



EE 284

PROJECT 1:

Voice over Wireless Ad-Hoc Network, A Hands-on SIP-based VoIP Experiments on: Call

Establishment, Busy Lines, Call on Hold, and Conference Calling

BY: AMITKUMAR H PATEL(011451284)
KUNAL BHAGCHANDANI(011478883)

Department Of Electrical Engineering

Table of Contents:

1.PART-I:	3
PHASE 1: ESTABLISH AND ANALYZE A SUCCESSFUL CALL BETWEEN 2 SIP CLIENTS.	6
PHASE 2: BUSY USER:.....	15
PHASE 3 - CALL ON HOLD:.....	18
PHASE 4 – CALL CONFERENCING:	29
2.PART-II BASIC SIP CLIENT	38
3.SIP PACKET FIELDS:.....	50
4.SIP 3 WAY HANDSHAKE:	52
5.AD-HOC NETWORKS:	53

1.PART-I:

For this part we used the virtual machine approach as creation of Ad-Hoc networks was not supported.

The below is the network settings available for VMWare fusion we used:

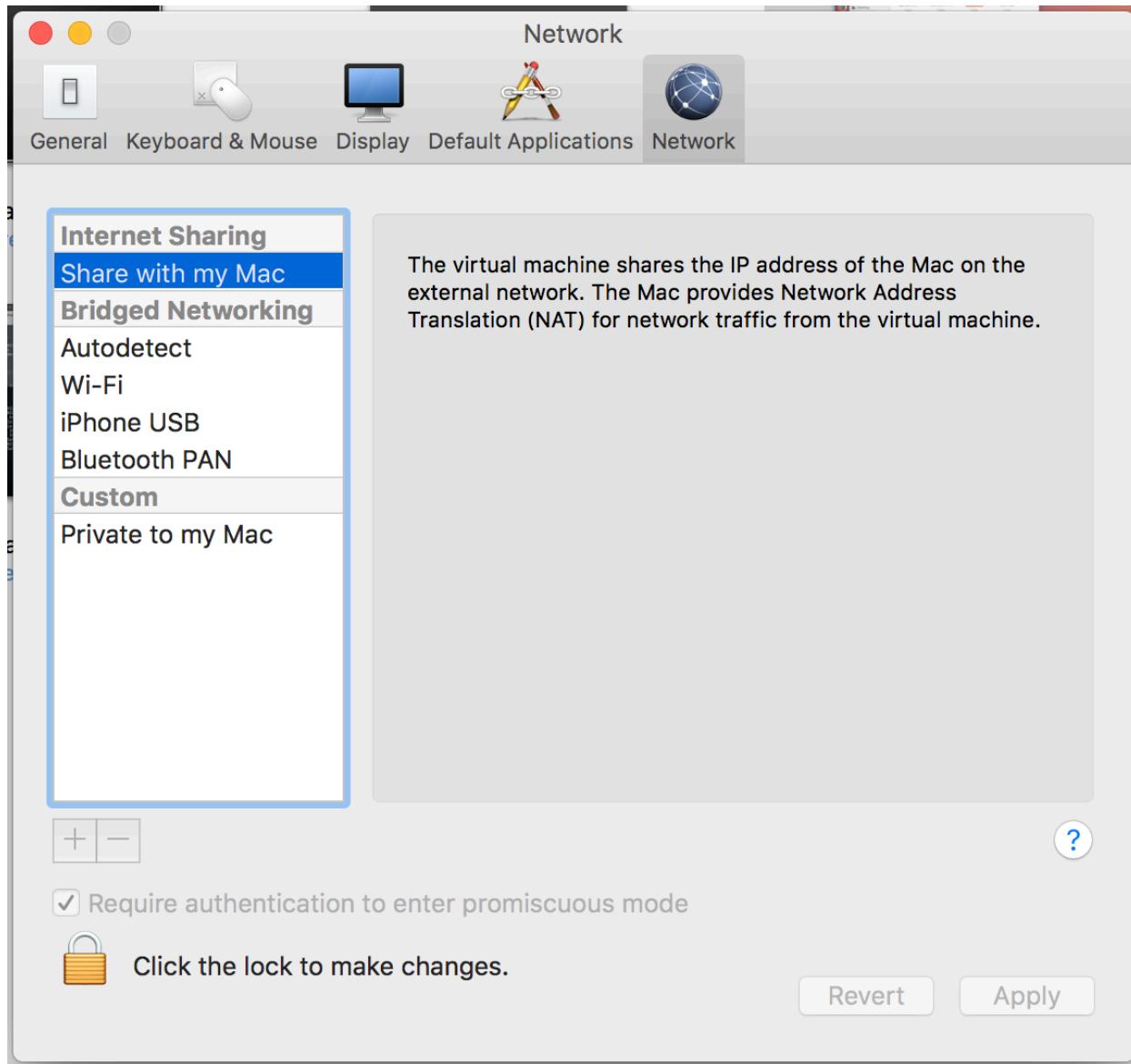
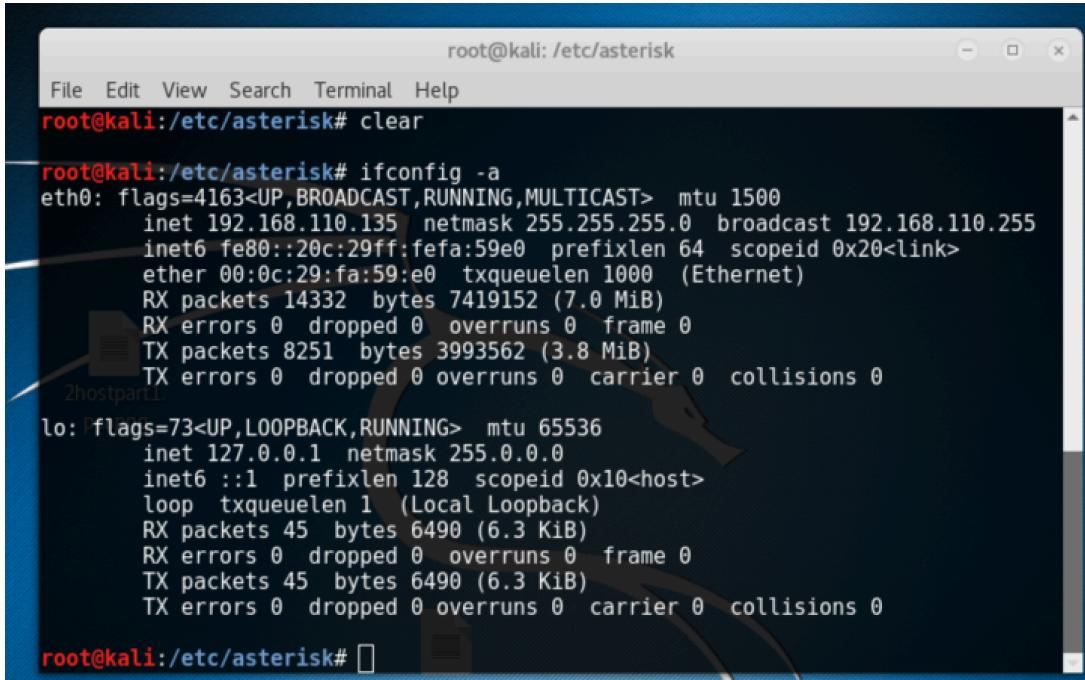


Figure : 1.0.1

The asterisk server is installed on Kali Linux the machine details of which are as under:



```
root@kali:/etc/asterisk
File Edit View Search Terminal Help
root@kali:/etc/asterisk# clear

root@kali:/etc/asterisk# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
        inet 192.168.110.135 netmask 255.255.255.0 broadcast 192.168.110.255
        inet6 fe80::20c:29ff:fe:59:e0 prefixlen 64 scopeid 0x20<link>
          ether 00:0c:29:fa:59:e0 txqueuelen 1000 (Ethernet)
            RX packets 14332 bytes 7419152 (7.0 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 8251 bytes 3993562 (3.8 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
  2hostpart1

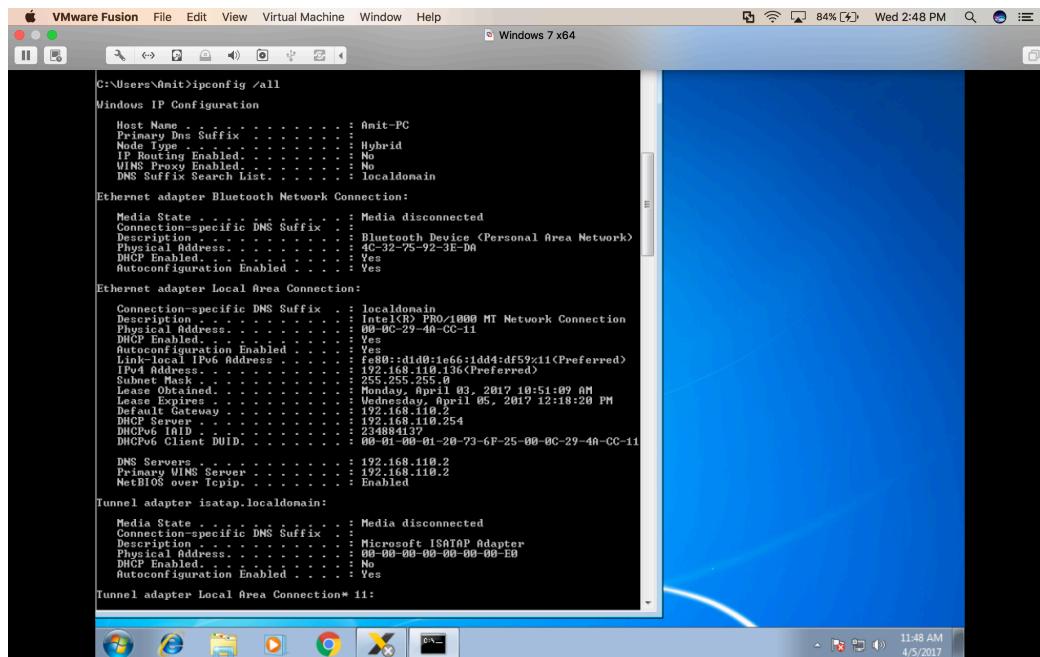
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
        inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
          loop txqueuelen 1 (Local Loopback)
            RX packets 45 bytes 6490 (6.3 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 45 bytes 6490 (6.3 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

root@kali:/etc/asterisk#
```

Figure : 1.0.2

Three clients as per requirement of project were created in VMWare Fusion with Windows 7 ultimate OS with usernames :2000, 2010 and 2009 respectively whose configs are as under:

2000:



```
C:\Users\Amit>ipconfig /all
Windows IP Configuration

Host Name . . . . . : Amit-PC
Primary Dns Suffix . . . . . :
Node Type . . . . . : Hybrid
WINS Proxy Enabled. . . . . : No
DNS Suffix Search List. . . . . :

Ethernet adapter Bluetooth Network Connection:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . . . :
  Description . . . . . : Bluetooth Device (Personal Area Network)
  Physical Address . . . . . : 4C-32-75-92-3E-DA
    DHCP Enabled . . . . . : Yes
    Autoconfiguration Enabled . . . . . : Yes

Ethernet adapter Local Area Connection:
  Connection-specific DNS Suffix . . . . . : localdomain
  Description . . . . . : Intel(R) PRO/1000 MT Network Connection
  Physical Address . . . . . : 00-0C-29-48-CC-11
    DHCP Enabled . . . . . : Yes
    Autoconfiguration Enabled . . . . . : Yes
    IPv4 Address . . . . . : fe80::1d0:1e66:1dd4:df59%11(PREFERRED)
                               192.168.110.136(Preferred)
    Subnet Mask . . . . . : 255.255.255.0
    Lease Obtained . . . . . : Monday, April 03, 2017 10:51:09 AM
    Lease Expires . . . . . : Wednesday, April 05, 2017 12:18:20 PM
    Default Gateway . . . . . : 192.168.110.254
    DHCPv6 IID . . . . . : 23484137
    DHCPv6 Client DUID . . . . . : 00-01-00-01-20-73-6F-25-00-0C-29-4A-CC-11
    DNS Servers . . . . . : 192.168.110.2
    Primary WINS Server . . . . . : 192.168.110.2
    NetBIOS over Tcpip . . . . . : Enabled

Tunnel adapter isatap.localdomain:
  Media State . . . . . : Media disconnected
  Connection-specific DNS Suffix . . . . . :
  Description . . . . . : Microsoft ISATAP Adapter
  Physical Address . . . . . : 00-00-00-00-00-00-E0
    DHCP Enabled . . . . . : No
    Autoconfiguration Enabled . . . . . : Yes
```

Figure : 1.0.3

2010:

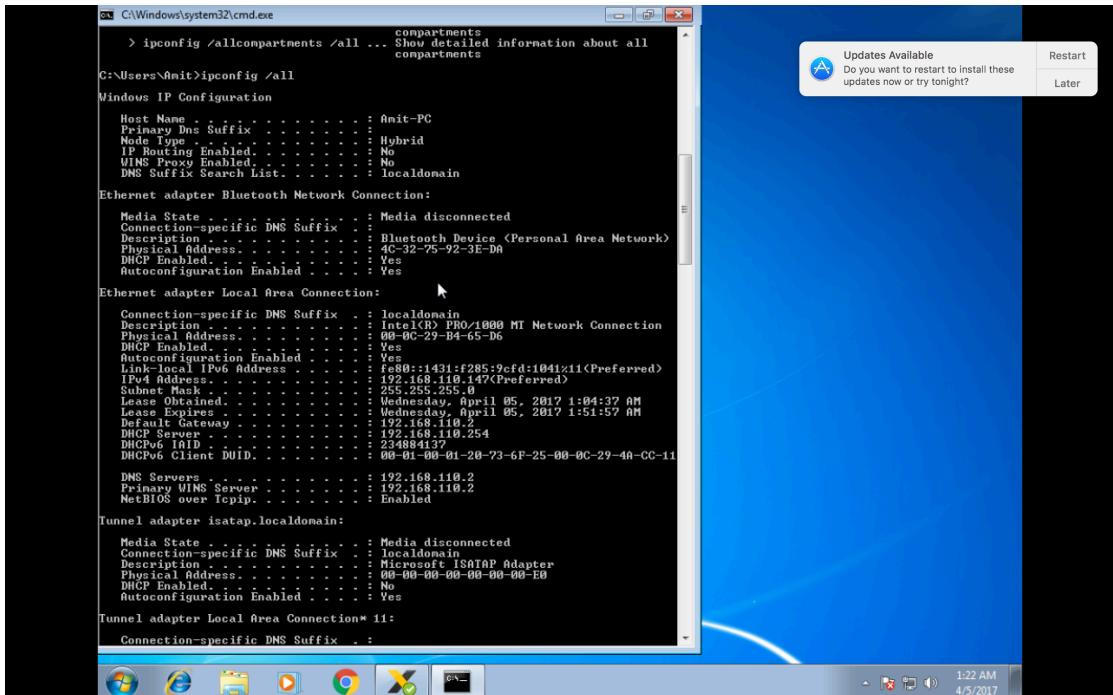


Figure : 1.0.4

2009:

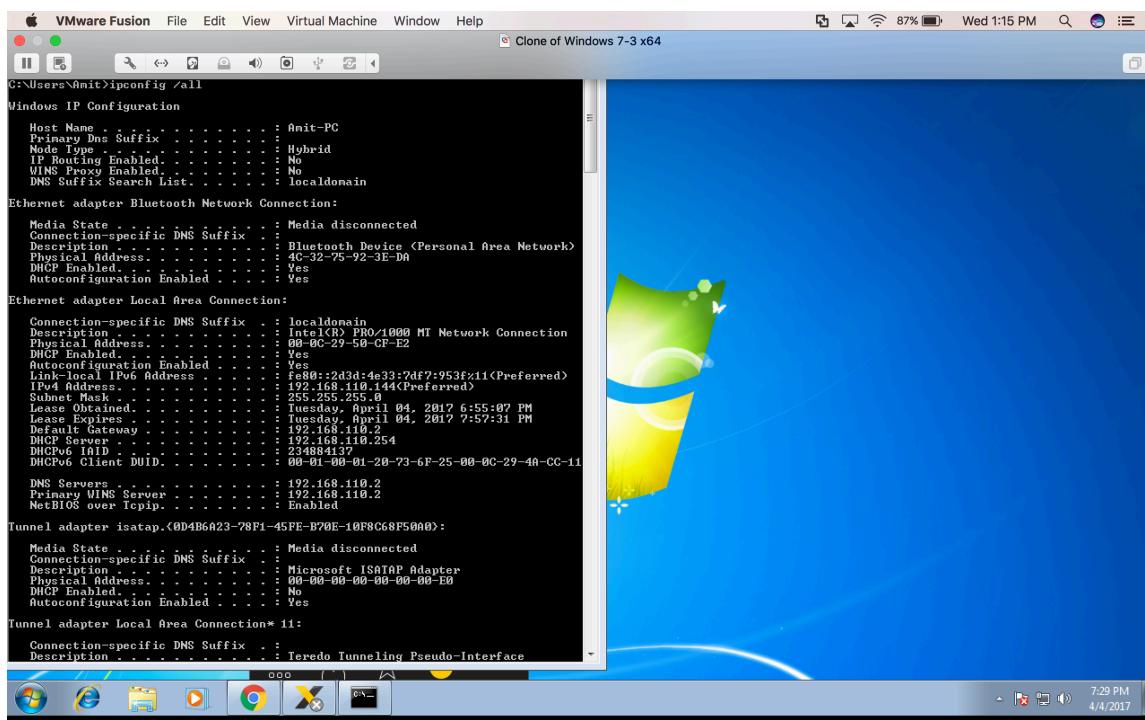
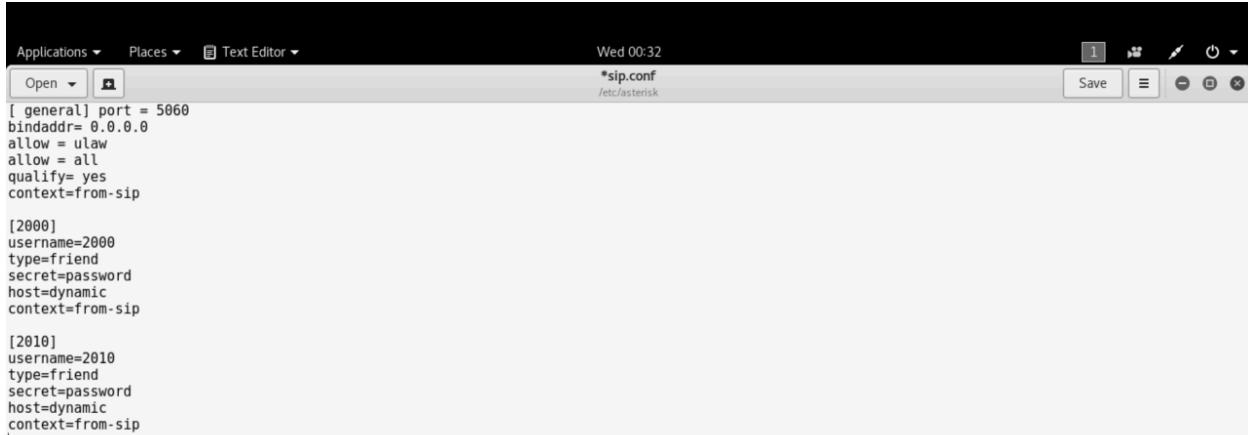


Figure : 1.0.5

PHASE 1: Establish and Analyze a Successful Call Between 2 SIP Clients.

Here we use two sip clients 2000 and 2010 and call from 2000 to 2010 in order to establish call.

Below figure 6 and 7 are sip.conf and extensions.conf files respectively.



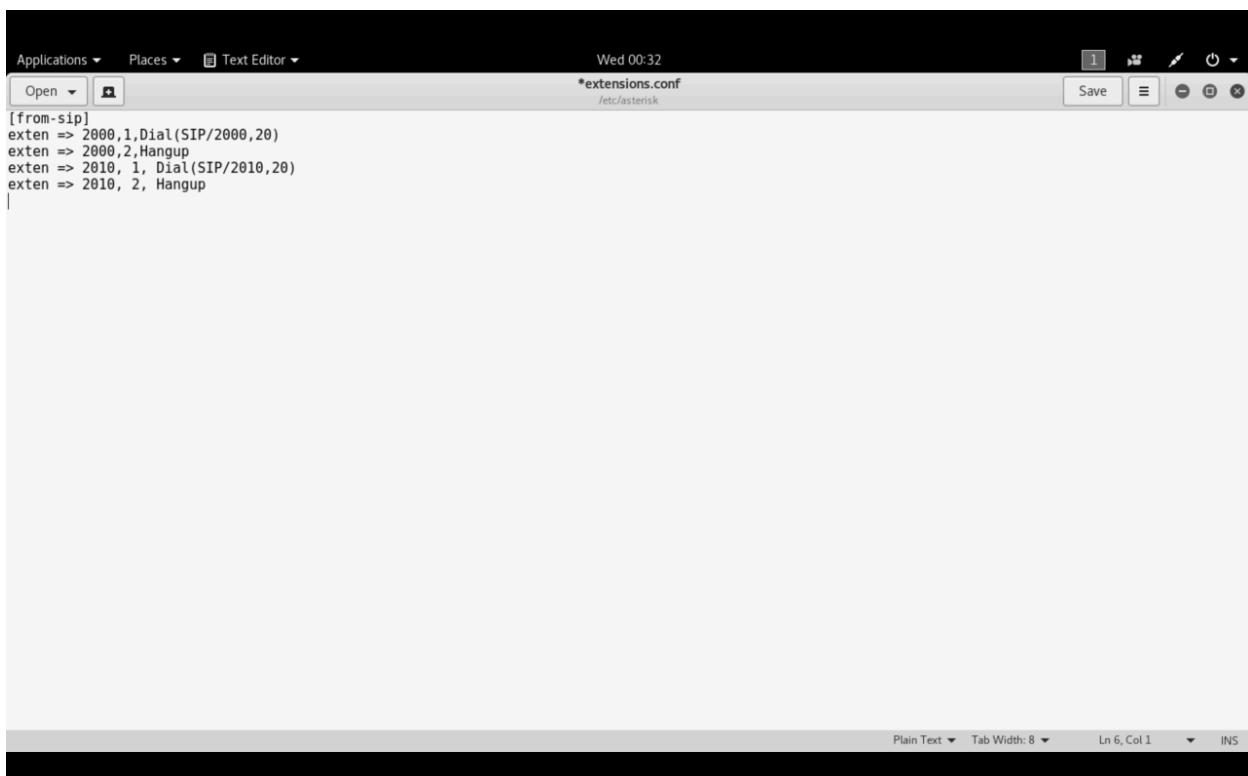
The screenshot shows a text editor window titled "sip.conf" located at "/etc/asterisk". The file contains the following configuration:

```
[general]
port = 5060
bindaddr= 0.0.0.0
allow = ulaw
allow = all
qualify= yes
context=from-sip

[2000]
username=2000
type=friend
secret=password
host=dynamic
context=from-sip

[2010]
username=2010
type=friend
secret=password
host=dynamic
context=from-sip
```

Figure : 1.1.1



The screenshot shows a text editor window titled "extensions.conf" located at "/etc/asterisk". The file contains the following configuration:

```
[from-sip]
exten => 2000,1,Dial(SIP/2000,20)
exten => 2000,2,Hangup
exten => 2010, 1, Dial(SIP/2010,20)
exten => 2010, 2, Hangup
```

Figure : 1.1.2

The below figure shows registration details of a Exlite SIP account for client 2010

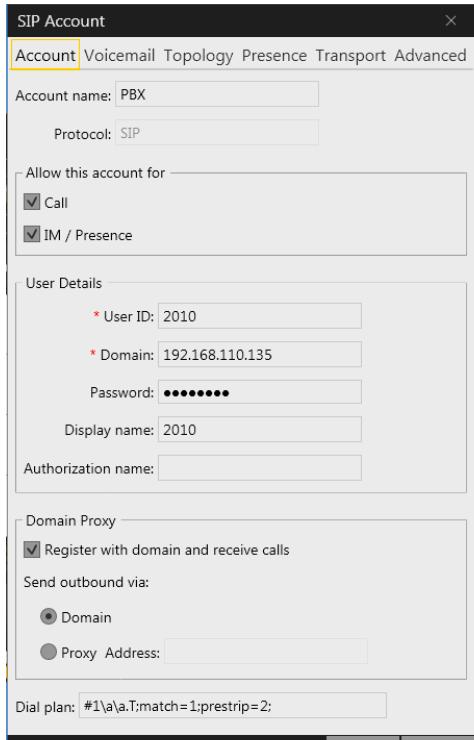


Figure : 1.1.3

This screenshot shows caller 2000 calling 2010



Figure : 1.1.4

2010 online:

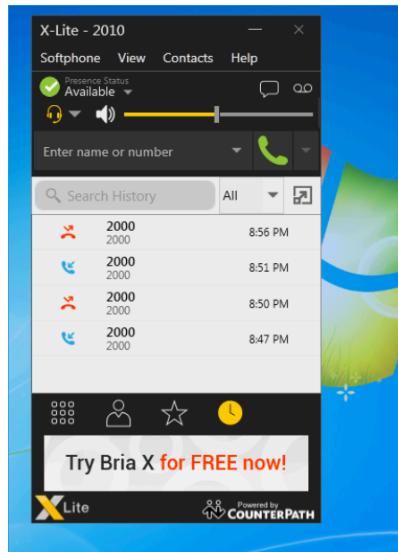


Figure : 1.1.5

Incoming call at 2010:

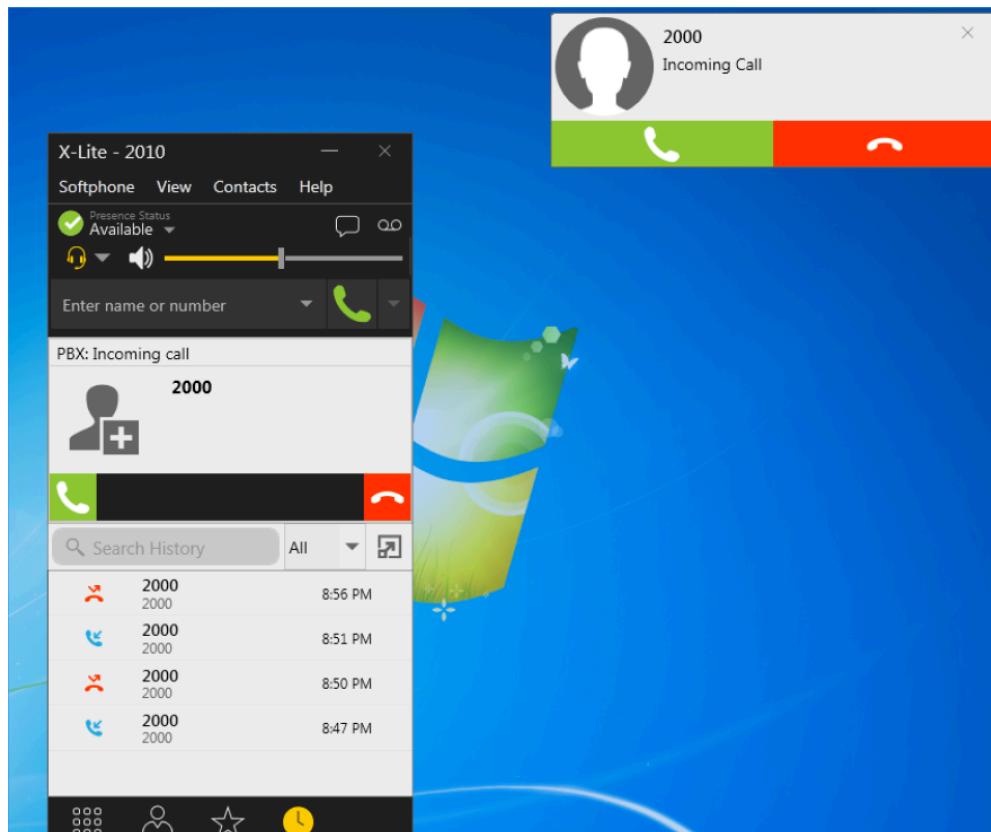


Figure : 1.1.6

Call established between 2000 and 2010:

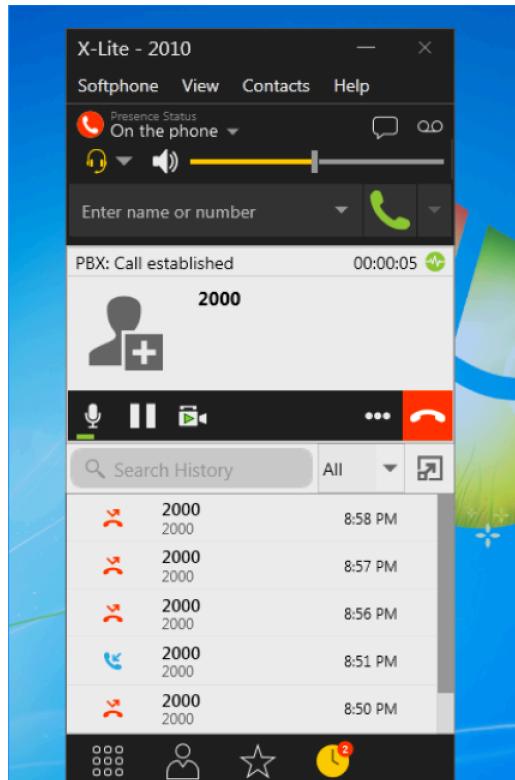


Figure : 1.1.7

Various events at asterisk server:

A screenshot of a terminal window on a Kali Linux system. The terminal title is "root@kali: ~". The window shows the output of the "sudo service asterisk start" command. The log output includes the Asterisk version (13.14.0-dfsg-1), copyright information, and a notice about the lack of warranty. It also shows several "NOTICE" and "WARNING" messages related to SIP channel handling and socket configuration. The terminal prompt "kali*CLI>" is visible at the bottom.

Figure : 1.1.8

Wireshark Capture during phase 1 with SIP filter applied:

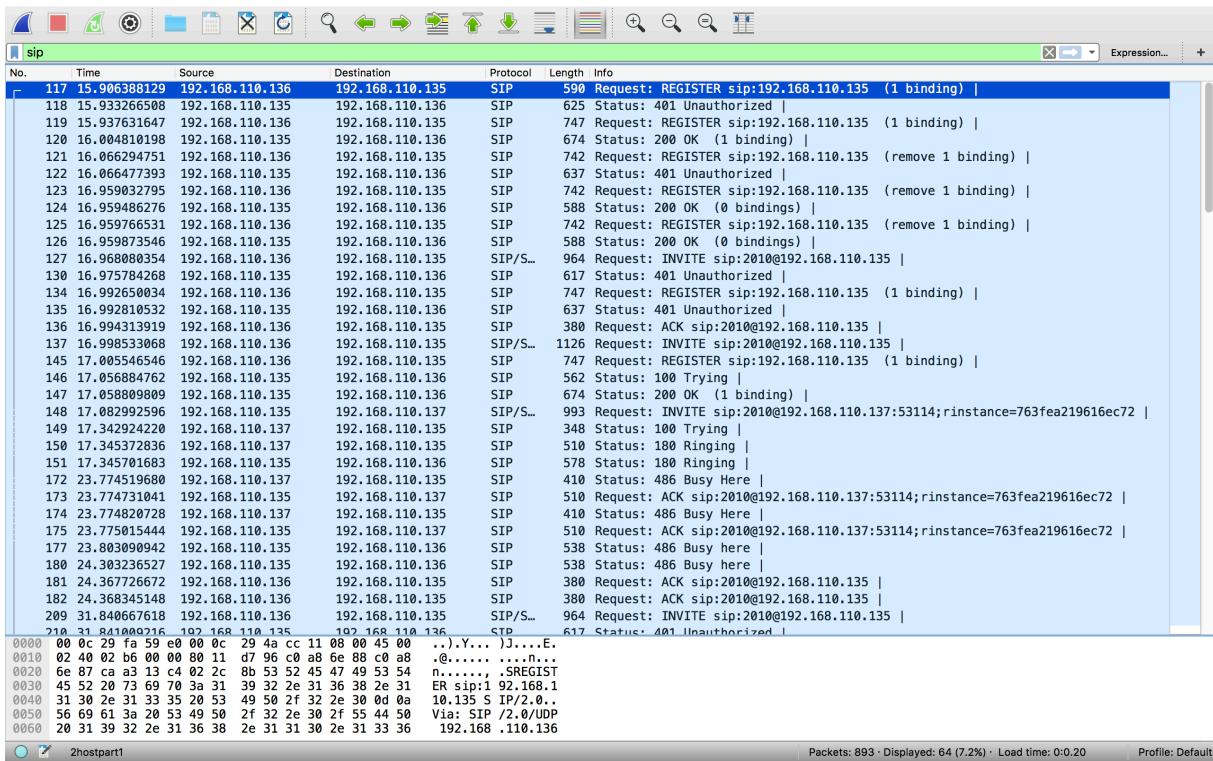


Figure : 1.1.9

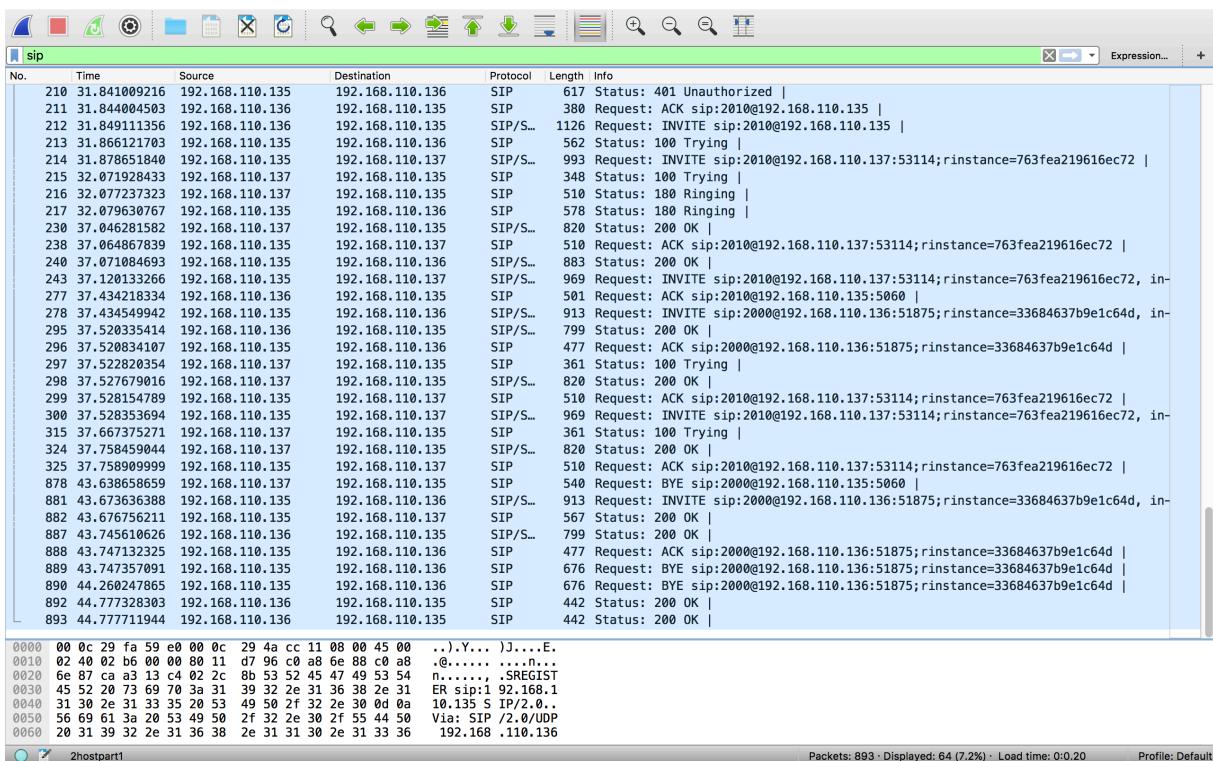


Figure : 1.1.10

Timing Diagrams of SIP packets:

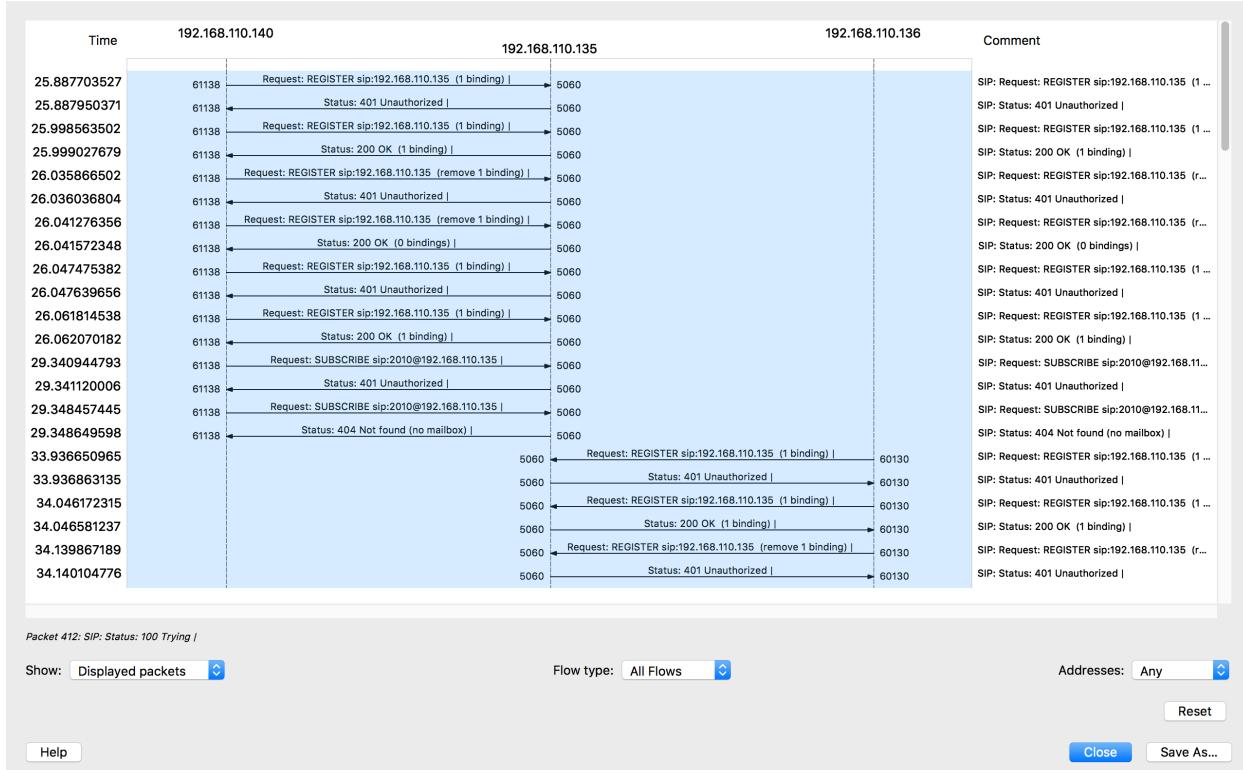


Figure : 1.1.11

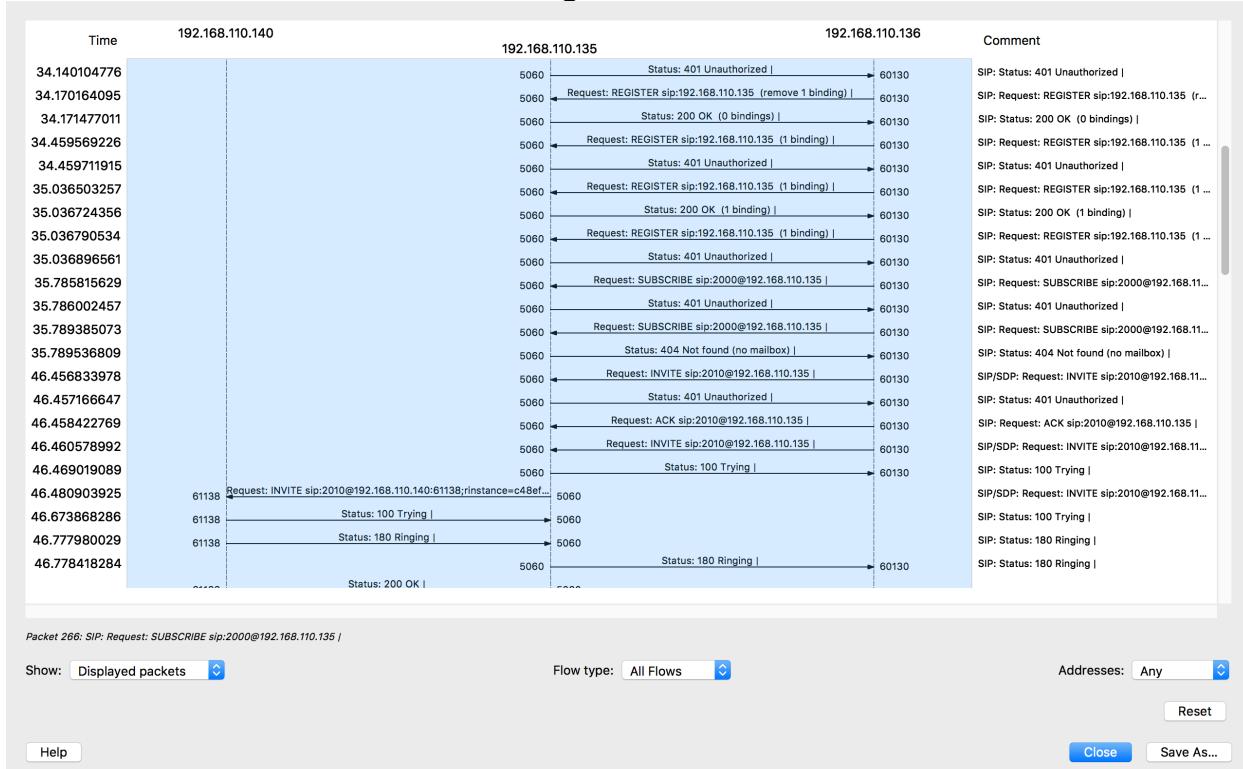


Figure : 1.1.12

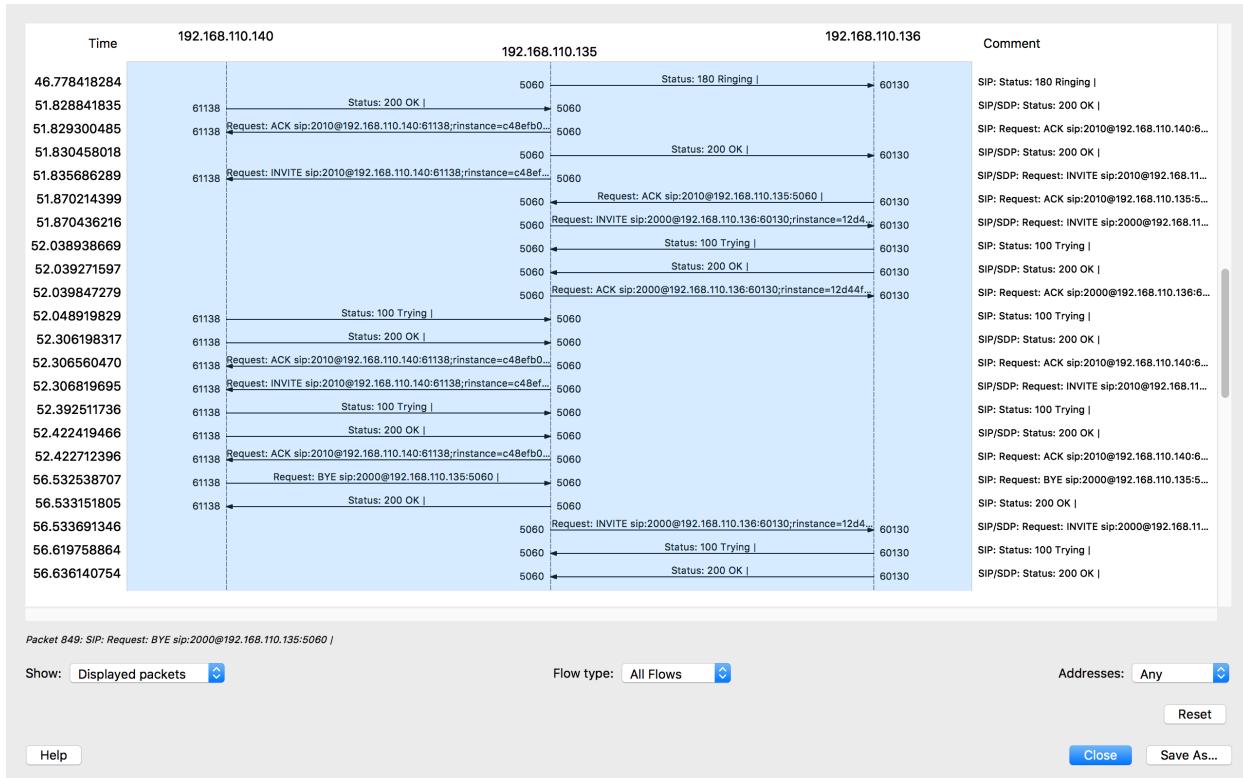


Figure : 1.1.13

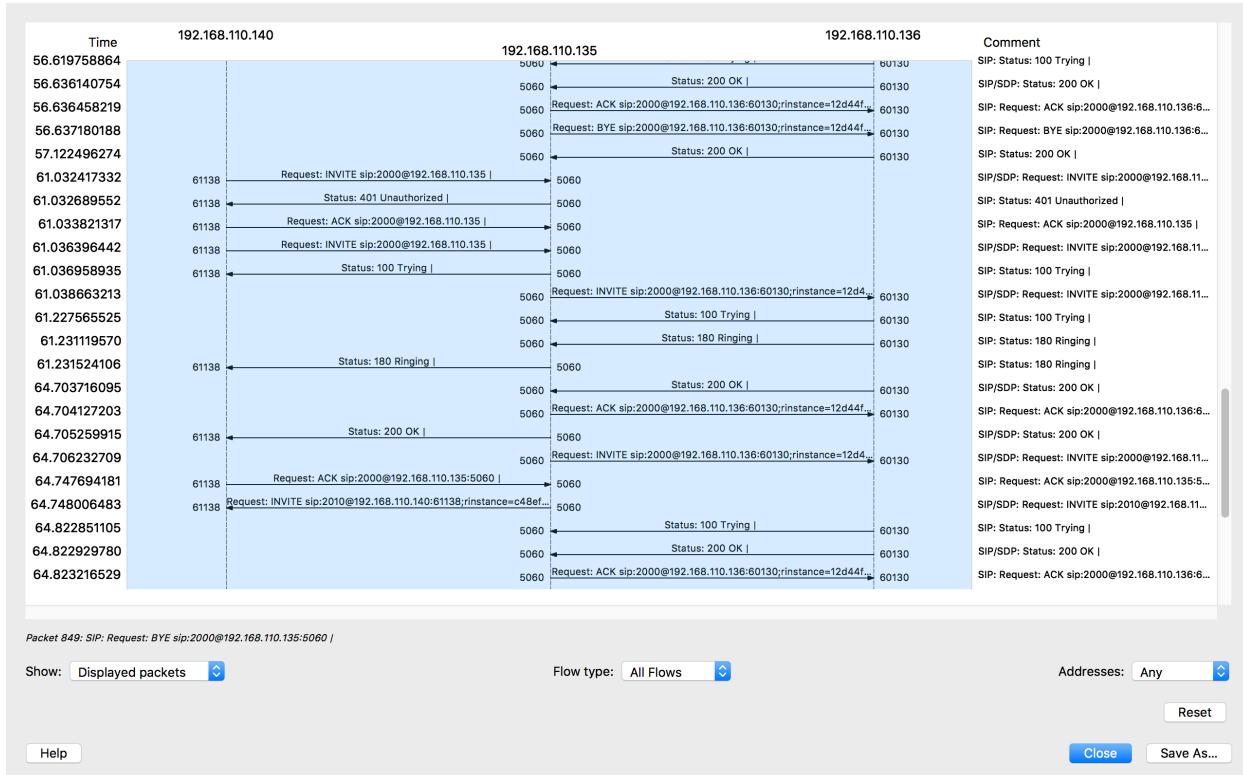


Figure : 1.1.14

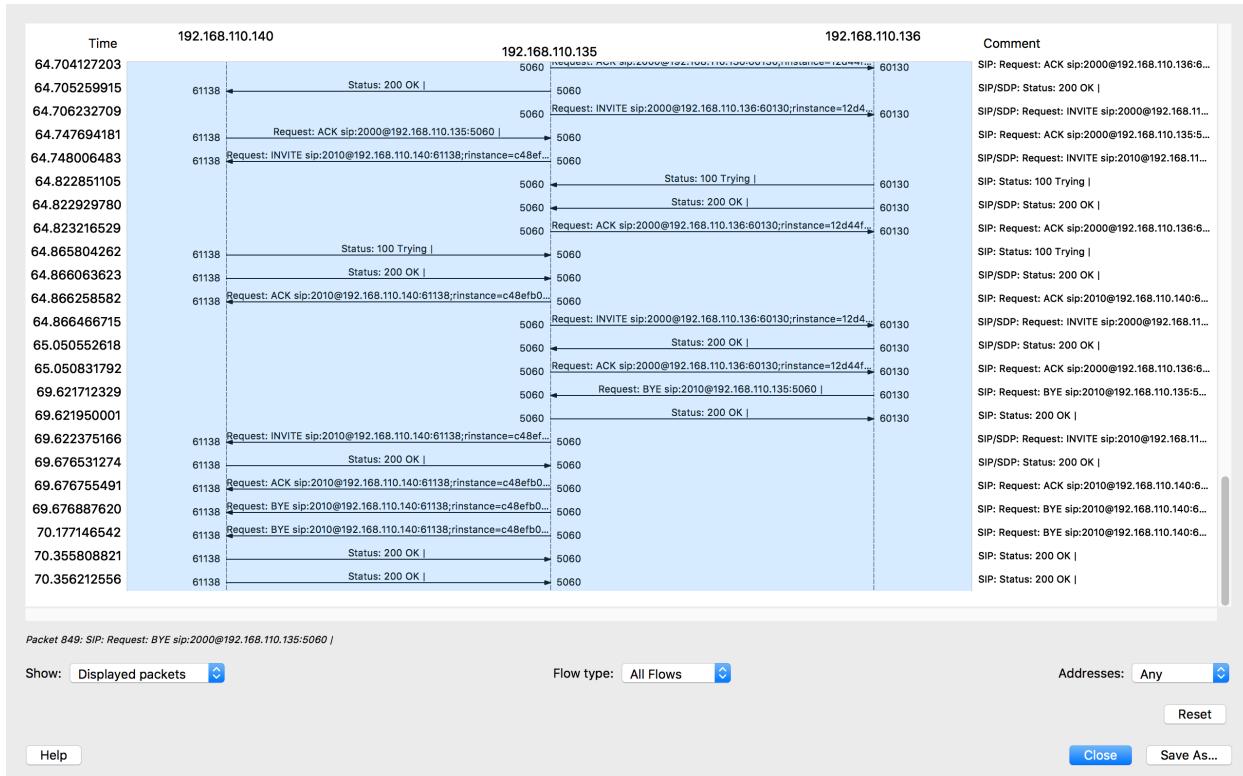


Figure : 1.1.15

Wireshark SIP Flows:

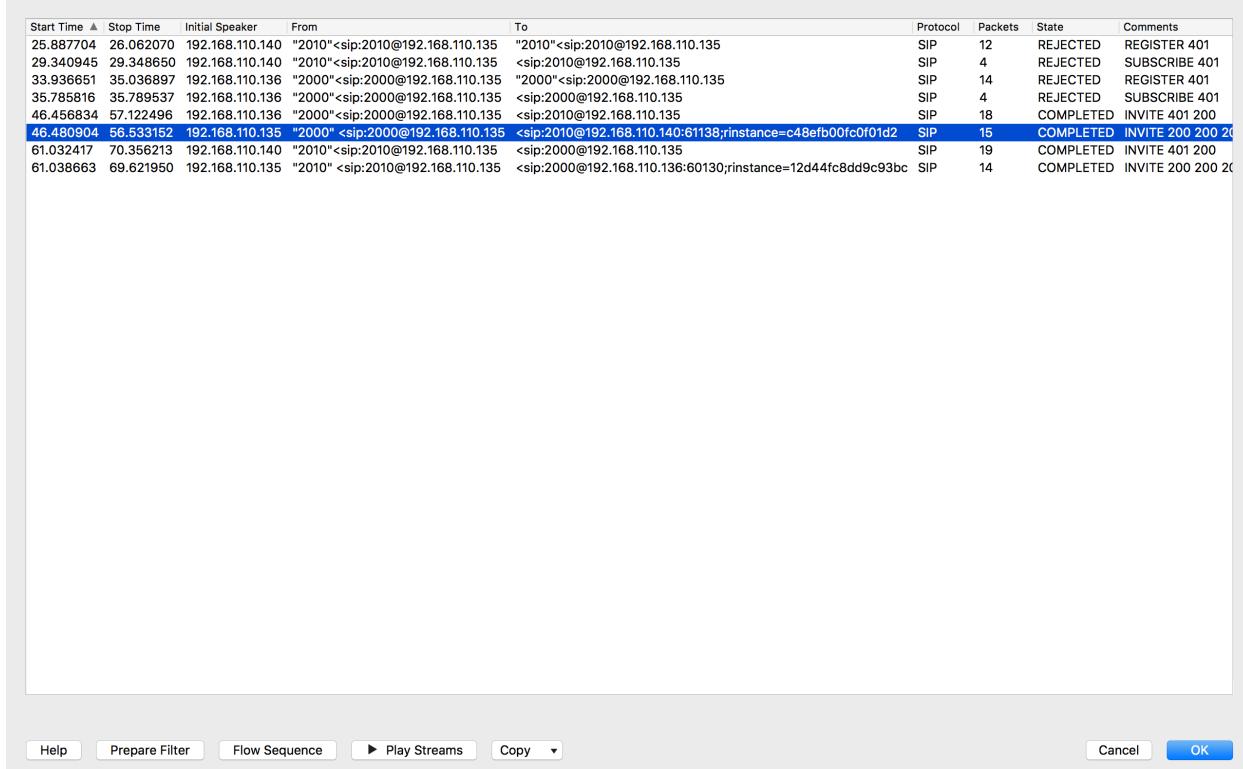


Figure : 1.1.16

The below are logical flow charts of our interest:

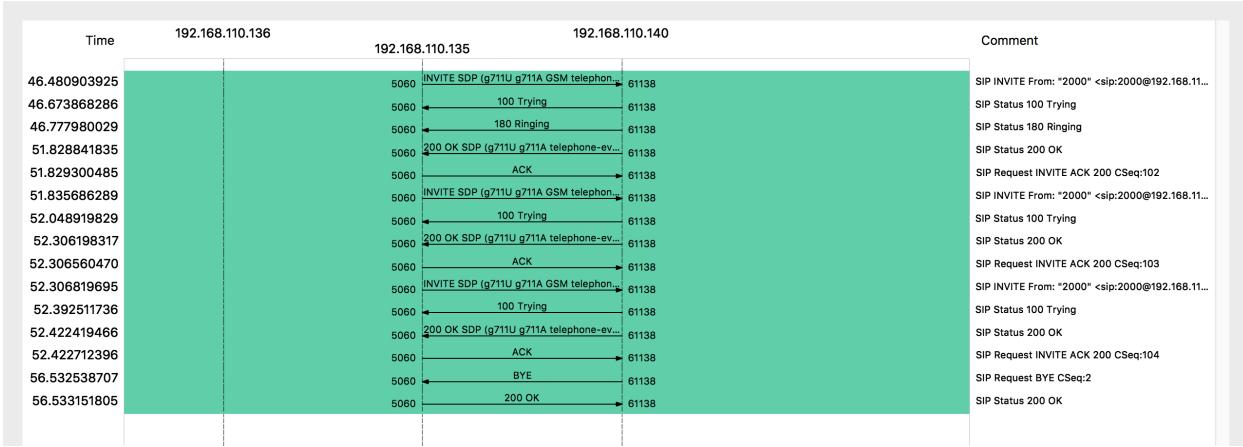


Figure : 1.1.17

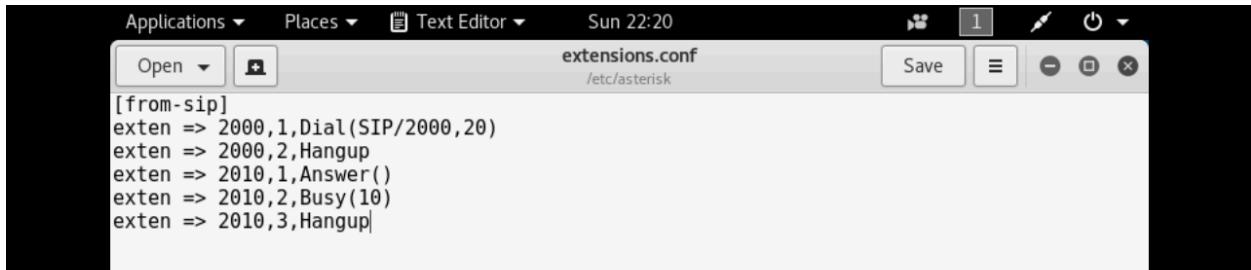


Figure : 1.1.18

The logical flow shows asterisk server notifying user 2000 about call request in the very first line of figure 1.1.17. Then 2010 sends Trying Ringing and OK to which the server acknowledges completing the SIP 3 way handshake. It is also shown that it uses G.711 codec and figure:1.1.18 shows the user 2010 joining the call by sending invite.

PHASE 2: Busy User:

The below is modified extensions.conf file to make busy user with delay of 10



```
[from-sip]
exten => 2000,1,Dial(SIP/2000,20)
exten => 2000,2,Hangup
exten => 2010,1,Answer()
exten => 2010,2,Busy(10)
exten => 2010,3,Hangup
```

Figure : 1.2.1

The below screenshot shows 2000 calling 2010 which is a busy user and the call ends after giving the busy tone instead of connecting the call

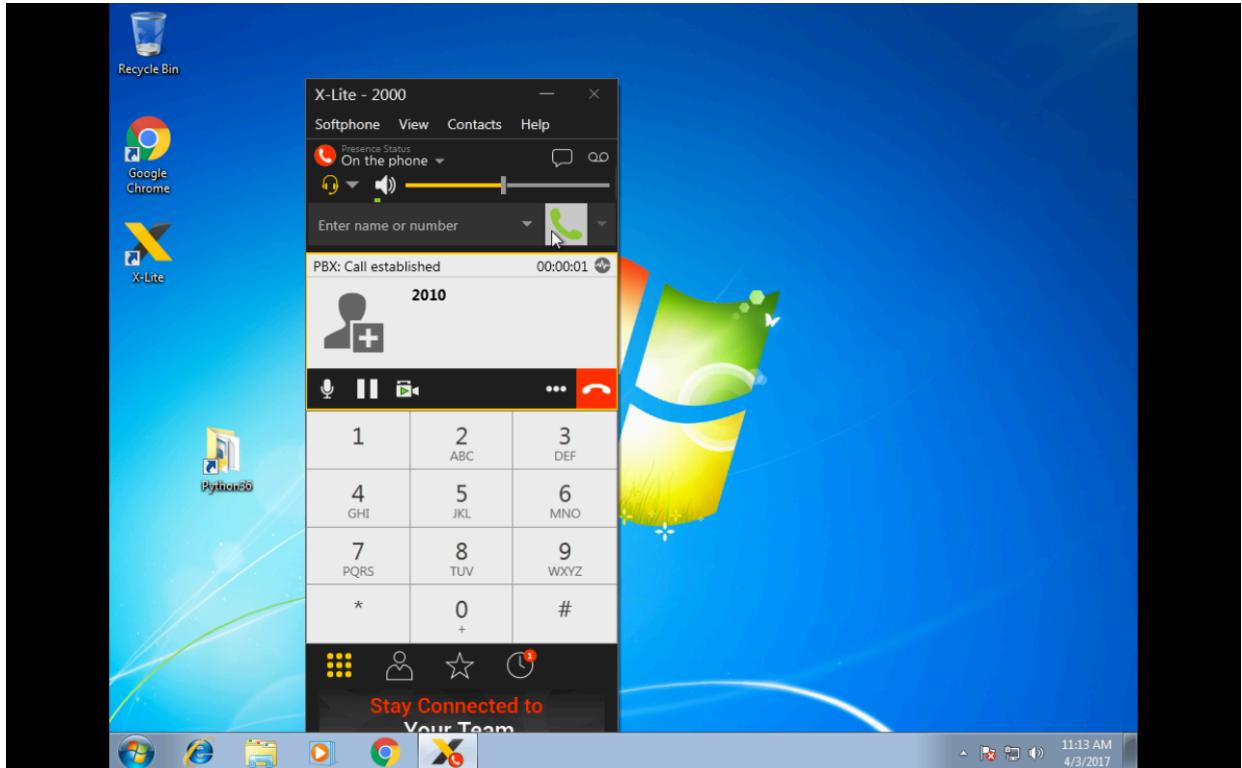


Figure : 1.2.2

The below are wireshark captures for busy user phase:

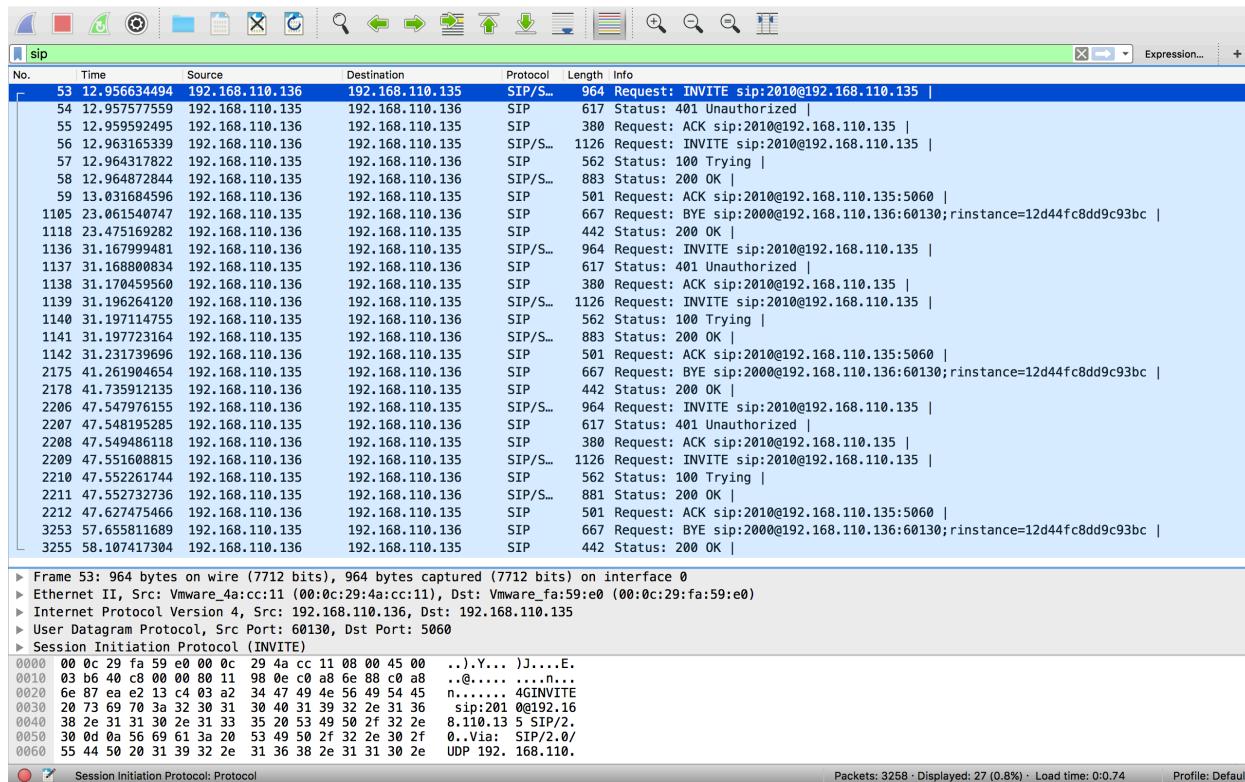


Figure : 1.2.3

Timing Diagram:

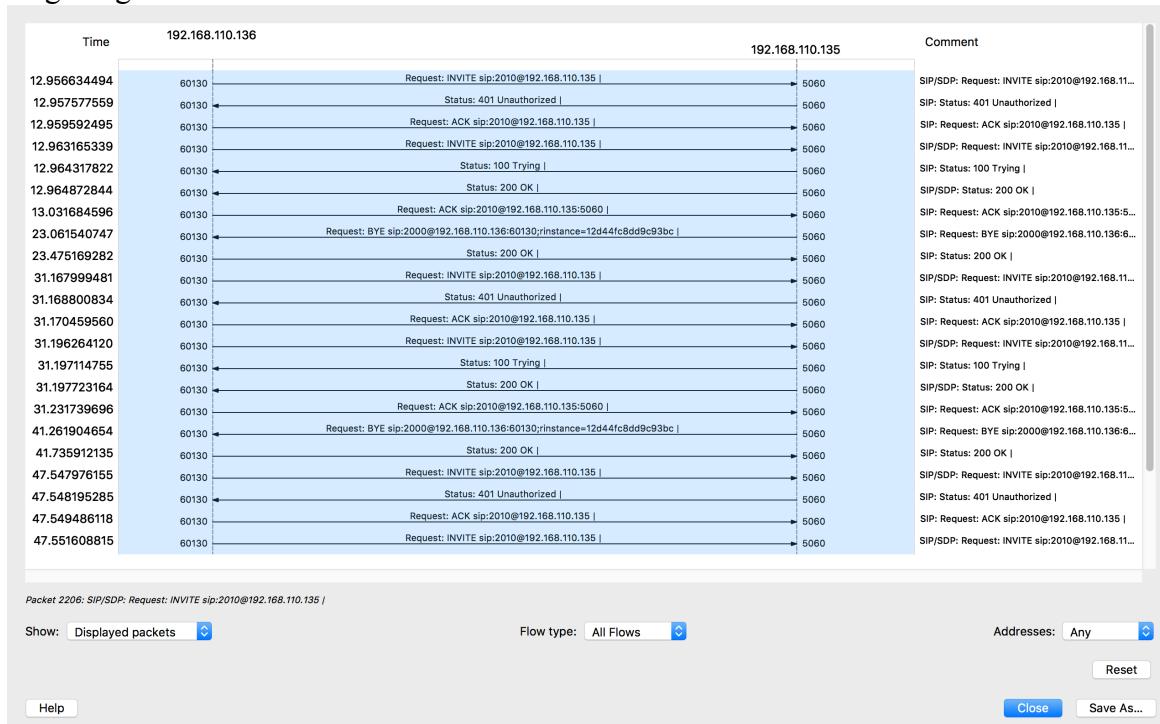


Figure : 1.2.4

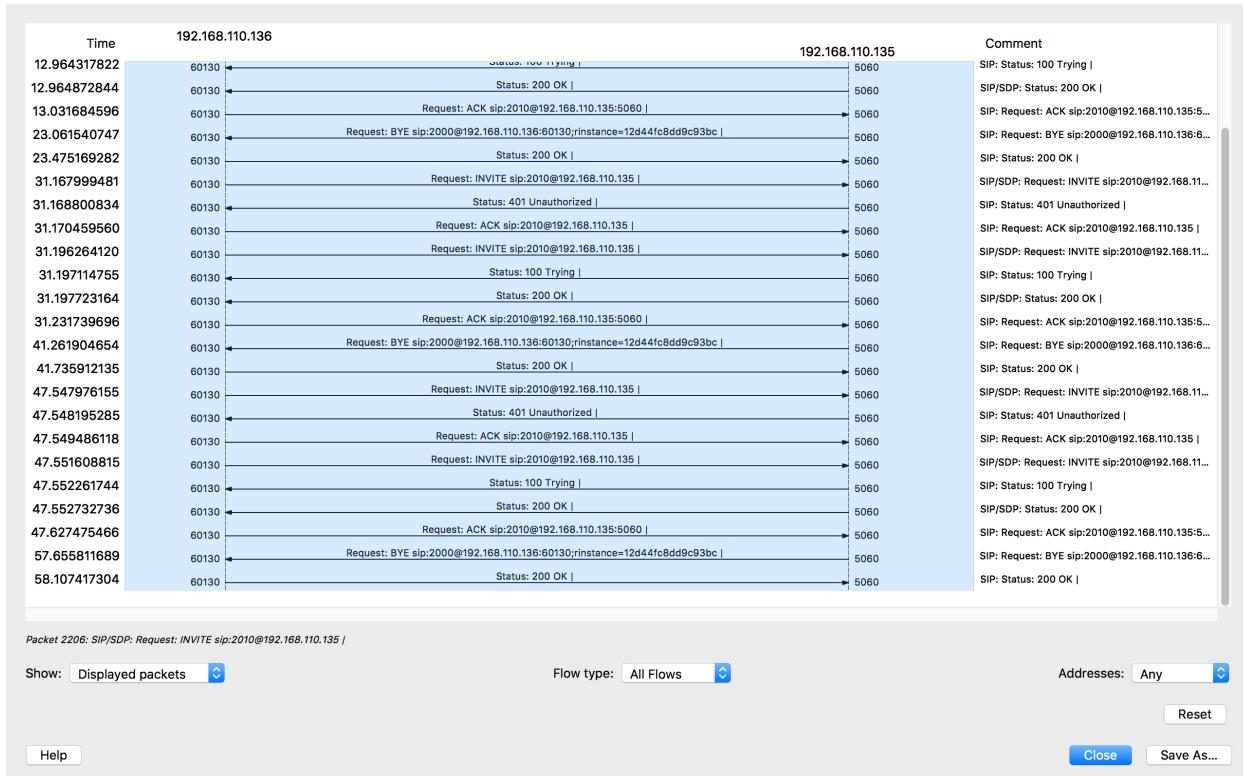


Figure : 1.2.5

SIP flows:

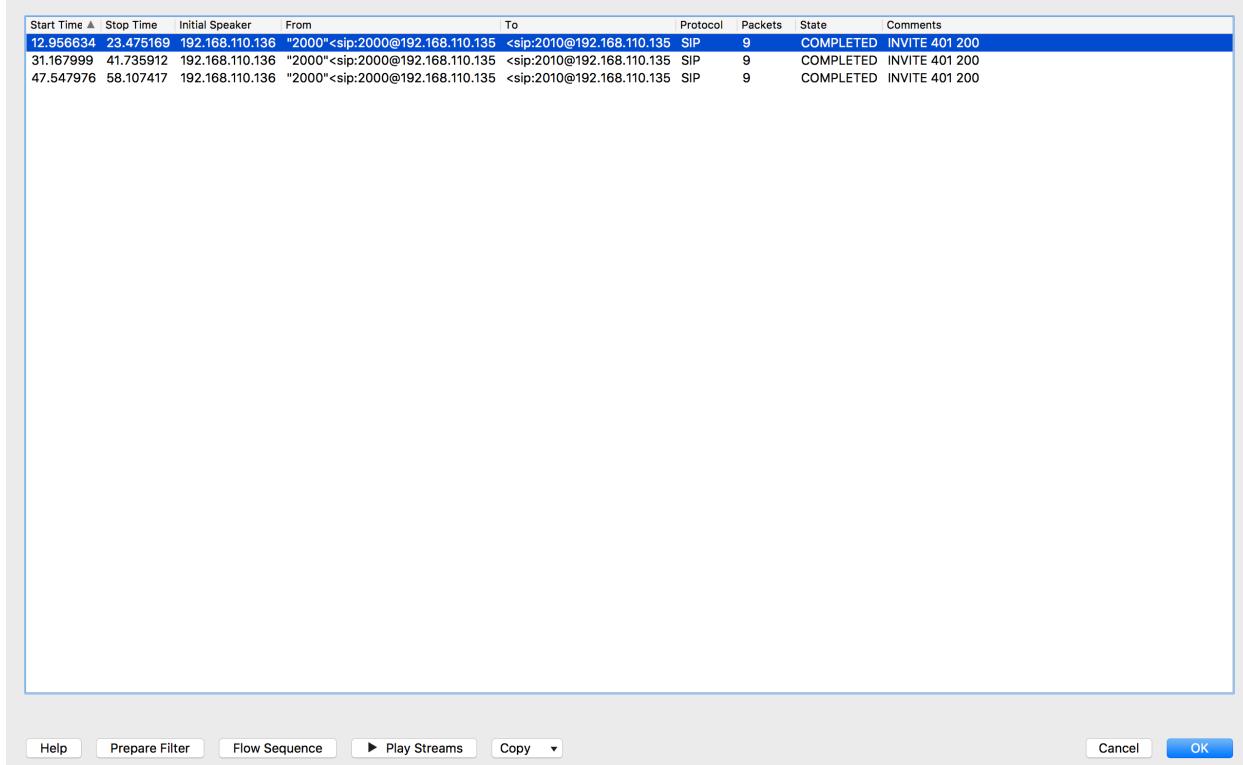


Figure : 1.2.6

User to asterisk:

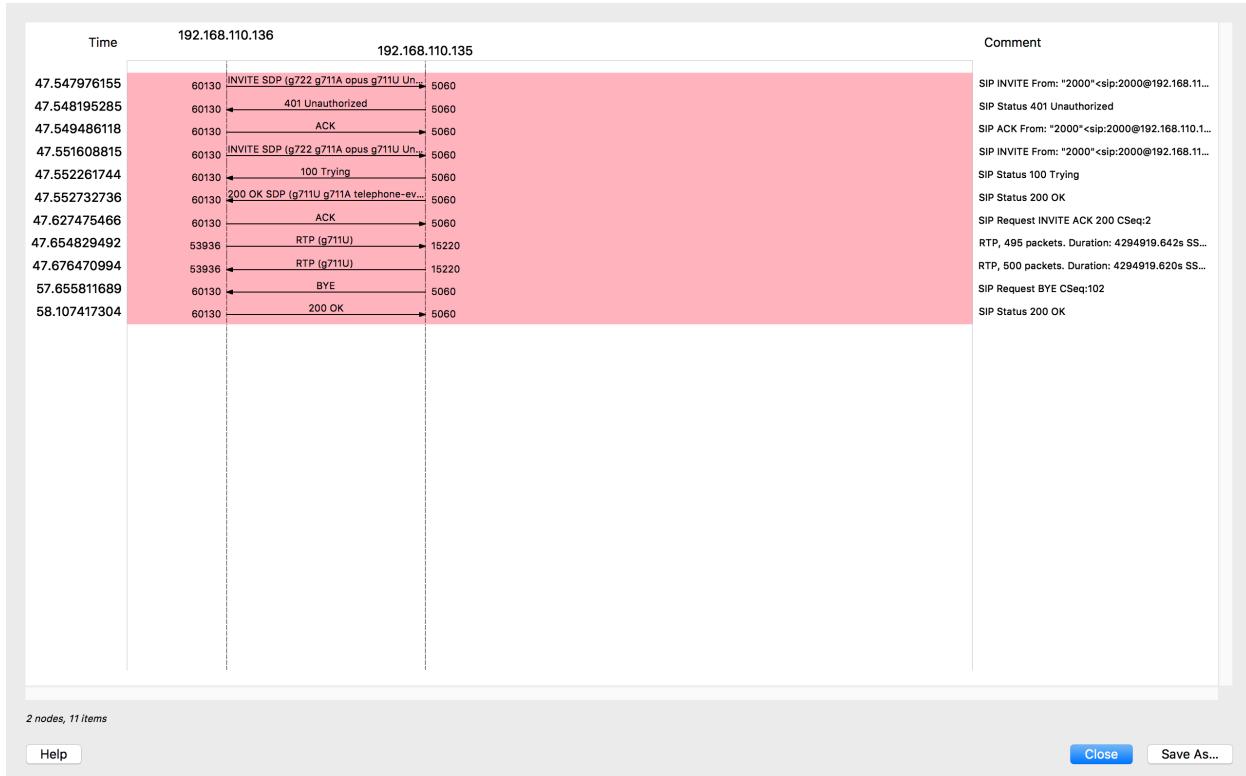


Figure : 1.2.7

It can be clearly seen in figure:1.2.7 that when user 2000 tries to call 2010 the server sends busy tone for 10secs from 47.65 and a BYE at the end at 57.65 to terminate the call according to the delay parameter specified.

PHASE 3 - Call on Hold:

Below is the modified extensions file to add third user 2009:

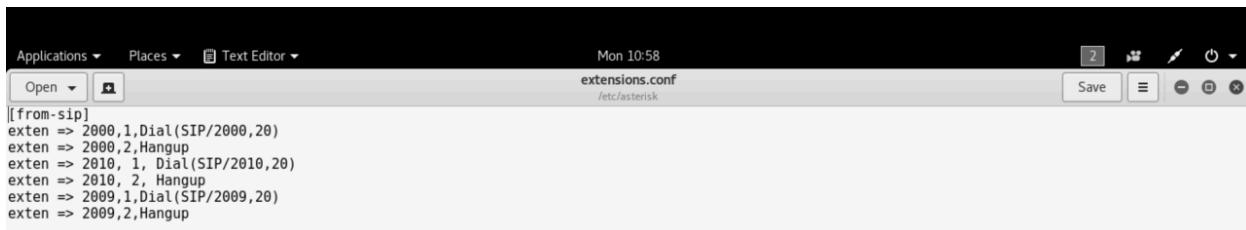
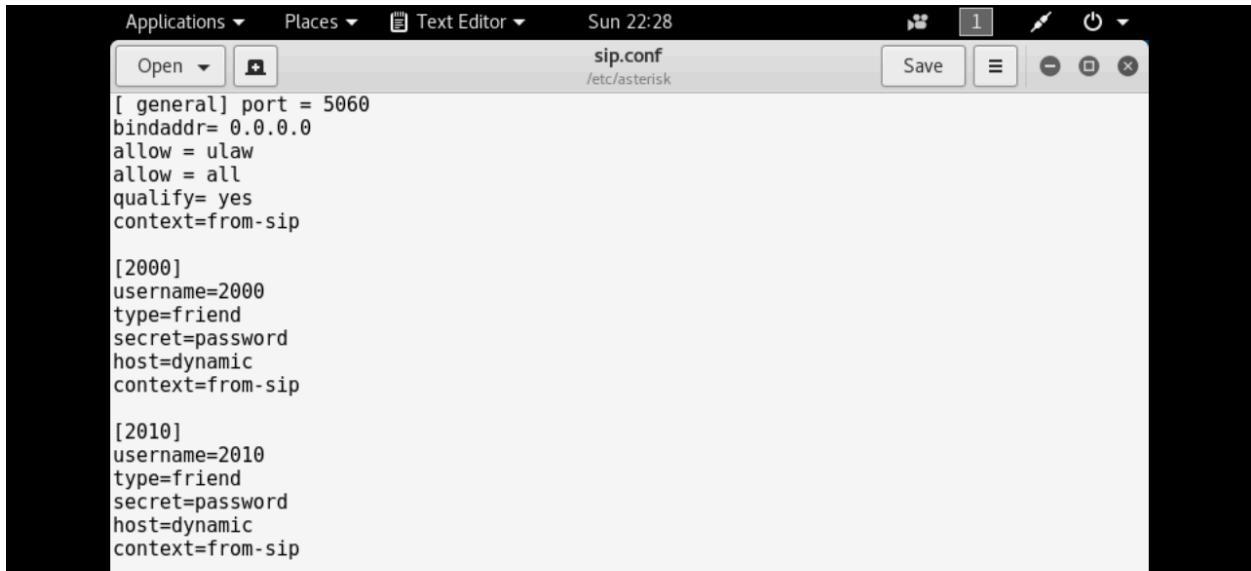


Figure : 1.3.1

The SIP.conf file with three users 2000,2010 and 2009



```
[ general] port = 5060
bindaddr= 0.0.0.0
allow = ulaw
allow = all
qualify= yes
context=from-sip

[2000]
username=2000
type=friend
secret=password
host=dynamic
context=from-sip

[2010]
username=2010
type=friend
secret=password
host=dynamic
context=from-sip
```

Figure : 1.3.2

The figure shows ongoing all between 2010 and 2000 at 2000's side:

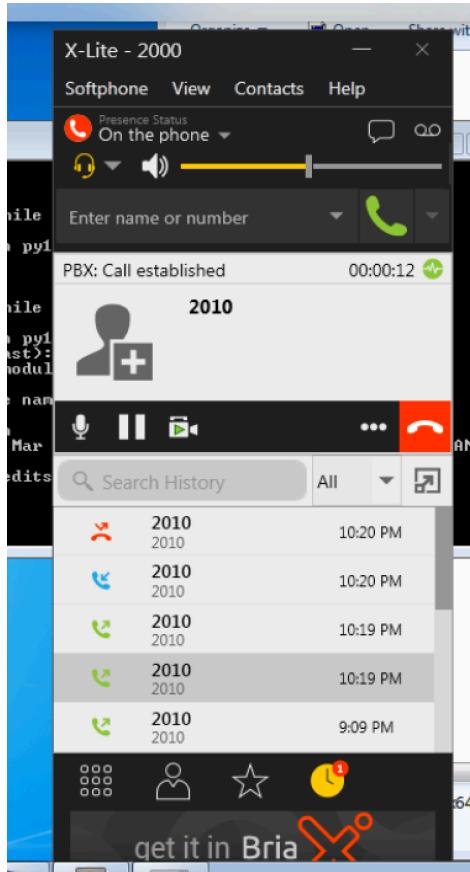


Figure : 1.3.3

2009 online:

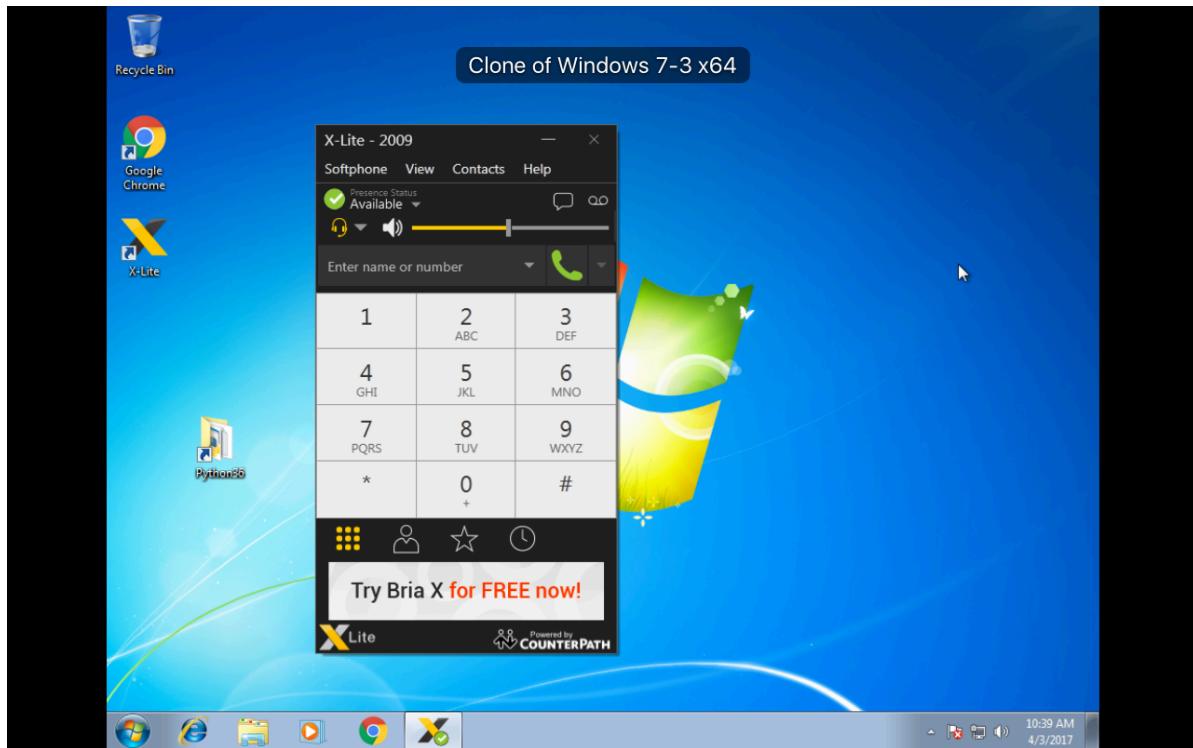


Figure : 1.3.4

2009 calling 2000:

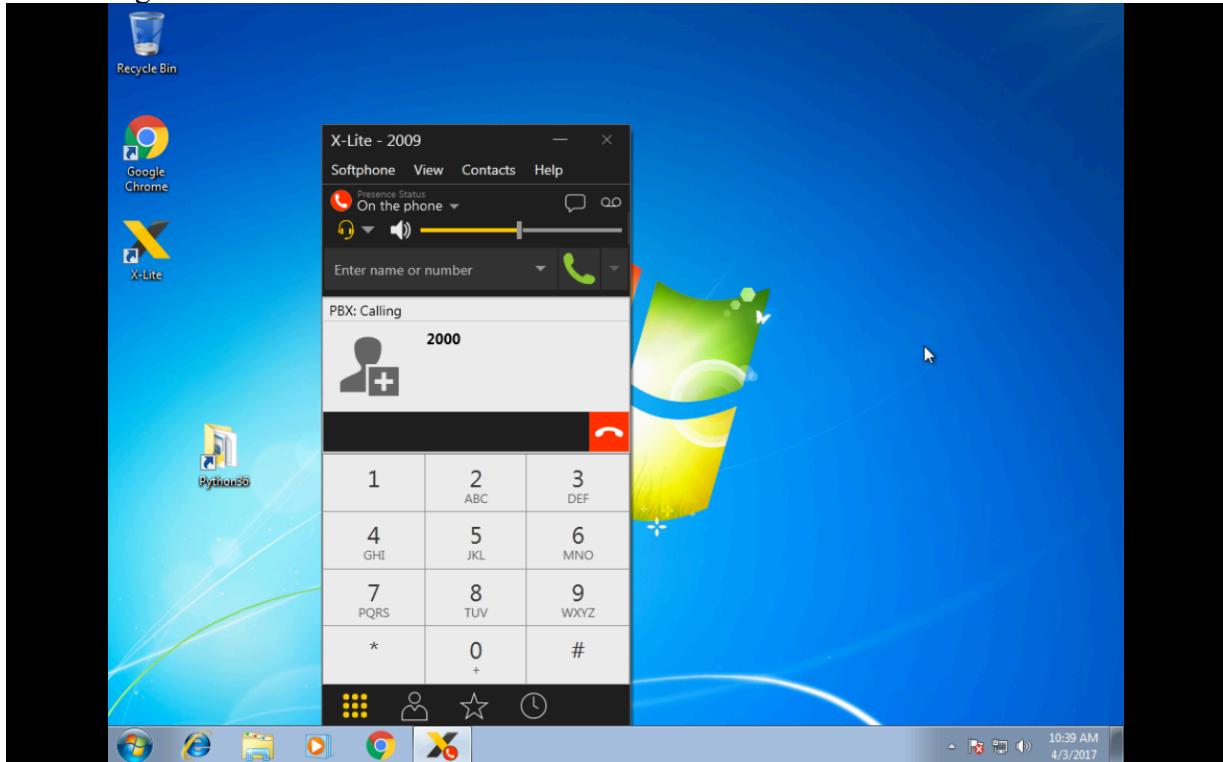


Figure : 1.3.5

2000 accepts call from 2009 while keeping 2010 on hold:

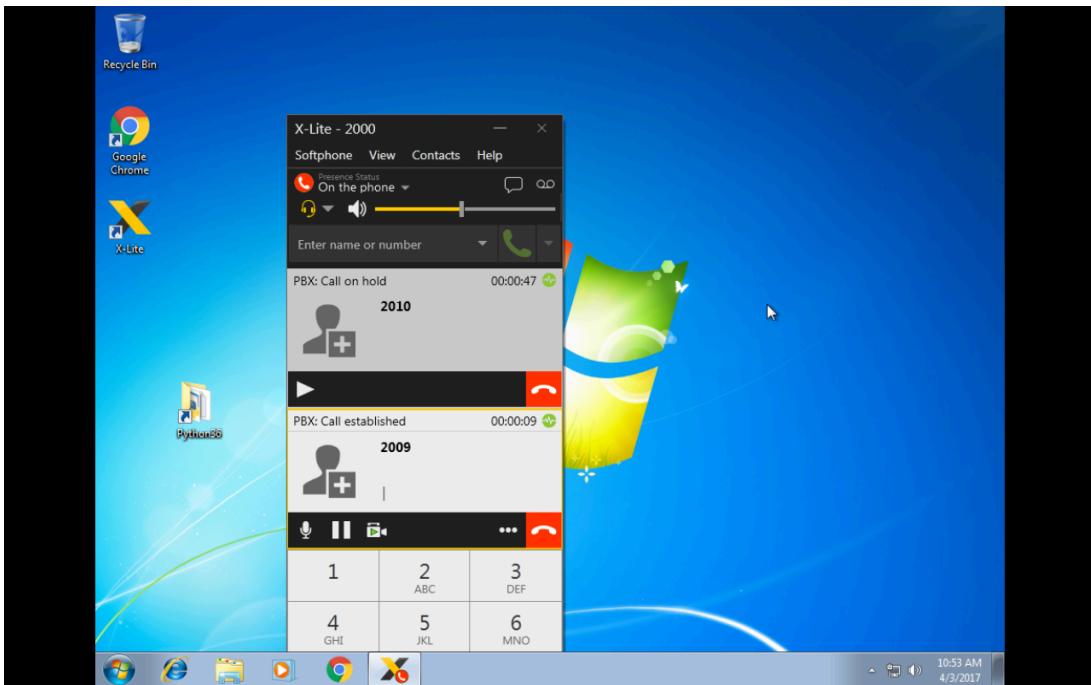


Figure : 1.3.6

2000 ends call with 2009 and 2010 still on hold:

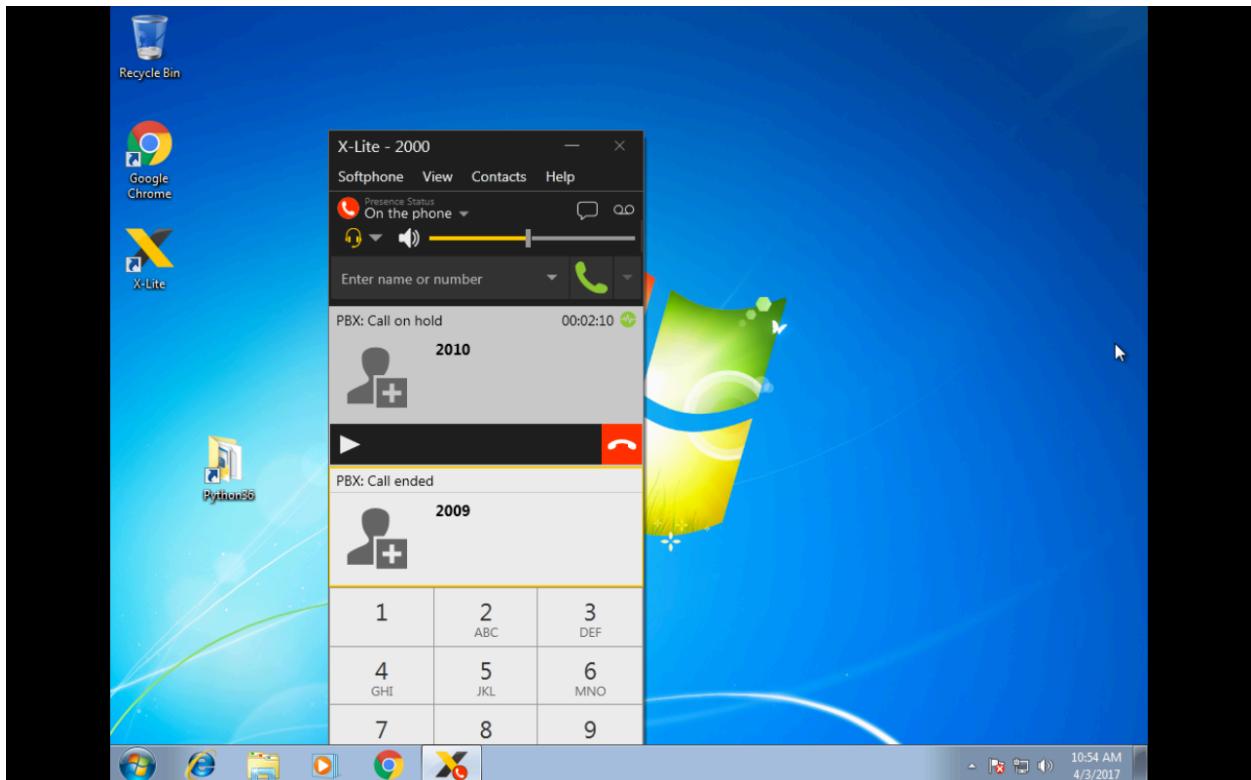


Figure : 1.3.7

2000 only has one call from 2010 on hold:

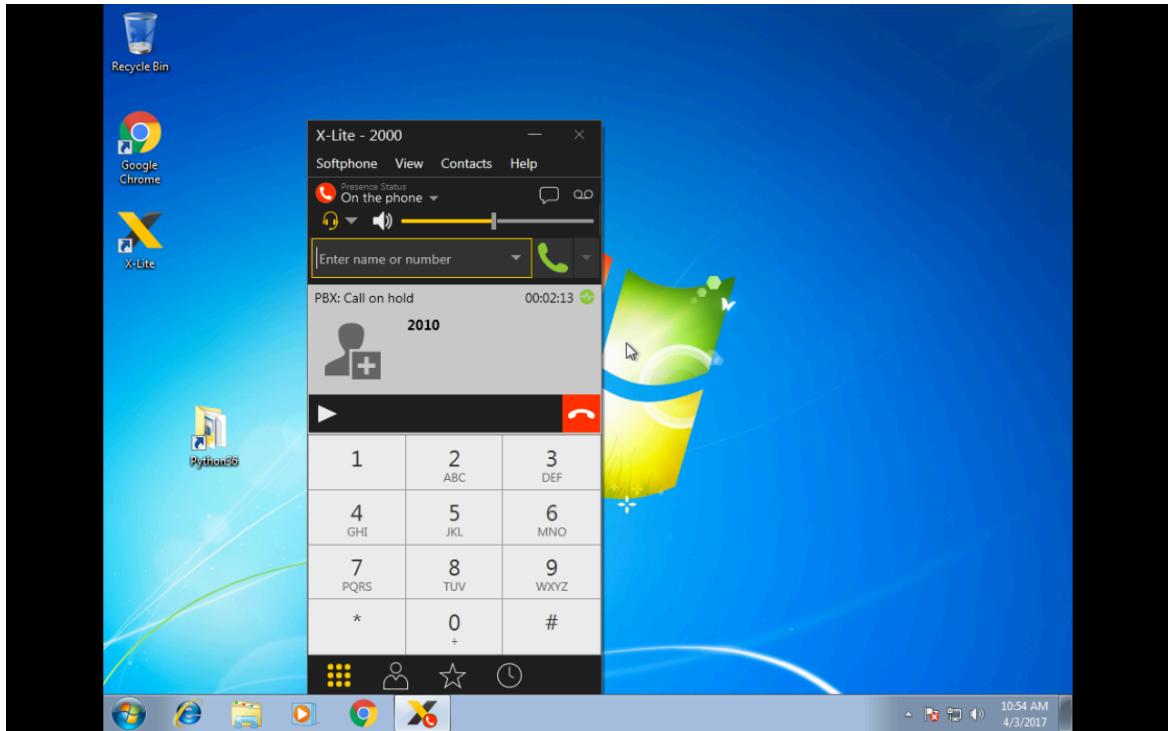


Figure : 1.3.8

2000 removes 2010 from hold:

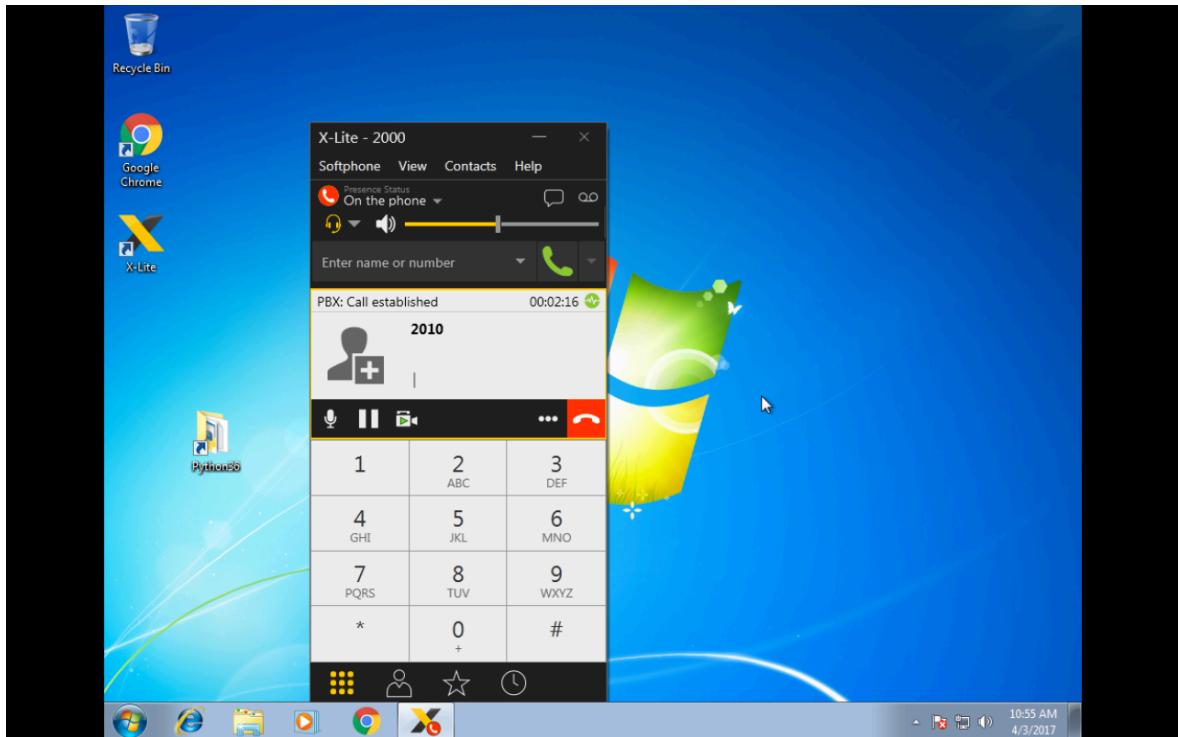


Figure : 1.3.9

Wireshark Captures:

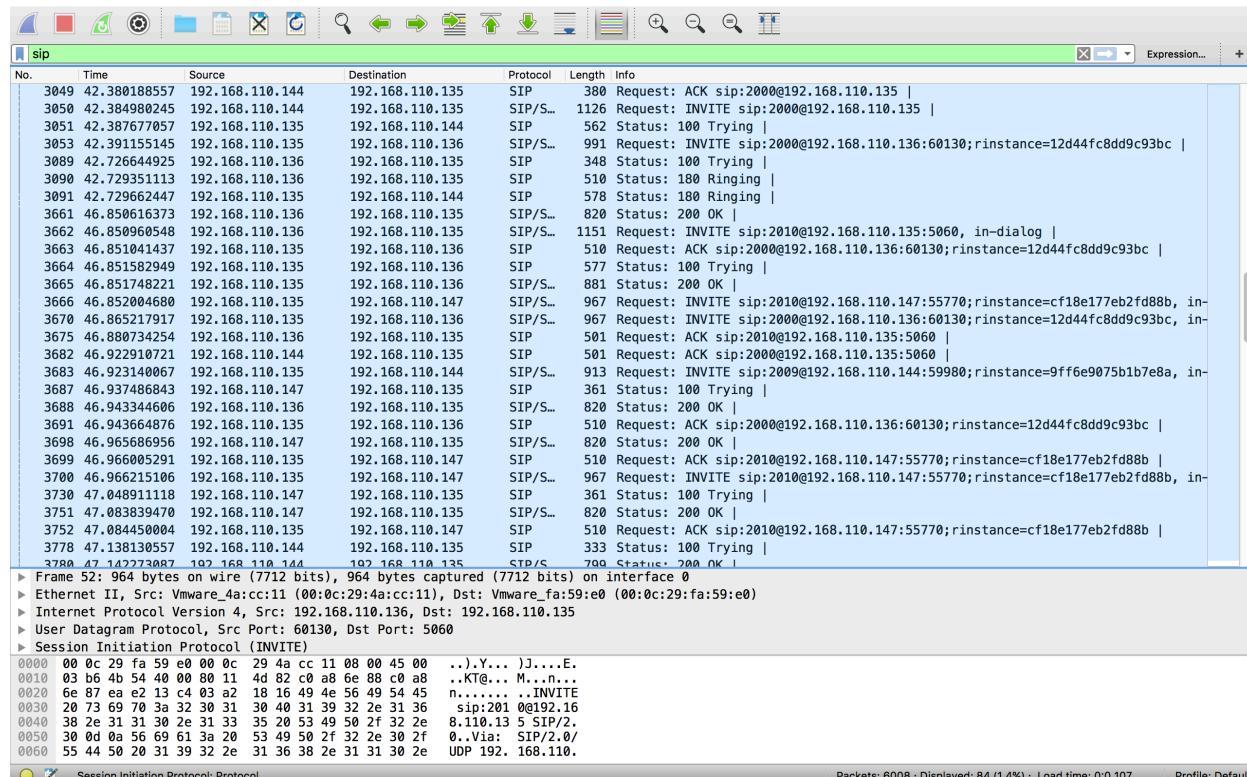


Figure : 1.3.4

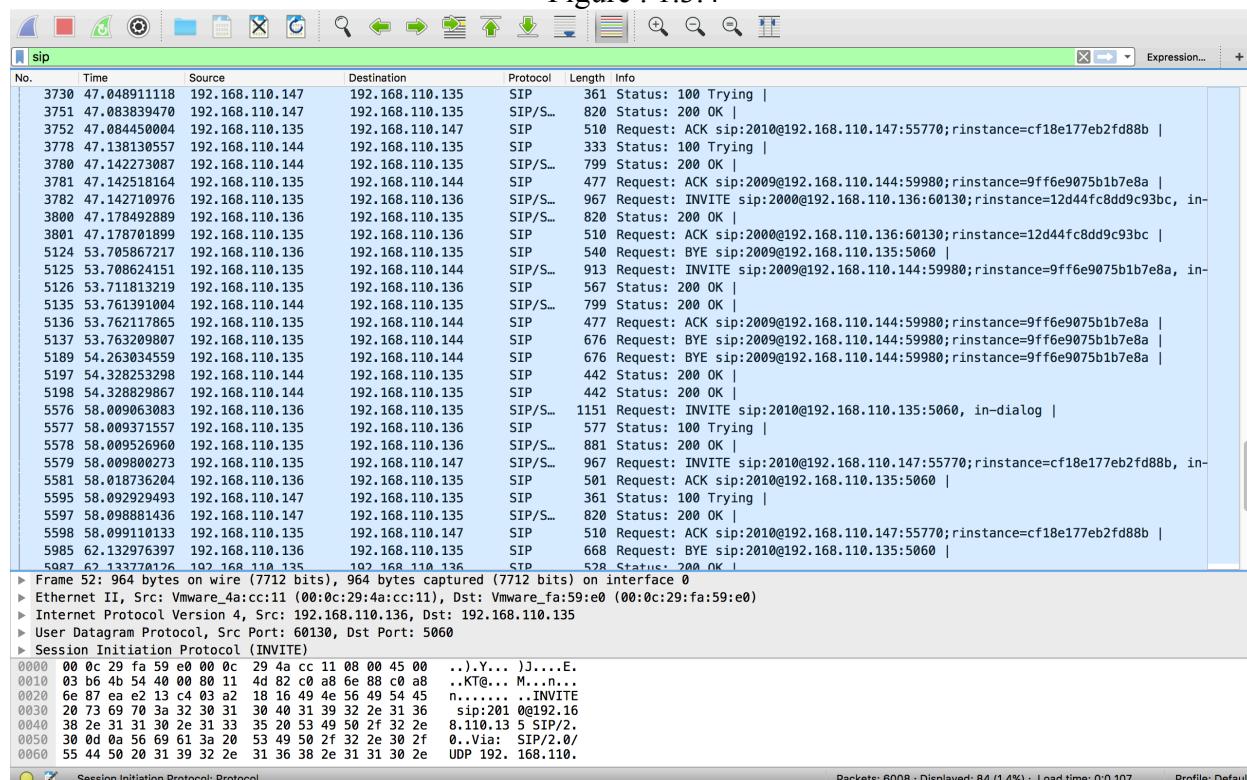


Figure : 1.3.5

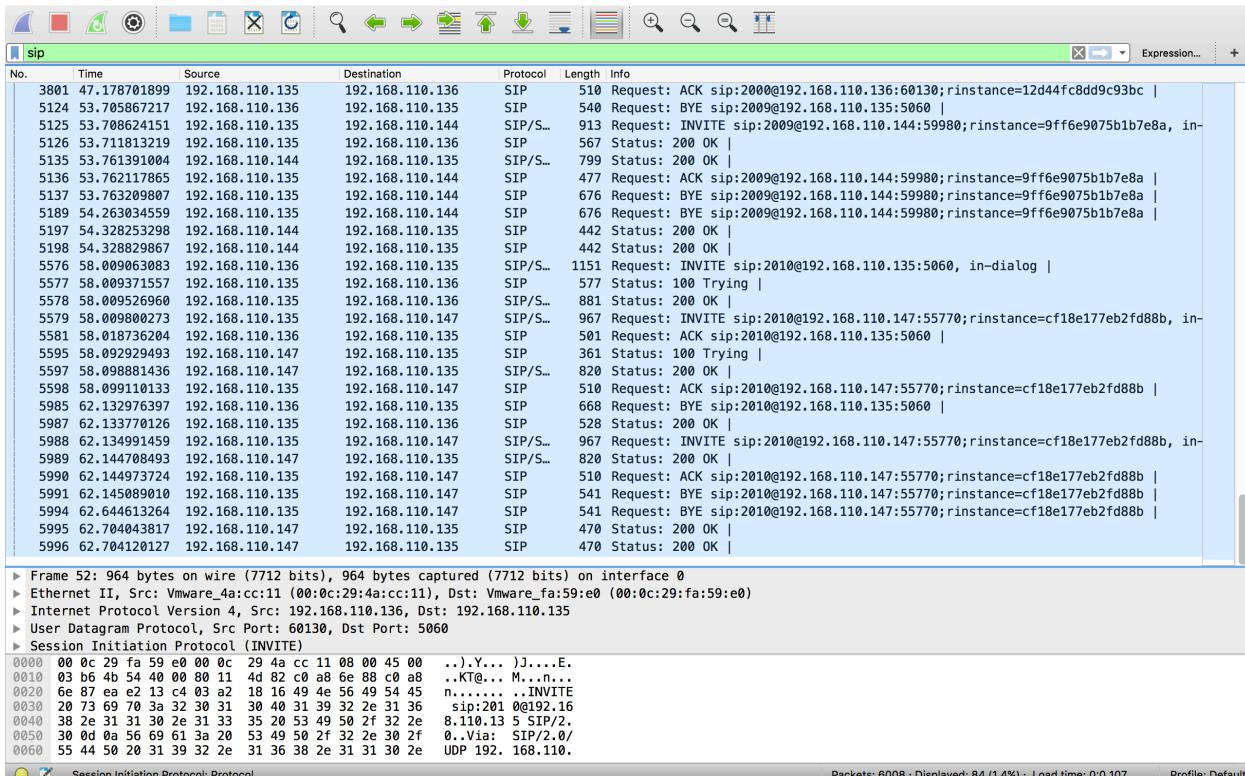


Figure : 1.3.6

Timing Diagrams:

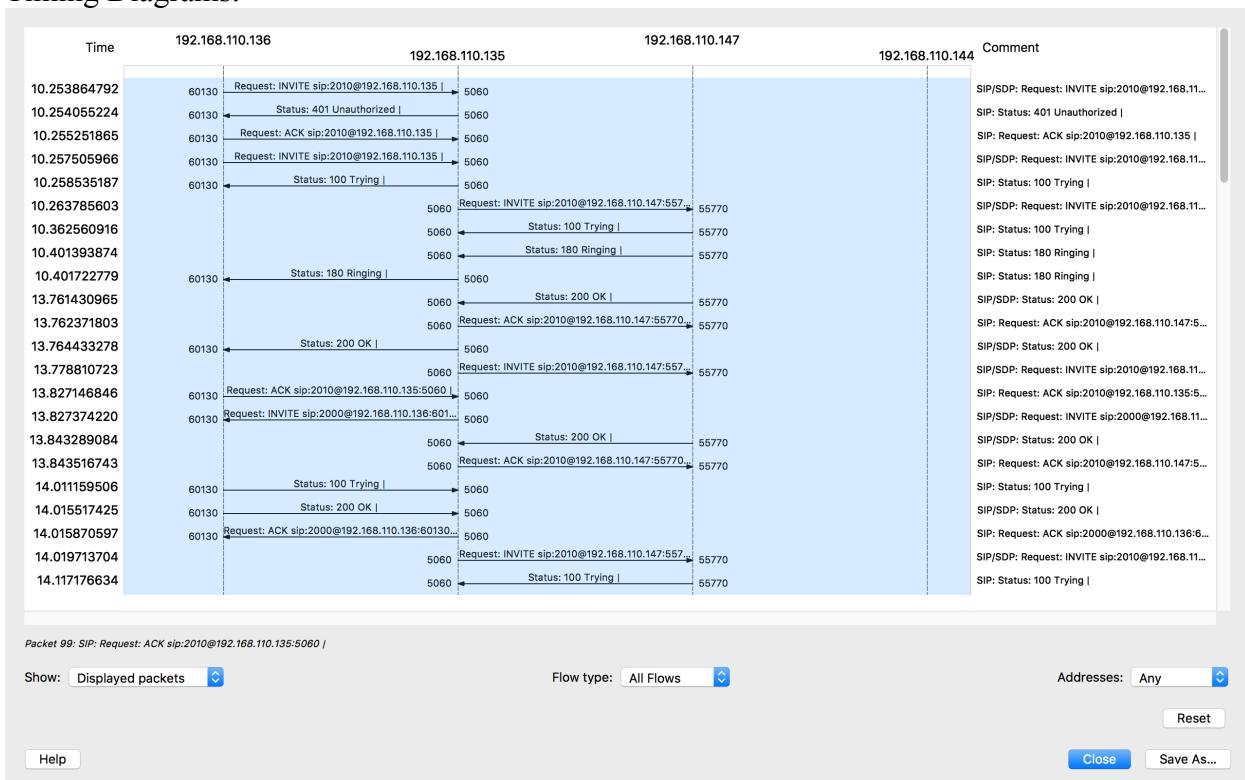


Figure : 1.3.7

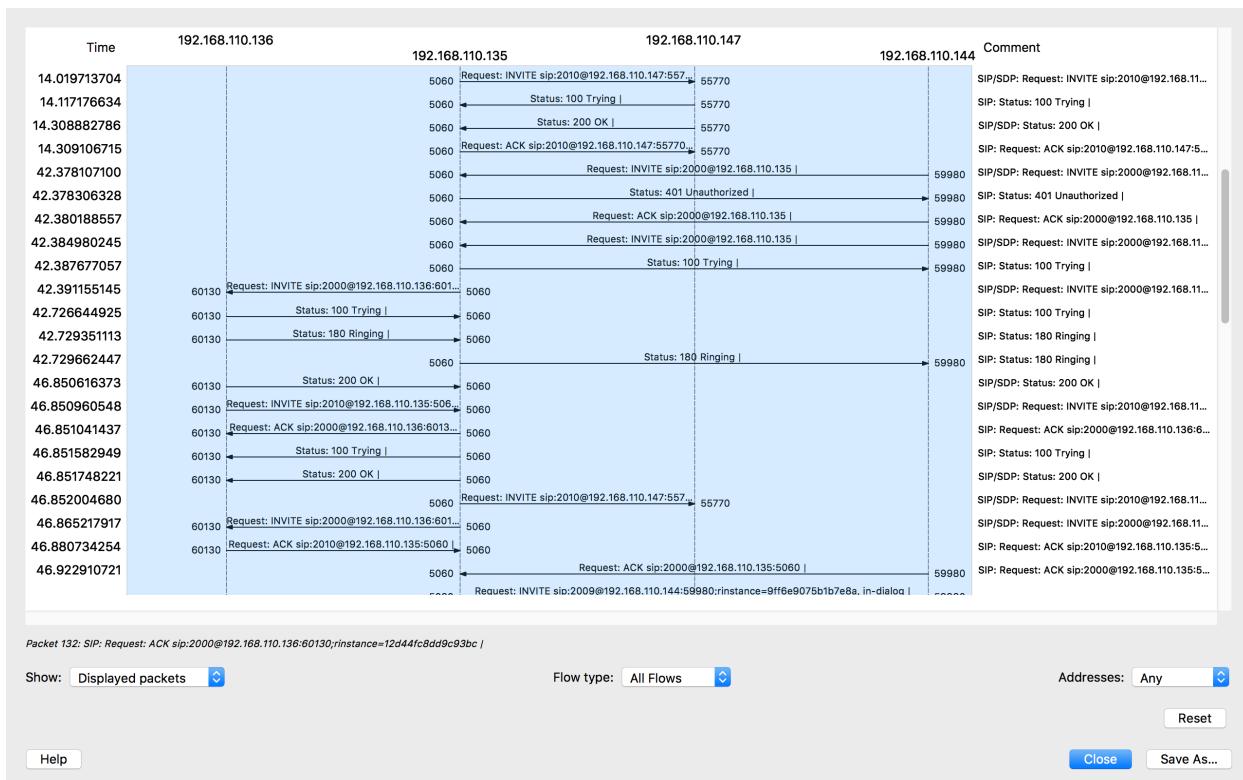


Figure : 1.3.8

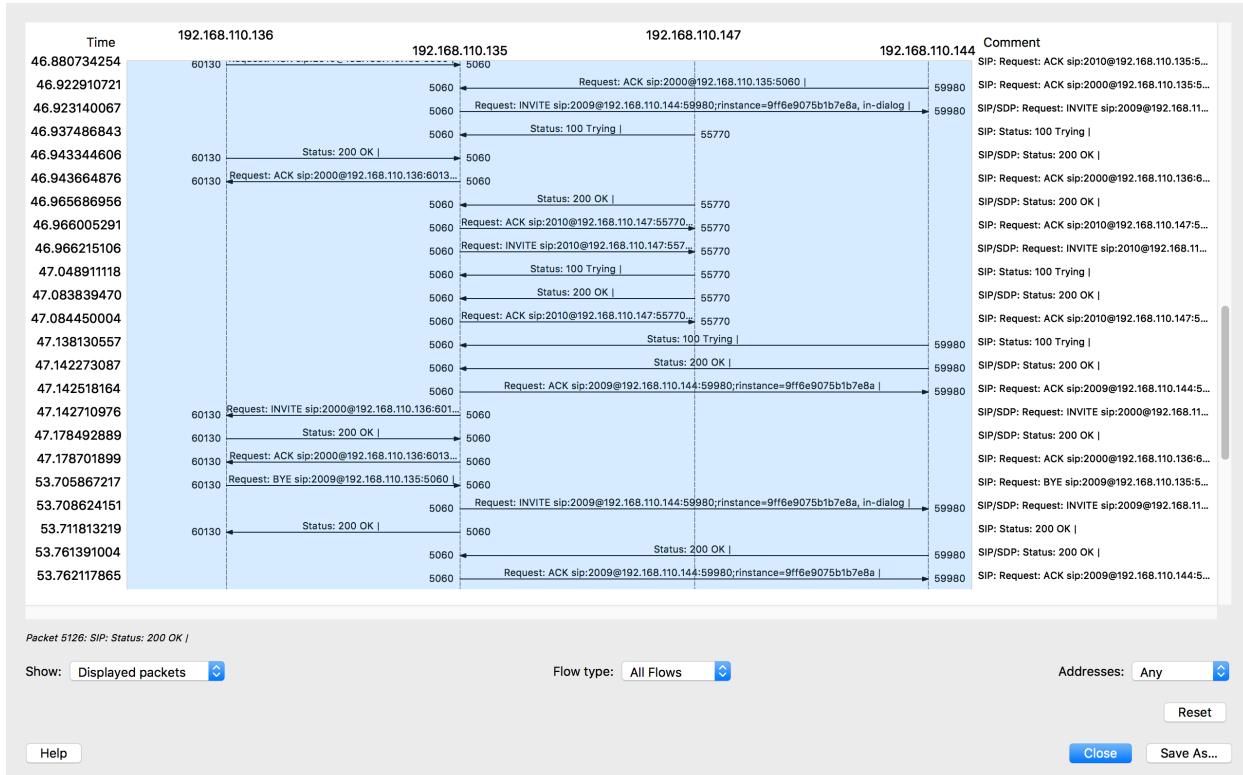


Figure : 1.3.9

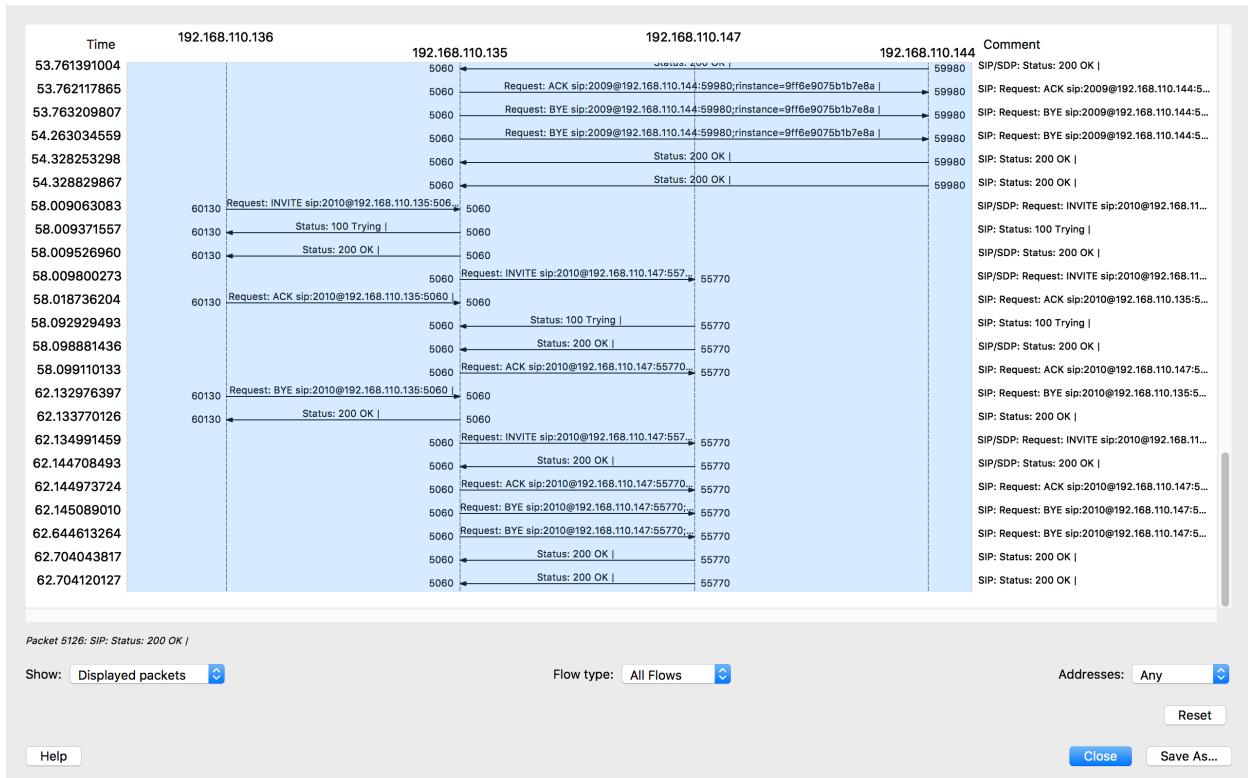


Figure : 1.3.10

SIP Flows:

Start Time	Stop Time	Initial Speaker	From	To	Protocol	Packets	State	Comments
10.253865	62.133770	192.168.110.136	"2000"<sip:2000@192.168.110.135	<sip:2010@192.168.110.135	SIP	22	COMPLETED	INVITE 401 200 20
10.263786	62.704120	192.168.110.135	"2000" <sip:2000@192.168.110.135	<sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b	SIP	31	COMPLETED	INVITE 200 200 20
42.378107	54.328830	192.168.110.144	"2009"<sip:2009@192.168.110.135	<sip:2000@192.168.110.135	SIP	18	COMPLETED	INVITE 401
42.391155	53.711813	192.168.110.135	"2009" <sip:2009@192.168.110.135	<sip:2000@192.168.110.136:60130;rinstance=12d44fc8dd9c93bc	SIP	13	COMPLETED	INVITE 200 200 20

Help Prepare Filter Flow Sequence Play Streams Copy ▾ Cancel OK

Figure : 1.3.11

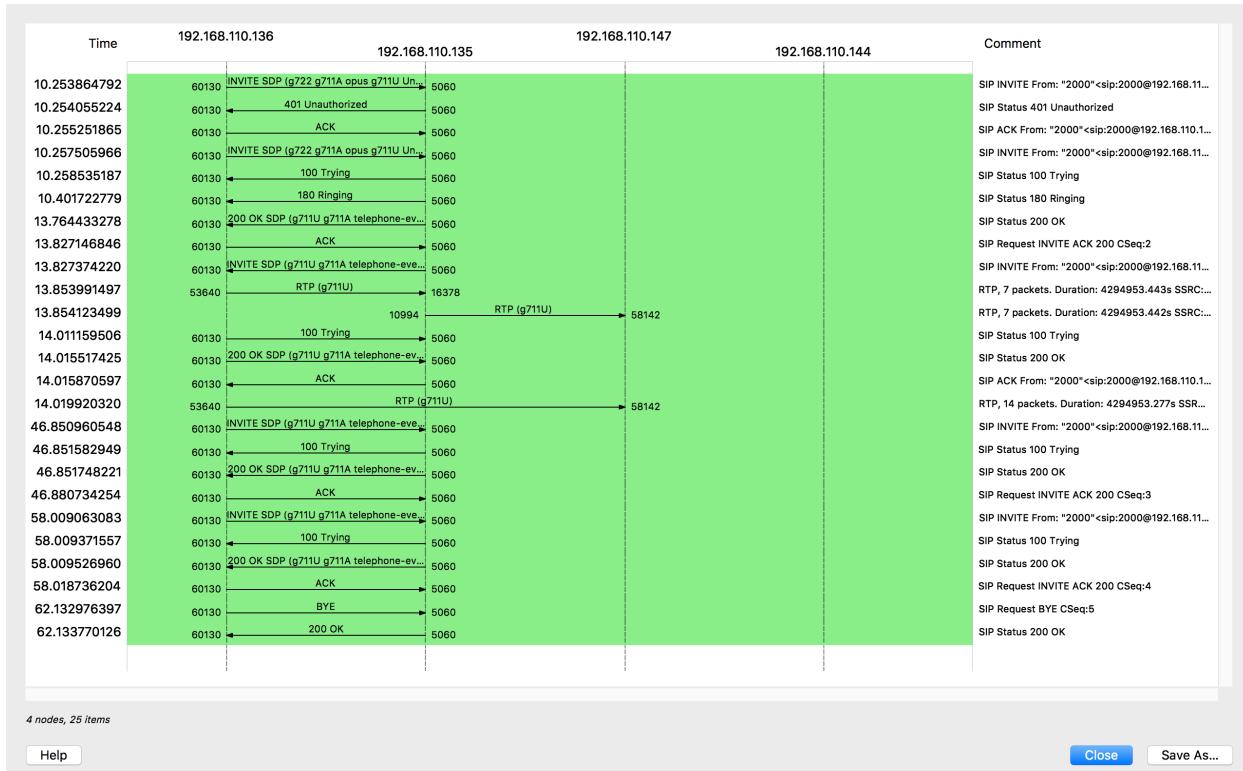


Figure : 1.3.12



Figure : 1.3.13



Figure : 1.3.14

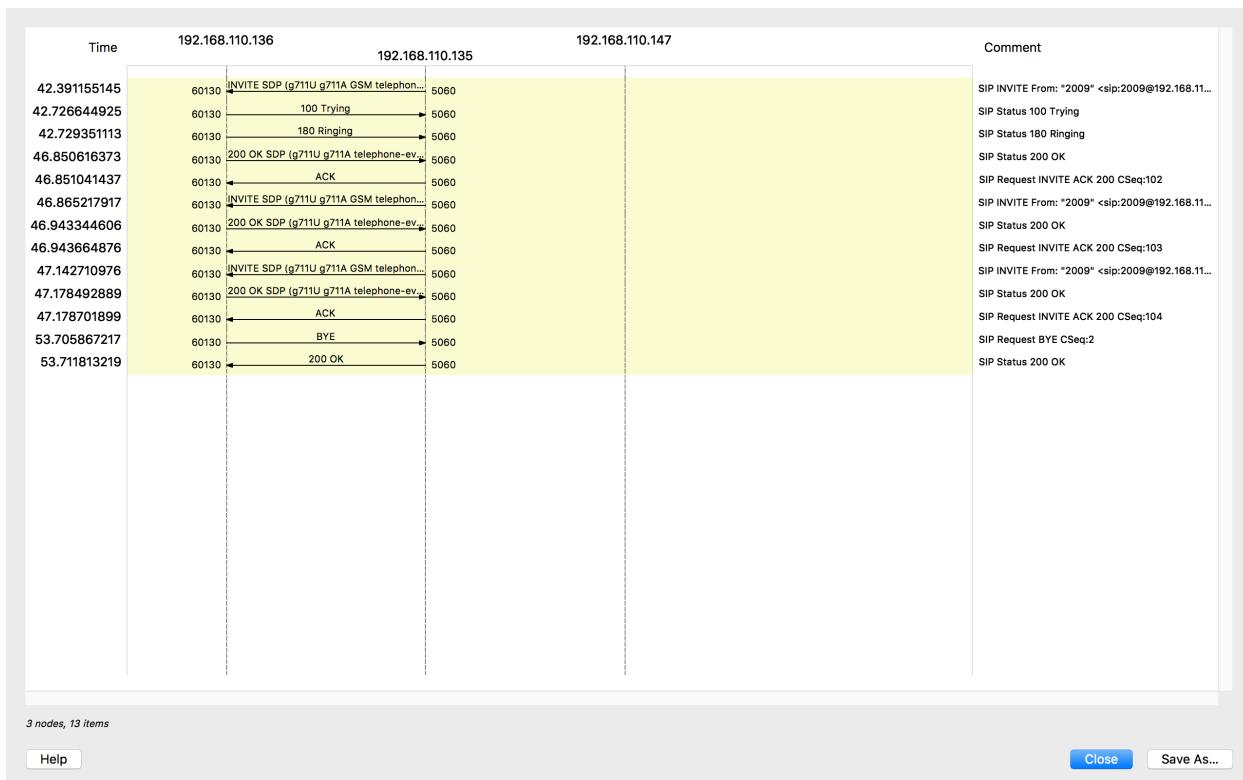


Figure : 1.3.15

We can see in fig:1.3.12 and 1.3.13 the call between 2000 and 2010 in progress from 2000 and 2010's side respectively. Fig 1.3.14 shows the call from 2009 is received by 2000 while keeping 2010 on hold and after completing call with 2009 again resuming call with 2010 as seen in fig: 1.3.15.

PHASE 4 – Call Conferencing:

The picture shows call in progress between 2000 and 2010 with 2009 calling 2000

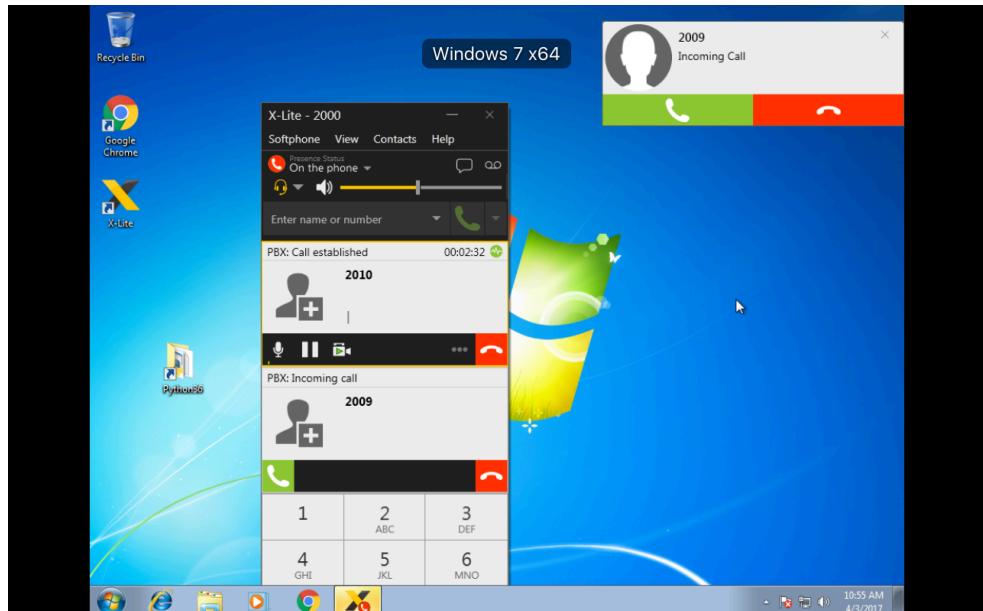


Figure : 1.4.1

Here when 2000 first accepts call 2010 goes on hold automatically as in real time cellular conferencing

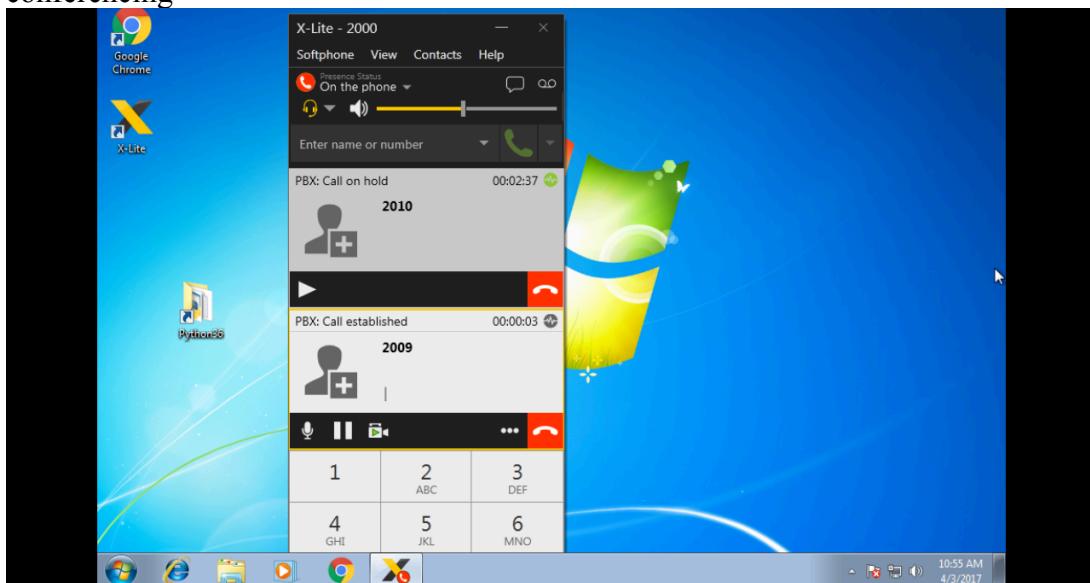


Figure : 1.4.2

This screenshot shows 2000 merging the two calls together:

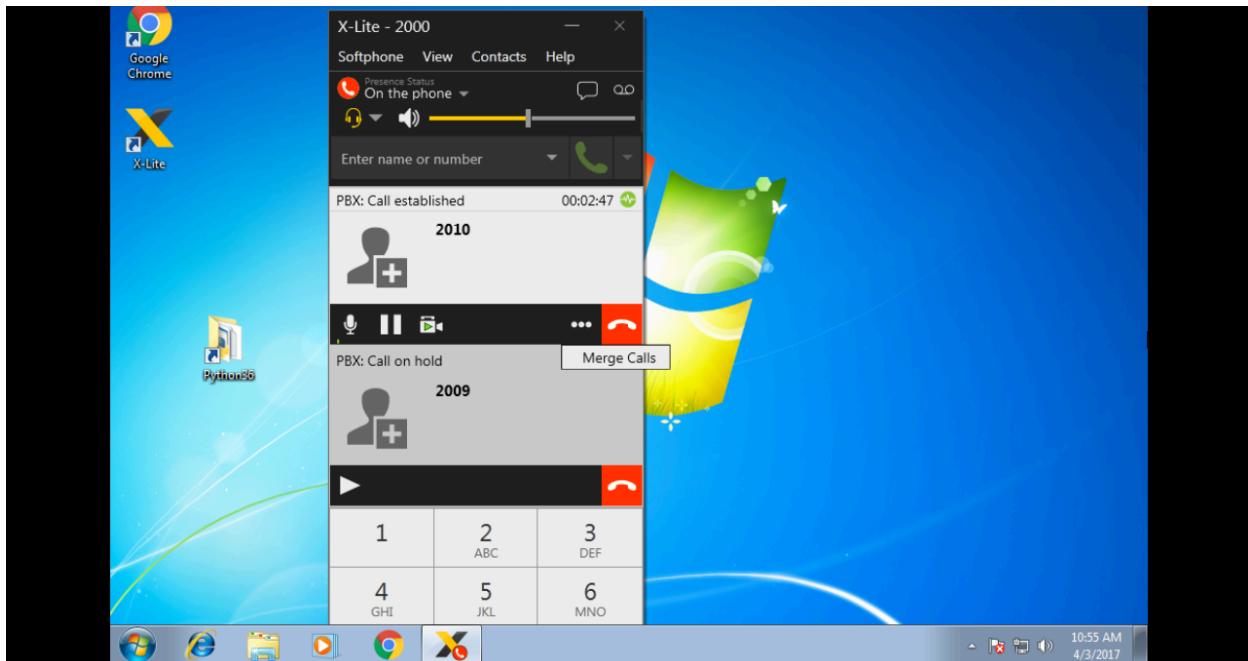


Figure : 1.4.3

This figure signify successful merger of calls with call conferencing:

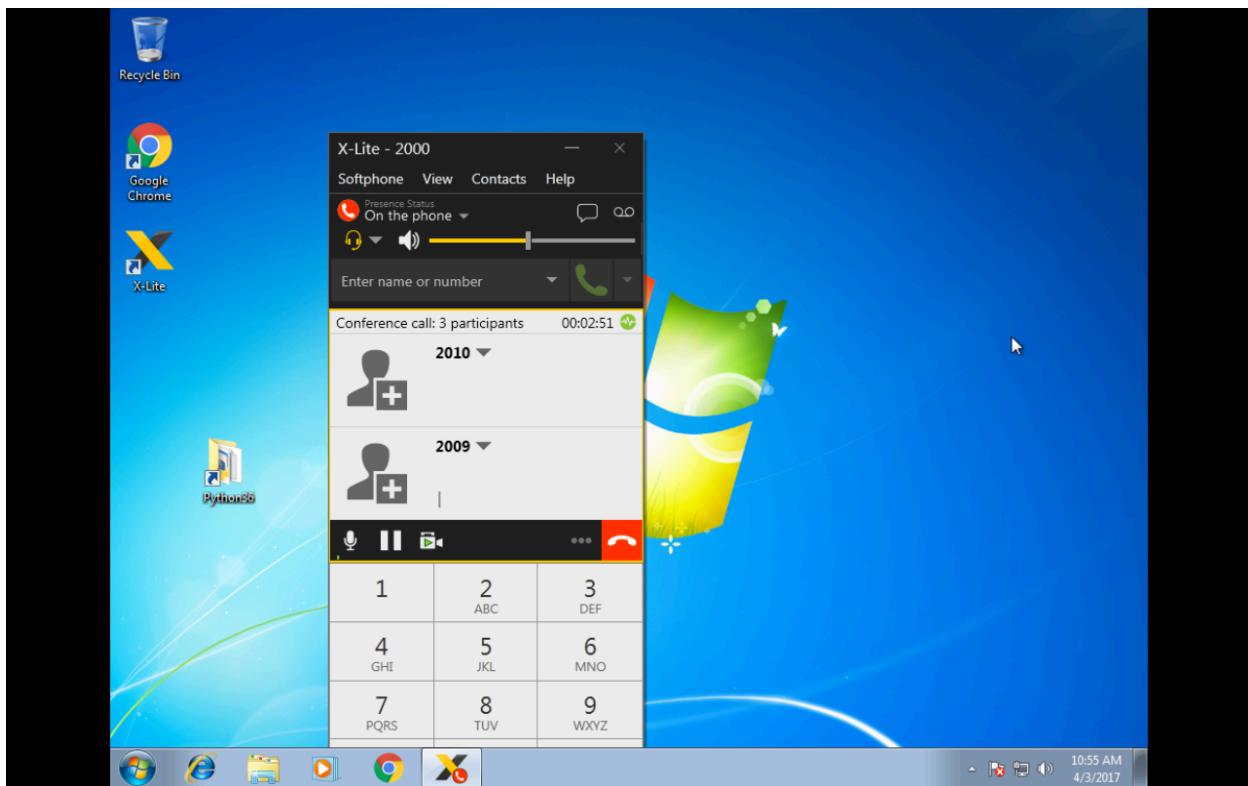


Figure : 1.4.4

This screenshot shows 2000 ending conference and both the calls simultaneously:

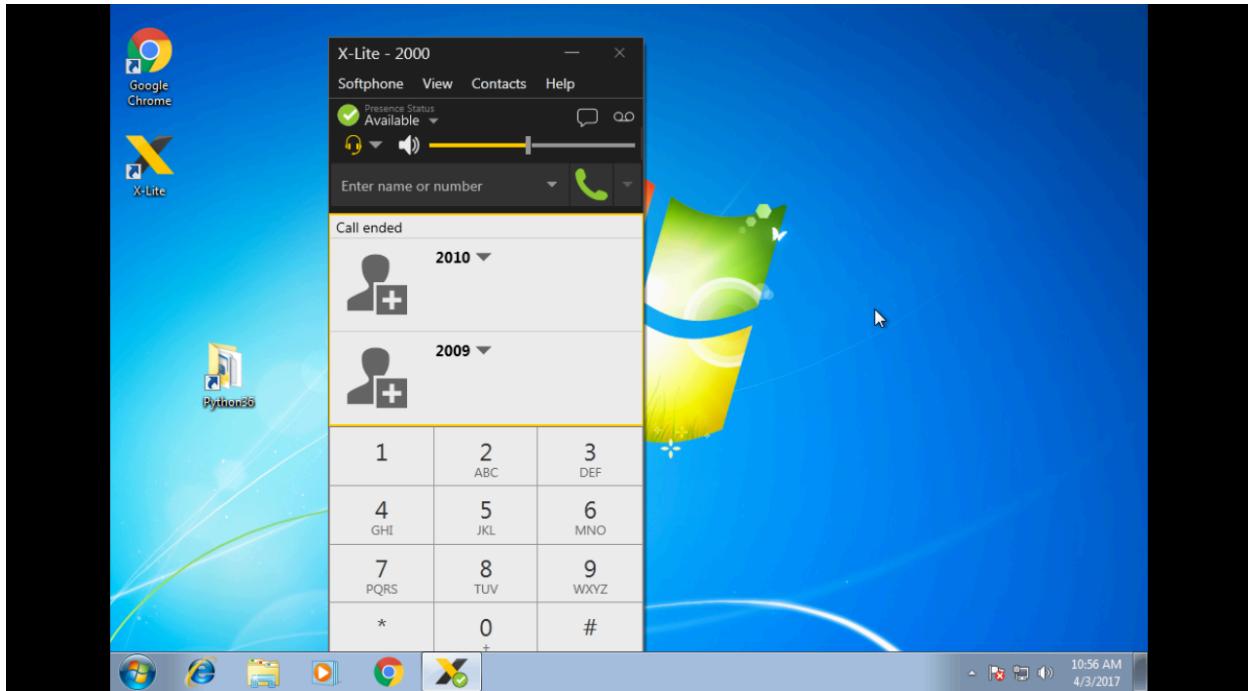


Figure : 1.4.5

Wireshark Captures:

Wireshark screenshot showing SIP traffic between two hosts, 192.168.110.136 and 192.168.110.135.

No.	Time	Source	Destination	Protocol	Length	Info
14	12.716493027	192.168.110.136	192.168.110.135	SIP	964	Request: INVITE sip:2010@192.168.110.135
15	12.716773624	192.168.110.135	192.168.110.136	SIP	617	Status: 401 Unauthorized
16	12.717877485	192.168.110.136	192.168.110.135	SIP	380	Request: ACK sip:2010@192.168.110.135
17	12.720331529	192.168.110.136	192.168.110.135	SIP/S...	1126	Request: INVITE sip:2010@192.168.110.135
18	12.721223081	192.168.110.135	192.168.110.136	SIP	562	Status: 100 Trying
19	12.724516813	192.168.110.135	192.168.110.147	SIP/S...	993	Request: INVITE sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b
20	12.806897120	192.168.110.147	192.168.110.135	SIP	348	Status: 100 Trying
21	13.007989249	192.168.110.147	192.168.110.135	SIP	510	Status: 180 Ringing
22	13.008320130	192.168.110.135	192.168.110.136	SIP	578	Status: 180 Ringing
47	17.799857990	192.168.110.147	192.168.110.135	SIP/S...	820	Status: 200 OK
48	17.800088272	192.168.110.135	192.168.110.147	SIP	510	Request: ACK sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b
49	17.800351355	192.168.110.135	192.168.110.136	SIP/S...	883	Status: 200 OK
50	17.801494305	192.168.110.135	192.168.110.147	SIP/S...	969	Request: INVITE sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b, in-
54	17.840244762	192.168.110.136	192.168.110.135	SIP	501	Request: ACK sip:2010@192.168.110.135:5060
56	17.840399284	192.168.110.135	192.168.110.136	SIP/S...	913	Request: INVITE sip:2000@192.168.110.136:60130;rinstance=12d44fc8dd9c93bc, in-
58	17.859700378	192.168.110.147	192.168.110.135	SIP/S...	820	Status: 200 OK
59	17.859924837	192.168.110.135	192.168.110.147	SIP	510	Request: ACK sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b
83	17.970890479	192.168.110.136	192.168.110.135	SIP	333	Status: 100 Trying
84	17.970926613	192.168.110.136	192.168.110.135	SIP/S...	799	Status: 200 OK
85	17.971192161	192.168.110.135	192.168.110.136	SIP	477	Request: ACK sip:2000@192.168.110.136:60130;rinstance=12d44fc8dd9c93bc
86	17.971711829	192.168.110.135	192.168.110.147	SIP/S...	969	Request: INVITE sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b, in-
101	18.124568490	192.168.110.147	192.168.110.135	SIP	361	Status: 100 Trying
115	18.272455625	192.168.110.147	192.168.110.135	SIP/S...	820	Status: 200 OK
116	18.272729577	192.168.110.135	192.168.110.147	SIP	510	Request: ACK sip:2010@192.168.110.147:55770;rinstance=cf18e177eb2fd88b
466	21.767971150	192.168.110.144	192.168.110.135	SIP/S...	964	Request: INVITE sip:2000@192.168.110.135
467	21.768167636	192.168.110.135	192.168.110.144	SIP	617	Status: 401 Unauthorized
470	21.771430640	192.168.110.144	192.168.110.135	SIP	380	Request: ACK sip:2000@192.168.110.135
472	21.788164760	192.168.110.144	192.168.110.135	SIP/S...	1126	Request: INVITE sip:2000@192.168.110.135

Frame 14: 964 bytes on wire (7712 bits), 964 bytes captured (7712 bits) on interface 0

► Ethernet II, Src: VMware_4a:c1:11 (00:0c:29:4a:c1:11), Dst: VMware_fa:59:e0 (00:0c:29:fa:59:e0)

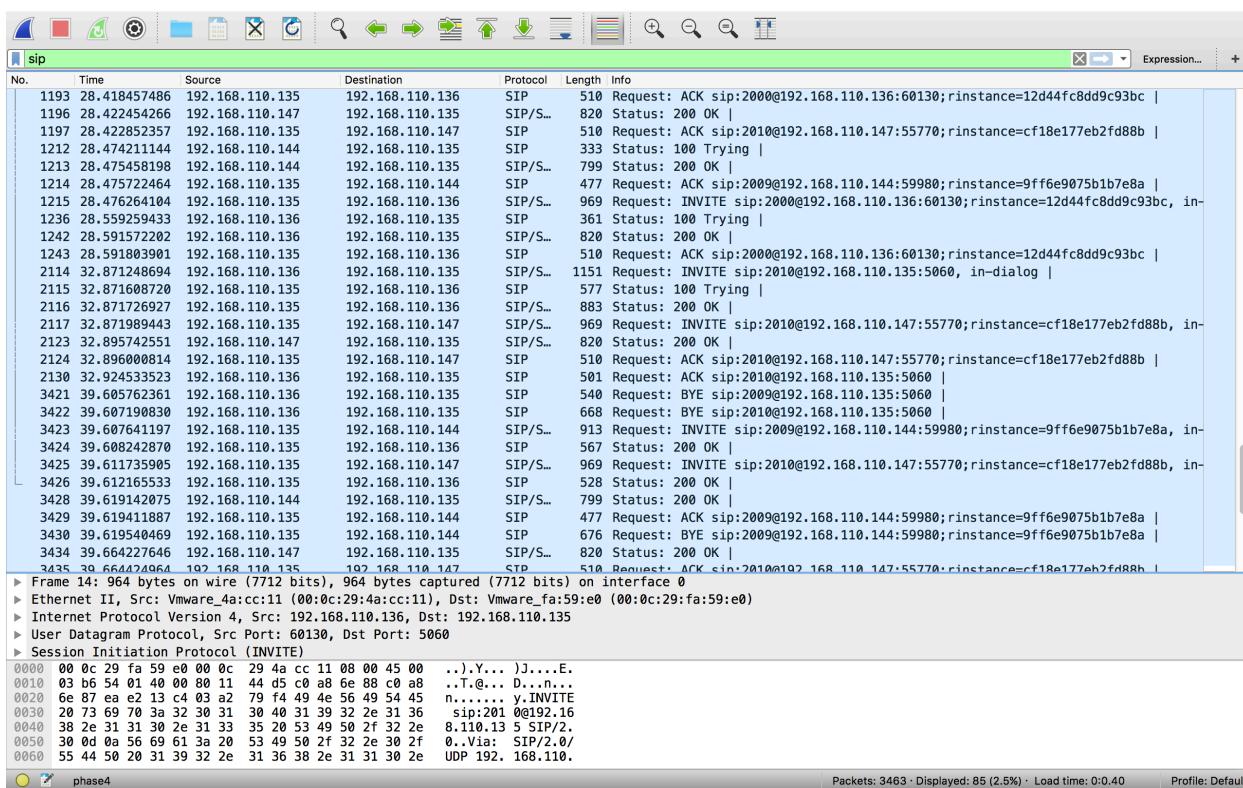
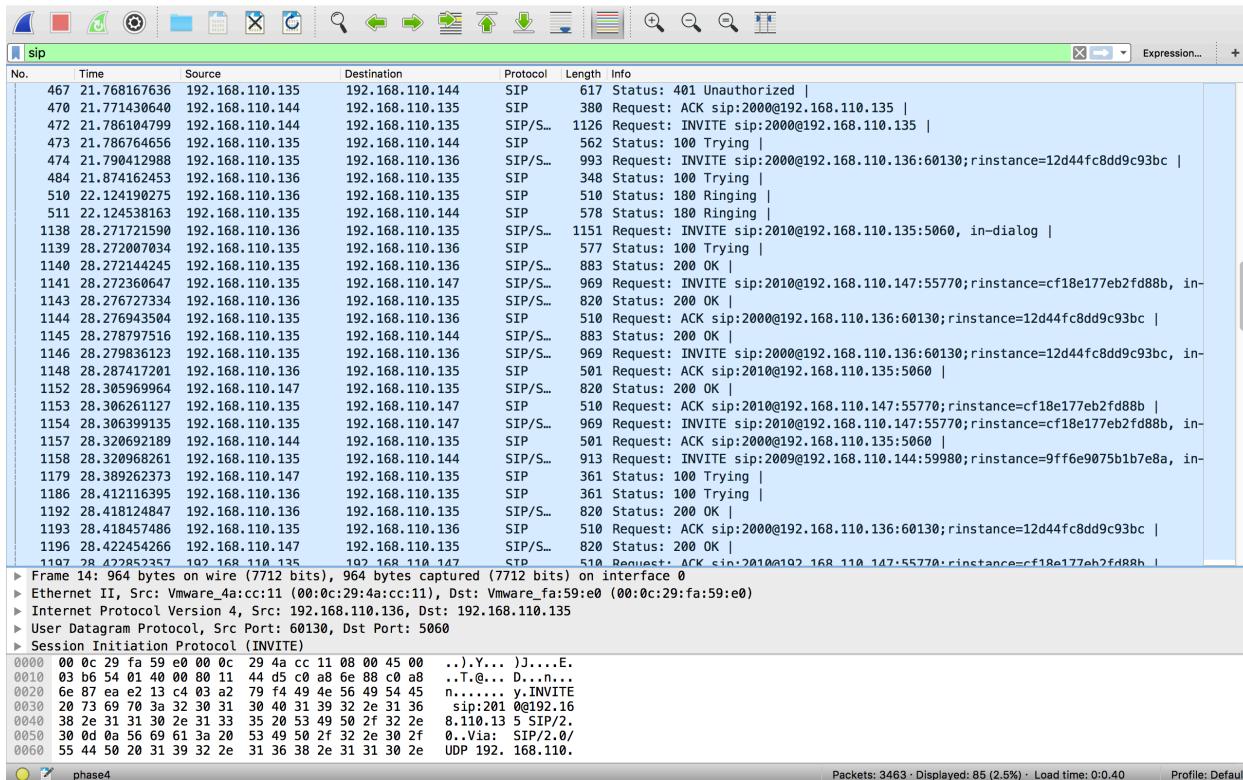
► Internet Protocol Version 4, Src: 192.168.110.136, Dst: 192.168.110.135

► User Datagram Protocol, Src Port: 60130, Dst Port: 5060

► Session Initiation Protocol (INVITE)

```
0000  00 0c 29 fa 59 e0 00 0c 29 aa cc 11 08 00 45 00  ...).Y.... )....E.
0010  03 b6 54 01 40 00 80 11 44 d5 c6 a8 68 88 c0 a8  ..T.@... D.....
0020  66 87 ea e2 13 c4 03 a2 79 f4 49 4e 56 49 54 45  n..... y.INVITE
0030  20 73 69 70 3a 32 30 31 30 40 31 39 32 2e 31 36  sip:201 0@192.16
0040  38 2e 31 31 30 2e 31 33 35 20 53 49 50 2f 32 2e 8.110.13 5 SIP/2,
0050  30 0d 0a 56 69 61 3a 20 53 49 50 2f 32 2e 30 2f 0..Via: SIP/2.0/
0060  55 44 50 20 31 39 32 2e 31 36 38 2e 31 31 30 2e UDP 192. 168.110.
```

Figure : 1.4.6



Timing Diagram:

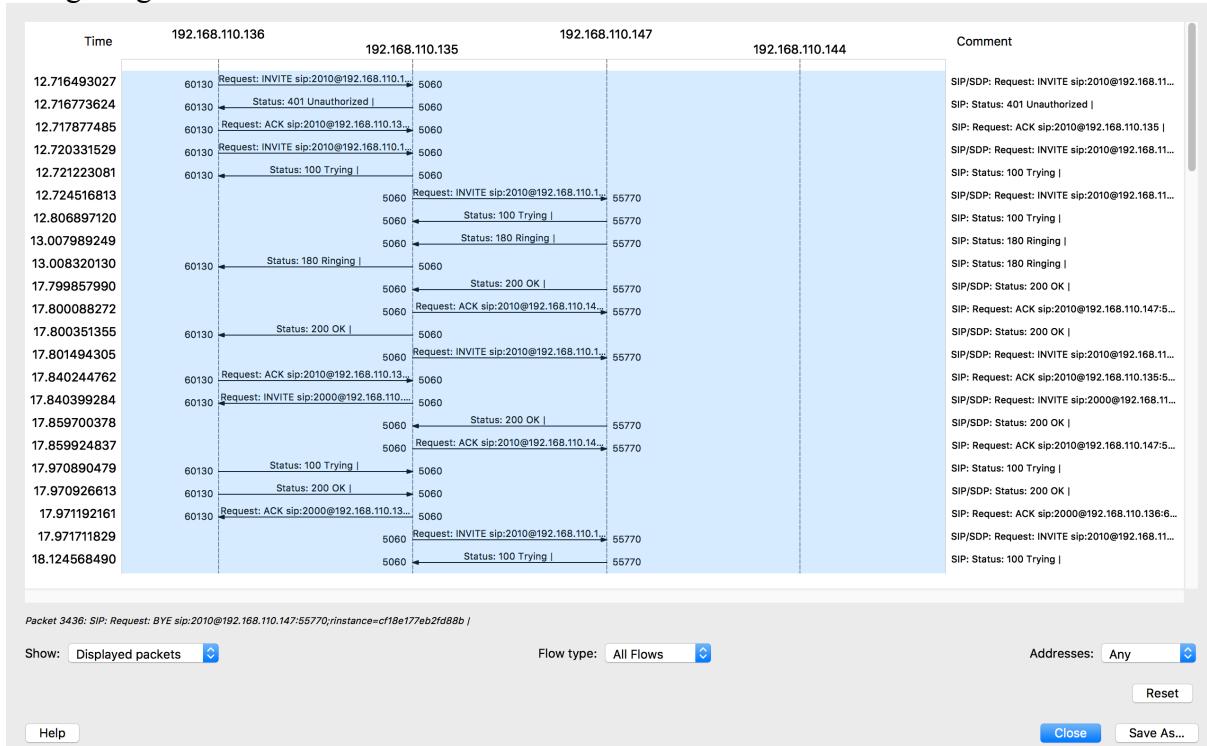


Figure : 1.4.9

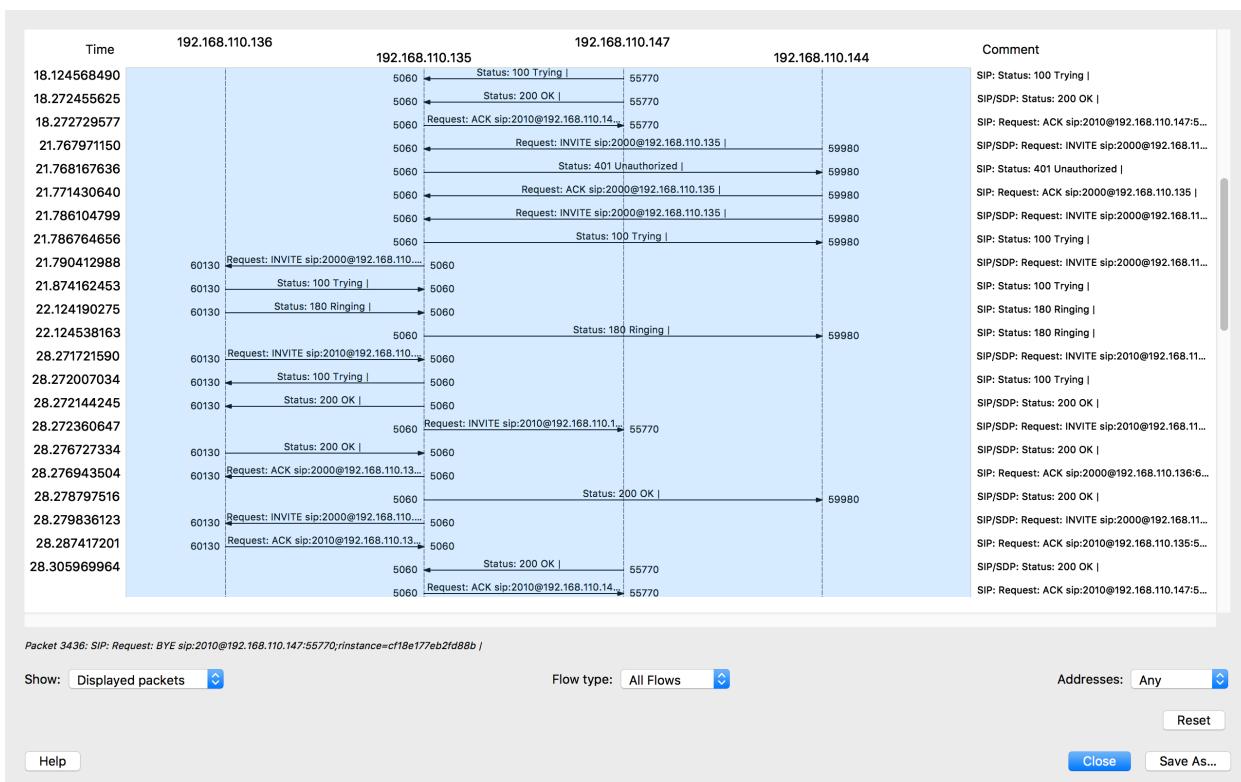


Figure : 1.4.10

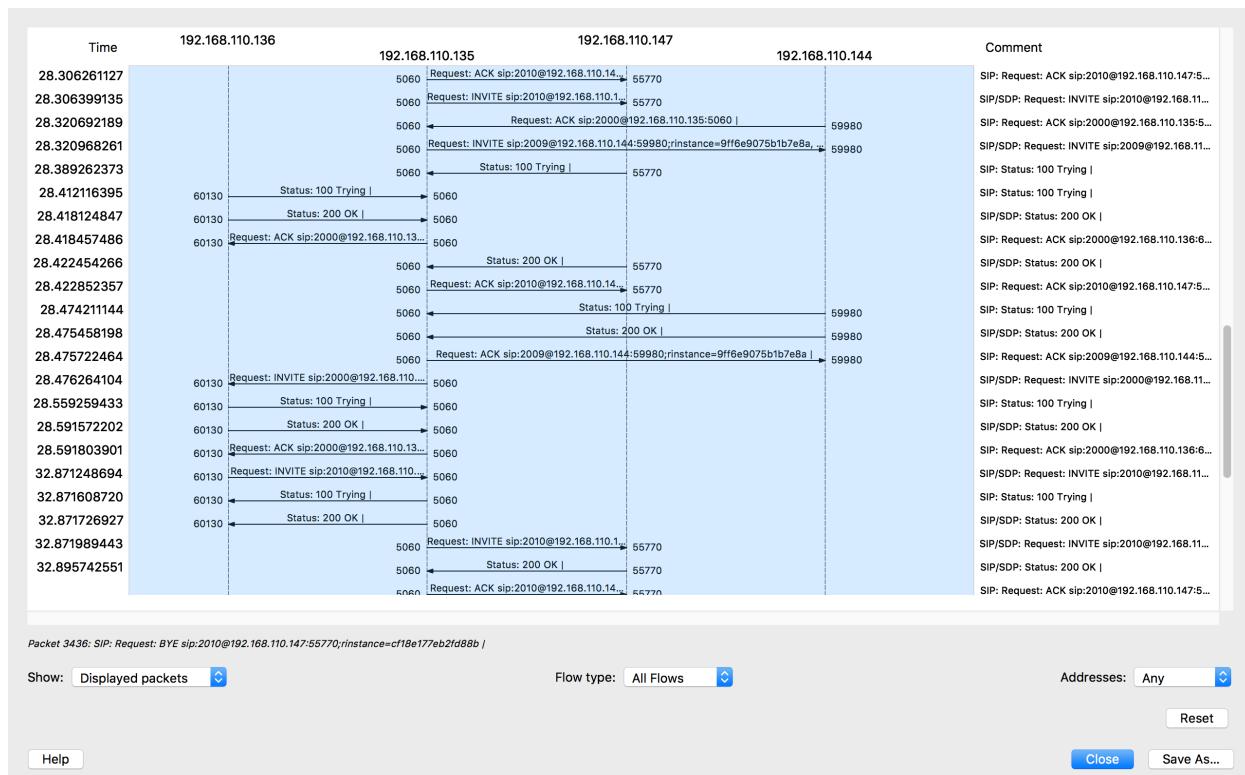


Figure : 1.4.11

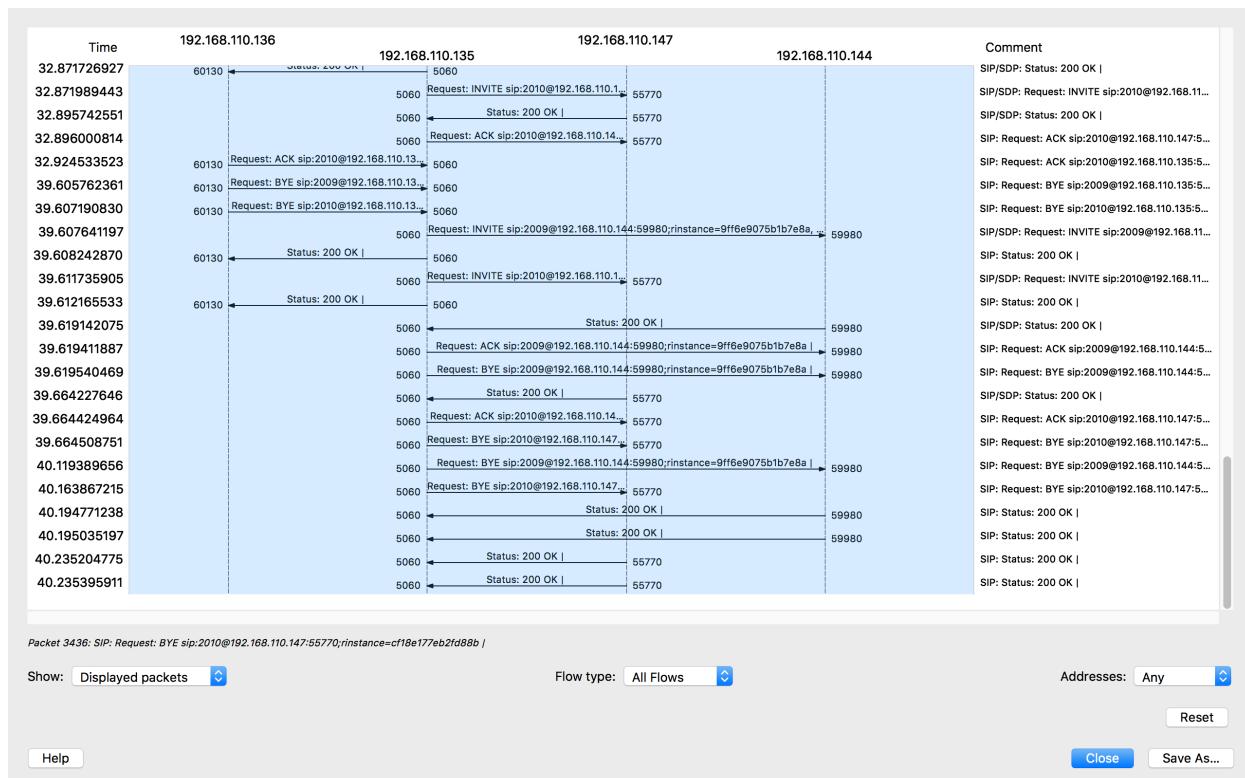


Figure : 1.4.12

SIP Flows:

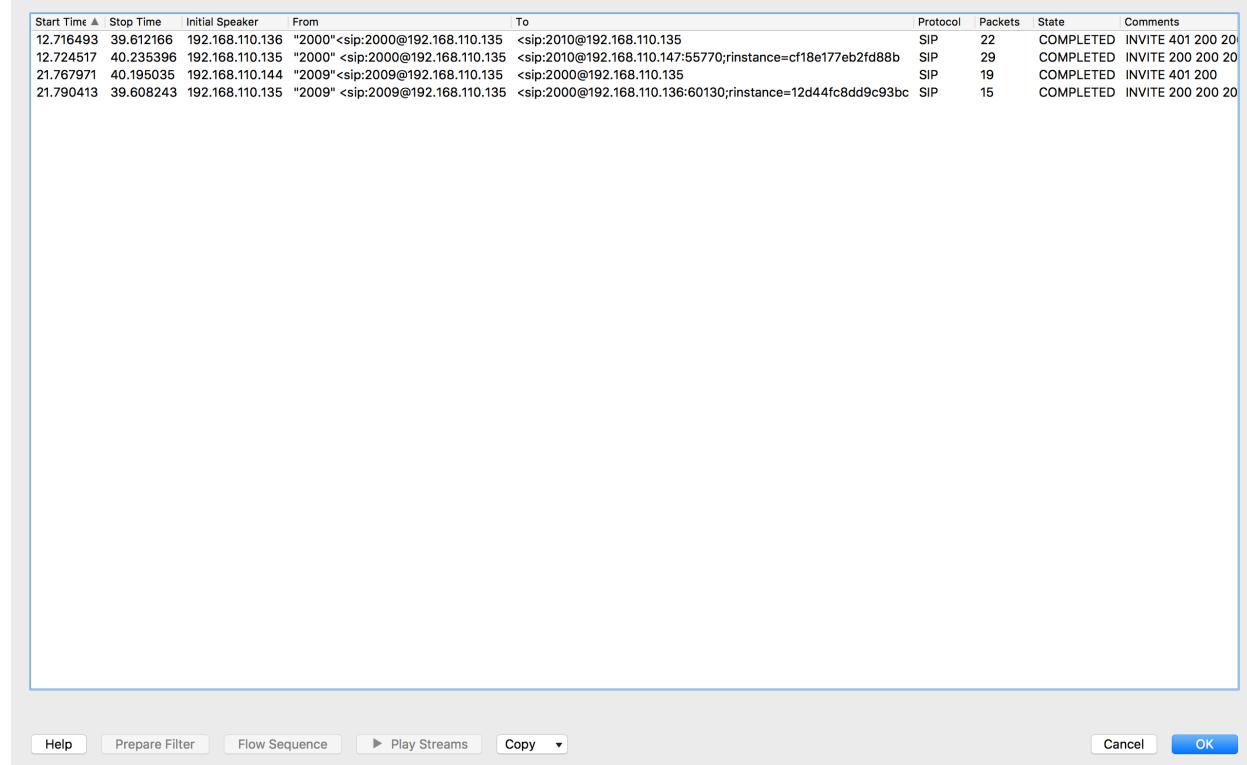


Figure : 1.4.13

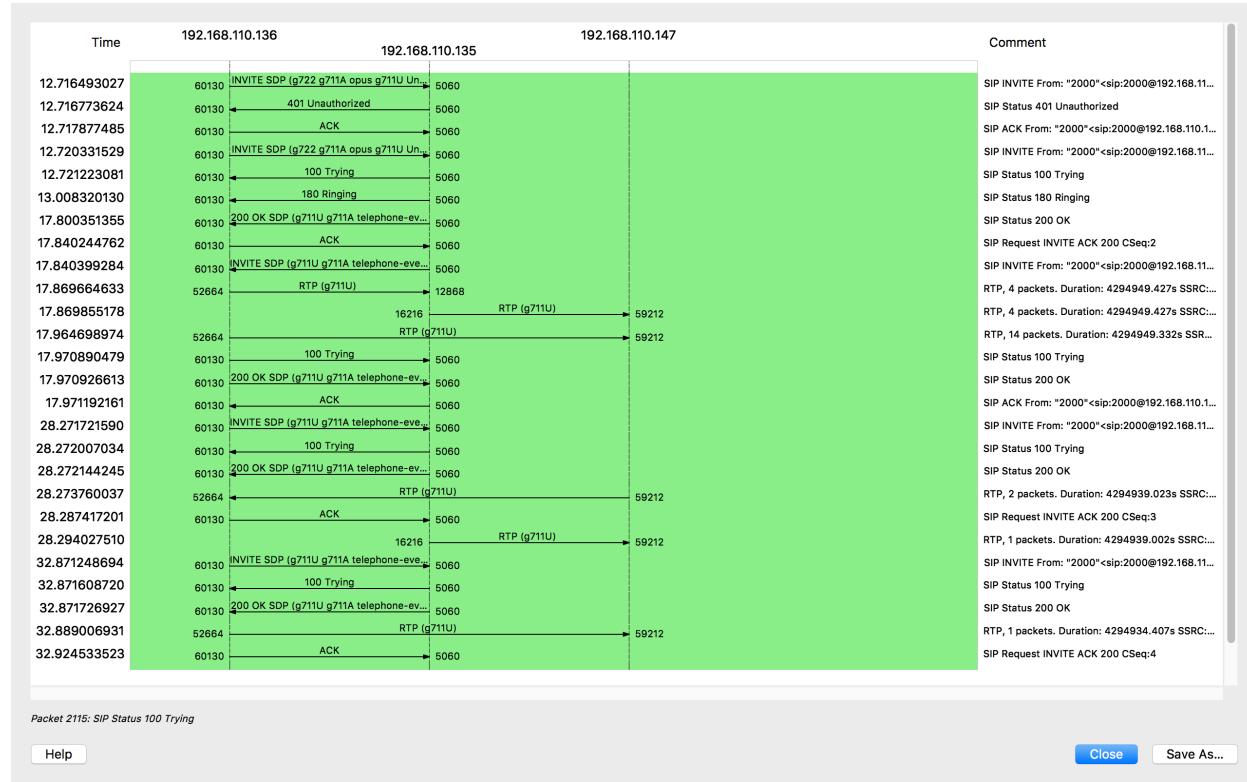


Figure : 1.4.14

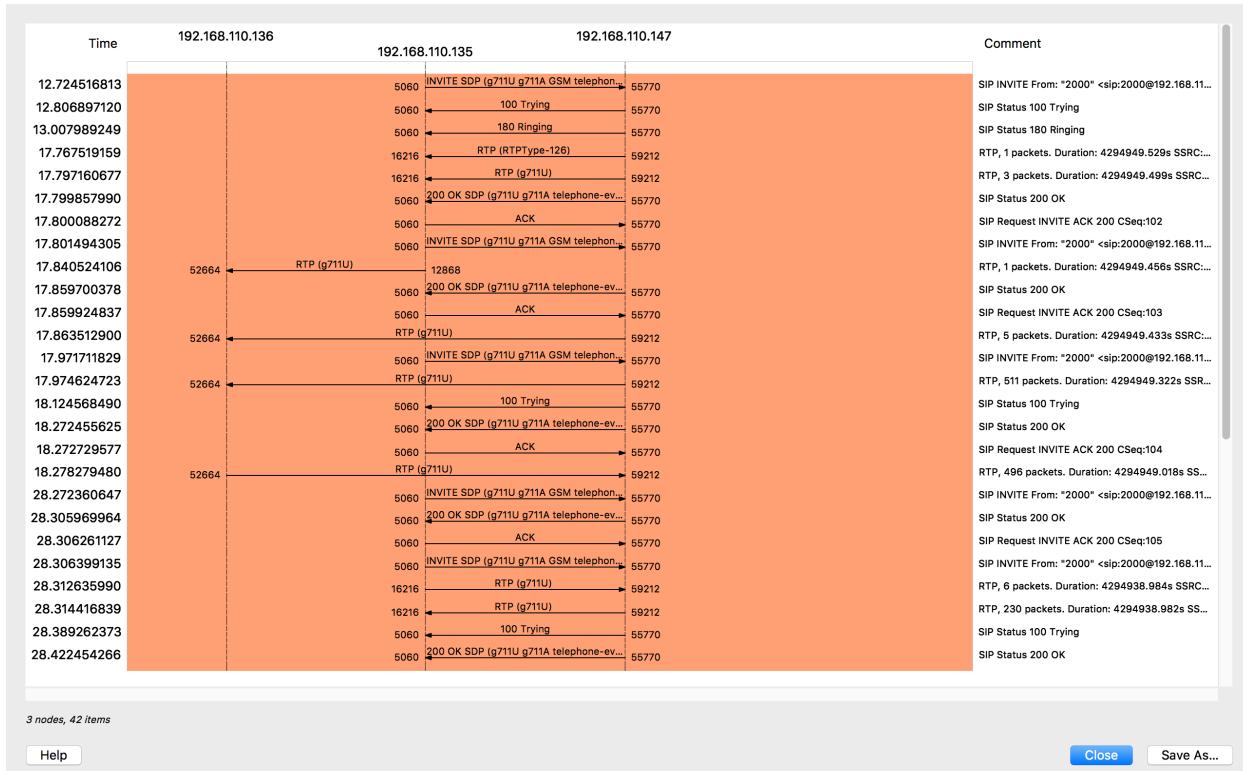


Figure : 1.4.15

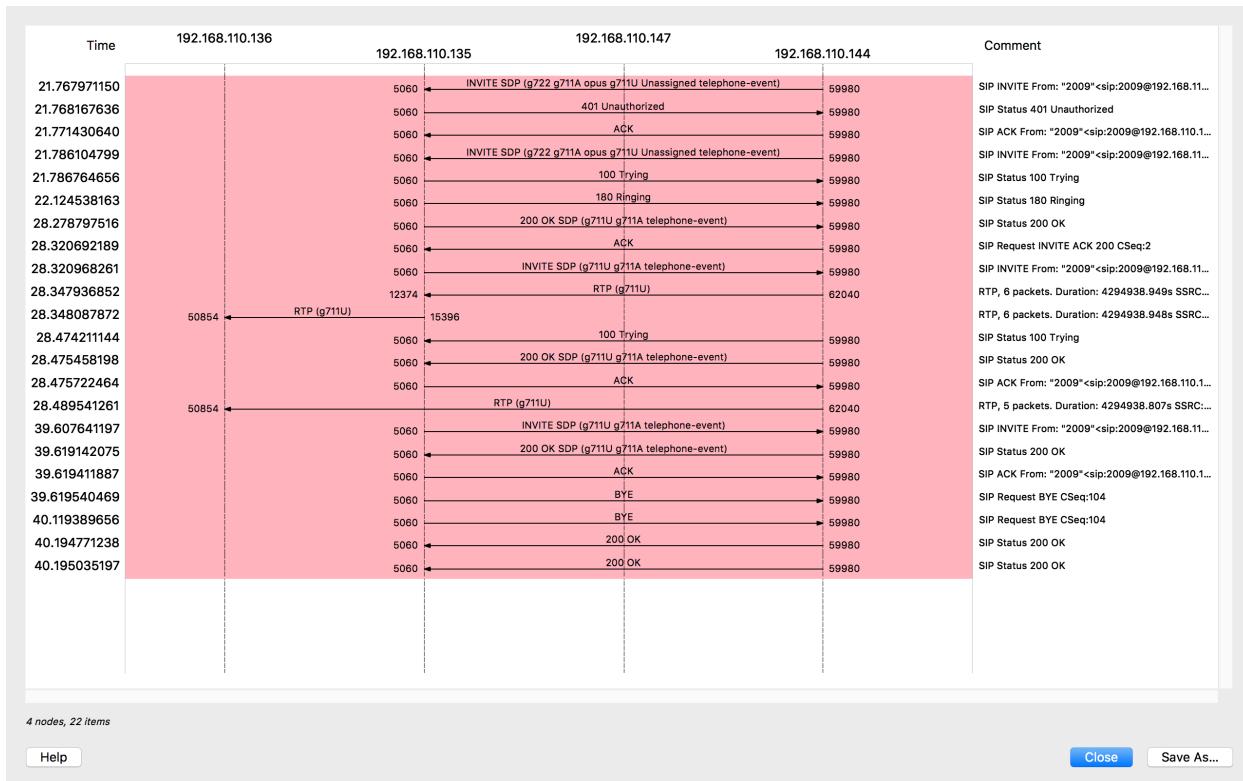


Figure : 1.4.16

Conference on:

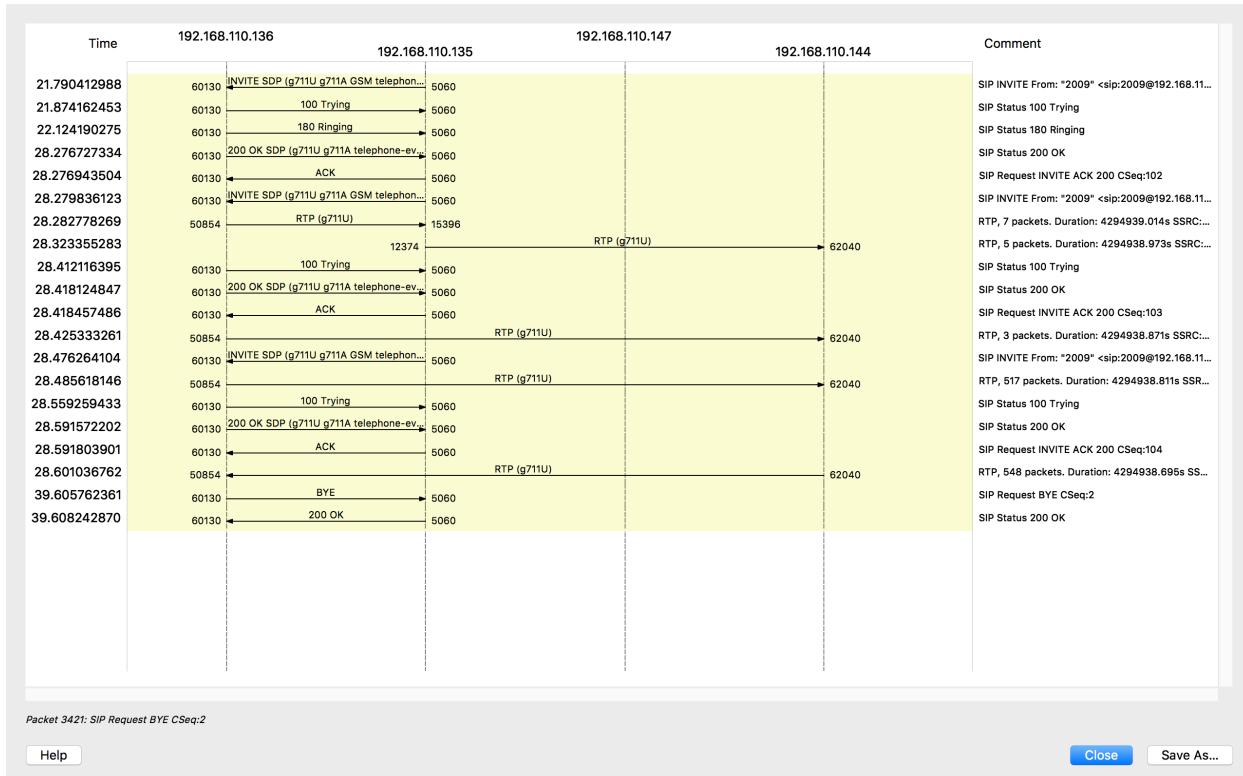


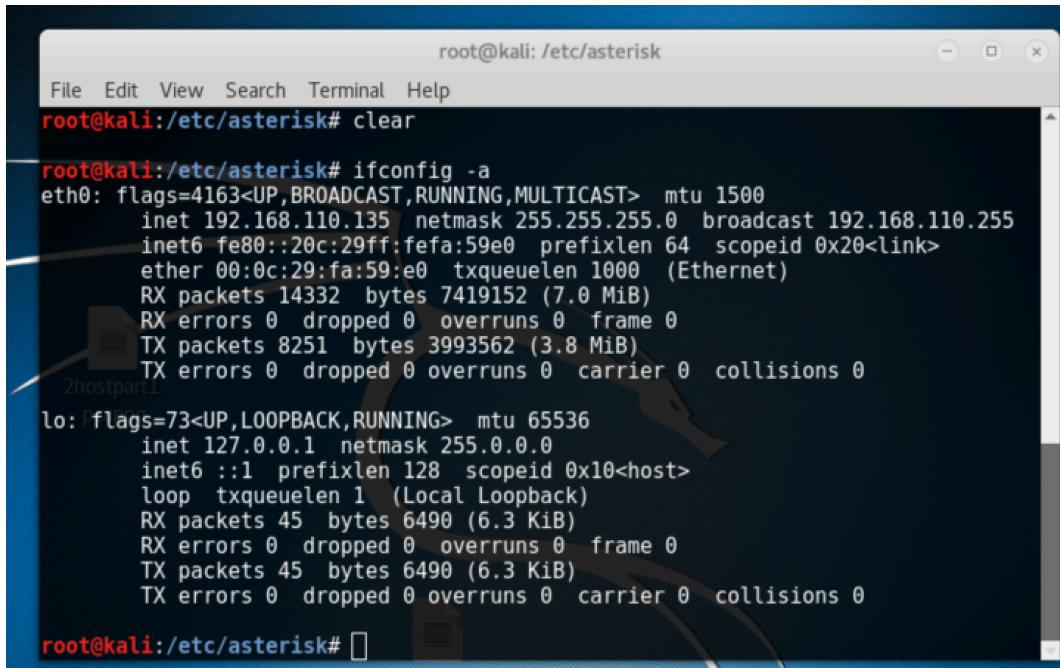
Figure : 1.4.17

Figure 1.4.14 shows establishment of calls between 2000 and 2010 and the messages exchanged at 2000's side figure:1.4.15 shows same on 2010's side. The figure:1.4.16 shows request from 2009 to 2000 to establish call when 2000 is already on call with 2010. The yellow flow graph shows commencement of conference as after accepting call from 2009 2000 invites 2010 for a conference which 2010 accepts and conference is established. The media flow can be seen in the RTP g.711 flow at 28.6 seconds in figure:1.4.17.

2.Part-II Basic SIP Client

For this part asterisk was installed on Kali Linux with one X-Lite client was loaded on Windows 7 and custom SIP client on Ubuntu their ip and MAC are as follows.

Asterisk:

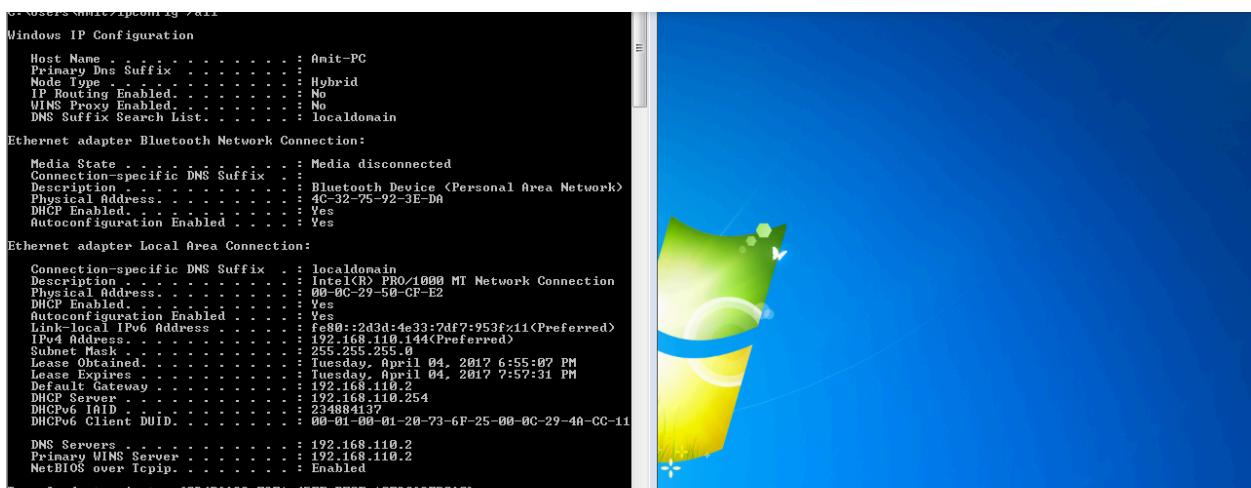


The screenshot shows a terminal window titled "root@kali: /etc/asterisk". The user has run the command "clear" to clear the screen. Then, they ran "ifconfig -a" to view all network interfaces. The output shows two interfaces: eth0 (Ethernet) and lo (loopback). The eth0 interface has an IP of 192.168.110.135 and a MAC address of 00:0c:29:fa:59:e0. The lo interface has an IP of 127.0.0.1 and a MAC address of ::1. Both interfaces show 0 errors and 0 dropped packets.

```
root@kali:/etc/asterisk# clear
root@kali:/etc/asterisk# ifconfig -a
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.110.135 netmask 255.255.255.0 broadcast 192.168.110.255
        inet6 fe80::20c:29ff:fea:59e0 prefixlen 64 scopeid 0x20<link>
            ether 00:0c:29:fa:59:e0 txqueuelen 1000 (Ethernet)
            RX packets 14332 bytes 7419152 (7.0 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 8251 bytes 3993562 (3.8 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1 (Local Loopback)
            RX packets 45 bytes 6490 (6.3 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 45 bytes 6490 (6.3 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
root@kali:/etc/asterisk#
```

Figure : 2.1

User on Windows

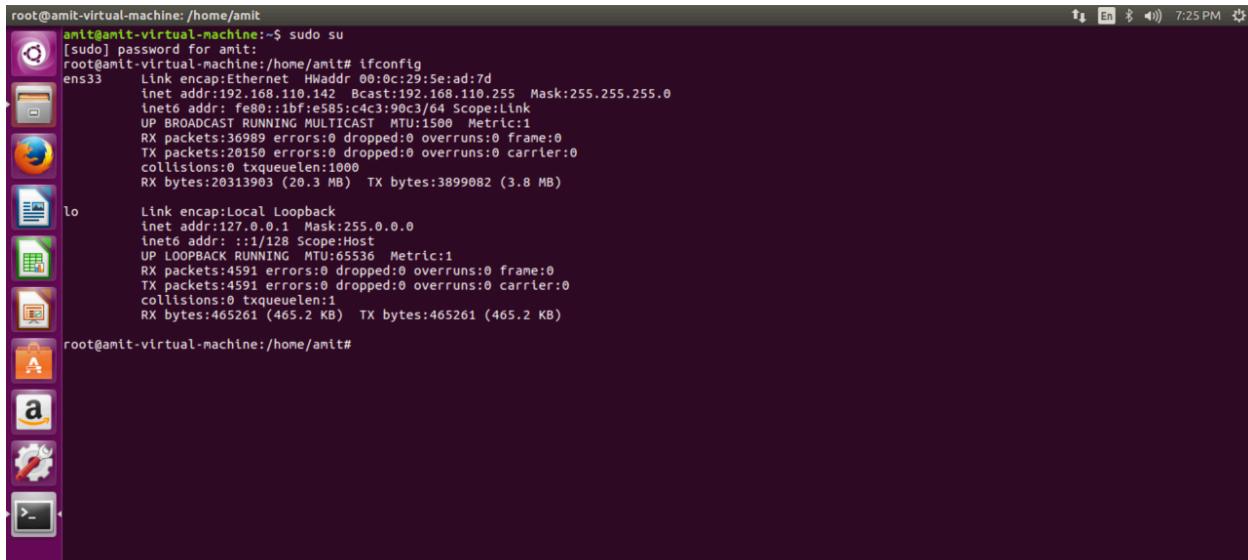


The screenshot shows the Windows Control Panel's "Network and Sharing Center". Under "Network Status", it says "This PC is connected to a network". Below that, under "Network and Sharing Center", it says "Windows is currently using Local Area Connection". On the left, there are icons for "Change adapter settings", "Change advanced sharing settings", and "Change power settings". The main pane displays network connection details for "Local Area Connection". It shows the connection is "Media disconnected" and lists the following properties:

Property	Value
Host Name	Amit-PC
Primary Dns Suffix	localdomain
Node Type	Hybrid
IP Routing Enabled	No
WINS Proxy Enabled	No
DNS Suffix Search List	localdomain
Ethernet adapter Bluetooth Network Connection:	
Media State	Media disconnected
Connection-specific DNS Suffix	localdomain
Description	Bluetooth Device <Personal Area Network>
Physical Address	4C-32-75-92-3E-DA
DHCP Enabled	Yes
Autoconfiguration Enabled	Yes
Ethernet adapter Local Area Connection:	
Connection-specific DNS Suffix	localdomain
Description	Intel(R) PRO/1000 MT Network Connection
Physical Address	00-0C-29-50-CF-E2
DHCP Enabled	Yes
Autoconfiguration Enabled	Yes
Link-local IPv6 Address	fe80::2d3d:4e33%11<Preferred>
IPv4 Address	192.168.110.144<Preferred>
Subnet Mask	255.255.255.0
Lease Obtained	Tuesday, April 04, 2017 6:55:07 PM
Lease Expires	Tuesday, April 04, 2017 7:57:31 PM
DNS Servers	192.168.110.254
DHCPv6 IID	234884137
DHCPv6 Client DUID	00-01-00-01-20-73-6F-25-00-0C-29-4A-CC-11
DNS Servers	192.168.110.2
Primary WINS Server	192.168.110.2
NetBIOS over Tcpip	Enabled

Figure : 2.2

Personalised client on Ubuntu:

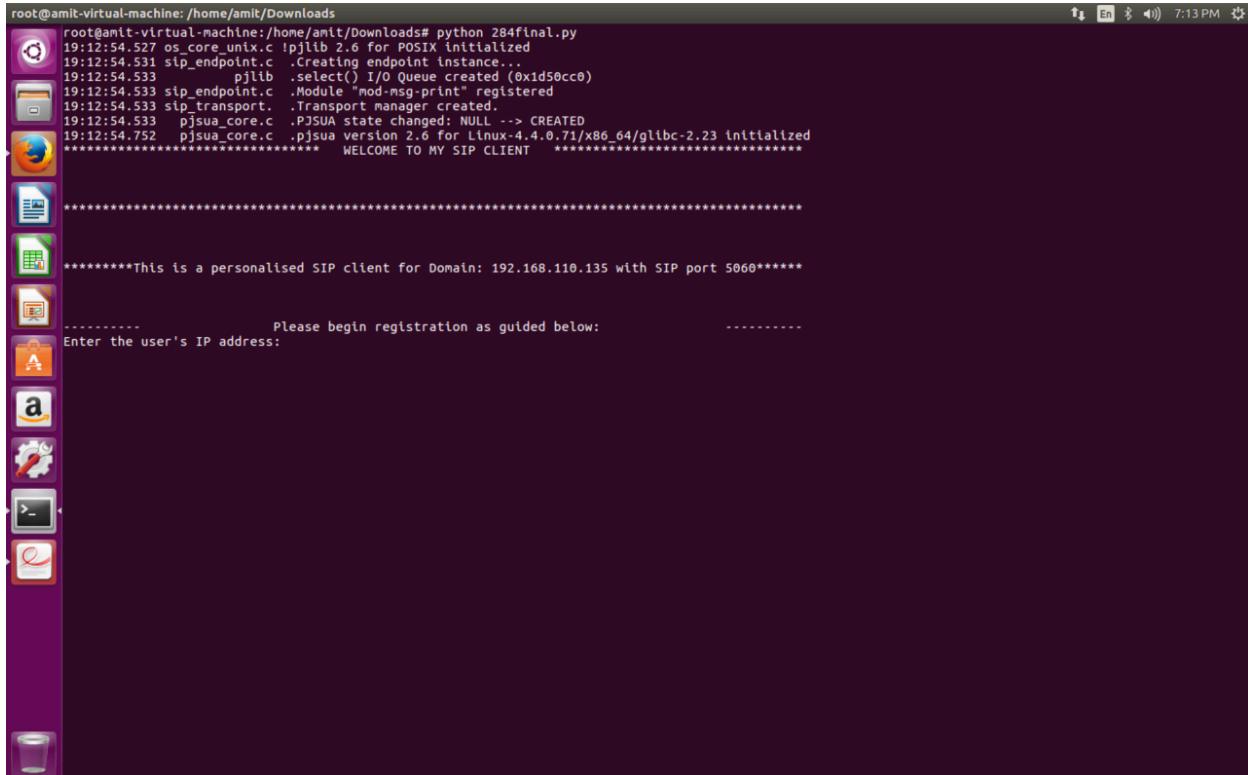


```
root@amit-virtual-machine:/home/amit
[amit@amit-virtual-machine:~]$ sudo su
root@amit-virtual-machine:/home/amit# ifconfig
ens33    Link encap:Ethernet HWaddr 00:0c:29:5e:ad:7d
          inet addr:192.168.110.142 Bcast:192.168.110.255 Mask:255.255.255.0
          inet6 addr: fe80::1bf1:e585:c4c3:90c3/64 Scope:link
             UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
             RX packets:36989 errors:0 dropped:0 overruns:0 frame:0
             TX packets:20150 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1000
             RX bytes:20313983 (20.3 MB) TX bytes:3899082 (3.8 MB)

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:4591 errors:0 dropped:0 overruns:0 frame:0
             TX packets:4591 errors:0 dropped:0 overruns:0 carrier:0
             collisions:0 txqueuelen:1
             RX bytes:465261 (465.2 KB) TX bytes:465261 (465.2 KB)
root@amit-virtual-machine:/home/amit#
```

Figure : 2.3

The below screenshot shows execution of custom code 284final.py file asking user to enter his IP

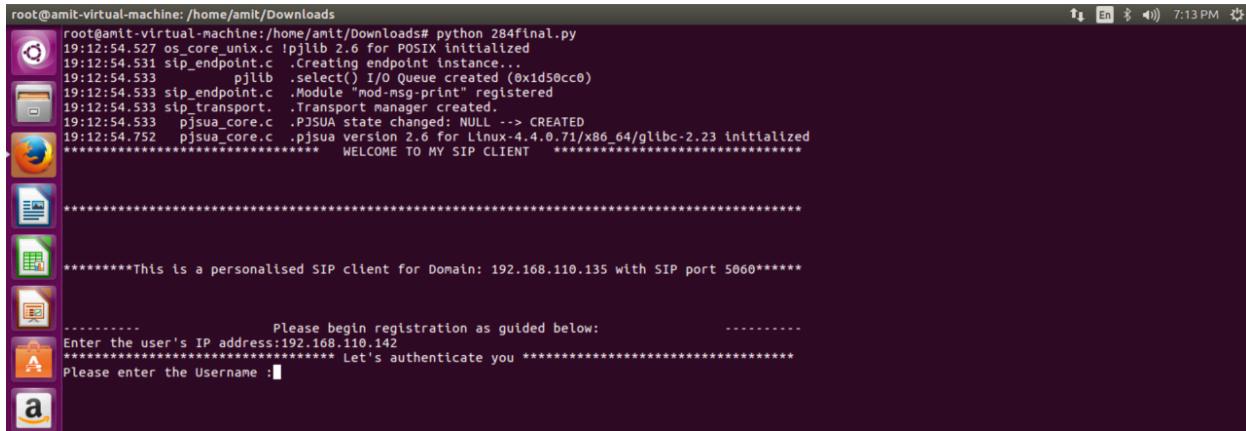


```
root@amit-virtual-machine:/home/amit/Downloads
root@amit-virtual-machine:/home/amit/Downloads# python 284final.py
19:12:54.527 os_core_unix.c [pjlib 2.6 for POSIX Initialized
19:12:54.531 stp_endpoint.c .Creating endpoint instance...
19:12:54.533     pjlib .select() I/O Queue created (0x1d50cc0)
19:12:54.533 stp_endpoint.c .Module "nod-msg-print" registered
19:12:54.533 stp_transport.c .Transport manager created.
19:12:54.533 pjsua_core.c .PJSUA state changed: NULL --> CREATED
19:12:54.752 pjsua_core.c .pjsua version 2.6 for Linux-4.4.0.71/x86_64/glibc-2.23 initialized
***** WELCOME TO MY SIP CLIENT *****

*****
*****This is a personalised SIP client for Domain: 192.168.110.135 with SIP port 5060*****
-----
----- Please begin registration as guided below: -----
Enter the user's IP address:
```

Figure : 2.4

Once the IP i.e 192.168.110.142 for client is entered authentication starts by asking the username

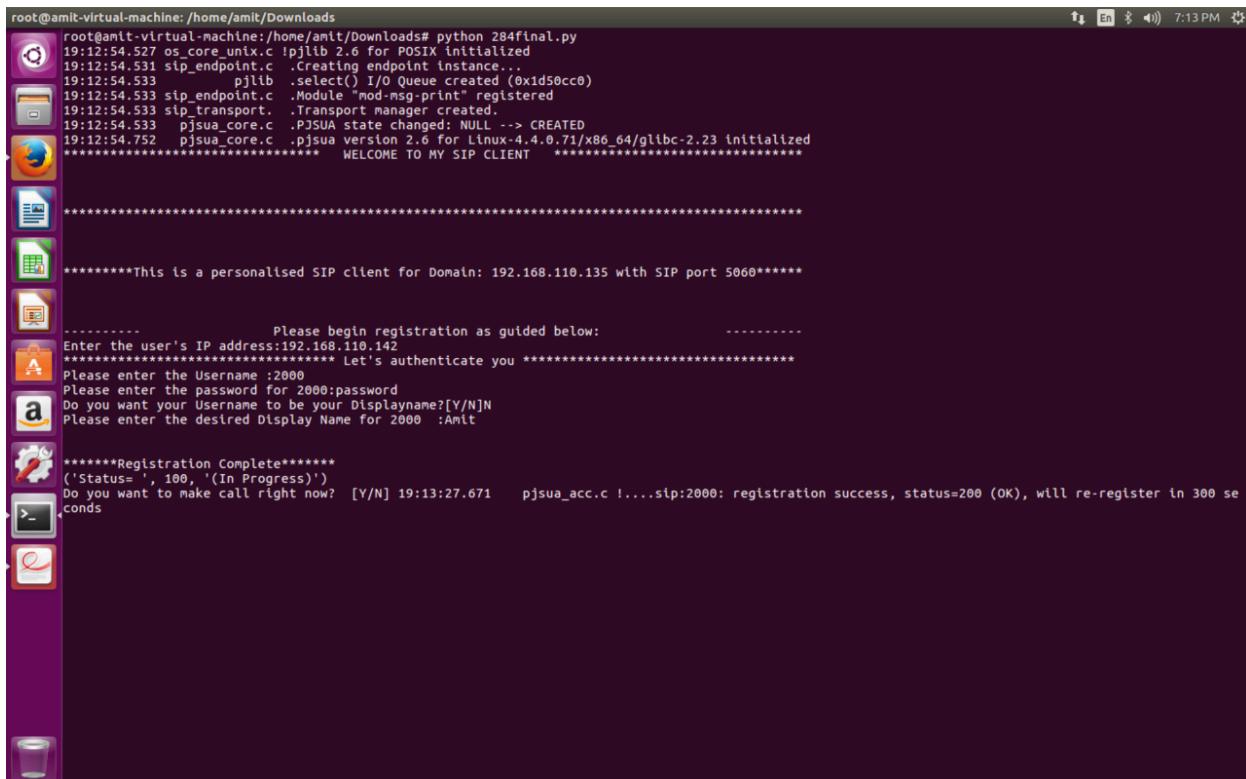


```
root@amit-virtual-machine:/home/amit/Downloads
root@amit-virtual-machine:/home/amit/Downloads# python 284final.py
19:12:54.527 os_core_unix.c |pjlib 2.6 for POSIX initialized
19:12:54.531 sip_endpoint.c .Creating endpoint instance...
19:12:54.533     pjlib .select() I/O Queue created (0x1d50cc0)
19:12:54.533 sip_endpoint.c .Module "mod-msg-print" registered
19:12:54.533 sip_transport.c .Transport manager created.
19:12:54.533 pjsua_core.c .PJSUA state changed: NULL --> CREATED
19:12:54.752 pjsua_core.c .pjsua version 2.6 for Linux-4.4.0.71/x86_64/glibc-2.23 initialized
***** WELCOME TO MY SIP CLIENT *****

*****
*****This is a personalised SIP client for Domain: 192.168.110.135 with SIP port 5060*****
-----
Please begin registration as guided below:
Enter the user's IP address:192.168.110.142
***** Let's authenticate you *****
Please enter the Username :■
```

Figure : 2.5

we enter the username 2000 which we earlier included in sip and extensions configuration files and the password and using displayname Amit instead of 2000. After this the client registers himself with the asterisk server and asks if the user wants to call.



```
root@amit-virtual-machine:/home/amit/Downloads
root@amit-virtual-machine:/home/amit/Downloads# python 284final.py
19:12:54.527 os_core_unix.c |pjlib 2.6 for POSIX initialized
19:12:54.531 sip_endpoint.c .Creating endpoint instance...
19:12:54.533     pjlib .select() I/O Queue created (0x1d50cc0)
19:12:54.533 sip_endpoint.c .Module "mod-msg-print" registered
19:12:54.533 sip_transport.c .Transport manager created.
19:12:54.533 pjsua_core.c .PJSUA state changed: NULL --> CREATED
19:12:54.752 pjsua_core.c .pjsua version 2.6 for Linux-4.4.0.71/x86_64/glibc-2.23 initialized
***** WELCOME TO MY SIP CLIENT *****

*****
*****This is a personalised SIP client for Domain: 192.168.110.135 with SIP port 5060*****
-----
Please begin registration as guided below:
Enter the user's IP address:192.168.110.142
***** Let's authenticate you *****
Please enter the Username :2000
Please enter the password for 2000:password
Do you want your Username to be your Displayname?[Y/N]N
Please enter the desired Display Name for 2000 :Amit

*****Registration Complete*****
('Status: ', 100, '(In Progress)')
Do you want to make call right now? [Y/N] 19:13:27.671  pjsua_acc.c !...sip:2000: registration success, status=200 (OK), will re-register in 300 se
conds
```

Figure : 2.6

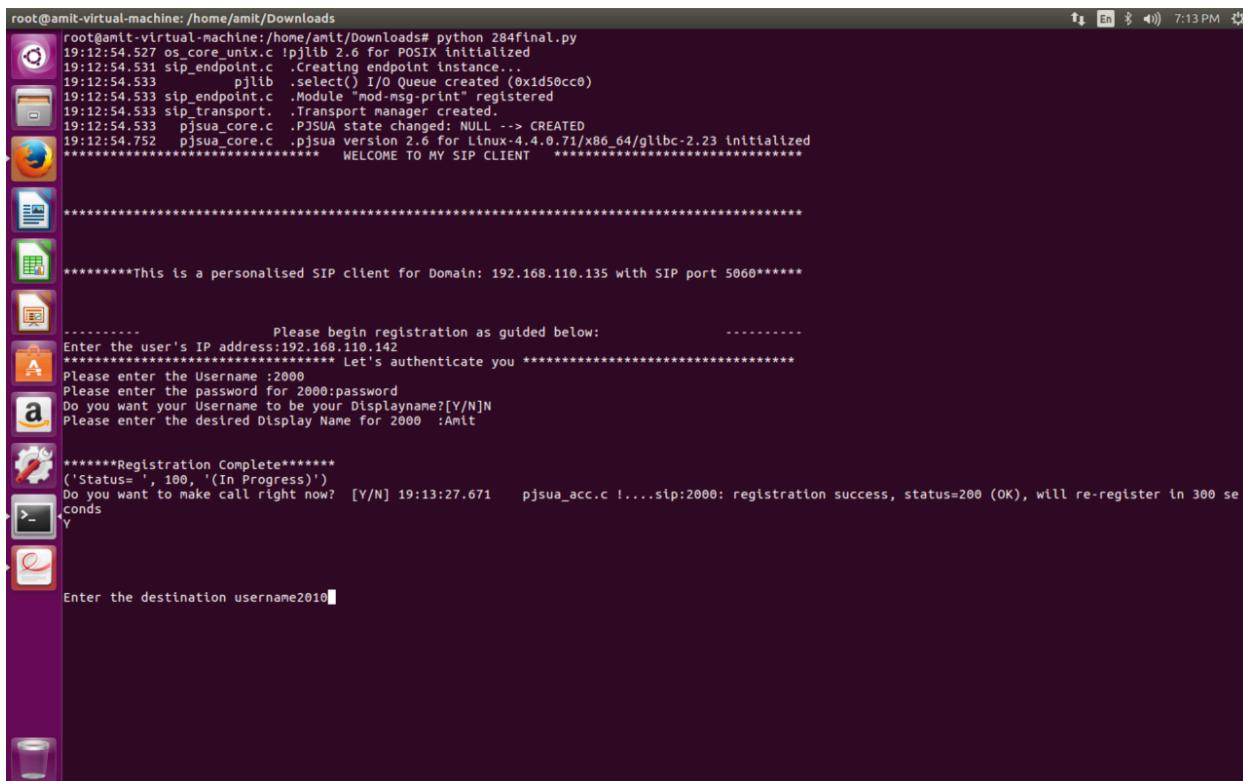
The screenshot of asterisk server showing 2000 registered :



```
root@kali:~# sudo asterisk -rvvv
Asterisk 13.14.0-dfsg-1, Copyright (C) 1999 - 2014, 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 13.14.0-dfsg-1 currently running on kali (pid = 1488)
-- Registered SIP '2000' at 192.168.110.142:5060
kali*CLI>
```

Figure : 2.7

On entering y client asks for username to be called at the asterisk server running on 192.168.110.135



```
root@amit-virtual-machine:/home/amit/Downloads#
root@amit-virtual-machine:/home/amit/Downloads# python 284final.py
19:12:54.527 os_core_unix.c [pjlib 2.6 for POSIX initialized
19:12:54.531 sip_endpoint.c .Creating endpoint instance...
19:12:54.533 pjlib .select() I/O Queue created (0x1d50cc0)
19:12:54.533 sip_endpoint.c .Module "mod-msg-print" registered
19:12:54.533 sip_transport.c .Transport manager created.
19:12:54.533 pjsua_core.c .PJSUA state changed: NULL --> CREATED
19:12:54.752 pjsua_core.c .pjsua version 2.6 for Linux-4.4.0.71/x86_64/glibc-2.23 initialized
***** WELCOME TO MY SIP CLIENT *****

*****
*****This is a personalised SIP client for Domain: 192.168.110.135 with SIP port 5060*****
*****
Please begin registration as guided below:
-----
Enter the user's IP address:192.168.110.142
***** Let's authenticate you *****
Please enter the Username :2000
Please enter the password for 2000:password
Do you want your Username to be your Displayname?[Y/N]N
Please enter the desired Display Name for 2000 :Amit

*****
Registration Complete*****
('Status= ', 100, '(In Progress)')
Do you want to make call right now? [Y/N] 19:13:27.671 pjsua_acc.c !...sip:2000: registration success, status=200 (OK), will re-register in 300 se
conds
Y

Enter the destination username2010
```

Figure : 2.8

We call 2010 from our client and the screenshot shows client calling 2010 and prompts to press enter to finish call:

```

root@amit-virtual-machine:/home/amit/Downloads#
root@amit-virtual-machine:/home/amit/Downloads# python 284final.py
19:12:54.527 os_core_unix.c [pjlib 2.6 for POSIX initialized
19:12:54.531 sip_endpoint.c .Creating endpoint instance...
19:12:54.533 pjlib .select() I/O Queue created (0x1d50cc0)
19:12:54.533 sip_endpoint.c .Module "mod-msg-print" registered
19:12:54.533 sip_transport.c .Transport manager created.
19:12:54.533 pjsua_core.c .PJSUA state changed: NULL --> CREATED
19:12:54.752 pjsua_core.c .pjsua version 2.6 for Linux-4.4.0.71/x86_64/glibc-2.23 initialized
***** WELCOME TO MY SIP CLIENT *****

*****
*****This is a personalised SIP client for Domain: 192.168.110.135 with SIP port 5060*****
***** Please begin registration as guided below: *****
Enter the user's IP address:192.168.110.142
***** Let's authenticate you *****
Please enter the Username :2000
Please enter the password for 2000:password
Do you want your Username to be your Displayname?[Y/N]N
Please enter the desired Display Name for 2000 :Amit

*****Registration Complete*****
('Status= ', 100, '(In Progress)')
Do you want to make call right now? [Y/N] 19:13:27.671 pjsua_acc.c !....sip:2000: registration success, status=200 (OK), will re-register in 300 se
conds
Y

Enter the destination username2010
Press <ENTER> to exit and destroy library

```

Figure : 2.9

The below figure shows asterisk server events where call from 2000 to 2010 has been initiated:

```

Applications ▾ Places ▾ Terminal ▾ Tue 19:13
Kali
root@kali:~# sudo asterisk -rvvvv
Asterisk 13.14.0-dfsg-1, Copyright (C) 1999 - 2014, 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 13.14.0-dfsg-1 currently running on kali (pid = 1488)
-- Registered SIP '2000' at 192.168.110.142:5060
-- Using SIP RTP CoS mark 5
-- Executing [2010@from-sip:1] Dial("SIP/2000-00000002", "SIP/2010,20")
-- Using SIP RTP CoS mark 5
-- Called SIP/2010
-- SIP/2010-00000003 is ringing
kali*CLI> 
ghostpart2.pcapng
phase3-4.pcapng
cus.pcapng

```

Figure : 2.10

Call from server at 2010

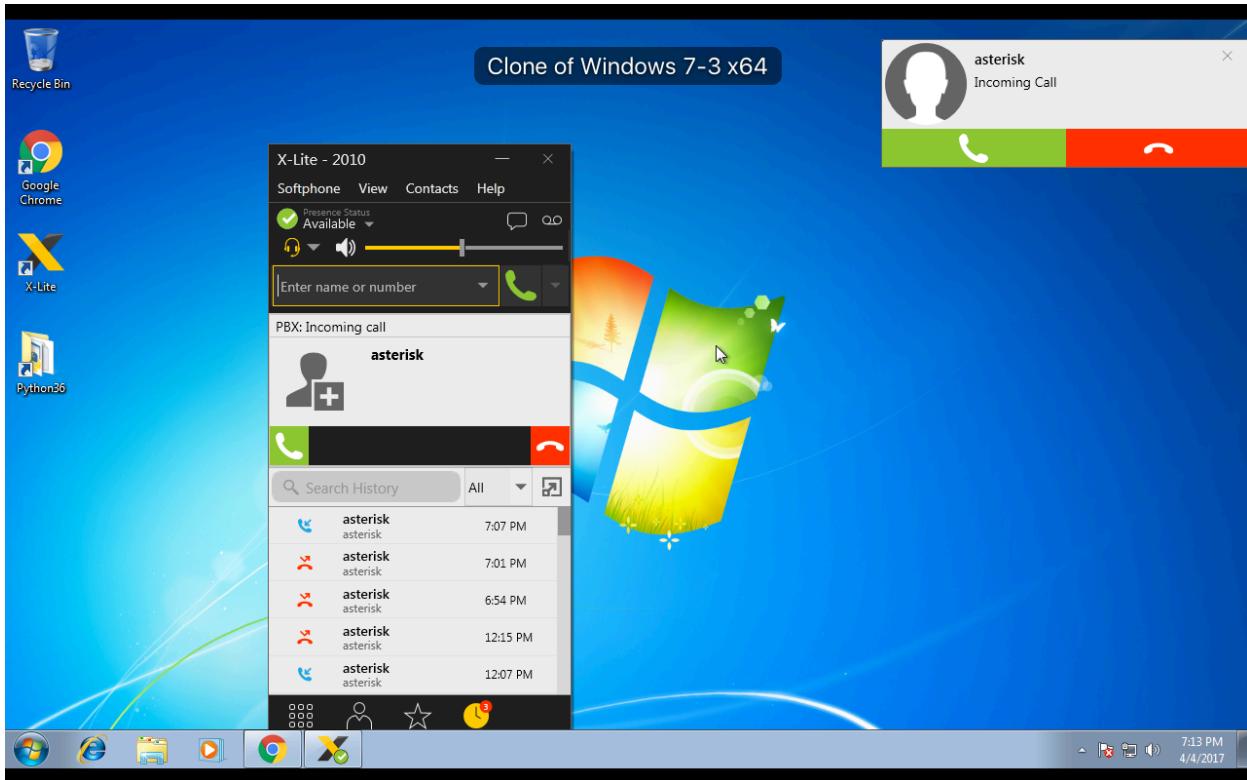


Figure : 2.11

2010 accepts call

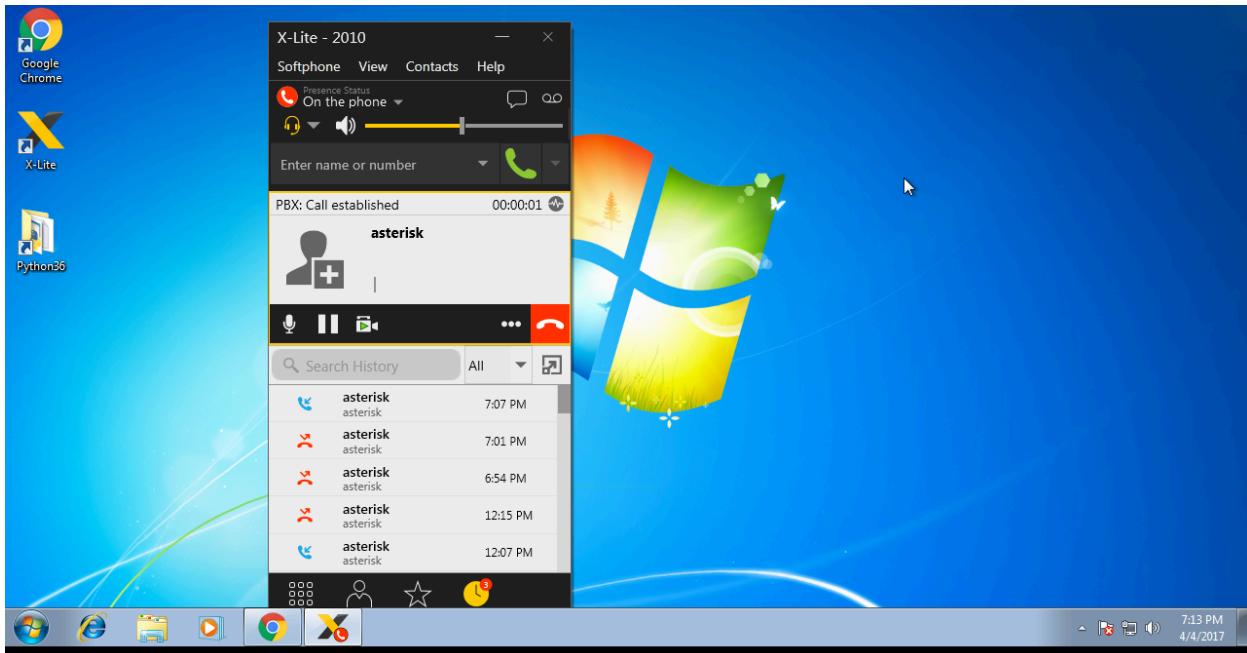
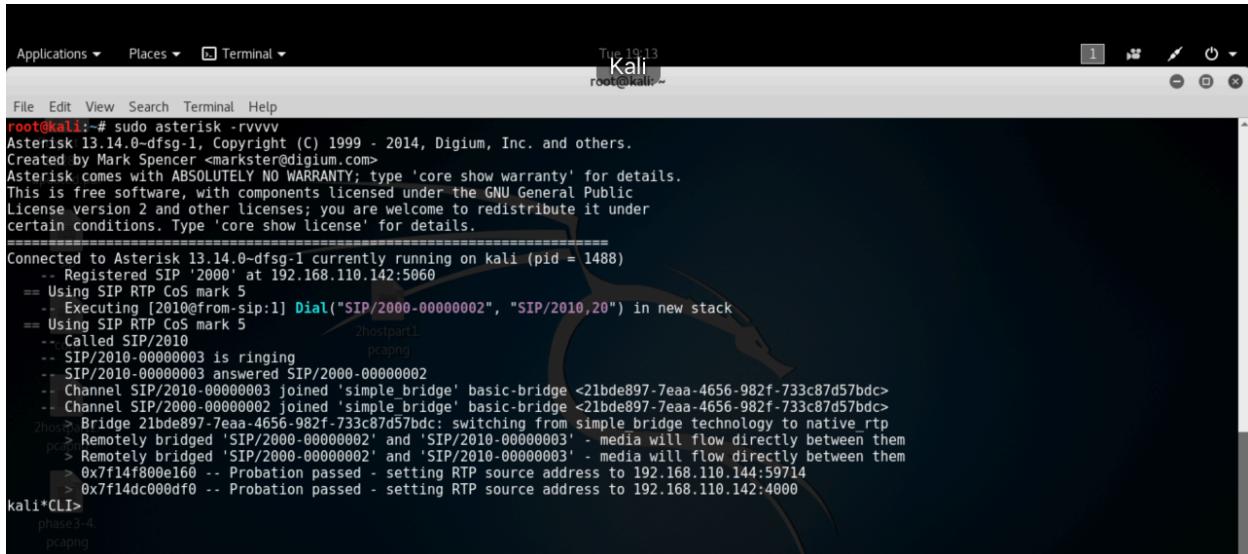


Figure : 2.12

The below is the events at asterisk server showing call established



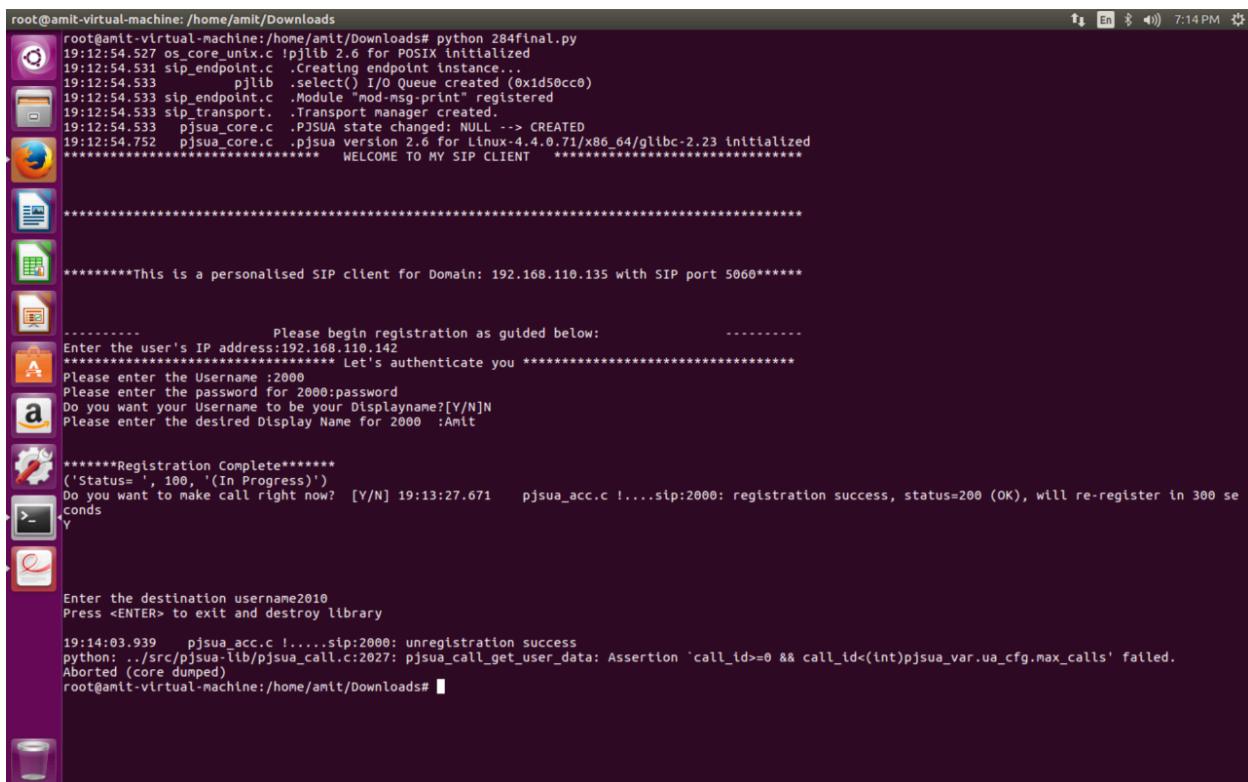
```

root@kali:~# sudo asterisk -rvvv
Asterisk 13.14.0-dfsg-1, Copyright (C) 1999 - 2014, 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 13.14.0-dfsg-1 currently running on kali (pid = 1488)
-- Registered SIP '2000' at 192.168.110.142:5060
== Using SIP RTP CoS mark 5
-- Executing [2010@from-sip:1] Dial("SIP/2000-00000002", "SIP/2010,20") in new stack
== Using SIP RTP CoS mark 5
-- Called SIP/2010
-- SIP/2010-00000003 is ringing
-- SIP/2010-00000003 answered SIP/2000-00000002
-- Channel SIP/2010-00000003 joined 'simple_bridge' basic-bridge <21bde897-7eaa-4656-982f-733c87d57bdc>
-- Channel SIP/2000-00000002 joined 'simple_bridge' basic-bridge <21bde897-7eaa-4656-982f-733c87d57bdc>
--> Remote bridge 21bde897-7eaa-4656-982f-733c87d57bdc: switching from simple bridge technology to native_rtp
--> Remotely bridged 'SIP/2000-00000002' and 'SIP/2010-00000003' - media will flow directly between them
--> Remotely bridged 'SIP/2000-00000002' and 'SIP/2010-00000003' - media will flow directly between them
--> 0x7f14f800e160 -- Probation passed - setting RTP source address to 192.168.110.144:59714
--> 0x7f14dc000df0 -- Probation passed - setting RTP source address to 192.168.110.142:4000
kali*CLI>
phase 3-4
pcapng

```

Figure : 2.13

The figure shows call ended from 2000's side and the libraries being destroyed.



```

root@amit-virtual-machine:/home/amit/Downloads#
root@amit-virtual-machine:/home/amit/Downloads# python 284final.py
19:12:54.527 os_core_unix.c [pjlib 2.6 for POSIX initialized
19:12:54.531 sip_endpoint.c .Creating endpoint instance...
19:12:54.533     pjlib .select() I/O Queue created (0x1d50cc0)
19:12:54.533 sip_endpoint.c .Module "mod-msg-print" registered
19:12:54.533 sip_transport.c .Transport manager created.
19:12:54.533 pjsua_core.c .PJSUA state changed: NULL --> CREATED
19:12:54.752 pjsua_core.c .pjsua version 2.6 for Linux-4.4.0.71/x86_64/glibc-2.23 initialized
***** WELCOME TO MY SIP CLIENT *****

*****
*****This is a personalised SIP client for Domain: 192.168.110.135 with SIP port 5060*****
*****

Please begin registration as guided below:
Enter the user's IP address:192.168.110.142
***** Let's authenticate you *****
Please enter the Username :2000
Please enter the password for 2000:password
Do you want your Username to be your Displayname?[Y/N]N
Please enter the desired Display Name for 2000 :Amit

*****Registration Complete*****
('Status= ', 100, '(In Progress)')
Do you want to make call right now? [Y/N] 19:13:27.671    pjsua_acc.c !....sip:2000: registration success, status=200 (OK), will re-register in 300 seconds
Y

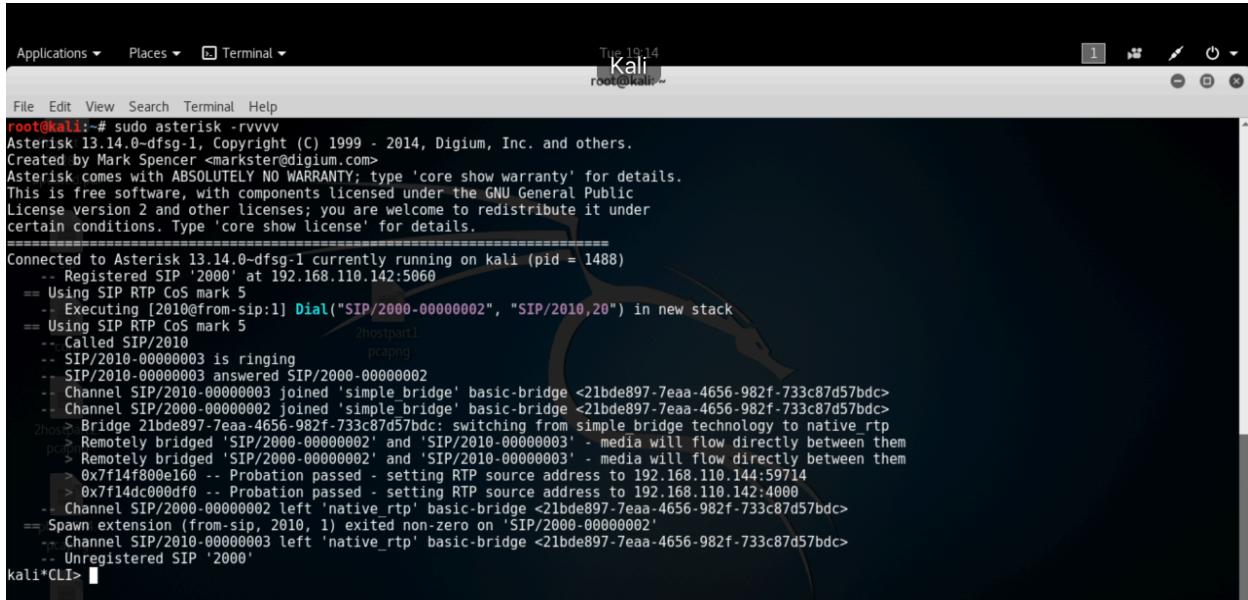
Enter the destination username2010
Press <ENTER> to exit and destroy library

19:14:03.939    pjsua_acc.c !....sip:2000: unregistration success
python: ../../src/pjsua-lib/pjsua_call.c:2027: pjsua_call_get_user_data: Assertion `call_id>=0 && call_id<(int)pjsua_var.ua_cfg.max_calls' failed.
Aborted (core dumped)
root@amit-virtual-machine:/home/amit/Downloads# 

```

Figure : 2.14

The asterisk server's output showing the whole process of registration ringing call and unregistration of 2000



```

root@kali:~# sudo asterisk -rvvv
Asterisk 13.14.0-dfsg-1, Copyright (C) 1999 - 2014, 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 13.14.0-dfsg-1 currently running on kali (pid = 1488)
-- Registered SIP '2000' at 192.168.110.142:5060
-- Using SIP RTP CoS mark 5
-- Executing [2010@from-sip:1] Dial("SIP/2000-00000002", "SIP/2010,20") in new stack
-- Using SIP RTP CoS mark 5
-- Called SIP/2010
-- SIP/2010-00000003 is ringing
-- SIP/2010-00000003 answered SIP/2000-00000002
-- Channel SIP/2010-00000003 joined 'simple_bridge' basic-bridge <2lbd897-7eaa-4656-982f-733c87d57bdc>
-- Channel SIP/2000-00000002 joined 'simple_bridge' basic-bridge <2lbd897-7eaa-4656-982f-733c87d57bdc>
--> Bridge 2lbd897-7eaa-4656-982f-733c87d57bdc: switching from simple bridge technology to native_rtp
--> Remotely bridged 'SIP/2000-00000002' and 'SIP/2010-00000003' - media will flow directly between them
--> Remotely bridged 'SIP/2000-00000002' and 'SIP/2010-00000003' - media will flow directly between them
--> 0x7f14dc0080f0 -- Probation passed - setting RTP source address to 192.168.110.142:4000
--> 0x7f14dc0080f0 -- Probation passed - setting RTP source address to 192.168.110.142:4000
-- Channel SIP/2000-00000002 left 'native_rtp' basic-bridge <2lbd897-7eaa-4656-982f-733c87d57bdc>
-- Spawn extension (from-sip, 2010, 1) exited non-zero on 'SIP/2000-00000002'
--> Channel SIP/2010-00000003 left 'native_rtp' basic-bridge <2lbd897-7eaa-4656-982f-733c87d57bdc>
-- Unregistered SIP '2000'
kali*CLI> 

```

Figure : 2.15

Wireshark Captures:

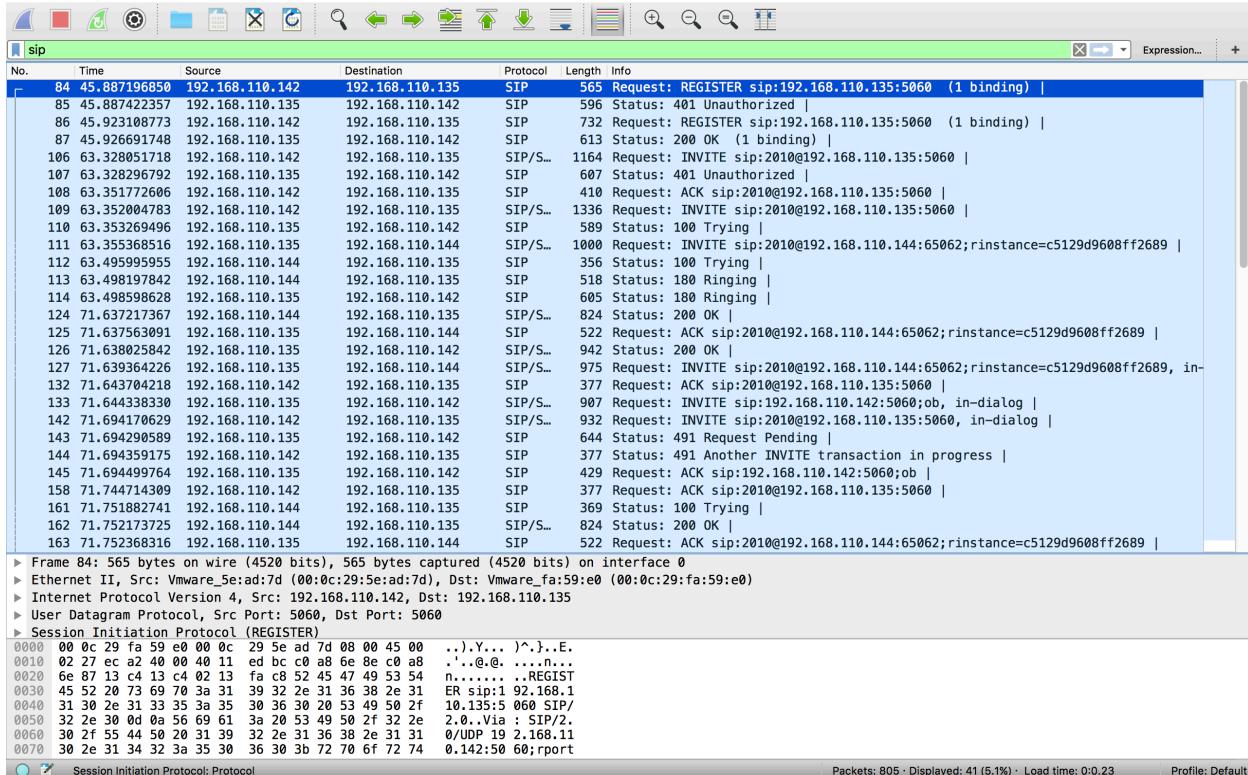


Figure : 2.16

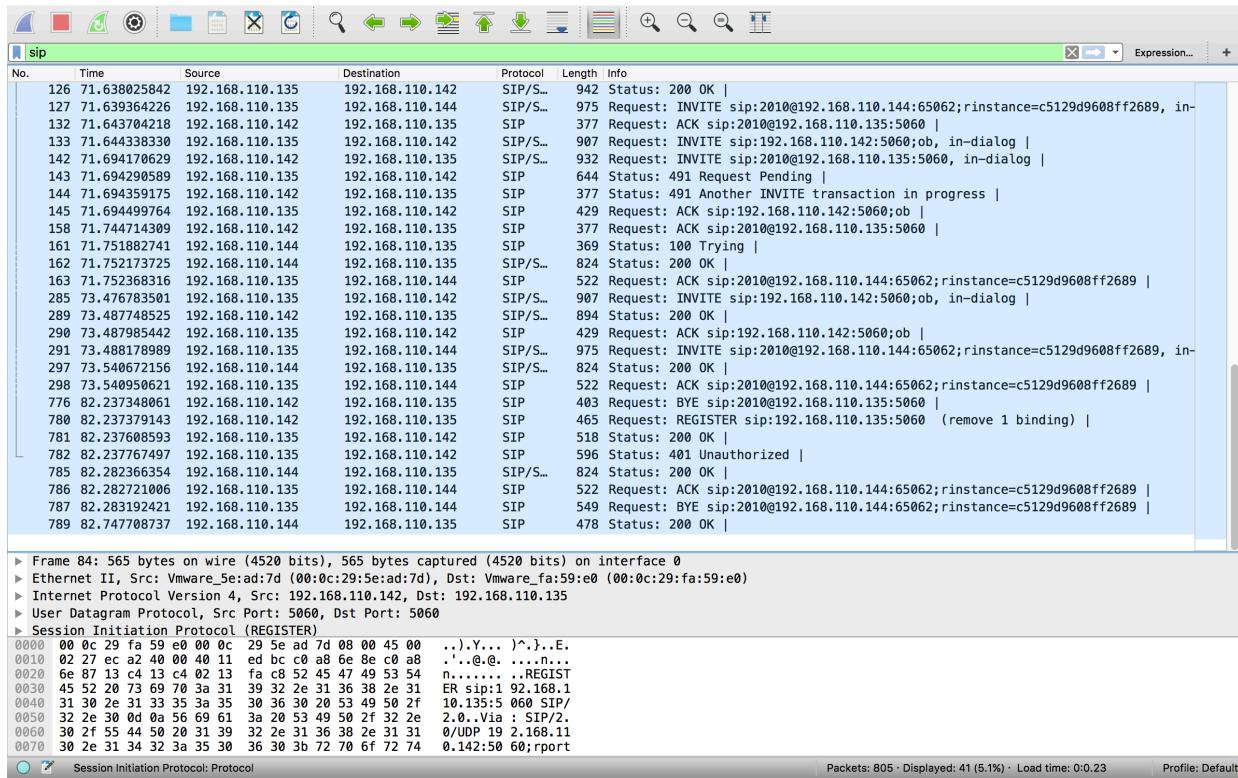


Figure : 2.17

Timing Diagram:

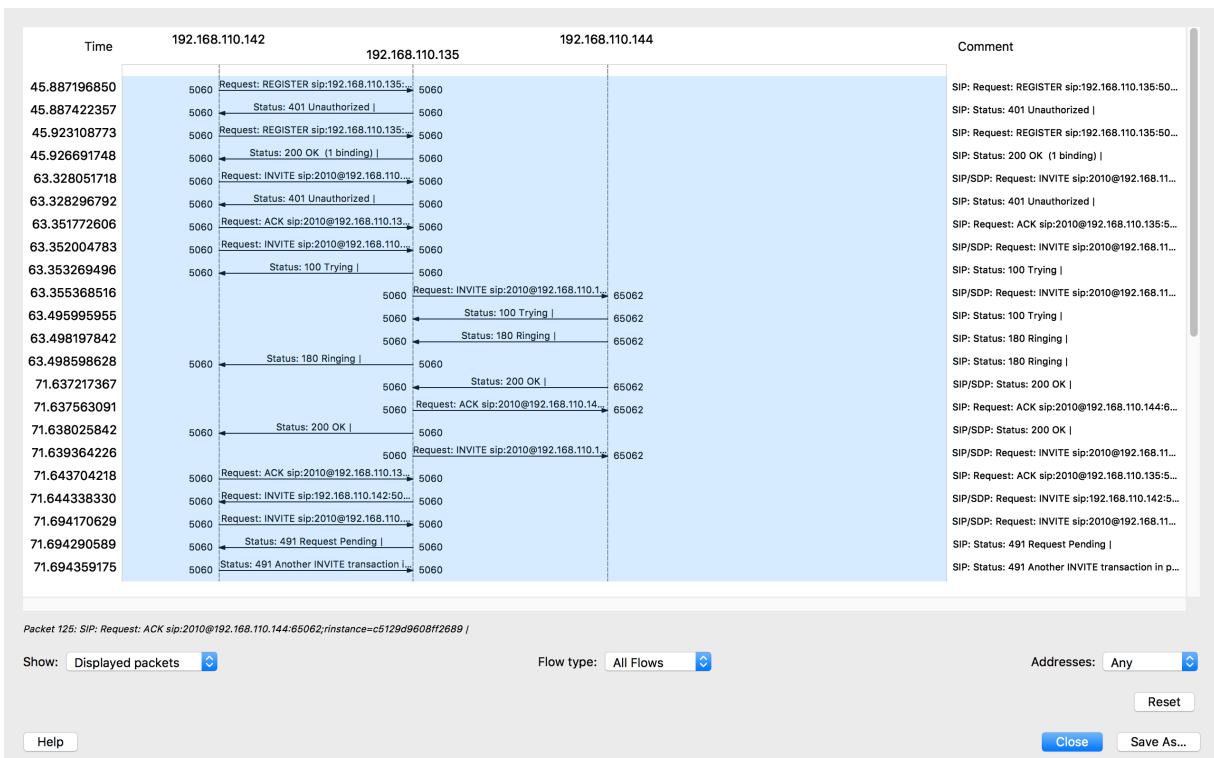


Figure : 2.18

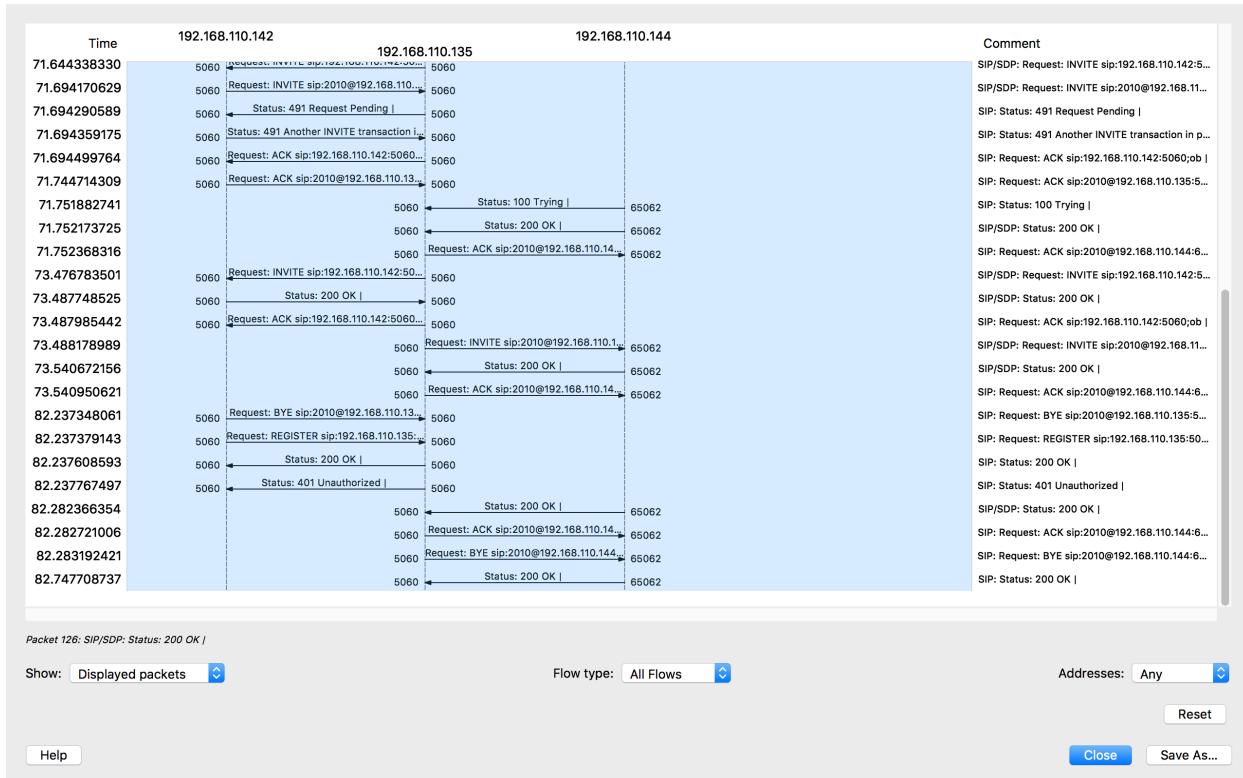


Figure : 2.19

SIP Flows:

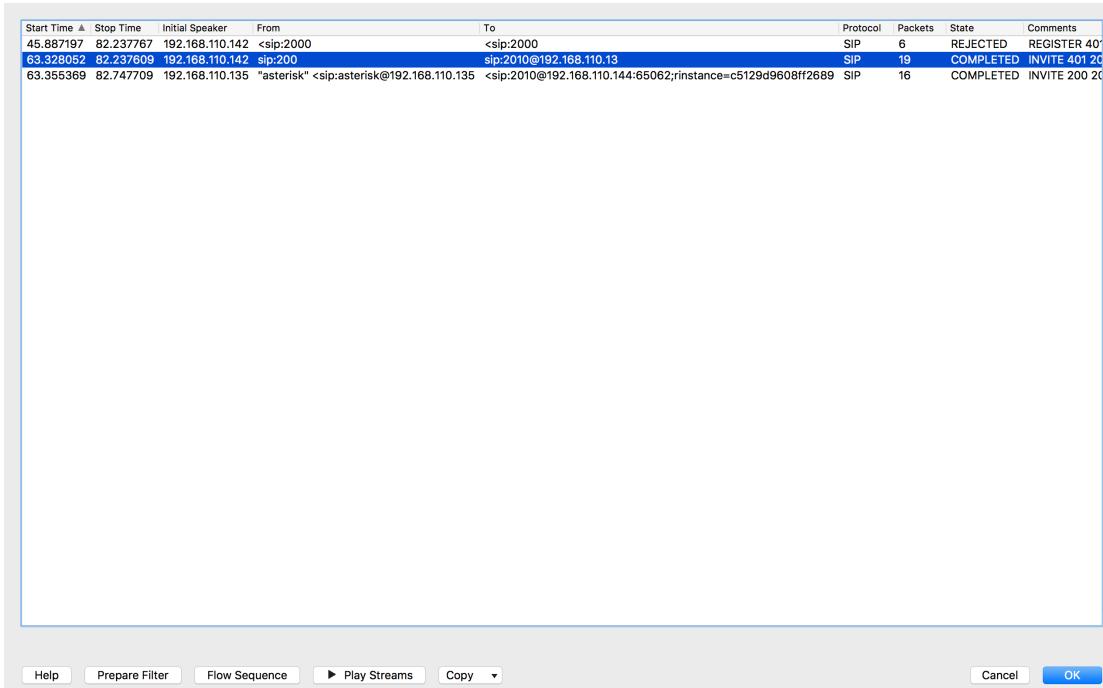


Figure : 2.20

The figure shows registration process of user 2000 via SIP Register and 200 OK

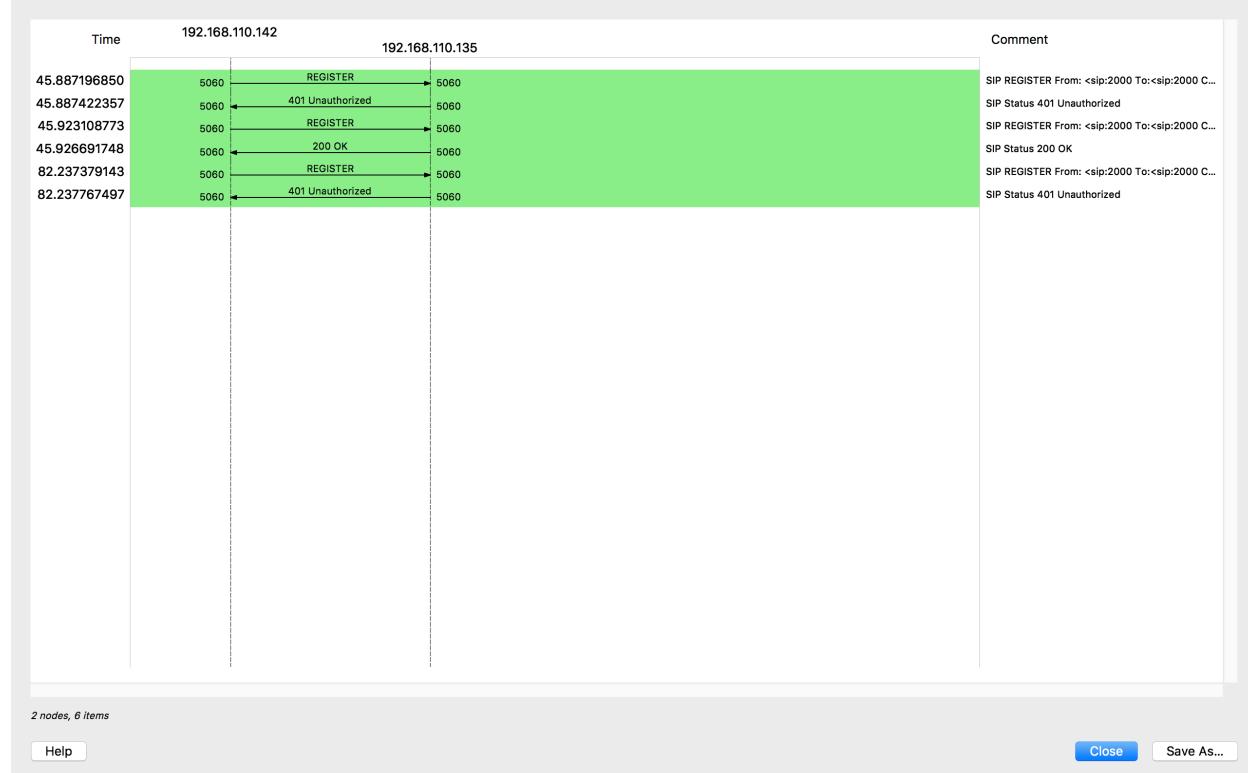


Figure : 2.21

The flow diagram shows INVITE from 2000 to 2010 for a call following the SIP 3 way handshake and establishment of call

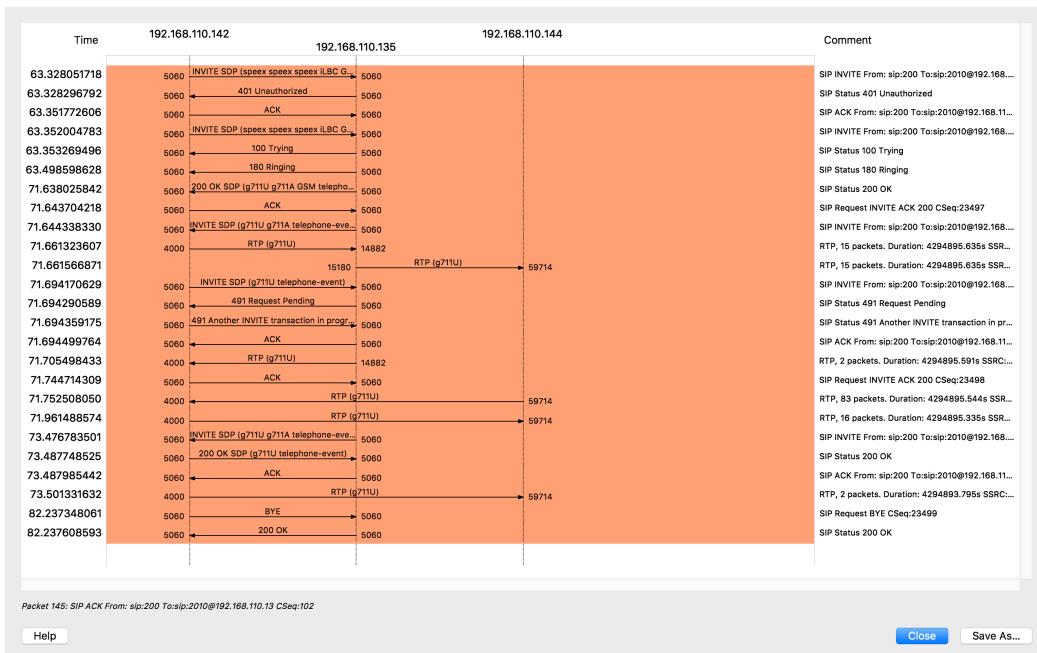


Figure : 2.22

This flow shows communication between asterisk server and SIP client 2010 for establishing a call with 2000 at 2010's side. The call is established at 73.54 seconds with the establishment of an RTP stream.

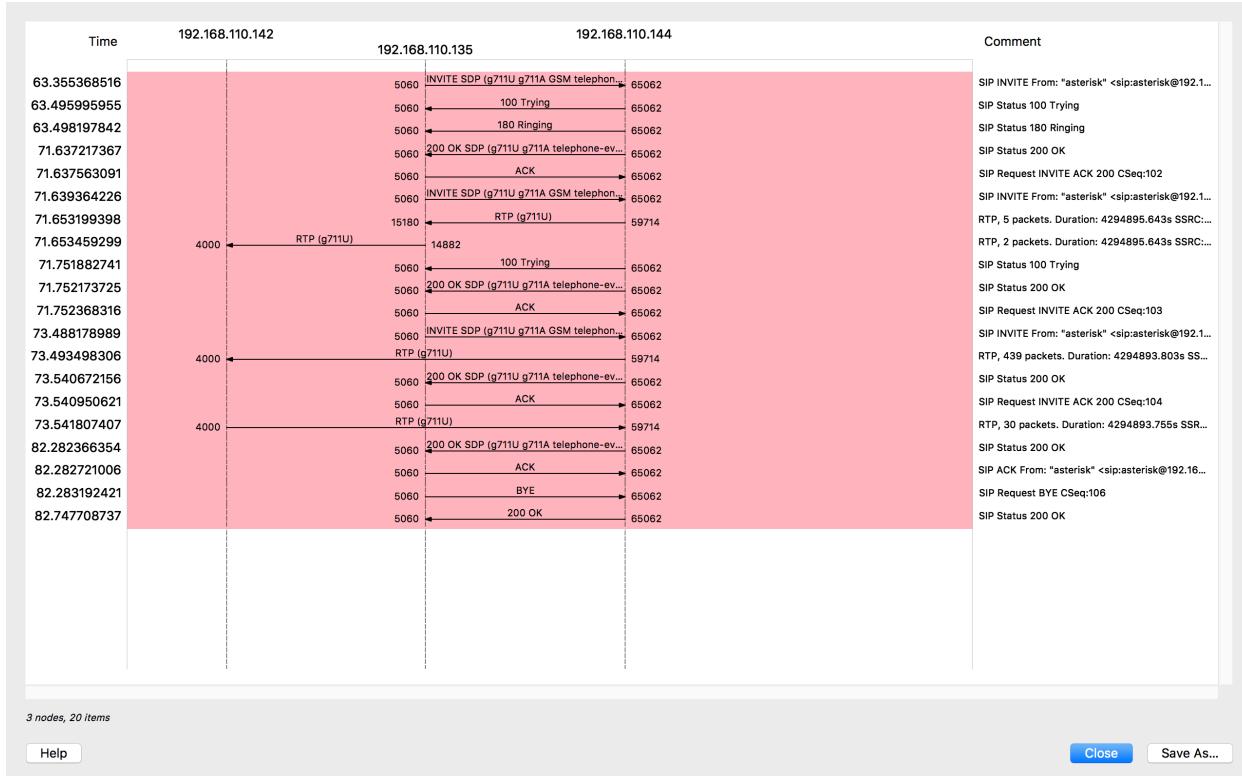
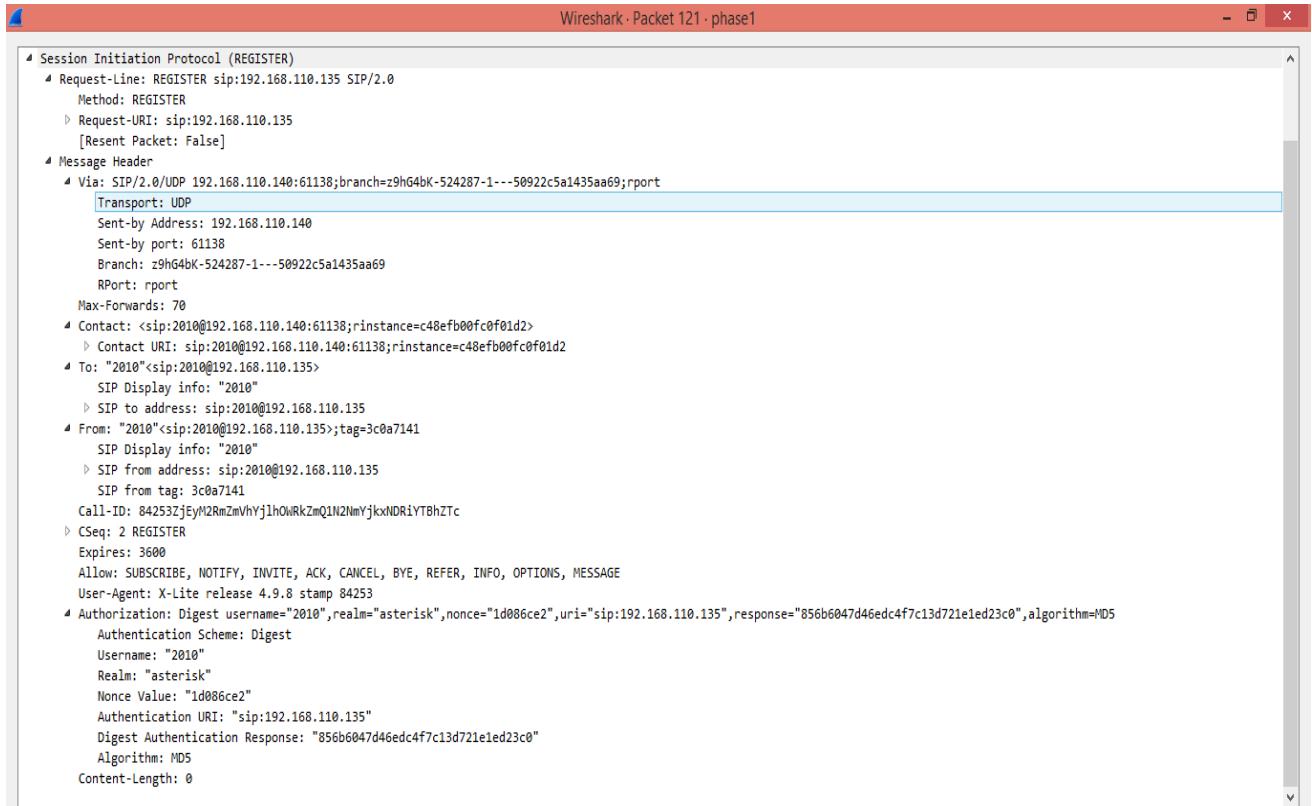


Figure : 2.23

3.SIP Packet Fields:



The screenshot shows a Wireshark window titled "Wireshark · Packet 121 · phase1". The packet details pane displays a SIP REGISTER message. Key fields include:

- Session Initiation Protocol (REGISTER)
- Request-Line: REGISTER sip:192.168.110.135 SIP/2.0
- Method: REGISTER
- Request-URI: sip:192.168.110.135
- [Resent Packet: False]
- Message Header:
 - Via: SIP/2.0/UDP 192.168.110.140:61138;branch=z9hG4bK-524287-1---50922c5a1435aa69;rport
 - Transport: UDP
 - Sent-by Address: 192.168.110.140
 - Sent-by port: 61138
 - Branch: z9hG4bK-524287-1---50922c5a1435aa69
 - RPort: rport
 - Max-Forwards: 70
 - Contact: <sip:2010@192.168.110.140:61138;rinstance=c48efb00fc0f01d2>
 - To: "2010"<sip:2010@192.168.110.135>
 - SIP Display info: "2010"
 - SIP to address: sip:2010@192.168.110.135
 - From: "2010"<sip:2010@192.168.110.135>;tag=3c0a7141
 - SIP Display info: "2010"
 - SIP from address: sip:2010@192.168.110.135
 - SIP from tag: 3c0a7141
 - Call-ID: 842532JeyM2RmZmVhYjh0WkZmQ1N2NmYjkxNDRiYTbhZTc
 - CSeq: 2 REGISTER
 - Expires: 3600
 - Allow: SUBSCRIBE, NOTIFY, INVITE, ACK, CANCEL, BYE, REFER, INFO, OPTIONS, MESSAGE
 - User-Agent: X-Lite release 4.9.8 stamp 84253
- Authorization: Digest username="2010",realm="asterisk",nonce="1d086ce2",uri="sip:192.168.110.135",response="856b6047d46edc4f7c13d721e1ed23c0",algorithm=MD5
 - Authentication Scheme: Digest
 - Username: "2010"
 - Realm: "asterisk"
 - Nonce Value: "1d086ce2"
 - Authentication URI: "sip:192.168.110.135"
 - Digest Authentication Response: "856b6047d46edc4f7c13d721e1ed23c0"
 - Algorithm: MD5
- Content-Length: 0

Figure : 3.1

Request Line

It shows the method of the SIP packet which is REGISTER in this case and gives the destination or the server address on which it wants to register. That is 192.168.110.135 Asterisk server address in our case.

Via

It shows the source IP address and port number of the SIP packet, transport protocol to be used and the version of SIP which is used.

Contact

This field contains the source or sender's SIP URI that can be used to contact the sender.

TO

It contains the address of the recipient of the request. It may or may not contain the address of the final destination. In our case display name 2010 is registering to Asterisk server thus it contains IP address of the asterisk server which is 192.168.110.135.

FROM

It consists of identity of the initiator of the request from the point of view of the server which of the form “user@domain” that is “2010@192.168.110.135” in our case

ALLOW

This field gives the information about the SIP Methods the caller can support which is SUBSCRIBE, BYE, ACK etc. in our case.

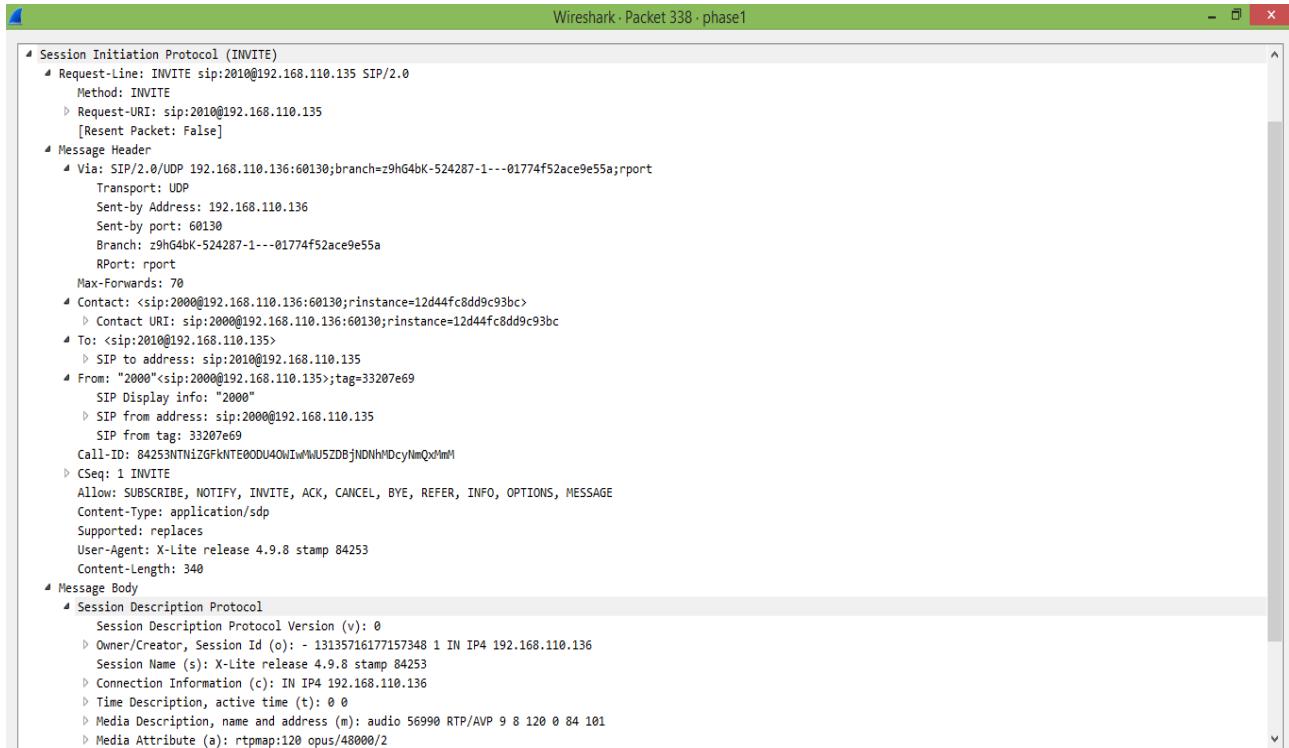


Figure : 3.2

For the above packet all the SIP Packet fields stays the same as it was for SIP Register packet but SIP Method used is INVITE instead of REGISTER.

Here source also shows the user agent it uses under the user agent field which is X-Lite in our case.

4.SIP 3 way handshake:

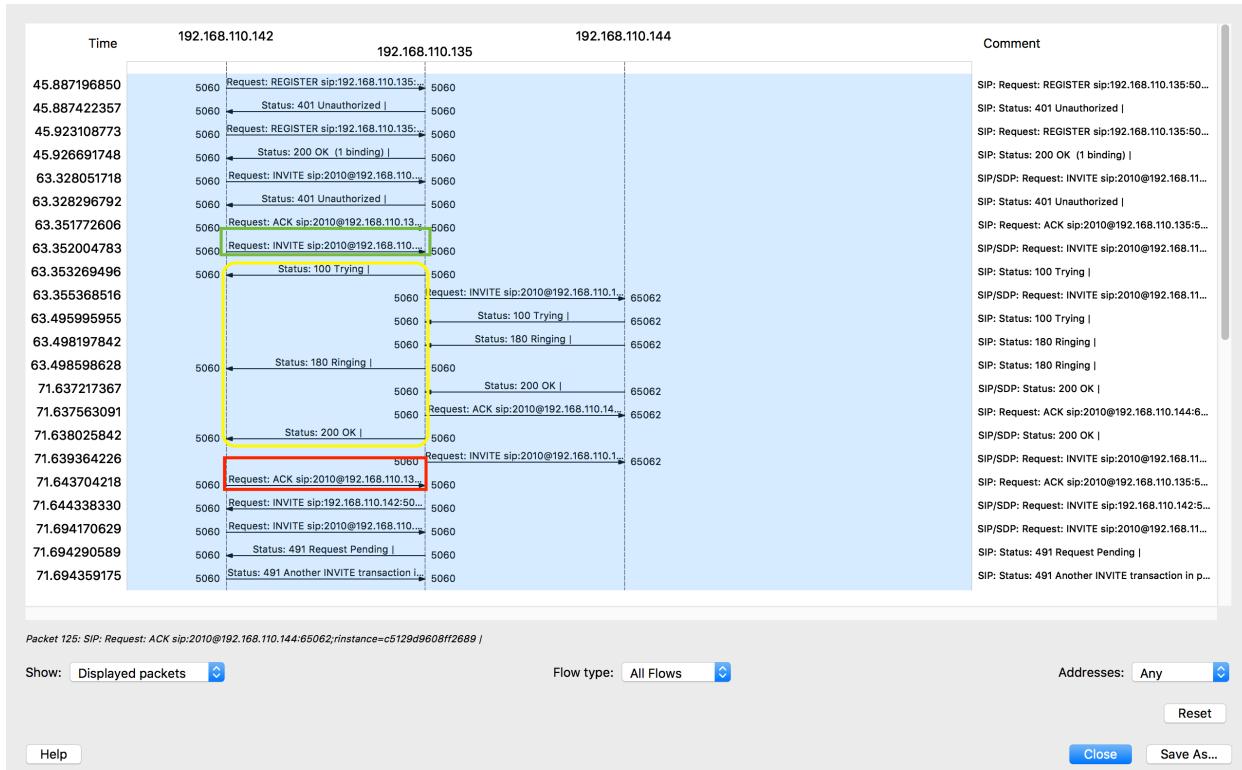


Figure : 4.1

The figure: 4.1 shows a typical SIP 3 way handshake observed during our experiment. The green box initiates the SIP 3 way handshake with the request INVITE to establish the call. The yellow box shows second part of 3 way handshake where the 100 Trying, 180Ringing and eventually 200 OK messages are sent from server to the calling client to which the calling client responds with Request_ACK as shown in the red box.

5.Ad-hoc Networks:

Ad hoc network

- Ad-hoc network is a network of devices that communicate with each other directly using Ethernet or wireless cards.
- Using this network devices can communicate without routers or access points.
- It is usually built spontaneously in Local Area Network (LAN) for a specific application or purpose i.e. it is a temporary network.
- Classification of Ad-hoc networks
 - Mobile ad-hoc networks (MANET)
 - Vehicular ad-hoc networks (VANET)
 - Internet Based mobile ad-hoc networks (iMANETS)
 - Military ad-hoc networks.
- Pros
 - Infrastructureless.
 - Low Cost.
 - Less Setup time.
 - Connects Automatically and adapts changes.