

VoIP

Computer Networks

Maunil Vyas Deep Patel Shreyas Patel Mihir Gajjar

**School of Engineering and Applied Science
Ahmedabad University**

Group - 15

Guidance By: Prof . Manish Chaturvedi

6 December, 2016

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

Implemented a VoIP system capable of :

Simple voice calling as well as conference call facility among clients on LAN, WAN, Internet.....

Implemented a VoIP system capable of :

Simple voice calling as well as conference call facility among clients on LAN, WAN, Internet.....

- Used the SIP and RTP protocols.

Implemented a VoIP system capable of :

Simple voice calling as well as conference call facility among clients on LAN, WAN, Internet.....

- Used the SIP and RTP protocols.
- Made soft-phone in Python language using support of PJSIP/PJSUA library.

Implemented a VoIP system capable of :

Simple voice calling as well as conference call facility among clients on LAN, WAN, Internet.....

- Used the SIP and RTP protocols.
- Made soft-phone in Python language using support of PJSIP/PJSUA library.
- Used Asterisk as our SIP server.

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

- Voice over Internet Protocol
- Transferring digitized voice over Internet in form of chunks/packets.

Pros:

- Reduced bandwidth with using more efficient compression mechanism which maintains quality.
- Efficient bandwidth utilization (Bandwidth can be decreased or increased based on requirement)
- Cheaper service
- Service can be setup easily

- Voice over Internet Protocol
- Transferring digitized voice over Internet in form of chunks/packets.

Pros:

- Reduced bandwidth with using more efficient compression mechanism which maintains quality.
- Efficient bandwidth utilization (Bandwidth can be decreased or increased based on requirement)
- Cheaper service
- Service can be setup easily

- Voice over Internet Protocol
- Transferring digitized voice over Internet in form of chunks/packets.

Pros:

- Reduced bandwidth with using more efficient compression mechanism which maintains quality.
- Efficient bandwidth utilization (Bandwidth can be decreased or increased based on requirement)
- Cheaper service
- Service can be setup easily

Cons:

- Voice delivery is not guaranteed (Packet may be lost).
- Unpredictable Quality of Service (IP uses Best-effort-service)

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

Why SIP ?

- There are mainly two famous open source protocols are available for call initiation facility SIP and H.323.

Functionality	H.323v3	SIP	Choice
Backward Compatibility	Compatible	Not Compatible	H.323
Fault Tolerance	Better	Good	H.323
Complexity	More	Less	SIP
Protocol Encoding	Binary(ASN.1)	Text	SIP
Debugging	Hard	Simple	SIP
Code Reuse	Difficult	Easy	SIP
Ease of Customization	Difficult	Easy	SIP
Call Control(Hold, Transfer, Forward, Wait)	Yes	Yes	Equivalent
Call Setup Delay	2-3 RTT	2-3 RTT	Equivalent
Transport Type	TCP/UDP	TCP/UDP	Equivalent
Packet Loss Recovery	Good	Good	Equivalent
Media Transport	RTP	RTP	Equivalent

Figure : Comparison of SIP and H.323

Our decision parameters for choosing SIP Over H.323

- Complexity.
 - Code Reuse
 - Call Control
 - Call Setup Delay
-
- Session Initiation Protocol [RFC 2543]. .

Our decision parameters for choosing SIP Over H.323

- Complexity.
 - Code Reuse
 - Call Control
 - Call Setup Delay
-
- Session Initiation Protocol [RFC 2543]. .
 - Designed in 1996 by Mark Handley, Henning Schulzrinne, Eve Schooler and Jonathan Rosenberg.

Our decision parameters for choosing SIP Over H.323

- Complexity.
 - Code Reuse
 - Call Control
 - Call Setup Delay
-
- **Session Initiation Protocol [RFC 2543].** .
 - Designed in 1996 by Mark Handley, Henning Schulzrinne, Eve Schooler and Jonathan Rosenberg.
 - It helps in setting up the session between calls.

Our decision parameters for choosing SIP Over H.323

- Complexity.
 - Code Reuse
 - Call Control
 - Call Setup Delay
-
- **Session Initiation Protocol [RFC 2543].** .
 - Designed in 1996 by Mark Handley, Henning Schulzrinne, Eve Schooler and Jonathan Rosenberg.
 - It helps in setting up the session between calls.
 - It is text based protocol similar to SMTP and HTTP.

Our decision parameters for choosing SIP Over H.323

- Complexity.
 - Code Reuse
 - Call Control
 - Call Setup Delay
-
- **Session Initiation Protocol [RFC 2543].** .
 - Designed in 1996 by Mark Handley, Henning Schulzrinne, Eve Schooler and Jonathan Rosenberg.
 - It helps in setting up the session between calls.
 - It is text based protocol similar to SMTP and HTTP.
 - Uses port 5060 for non-encrypted traffic and port 5061 for encrypted traffic.

Components of SIP

- User Agent .

Components of SIP

- User Agent .
- Registrar Server (Registers User Agent).

Components of SIP

- User Agent .
- Registrar Server (Registers User Agent).
- Location Server (Provides location of user to proxy servers).

Components of SIP

- User Agent .
- Registrar Server (Registers User Agent).
- Location Server (Provides location of user to proxy servers).
- Redirect Server (Redirects the SIP requests to new address)

Components of SIP

- User Agent .
- Registrar Server (Registers User Agent).
- Location Server (Provides location of user to proxy servers).
- Redirect Server (Redirects the SIP requests to new address)
- Proxy Server (Forwards SIP requests to other servers)

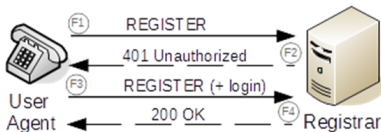


Figure : User Agent Registration Process

Components of SIP

- User Agent .
- Registrar Server (Registers User Agent).
- Location Server (Provides location of user to proxy servers).
- Redirect Server (Redirects the SIP requests to new address)
- Proxy Server (Forwards SIP requests to other servers)

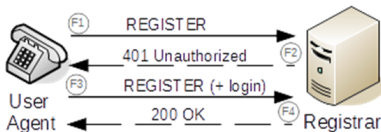


Figure : User Agent Registration Process

SIP call placing mechanism

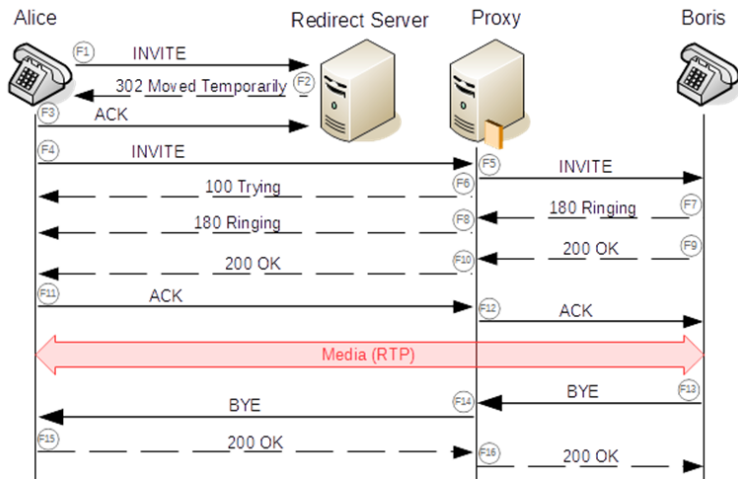


Figure : Call placing mechanism in SIP

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- **RTP**

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

- Real-time Transport Protocol [RFC 3550].

- Real-time Transport Protocol [RFC 3550].
- First published in 1996.

- Real-time Transport Protocol [RFC 3550].
- First published in 1996.
- Used to deliver audio and video between users.

- Real-time Transport Protocol [RFC 3550].
- First published in 1996.
- Used to deliver audio and video between users.
- Major RTP implementations built on UDP as we need real time streaming.

- Real-time Transport Protocol [RFC 3550].
- First published in 1996.
- Used to deliver audio and video between users.
- Major RTP implementations built on UDP as we need real time streaming.

Sub Protocols

- RTP : Data Transfer.
- RTCP : To control transmission by providing feedback and synchronization

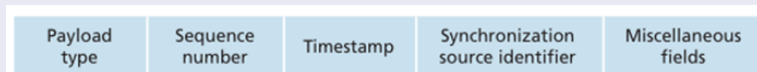
RTP

- Real-time Transport Protocol [RFC 3550].
- First published in 1996.
- Used to deliver audio and video between users.
- Major RTP implementations built on UDP as we need real time streaming.

Sub Protocols

- RTP : Data Transfer.
- RTCP : To control transmission by providing feedback and synchronization

RTP Packet



Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize



Asterisk

- We have used **Asterisk Server**.
- The single package of asterisk has a functionality of Registrar, Proxy Server, Location Server and Redirect Server.



Asterisk

- We have used **Asterisk Server**.
- The single package of asterisk has a functionality of Registrar, Proxy Server, Location Server and Redirect Server.

Installation and Configuration in Ubuntu

- Write **sudo apt-get install asterisk** in console to install asterisk.
- Info related to Configuration: **[3]**

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- **Soft-Phone**
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

Soft-Phone

- Built on Python using **PJSUA/PJSIP library**.

Soft-Phone

- Built on Python using **PJSUA/PJSIP library**.

How to use Soft Phone in our application?

- Enter your name and password.
- Enter **m** to make call, **a** to answer incoming call, **h** to hangup the call, and **q** to quit soft phone.
- After pressing **m** enter the name of user to whom you want to call.

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis

4 Summary

- Epitomize

Images of working model

```
maunil@maunil-ThinkPad-L430: ~/Desktop/cn_pro
10:24:15.469 pjsua_core.c .No SIP worker threads created
10:24:15.469 pjsua_core.c .pjsua version 2.5.5 for Linux-3.13.0.101/x86_64/glibc-2.19 initialized
10:24:15.469 pjsua_core.c .PJSUA state changed: CREATED --> INIT
10:24:15.469 pjsua_core.c SIP UDP socket reachable at 10.20.37.127:5080
10:24:15.469 udp0x2838500 SIP UDP transport started, published address is 10.20.37.127:5080
10:24:15.470 pjsua_core.c PJSUA state changed: INIT --> STARTING
10:24:15.470 sip_endpoint.c .Module "mod-unsolicited-mwi" registered
10:24:15.470 pjsua_core.c .PJSUA state changed: STARTING --> RUNNING

-----
Welcome into VOIP Service
-----
Please Enter Asterisk Server IP:>10.20.11.12
Please Enter Your Username:>maunil
Please Enter your Password:>maunil
```

Figure : User 1 Registration

Images of working model

```
maunil@maunil-ThinkPad-L430: ~/Desktop/cn_pro
Contact: <sip:maunil@10.20.37.127:5080;ob>;expires=300
Date: Tue, 06 Dec 2016 04:59:40 GMT
Content-Length: 0

--end msg--

10:24:53.478 pjsua_acc.c ....SIP outbound status for acc 0 is not active
10:24:53.478 pjsua_acc.c ....<sip:maunil@10.20.11.12>: registration success,
status=200 (OK), will re-register in 300 seconds
10:24:53.478 pjsua_acc.c ....Keep-alive timer started for acc 0, destination
:10.20.11.12:5060, interval:15s

Registration complete, status= 200 (OK)

Press ENTER to quit

Menu: m=make call, h=hangup call, a=answer call, q=quit
m
Enter destination URI to call: deep
```

Figure : Making Call to deep

Images of working model

```
maunil@maunil-ThinkPad-L430: ~/Desktop/cn_pro
--end msg--

10:28:48.933 pjsua_core.c .RX 560 bytes Response msg 180/INVITE/cseq=644 (rd
ta0x7f2420002988) from UDP 10.20.11.12:5060:
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP 10.20.37.127:5080;branch=z9hG4bKPjw6XltWhYl38FcuJLCfEk6mh12wAku
y7W;received=10.20.37.127;rport=5080
From: <sip:maunil@10.20.11.12>;tag=v3szmoUh-ub-qbdzqXAceGPzBgYQ7cPt
To: sip:deep@10.20.11.12;tag=as7c8d55b2
Call-ID: 10GyP4GufS5fstZDgGX-qre2Cc7kUvh9
CSeq: 644 INVITE
Server: Asterisk PBX 11.7.0-dfsg-1ubuntu1
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLIS
H
Supported: replaces, timer
Session-Expires: 1800;refresher=uas
Contact: <sip:deep@10.20.11.12:5060>
Content-Length: 0

--end msg--

Call with sip:deep@10.20.11.12 is EARLY last code = 180 (Ringing)
```

Figure : Ringing info at user1

Images of working model

```
ramkabin@Ramkabin: ~/Documents/CN_PRJ
Expires: 300
Contact: <sip:deep@10.20.11.17:5080;ob>;expires=300
Date: Tue, 06 Dec 2016 05:02:49 GMT
Content-Length: 0

--end msg--

10:28:02.943    pjsua_acc.c ....SIP outbound status for acc 0 is not active

10:28:02.943    pjsua_acc.c ....<sip:deep@10.20.11.12>: registration success, s
tatus=200 (OK), will re-register in 300 seconds

10:28:02.943    pjsua_acc.c ....Keep-alive timer started for acc 0, destination
:10.20.11.12:5060, interval:15s

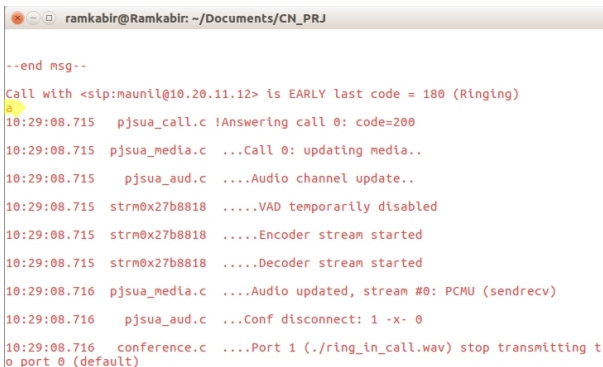
Registration complete, status= 200 (OK)

Press ENTER to quit

Menu: m=make call, h=hangup call, a=answer call, q=quit
10:28:48.948    pjsua_core.c .RX 863 bytes Request msg INVITE/cseq=102 (rdata0x7
```

Figure : User 2 Registration

Images of working model



```
ramkabr@Ramkabr: ~/Documents/CN_PRJ

--end msg--
Call with <sip:maunil@10.20.11.12> is EARLY last code = 180 (Ringing)
10:29:08.715 pjsua_call.c !Answering call 0: code=200
10:29:08.715 pjsua_media.c ...Call 0: updating media..
10:29:08.715 pjsua_aud.c ....Audio channel update..
10:29:08.715 str0x27b8818 .....VAD temporarily disabled
10:29:08.715 str0x27b8818 .....Encoder stream started
10:29:08.715 str0x27b8818 .....Decoder stream started
10:29:08.716 pjsua_media.c ....Audio updated, stream #0: PCMU (sendrecv)
10:29:08.716 pjsua_aud.c ...Conf disconnect: 1 -x- 0
10:29:08.716 conference.c ....Port 1 (./ring_in_call.wav) stop transmitting t
o port 0 (default)
```

Figure : User 2 Answering

Images of working model

```
ramkabr@Ramkabr: ~/Documents/CN_PRJ
a=fmtp:101 0-16

--end msg--

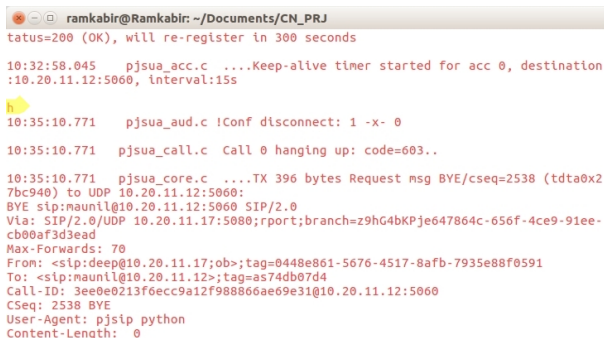
Call with <sip:maunil@10.20.11.12> is CONNECTING last code = 200 (OK)
Menu: m=make call, h=hangup call, a=answer call, q=quit
10:29:08.755 pjsua_core.c !RX 425 bytes Request msg ACK/cseq=102 (rdata0x7f6d
e0003d68) from UDP 10.20.11.12:5060:
ACK sip:deep@10.20.11.17:5080;ob SIP/2.0
Via: SIP/2.0/UDP 10.20.11.12:5060;branch=z9hG4bK229cb2aa
Max-Forwards: 70
From: <sip:maunil@10.20.11.12>;tag=as74db07d4
To: <sip:deep@10.20.11.17:5080;ob>;tag=0448e861-5676-4517-8afb-7935e88f0591
Contact: <sip:maunil@10.20.11.12:5060>
Call-ID: 3ee0e0213f6ecc9a12f988866ae69e31@10.20.11.12:5060
CSeq: 102 ACK
User-Agent: Asterisk PBX 11.7.0-dfsg-1ubuntu1
Content-Length: 0

--end msg--

Call with <sip:maunil@10.20.11.12> is CONFIRMED last code = 200 (OK)
```

Figure : Call connection info at User 2

Images of working model



A terminal window titled 'ramkabit@Ramkabit: ~/Documents/CN_PRJ' displays SIP call logs. The logs show a call starting at 10:32:58.045, followed by a hangup at 10:35:10.771. The hangup sequence includes a BYE message from the user, a 200 OK response from the server, and a 486 Busy Here response from the user. The terminal text is as follows:

```
ramkabit@Ramkabit: ~/Documents/CN_PRJ
tatus=200 (OK), will re-register in 300 seconds

10:32:58.045    pjsua_acc.c    ....Keep-alive timer started for acc 0, destination
:10.20.11.12:5060, interval:15s

h
10:35:10.771    pjsua_aud.c    !Conf disconnect: 1 -x- 0

10:35:10.771    pjsua_call.c    Call 0 hanging up: code=603..

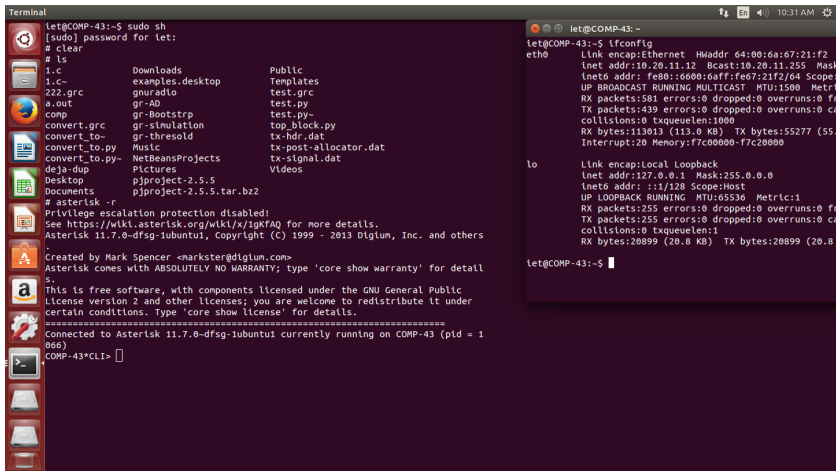
10:35:10.771    pjsua_core.c    ....TX 396 bytes Request msg BYE/cseq=2538 (tdta0x2
7bc940) to UDP 10.20.11.12:5060:
BYE sip:maunil@10.20.11.12:5060 SIP/2.0
Via: SIP/2.0/UDP 10.20.11.17:5080;rport;branch=z9hG4bKPje647864c-656f-4ce9-91ee-
cb00af3d3ead
Max-Forwards: 70
From: <sip:deep@10.20.11.17;ob>;tag=0448e861-5676-4517-8afb-7935e88f0591
To: <sip:maunil@10.20.11.12>;tag=as74db07d4
Call-ID: 3ee0e0213f6ecc9a12f988866ae69e31@10.20.11.12:5060
CSeq: 2538 BYE
User-Agent: pjsip python
Content-Length: 0

10:35:10.771    pjsua_core.c    ....RX 104 bytes Response msg 200 OK (tdta0x2
7bc940) from UDP 10.20.11.12:5060:
200 OK SIP/2.0
Via: SIP/2.0/UDP 10.20.11.12:5060;branch=z9hG4bKPje647864c-656f-4ce9-91ee-
cb00af3d3ead
Max-Forwards: 70
From: <sip:maunil@10.20.11.12>;tag=as74db07d4
To: <sip:deep@10.20.11.17;ob>;tag=0448e861-5676-4517-8afb-7935e88f0591
Call-ID: 3ee0e0213f6ecc9a12f988866ae69e31@10.20.11.12:5060
CSeq: 2538 BYE
User-Agent: pjsip python
Content-Length: 0

10:35:10.771    pjsua_core.c    ....TX 104 bytes Response msg 486 Busy Here (tdta0x2
7bc940) to UDP 10.20.11.12:5060:
486 Busy Here SIP/2.0
Via: SIP/2.0/UDP 10.20.11.12:5060;branch=z9hG4bKPje647864c-656f-4ce9-91ee-
cb00af3d3ead
Max-Forwards: 70
From: <sip:maunil@10.20.11.12>;tag=as74db07d4
To: <sip:deep@10.20.11.17;ob>;tag=0448e861-5676-4517-8afb-7935e88f0591
Call-ID: 3ee0e0213f6ecc9a12f988866ae69e31@10.20.11.12:5060
CSeq: 2538 BYE
User-Agent: pjsip python
Content-Length: 0
```

Figure : User 2 Hangup

Images of working model



The image shows a Linux terminal window with a dark background and a sidebar of application icons on the left. The terminal output is as follows:

```
let@COMP-43:~$ sudo sh
[sudo] password for let:
# clear
# ls
1.c          Downloads      Public
1.c~         examples.desktop Templates
222.grc      gnuradio      test.grc
a.out        gr-AD         test.py
comp         gr-Bootstrp   test.py~
convert.grc  gr-simulation top_block.py
convert_io~  gr-threshold  tx-hdr.dat
convert_to.py Music         tx-post-allocator.dat
convert_to.py~ NetBeansProjects tx-signal.dat
deja-dup     Pictures       Videos
Desktop      pjproject-2.5.5
Documents    pjproject-2.5.5.tar.bz2
# asterisk -r
Privilege escalation protection disabled!
See https://wiki.asterisk.org/wiki/x/1gKFAQ for more details.
Asterisk 11.7.0-dfsg-1ubuntu1, Copyright (C) 1999 - 2013 Digium, Inc. and others
.
Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details
.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 11.7.0-dfsg-1ubuntu1 currently running on COMP-43 (pid = 1
866)
COMP-43*CLI> 
```

On the right side of the terminal window, there is a separate window titled 'let@COMP-43: ~' showing the output of the 'ifconfig' command:

```
let@COMP-43:~$ ifconfig
eth0      Link encap:Ethernet  HWaddr 64:00:6a:07:21:f2
          inet addr:10.20.11.12  Bcast:10.20.11.255  Mask
          inet6 addr: fe80::6600:6aff:fe67:21f2/64 Scope:
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metr
          RX packets:581 errors:0 dropped:0 overruns:0 fr
          TX packets:439 errors:0 dropped:0 overruns:0 ca
          collisions:0 txqueuelen:1000
          RX bytes:113013 (113.0 KB)  TX bytes:55277 (55.
          Interrupt:20 Memory:f7c00000-f7c20000

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:255 errors:0 dropped:0 overruns:0 fr
          TX packets:255 errors:0 dropped:0 overruns:0 ca
          collisions:0 txqueuelen:1
          RX bytes:20899 (20.8 KB)  TX bytes:20899 (20.8
```

Figure : Asterisk Server

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

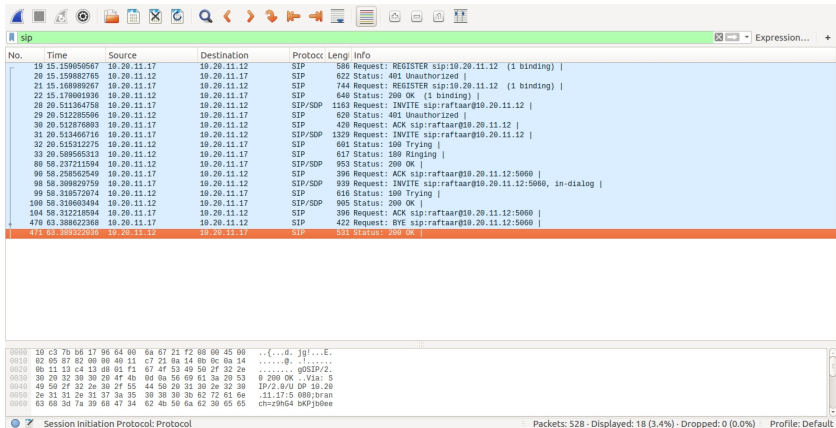
3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- **Wire shark Analysis**

4 Summary

- Epitomize

Wire shark Analysis



No.	Time	Source	Destination	Protocol	Length	Info
19	15.159050567	10.20.11.17	10.20.11.12	SIP	586	Request: REGISTER sip:10.20.11.12 (1 binding)
20	15.159882765	10.20.11.12	10.20.11.17	SIP	622	Status: 401 Unauthorized
21	15.160909267	10.20.11.17	10.20.11.12	SIP	744	Request: REGISTER sip:10.20.11.12 (1 binding)
22	15.178691936	10.20.11.12	10.20.11.17	SIP	640	Status: 200 OK (1 binding)
28	20.511364758	10.20.11.17	10.20.11.12	SIP/SDP	1163	Request: INVITE sip:raftaar@10.20.11.12
29	20.512285596	10.20.11.12	10.20.11.17	SIP	620	Status: 401 Unauthorized
30	20.512876803	10.20.11.17	10.20.11.12	SIP	420	Request: ACK sip:raftaar@10.20.11.12
31	20.513466716	10.20.11.17	10.20.11.12	SIP/SDP	1329	Request: INVITE sip:raftaar@10.20.11.12
32	20.515312275	10.20.11.12	10.20.11.17	SIP	601	Status: 100 Trying
33	20.5095055313	10.20.11.12	10.20.11.17	SIP	617	Status: 180 Ringing
80	58.237211594	10.20.11.12	10.20.11.17	SIP/SDP	953	Status: 200 OK
90	58.258562549	10.20.11.17	10.20.11.12	SIP	396	Request: ACK sip:raftaar@10.20.11.12:5060
98	58.309829759	10.20.11.17	10.20.11.12	SIP/SDP	939	Request: INVITE sip:raftaar@10.20.11.12:5060, in-dialog
99	58.310572074	10.20.11.12	10.20.11.17	SIP	616	Status: 100 Trying
100	58.310603494	10.20.11.12	10.20.11.17	SIP/SDP	905	Status: 200 OK
104	58.312218594	10.20.11.17	10.20.11.12	SIP	396	Request: ACK sip:raftaar@10.20.11.12:5060
470	63.388622368	10.20.11.17	10.20.11.12	SIP	422	Request: BYE sip:raftaar@10.20.11.12:5060
471	63.389322036	10.20.11.12	10.20.11.17	SIP	551	Status: 200 OK

0000	10 c3 7b b6 17 96 64 00 6a 67 21 f2 08 00 45 00	..{...d. jg!...E.
0010	02 05 87 82 00 00 40 11 c7 21 0a 14 0b 0c 0a 14@..!.....
0020	0b 11 13 c4 c5 10 08 61 f1 67 4f 53 49 50 2f 32 2eg0SIP/2.
0030	30 20 32 30 30 20 4f 4b 04 0a 56 69 61 3a 20 53	0 200 OK Via: S
0040	49 50 2f 32 2e 30 2f 55 44 50 20 31 30 2e 32 30	IP/2.0/UDP 10.20
0050	2e 31 31 2e 31 37 3a 35 30 30 30 3b 62 72 61 6e	.11.17:5 000;bran
0060	63 68 3d 7a 39 68 47 34 62 4b 50 6a 62 39 65 65	ch=z9hG4 bKpJb6ee

Session Initiation Protocol: Protocol Packets: 528 - Displayed: 18 (3.4%) - Dropped: 0 (0.0%) Profile: Default

Figure : Call Initiation Using SIP

Wire shark Analysis

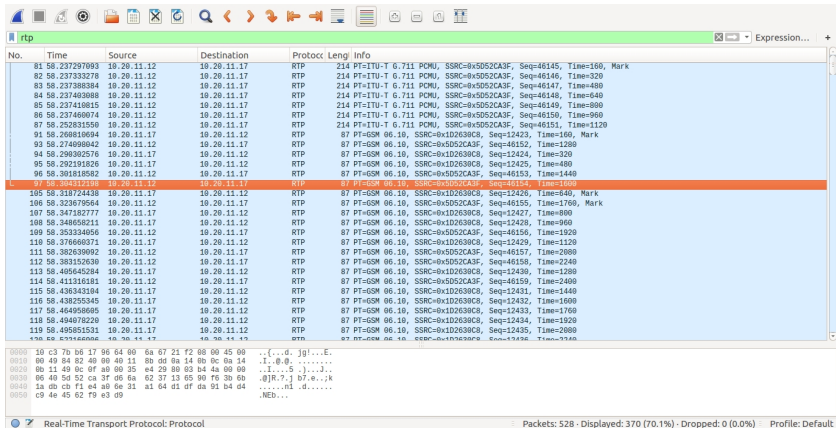


Figure : RTP

Wire shark Analysis

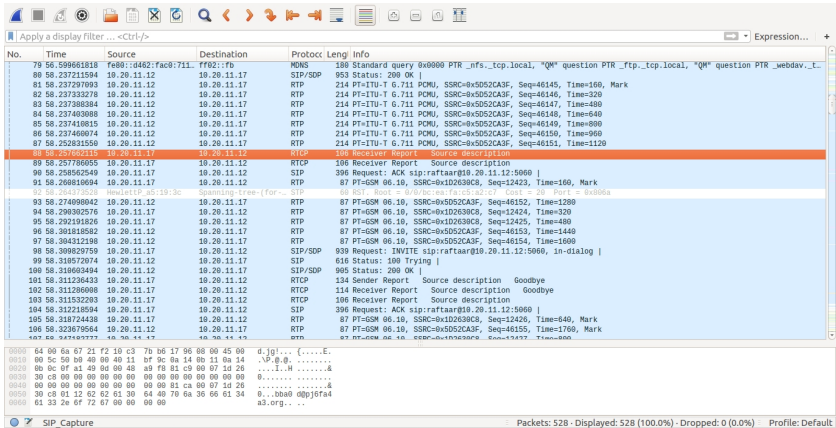


Figure : RTCP

Outline

1 Project Overview

- Abstract

2 Protocols

- VoIP
- SIP
- RTP

3 Project Parameters

- Setting Up SIP Server
- Soft-Phone
- Images of working model
- Wire shark Analysis


4 Summary

- Epitomize


Summary


- Successfully Implemented VoIP.
- Future Goals:
 - Android Application with better voice quality.
 - Video Calling.

References I

 Jim Kurose, Keith W. Ross,
Computer Networking: A Top-Down Approach.
PEARSON, November 2000.

 WIKIPEDIA.
For Images.

 Dalgic I, Fang H. Comparison of H. 323 and SIP for IP Telephony
Signaling. In Photonics East'99 1999 Nov 22 (pp. 106-122).
International Society for Optics and Photonics.
For Asterisk Configuration.

 Papageorgiou, P., 2001. A comparison of H. 323 vs SIP. University of
Maryland.
For Comparison of H.323 and SIP

References II



[http : //most –
VoIP.readthedocs.io/en/latest/asteriskdocs/asteriskconfiguration.html](http://most-VoIP.readthedocs.io/en/latest/asteriskdocs/asteriskconfiguration.html)

For Asterisk Configuration.



<http://www.pjsip.org/python/pjsua.html>.
PJSUA Python library Support