried to configure the interface of the app.

Finally, we were able to set up communication and were able to share voice, text and images. The work can be further carried forward to build our own customized software for voice communication which can cater for n number of users with very minmal latency, high quality and security.

8. References

- https://github.com/
- https://www.youtube.com/
- https://www.wikipedia.org/
- https://www.researchgate.net
- https://wiki.mumble.info/wiki/Installing_Mumble
- https://www.linphone.org/
- https://www.forcepoint.com/cyber-edu/osi-model
- https://github.com/qTox/qTox/tree/master/doc
- https://ozeki.hu/p_231-download-ozeki-software-products.html
- https://bonjour.en.softonic.com/