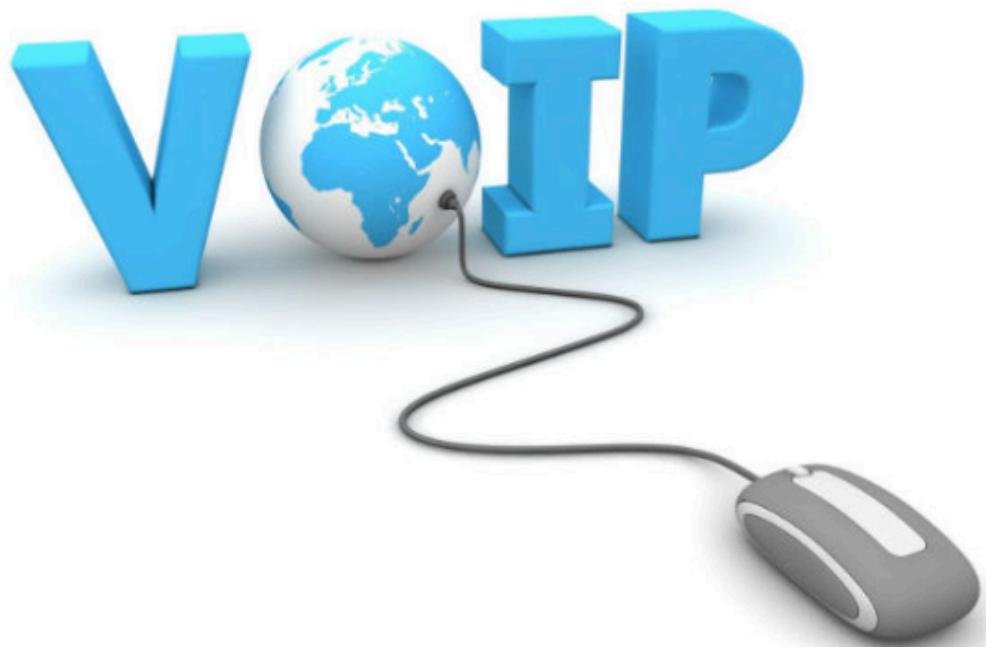


# XAMU LAB 5 - TELEFONÍA IP



Estudiants: Jia Le Chen

Cristina Sanchez-Mora Gassol

Nawal Bouallala Safyoun

Mariona Farré Tapias

Asignatura: XAMU

Cuatrimestre: Primavera

Curso: 2024-2025

Profesor: David Rincon Rivera

# ÍNDEX

Introducció	3
<b>EXERCICIS PREVIS</b>	<b>4</b>
Exercici previ 1	4
Exercici previ 2	8
Exercici previ 3	16
Exercici previ 4	19
Exercici previ 5	21
Exercici previ 6	23
Exercici previ 7	30
Exercici previ 8	36
<b>EXERCICIS LABORATORI</b>	<b>47</b>
Exercici 1	47
Exercici 2	54
Exercici 3	56
Exercici 4	58
Exercici 5	60
Exercici 6	61
Exercici 7	63
Exercici 8	64
Exercici 10	68
Exercici 11	82
Conclusió	87

## **Introducció**

La telefonia IP, o VoIP (Voice over Internet Protocol), és una tecnologia que permet la transmissió de veu i multimèdia a través de xarxes de dades, com Internet, utilitzant el protocol IP. Aquest avanç tecnològic ha revolucionat la manera en què ens comuniquem, oferint una alternativa eficient i econòmica a les tradicionals xarxes telefòniques de commutació de circuits. En el context d'aquesta pràctica de laboratori, explorarem diferents aspectes tècnics de la telefonia IP, incloent la configuració de clients i servidors SIP, l'anàlisi de paquets de dades amb eines com Wireshark, i l'avaluació de la qualitat del servei sota diverses condicions de xarxa.

## **EXERCICIS PREVIS**

### **Exercici previ 1**

**Llegiu tots els estudis previs i exercicis de la pràctica, feu-vos una idea de cada escenari, i planifiqueu amb el vostre grup com els fareu: màquines disponibles (4 al laboratori + portàtils i/o telèfons mòbils), quins sistemes operatius corre cada màquina, quins clients i servidors es podran instal·lar, com es duran a terme els exercicis que requereixen coordinació (com assignareu els comptes SIP, qui trucarà o serà trucat, etc).**

Per als exercicis previs farem:

1. Per al 1, no necessitem cap software, només un dispositiu per a accedir al document i escriure.
2. Per al 2, es pot realitzar amb un ordinador amb els softwares: MiscroSIP, Linphone i Zoiper instal·lat A més d'un dispositiu mòbil. No es requereix més d'una persona.
3. Per al 3, és necessari només un dispositiu amb accés a internet per a accedir a la web  
[https://www.myvoipapp.com/docs/faq/setup\\_ippbx\\_for\\_small\\_business\\_step\\_by\\_step/index.html](https://www.myvoipapp.com/docs/faq/setup_ippbx_for_small_business_step_by_step/index.html). No es requereix més d'una persona.
4. Per al 4 és necessari un dispositiu amb accés a internet, recomanable un ordinador, per a fer crear un compte SIP. També és necessari un dispositiu mòbil per a fer una trucada al número del compte SIP, per tal de fer les proves adhients. Com que tots tenim un portàtil o ordinador i un telèfon mòbil, no es requereix més d'una persona per a realitzar l'exercici.
5. L'exercici previ 5 només necessita accés a internet.
6. Per al 6, utilitzem un portàtil linux amb l'eina dig. No es requereix més d'una persona.
7. Per al 7, necessitem un ordinador amb wireshark. No es requereix més d'una persona.
8. Per al 8, es requereixen 2 terminals, un Linphone i un MicroSIP i al menys un d'ells a casa, per tal de passar per un NAT. És millor realitzar l'exercici amb dos persones. Un ordinador amb el terminal MicroSIP i un altre amb el terminal Linphone. Els dos ordinadors han de tenir Wireshark i han de filtrar per protocols SIP y DNS.

Per als exercicis del laboratori:

Tenim 4 màquines al laboratori + 1 portàtil per alumne + 2 telèfons Android + 2 telèfons IPhone + 3 adreces UPC

Els 4 ordinadors del laboratori tenen Windows i Linux.

Tenim 1 portàtil amb Windows i 1 portàtil al Linux i 2 portàtils amb Linux i Windows.

Asignarem 4 rols, un per alumne, per a facilitar el treball al laboratori:

- Alumne 1: Nawal Bouallala Safyoun
  - Portàtil Windows
    - Software instal·lat: Wireshark, MicroSIP, Linphone (per si de cas)
    - Ordinador laboratori ¿Windows/Linux?
    - Dispositiu movil Android
- Alumne 2: Jia Le Chen
  - Portàtil Linux
    - Software instal·lat: Wireshark, Linphone, MyVoIPapp MiniSIPserver
    - Ordinador laboratori ¿Windows/Linux?
    - Dispositiu movil IPhone
- Alumne 3: Mariona Farré Tapias
  - Portàtil Linux
    - Software instal·lat: dig, Wireshark, Linphone, netem, Zoiper
    - Ordinador laboratori ¿Windows/Linux?
    - Dispositiu movil IPhone
- Alumne 4: Cristina Sánchez-Mora Gassol
  - Portàtil Windows
    - Software instal·lat: MicroSIP, Linphone, Zoiper, Wireshark
    - Ordinador laboratori ¿Windows/Linux?
    - Dispositiu movil Android

Llegenda colors:

Linux   Windows   Parella 1   Parella 2   Tots

1. Per al exercici 1, ens dividim en dues parelles. Cada parella Linux-Windows obrirà Linphone-MicroSIP. Capturant amb Wireshark a les dues màquines, és realitzarà una trucada d'uns segons. La parella 1 farà de Linphone a MicroSIP, MicroSIP penja. La parella 2 farà de MicroSIP a Linphone, Linphone penja. S'ha d'apuntar l'IP de cada terminal.

2. Per al exercici 2, amb el mateix escenari d'abans. Hem d'utilitzar almenys dos còdecs i un rellotge diferents. (Dos captures mínim).
3. Per a l'exercici 3, aprofitem les captures dels exercicis 1 i 2.
4. Per a l'exercici 4, utilitzem de nou el mateix escenari (Capturar en wireshark cada apartat). Però fem alguns canvis a l'hora de fer les trucades.
  - 4.1. Linphone truca a MicroSIP; MicroSIP refusa. MicroSIP truca a Linphone; Linphone refusa.
  - 4.2. Linphone truca a MicroSIP; MicroSIP no despenja i expira el temporitzador. MicroSIP truca a Linphone; Linphone no despenja i expira el temporitzador.
  - 4.3. Linphone truca a MicroSIP, deixem sonar; Linphone cancel·la. MicroSIP truca a Linphone, deixem sonar; MicroSIP cancel·la.
5. Per a l'exercici 5, per a cada parella configurar els dos terminals per a que acceptin còdecs d'àudio diferents entre ells. Repetir escenari exercici 1.
6. En aquest exercici és important utilitzar els nostres portàtils per a poder utilitzar la webcam. Connectem les webcams als terminals. Configurem els còdecs d'àudio i vídeo per a que siguin compatibles (entre els terminals de la parella). Capturem amb Wireshark. Fem la trucada del exercici 1.
7. Per a aquest exercici, ja no treballem en parelles, si no en grup. Necessitem tres terminals. Dos terminals han de tenir configurat el mateix usuari SIP (usuari A), l'altre terminal ha de tenir un altre usuari SIP different (usuari B).
  - 7.1. Usuari B truca a usuari A.
  - 7.2. Capturar en wireshark desde els tres terminals. Usuari B truca a usuari A, un d'ells despenja.
8. Un terminal ha de corre sense clients SIP, ha de arrencar el servidor MyVoIPapp MiniSIPserver. Els altres tres terminals han de configurar-se per a registrar cap al servidor en comptes del proxy. Els terminals amb clients es faràn trucades, capturant en wireshark.
  - 8.1. Terminal 1 truca terminal 2.
  - 8.2. Terminal 2 truca terminal 3.
  - 8.3. Terminal 3 truca terminal 1.
  - 8.4. Terminal 1 truca terminal 3.
  - 8.5. Terminal 3 truca terminal 2.
  - 8.6. Terminal 2 truca terminal 1.
9. Matrix escenari que l'exercici anterior. El servidor ha de configurar la línia exterior amb el compte de GoTrunk del exercici previ 4.
  - 9.1. Terminal 1 truca al mòvil d'un de nostaltres.

- 9.2. Configurar servidor per a reencaminar trucades del exterior.  
Capturar el servidor amb Wireshark. Trucar d'un dels nostres mòvils al terminal 1.
  - 9.3. Anar al web de GoTrunk, descarregar el PCAP.
10. Desde un terminal Linux, capturem amb wireshark, iniciem un trucada amb Linphone, amb netem introduïm problemes de QoS un a un.
  - 10.1. Introduïm pèrdues entre el 1% i el 20%:
  - 10.2. Introduïm retard absolut de 0.5s:
  - 10.3. Introduïm retard absolut d'1s:
  - 10.4. Introduïm jitter 100ms +/- 50ms:
  - 10.5. Buscar límit de correcció de jitter de Linphone.
  - 10.6. Cambiar el client a Zoiper i buscar el seu límit.
  - 10.7. Introduïm pèrdues 50%:
11. Amb l'escenari del exercici 8: 1 terminal servidor. 2 terminals clients.
  - 11.1. Capturar en wireshark en terminal client. Configurem el client en windows de MicroSIP, amb STUN apuntat al servidor STUN del nostre servidor.
  - 11.2. Capturar en wireshark. Configurar el client per a apuntar al servidor STUN públic. Desconfigurar el STUN del client.
  - 11.3. Configurar dos clients amb TURN (Linphone), capturar amb wireshark un d'ells.
  - 11.4. Capturar en wireshark. Trucar desde un client TURN a l'altre.

## **Exercici previ 2**

**Idealment cada grup hauria de disposar, entre tots els membres, d'una instal·lació de MicroSIP, una de Linphone, una de Zoiper, i un client Android o iPhone. Per a cadascun d'ells, analitzeu els menús de configuració en aquests dos aspectes:**

**a) Còdecs (àudio i vídeo): de quins disposa? Quins paràmetres addicionals tenen? (detecció de silenci, comfort noise, etc). Algun paràmetre addicional que us cridi l'atenció?**

MicroSIP:

- Còdecs d'àudio: Opus 24 kHz, G. 711 A-law, G. 711 u-law, G. 722 16 kHz, G. 722.1 16 kHz, G. 722.1 32 kHz, G.723 8 kHz, G.729 8 kHz, GSM 8 kHz, AMR 8 kHz, AMR-WB 16 kHz, iLBC 8 kHz, Speex 32 kHz, Speex 16 kHz, Speex 8 kHz, SILK 16 kHz, SILK 8 kHz, LPCM 8 kHz, LPCM 8 kHz Stereo, LPCM 16 kHz, LPCM 16 kHz Stereo, LPCM 44 kHz, LPCM 44 kHz Stereo, LPCM 48 kHz, LPCM 48 kHz Stereo
- Còdecs de vídeo: H264/99, H263-1998/98, VP8/100, VP9/101
- Parámetros adicionales: Voice Activity Detection (VAD), Echo Cancellation (EC), definir el códec de la trucada quan no ets l'emisor (Forzar codec en llamadas entrantes)

Linphone:

- Còdecs d'àudio:

Nombre	Descripción	Tasa (Hz)	Tasa de bits (Kbit...)	Parámetros	Estatus
opus	An opus encoder.	48000	50	+ useinbandfec=1	<input checked="" type="checkbox"/>
speex	The free and wonderful speex codec	16000	40	+ vbr=on	<input checked="" type="checkbox"/>
speex	The free and wonderful speex codec	8000	32	+ vbr=on	<input checked="" type="checkbox"/>
PCMU	ITU-G.711 ulaw encoder	8000	80		<input checked="" type="checkbox"/>
PCMA	ITU-G.711 alaw encoder	8000	80		<input checked="" type="checkbox"/>
GSM	The GSM full-rate codec	8000	30		<input type="checkbox"/>
G722	The G.722 wideband codec	8000	80		<input type="checkbox"/>
iLBC	WebRtc's iLBC encoder	8000	24	+ mode=30	<input type="checkbox"/>
G729	G729 audio encoder filter	8000	24	+ annexb=yes	<input checked="" type="checkbox"/>
speex	The free and wonderful speex codec	32000	40	+ vbr=on	<input type="checkbox"/>
BV16	The BV16 full-rate codec	8000	32		<input type="checkbox"/>
L16	L16 dummy encoder	44100	1428		<input type="checkbox"/>
L16	L16 dummy encoder	44100	722		<input type="checkbox"/>

**OK**

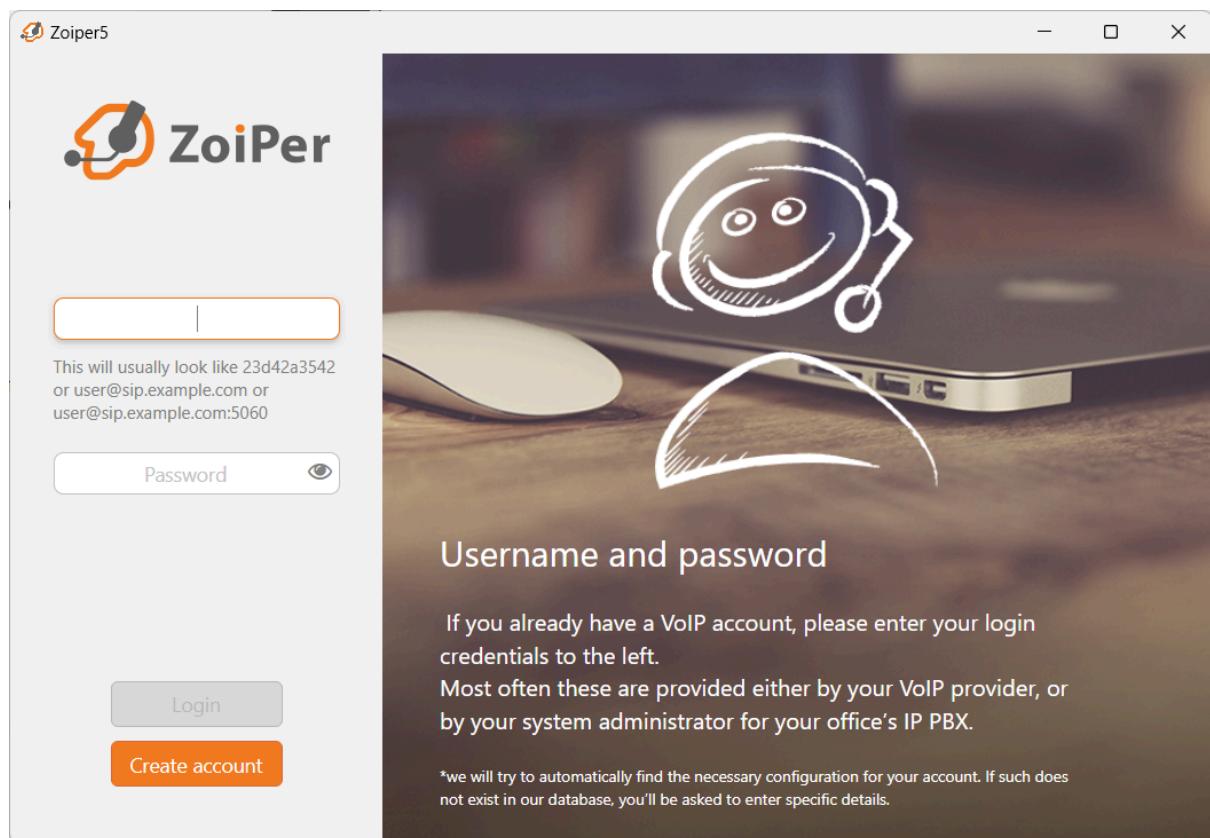
- Còdecs de vídeo:

Nombre	Descripción	Tasa (Hz)	Tasa de bits (Kbit...)	Parámetros	Estatus
AV1	An AV1 encoder based on aom.	90000	1500		<input checked="" type="checkbox"/>
VP8	A VP8 video encoder using libvpx library.	90000	1500		<input checked="" type="checkbox"/>
H264	Provided by CISCO SYSTEM,INC				<input type="checkbox"/>

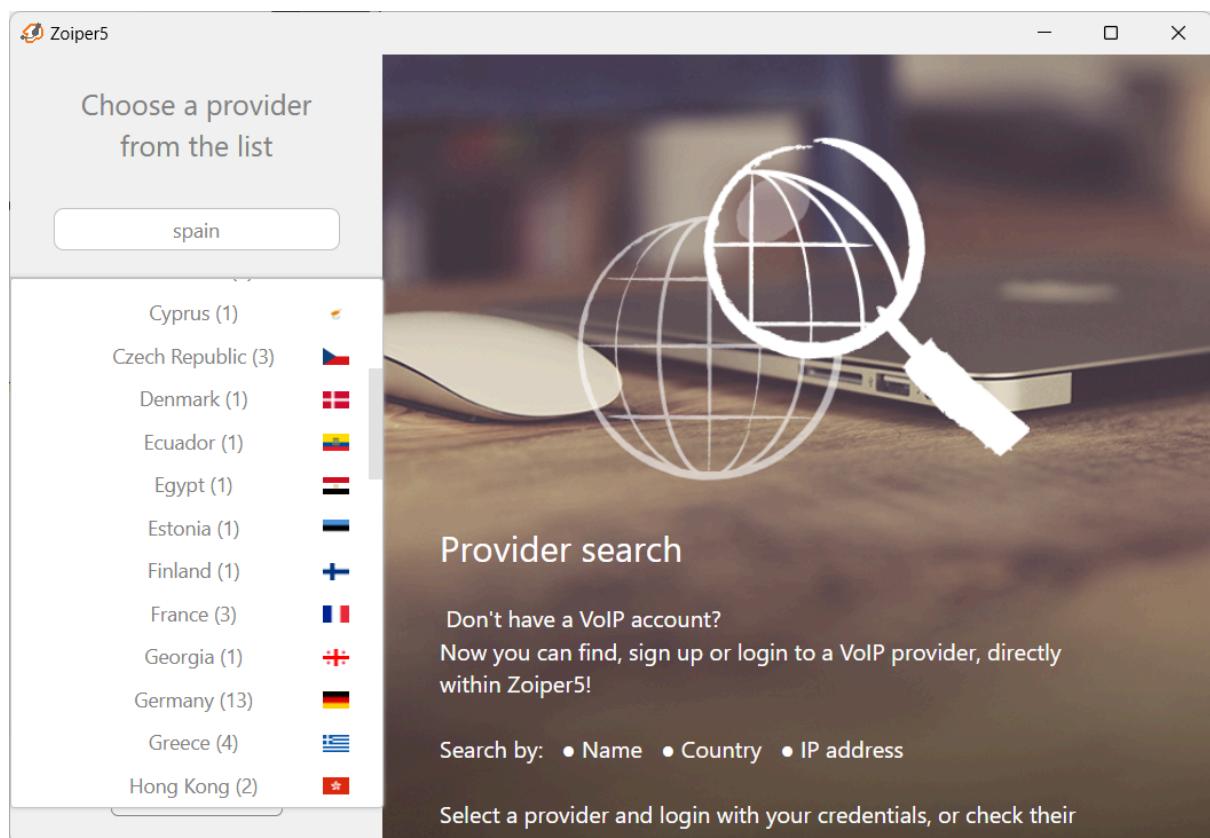
- Paràmetres adicionals: VBR, EC, Control de congestió i jitter amb búffer adaptatiu.

Zoiper:

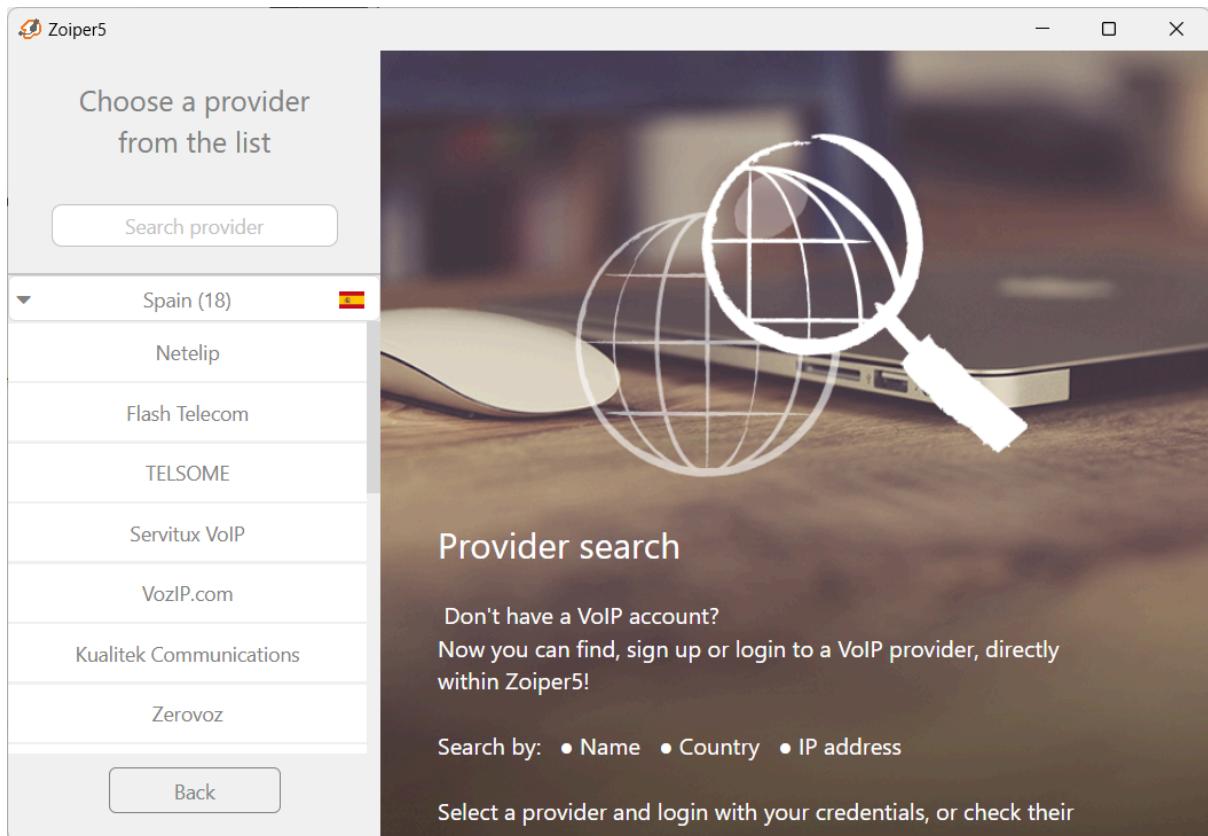
Per a utilitzar Zoiper ens requereix crear un compte:



Hem de sel·lecccionar un proveïdor:



En sel·leccionar Spain, ens mostra un llistat de proveïdors:



Però no sabem exactament quin utilitzar. Per tant treurem l'informació d'internet i preguntarem al laboratori.

Aquests són els códecs d'àudio i vídeo:

#### Supported audio codecs

Zoiper 5 supports the following audio codecs:

- *GSM*
- *ITU G.711 (PCMU/PCMA)* Comes in two flavors: *a-law and mu-law*.
- *iLBC20*
- *iLBC30*
- *ITU G.729 \**
- *Speex (8000/16000/32000) \**
- *G.726 \**
- *Opus (8000/16000/24000/48000) \**

#### Supported video codecs: \*

Zoiper 5 supports the following video codecs:

- *H264*
- *VP8*

Paràmetres adicionals: no ho sabem, no apareix a la web.

**b) Protocols: comenteu com a mínim aquests aspectes, i relacioneu-los amb el que heu vist a teoria.**

**a. Com es fa la configuració del compte SIP? Quins paràmetres necessita?**

**b. Quins paràmetres relacionats amb SIP es poden escollir?**

(Exemples: tons DTMF, opcions de registre, temporitzadors, protocol de transport... )

**c. Quins paràmetres relacionats amb RTP i RTCP es poden escollir?**

**d. Quins paràmetres relacionats amb el salt de NATs es poden escollir? (Exemples: rport, STUN, TURN, ICE...)**

MicroSIP:

Podem afegir un compte SIP, desplegant el menú de la dreta > Añadir cuenta... Aquest és el menú per afegir un compte, amb els paràmetres que hem d'afegir:

Cuenta

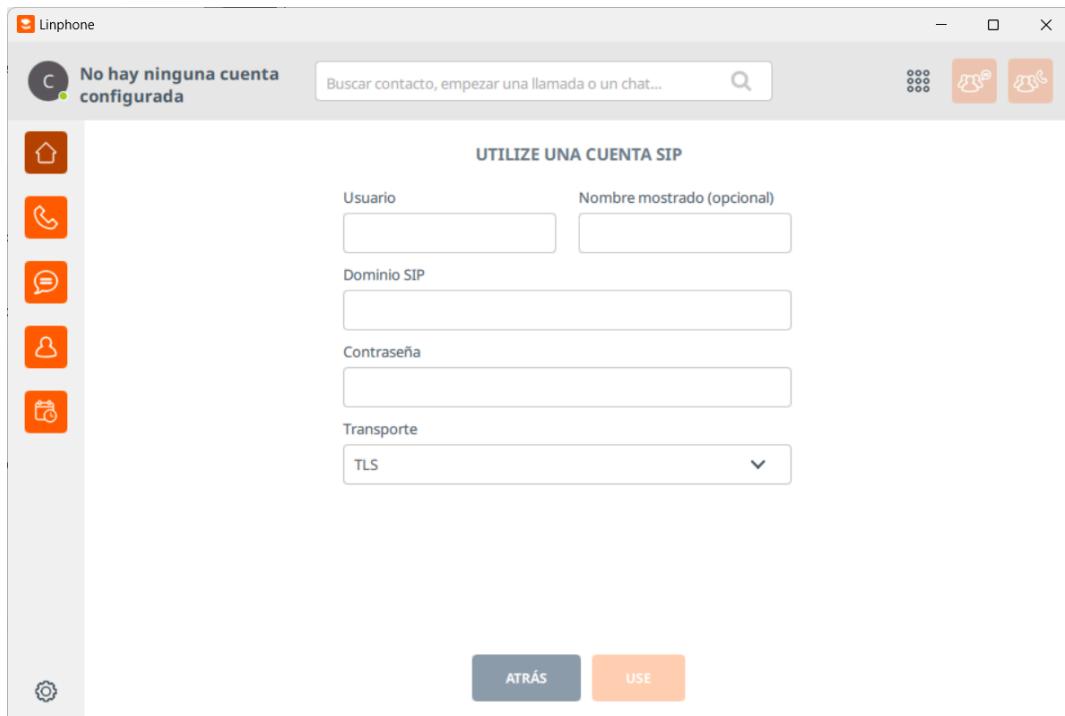
Nombre de cuenta	<input type="text"/>
Servidor SIP	<input type="text"/>
Proxy SIP	<input type="text"/>
Usuario *	<input type="text"/>
Dominio *	<input type="text"/>
Iniciar sesión	<input type="text"/>
Contraseña	<input type="password"/> <a href="#">Mostrar contraseña</a>
Nombre para mostrar	<input type="text"/>
Núm. buzón de voz	<input type="text"/>
Prefijo de Marcación	<input type="text"/>
Plan de marcado	<input type="text"/>
<input type="checkbox"/> Hide Caller ID	<input type="text"/>
Comunicación cifrada	<input type="button" value="Desactivado"/>
Transporte	<input type="button" value="UDP"/>
Dirección pública	<input type="button" value="Automático"/>
Refresco de Registro	<input type="button" value="300"/> Mantener Conexión <input type="button" value="15"/>
<input type="checkbox"/> Publicar presencia	<input type="text"/>
<input type="checkbox"/> Permitir reescritura IP	<input type="text"/>
<input type="checkbox"/> ICE	<input type="text"/>
<input type="checkbox"/> Desactivar temporiz. de sesión	<input type="text"/>
<input type="button" value="Guardar"/> <input type="button" value="Cancelar"/>	

A més a més, hi ha altres paràmetres relacionats amb SIP que podem marcar, com el protocol de transport, el xifrat en la comunicació, la direcció

pública, el refreshment de registre, el temps per a mantenir una connexió, si volem publicar presència, permetre reescritura IP o desactivar la temporització de la sessió. També ens permet configurar els protocols SIP i STUN/ICE.

Linphone:

Per a configurar un compte, això es el que ens demana:



Però podem afegir un compte nou desde el menú de preferencias > Cuentas SIP > Añadir cuenta. Aquí trobem molts més paràmetres:

## Ajustes principales de cuenta SIP

Dirección SIP*	sip:@sip.linphone.org
Dirección del servidor SIP*	<sip:sip.linphone.org;transport=tls>
Duración del registro (seg)	600
Transporte	TLS
Ruta	
Conference URI	sip:conference-factory@sip.linphone.org
Video Conference URI	sip:videoconference-factory@sip.linphone.org
E2E encryption keys server URL	https://lime.linphone.org/lime-server/lime-server.php
Parámetros de contacto	message-expire=2419200
Intervalo RTP regular de AVPF ...	1
Registrar	<input checked="" type="checkbox"/>
Publicar información de presen...	<input checked="" type="checkbox"/>
	Publish duration (sec) <input type="text" value="120"/>
Habilitar AVPF	<input checked="" type="checkbox"/>
Prefix for your country	<input type="text"/> <input type="button" value="+"/> <input type="button" value="-"/>
	Replace '+' by '00' <input type="checkbox"/>
Apply prefix for outgoing calls and chats	<input checked="" type="checkbox"/>

## NAT y cortafuegos

Activar ICE	<input checked="" type="checkbox"/>
Activar TURN	<input type="checkbox"/>
Servidor STUN/TURN	stun.linphone.org
Usuario TURN	
Contraseña TURN	

## Advanced

Bundle mode

D'aquestes opcions, relacionats amb NATs tenim l'opció per activar IVE, TURN, STUN/TURN, Port NAT, Direcció IP privada.

Referents a RTP/RTPC, tenim els paràmetres del interval RTP, Interval RTCP regular d'AVPF i opció d'habilita AVPF.

Zoiper:

Al no tenir accés a la intereficie de l'aplicació, no podem veure els paràmetres exactes.

**c) Compareu els clients entre sí, i avaleu quin sembla més complet.**

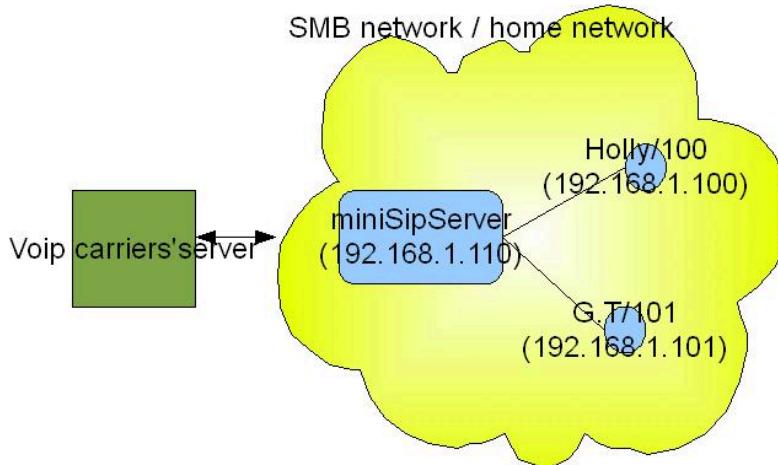
De Zoiper no podem donar una crítica real, ja que no hem pogut veure tots els seus paràmetres. Però a nivell de còdecs sembla que MicroSIP i Linphone donen més opcions. A més a més, sembla que Linuphone és molt més configurable que MicroSIP. No només aixó, si no que Linuphone té l'opció de configurar TURN mentre que MicroSIP no.

## Exercici previ 3

Durant la pràctica instal·lareu una instància del servidor MyVoIPapp MiniSIPserver al laboratori, que farà de centraleta, pel que heu d'aprendre a configurar-lo.

a) Llegiu els punts 3 (Scenario) i 4 (Configuration) de la guia ràpida de configuració de MiniSIPserver.

- Punt 3 (Scenario): És un entorn simple per petites empreses o negocis basats a la llar:



- Punt 4 (Configuration): És una explicació pas a pas per configurar MiniSIPserver.

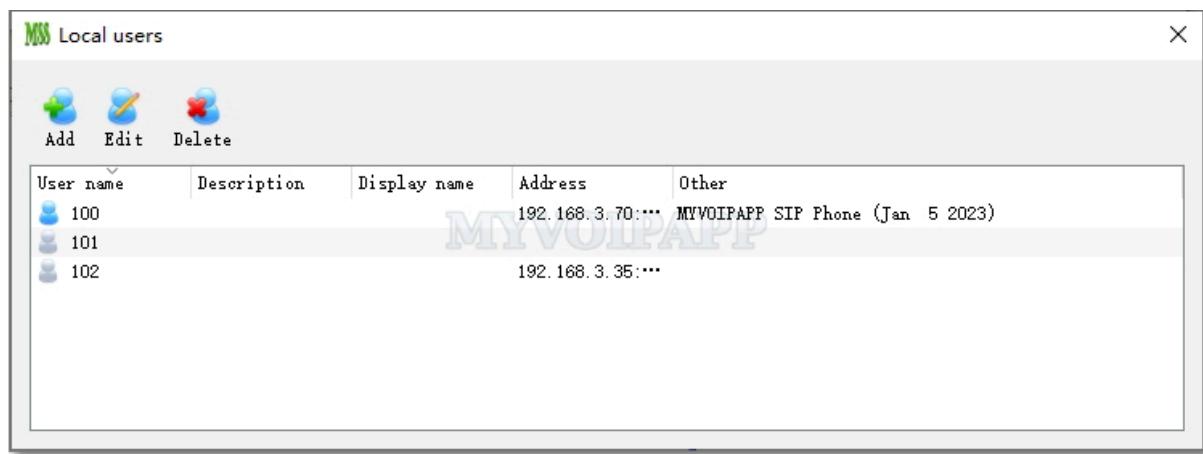
b) Analitzeu els menús de configuració del MiniSIPserver. Com es poden crear comptes SIP? Fixeu-vos que ja en té tres configurats per defecte.

Així es veu quan s'inicia el MiniSIPserver.

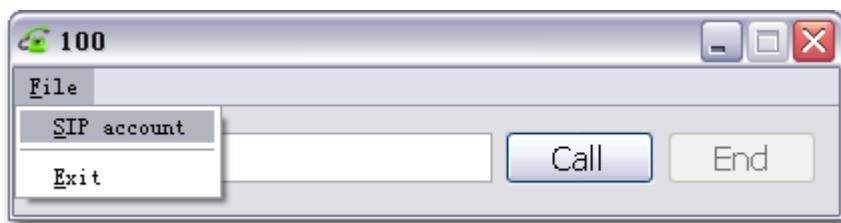
The screenshot shows the main window of the miniSIPserver V40 application. The title bar reads "miniSIPserver V40 (20 clients) build 20231128". The menu bar includes File, Data, Dial Plan, Services, Maintain, Window, and Help. Below the menu is a toolbar with icons for System, Local users, External lines, and Analyze called number. The main area displays a log of server startup messages:

```
2023-11-29 12:23:23 | Local STUN server is '192.168.3.70'.
2023-11-29 12:23:23 | STUN server is ready.
2023-11-29 12:23:23 | SIP server information:
2023-11-29 12:23:23 |   main address (ipv4) is '192.168.3.70'.
2023-11-29 12:23:23 |   main address (ipv6) is 'fe80::2c9f:b126:5ff8:9c2a'.
2023-11-29 12:23:23 |   UDP port is 5060.
2023-11-29 12:23:23 |   TCP port is 5060.
2023-11-29 12:23:23 |   TLS port is 5061.
2023-11-29 12:23:23 | HTTP server is running at port 8080.
2023-11-29 12:23:23 | All data are stored in 'C:\Users\Admin\AppData\Roaming\minisipserver\'.
2023-11-29 12:23:23 | This version is 'V40 (20 clients) build 20231128, win'.
2023-11-29 12:23:23 | SIP server is ready now.
```

Si li cliquem a Local Users, sortirà aquesta finestra el qual ens deixa crear comptes SIP, ja venen per defecte 3 usuaris (100,101,102) amb contrasenyes per defecte (100,101,102).



Per configurar un usuari només hem de fer clic a l'usuari que vulguem editar, es pot editar l'adreça del servidor SIP, el port, el username i la contrasenya.

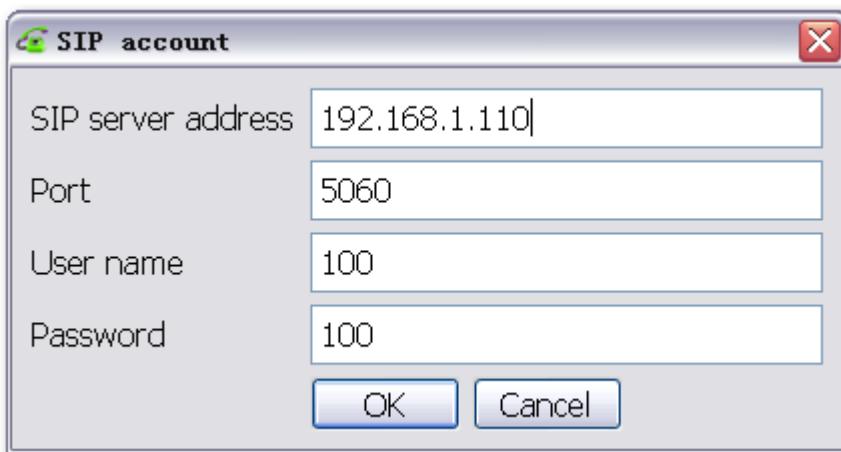


**SIP server address:** Aquesta és l'adreça IP del servidor SIP al qual el client es connectarà. En aquest cas, l'adreça és 192.168.1.110, la qual cosa indica que el servidor està en una xarxa local privada.

**Port:** Aquest és el port de xarxa a través del qual es realitzarà la connexió al servidor SIP. El port 5060 és el port estàndard per al trànsit SIP no xifrat.

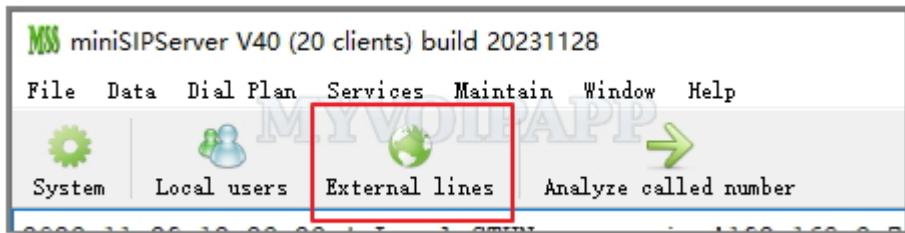
**User name:** Aquest és el nom d'usuari o identificació del client SIP. En aquest cas, el nom d'usuari és 100.

**Password:** Aquest és el camp on s'introduceix la contrasenya corresponent al nom d'usuari per autenticar-se en el servidor SIP.

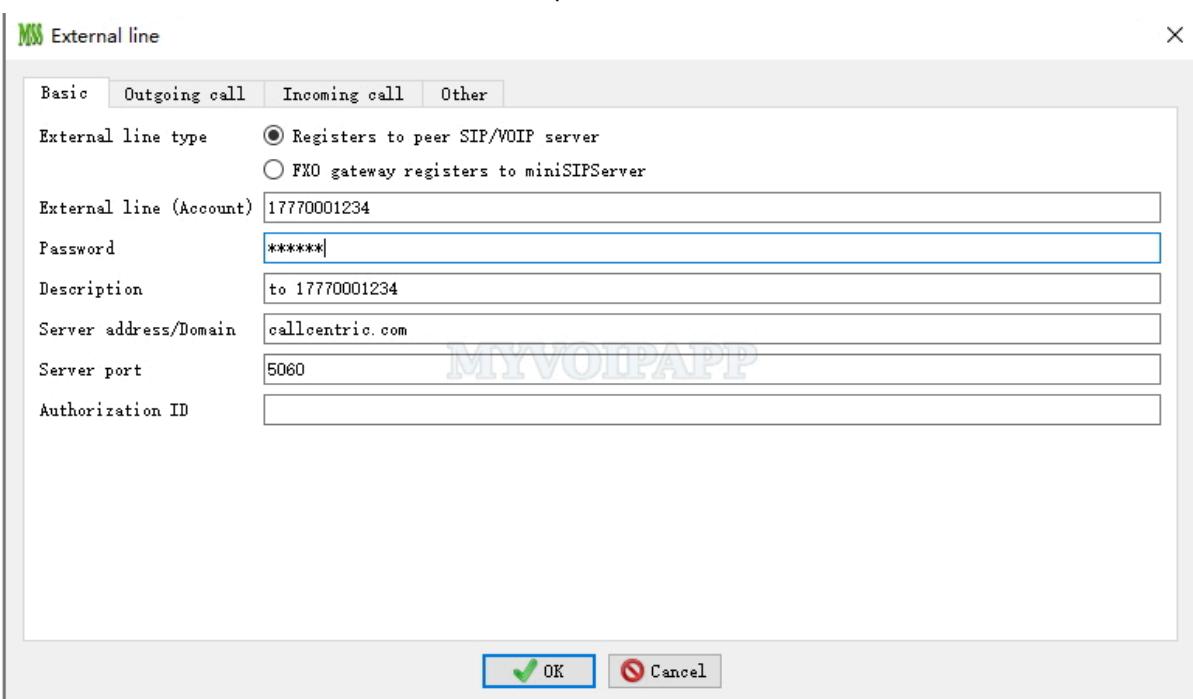


**c) Analitzeu la configuració de les línies externes (SIP Trunk). Quines dades necessitareu?**

Per conectar el miniSIPServer al proveïdor de VOIP li donem clic a External Links.



A la finestra emergent, es fa clic al botó "Afegeix" per afegir una línia externa amb la informació del compte de CallCentric.



En aquest moment ja està configurat per fer i rebre trucades.

## Exercici previ 4

Visiteu la web del proveïdor de SIP trunking <https://gotrunk.es/> i aconseguiu, per a cada grup, un compte SIP trunk gratuïts de 30 dies identificant-vos amb el vostre correu UPC. Si us demana informació d'adreça, poseu l'adreça de l'Escola (Av. Víctor Balaguer,1, 08800 Vilanova i la Geltrú) i deixeu la informació fiscal sense omplir. Activeu el compte i aconseguiu els paràmetres de configuració del SIP trunk (SIP endpoint). Obtindreu una configuració similar a la de la figura 9, amb un proxy (escolliu l'europeu), un usuari SIP i un password. Un cop fet això, assigneu-li al vostre compte SIP Trunk un número de telèfon espanyol d'àmbit geogràfic (prefixos +3491 -Madrid- o +3493 -Barcelona -, NO un 900, 902 o +3451, que són no geogràfics). Proveu podeu configurar un dels terminals SIP amb les dades del compte SIP trunk, i per tant tindrà associat el número de telèfon +349x.... Proveu a fer una trucada des del vostre telèfon mòbil cap al terminal SIP marcant el número +349x (no despenseu si no voleu que us cobrin, tot i que no serà més que alguns cèntims) i a l'inrevés, en aquest cas amb prefix 0034xxx (i la trucada serà gratuïta, ja que GoTrunk us dóna un crèdit gratuït de 4 euros).

Anem a la pagina web <https://gotrunk.es/> i ens registrem amb el compte d'usuari de la universitat. S'ha de confirmar el registre mitjançant un correu electrònic.

Email de trabajo  
jiale.chen@estudiantat.upc.edu

Este valor no debe estar en blanco.

Contraseña  
.....

Este valor no debe estar en blanco.

Acepto los Términos y condiciones  
Es necesario aceptar los términos y condiciones.

REGISTRO

Es configura el perfil per activar el compte.

Es configura una terminació SIP.

## Terminaciones SIP (1)



+ AÑADIR

Nombre ↑	Tipo	Dirección IP	Localización	Autenticación	POP
XAMU	Generic SIP ...	IP dinámica	Localización ...	Credenciales...	Europe

## **Exercici previ 5**

**Referent als servidors que ens ajuden amb el problema del NAT (STUN i TURN),**

**a) Busqueu si els clients SIP us suggereixen algun servidor STUN.**

MicroSIP ens suggereix el servidor STUN “stun.l.google.com:19302”

Linphone ens suggereix el servidor STUN “stun.linphone.org”

Zoiper ens suggereix el servidor STUN “stun.zoiper.com”

**b) Busqueu servidors STUN públics (suggeriment: Google en té, pels seus serveis WebRTC).**

Servidor STUN de Google:

- stun.l.google.com:19302
- stun1.l.google.com:19302
- stun2.l.google.com:19302
- stun3.l.google.com:19302
- stun4.l.google.com:19302

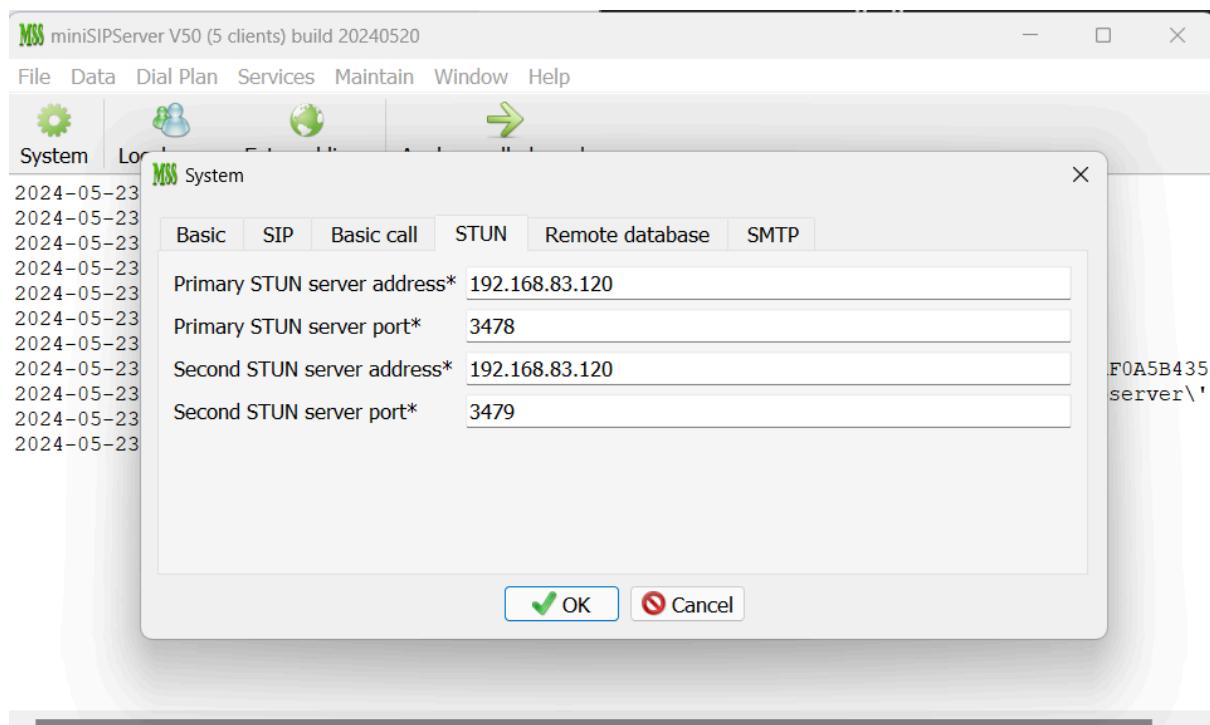
Altres servidors públics STUN:

- stun.sipgate.net:3478
- stun.myvoiptraffic.com:3478
- stun.mywatson.it:3478
- stun.nas.net:3478
- stun.neotel.co.za:3478
- stun.netappel.com:3478
- stun.netappel.fr:3478
- stun.netgsm.com.tr:3478
- stun.nfon.net:3478
- stun.noblogs.org:3478
- stun.noc.ams-ix.net:3478

Podem consultar un gran llistat de servidors públics de STUN en aquest repositori de github:

<https://gist.github.com/mondain/b0ec1cf5f60ae726202e>

**c) Mireu quines capacitats com servidor STUN i/o TURN té el MiniSIPserver, i com configurar-los.**



Aquestes són les opcions STUN que ofereix MiniSIPserver.

- Primary STUN server IP address (MIP), espera un valor de tipus string que sigui l'IP primaria del servidor STUN amb la que es connectarà miniSIPserver.
- Primary STUN server port (MPORT), espera un valor de tipus integer, aquest correspon al port primari del servidor STUN amb el que el miniSIPserver es connectarà. El valor per defecte es 3478, que és l'estàndar d'STUN.
- Second STUN server IP address (SIP), espera un valor string, en aquest cas és el mateix que el primer paràmetre però ara és la segona IP del servidor, en comptes de la primaria.
- Second STUN server Port (SPORT), espera un valor integer, és el port secundari al que es connectarà el miniSIPserver. Ha de ser diferent al port primari.

**d) És més complicat trobar servidors TURN públics. Mireu**

**<https://www.metered.ca/tools/openrelay/> i obteniu un parell d'usuaris per grup.**

Hem creat dos usuaris per al servidor TURN d'Openrelay:

- User 1:
  - email: cristina.sanchez-mora@estudiantat.upc.edu
  - password: 2@dh7ThC@.ASw8L

## Exercici previ 6

A un PC Linux, amb l'eina dig, investigueu els registres NAPTR i SRV del domini upc.edu.

L'eina dig és nativa en Linux, i per Windows us proporcionem a Atenea la implementació dig.exe per executar des de la línia de comandes (heu de descomprimir i copiar la carpeta a qualsevol path, per exemple c:\dig).

- a) Comenteu el que reporten les ordres dig -t ANY upc.edu (de vegades, sobretot si esteu connectats des de fora de la xarxa UPC, heu de demanar pels registres concrets, per exemple dig -t SRV \_sip.\_udp.upc.edu).

En un pc de linux, una vegada amb la eina de dig descarregada i verificada que esta correctament instal·lada amb la comanda : dig -v

DIG 9.16.48-Ubuntu

Els resultats de la consulta de dig -t ANY upc.edu connectat a una wifi de la universitat són:

```
c6890730@aul-1927:~/home-ubwan-c6890730/XAMU$ dig -t ANY upc.edu

; <>> DIG 9.16.48-Ubuntu <>> -t ANY upc.edu
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 7125
;; flags: qr rd ra; QUERY: 1, ANSWER: 19, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;upc.edu.           IN      ANY

;; ANSWER SECTION:
upc.edu.        172800  IN      NS      ns2.upc.edu.
upc.edu.        172800  IN      NS      ns1.cesca.cat.
upc.edu.        172800  IN      NS      ns1.upc.edu.
upc.edu.        172800  IN      NS      ns2.cesca.cat.
upc.edu.        172800  IN      A       147.83.2.135
upc.edu.        3600   IN      TXT    "facebook-domain-verification=njmvfgzqisovaytxj5f8up1ovvrf3q"
upc.edu.        3600   IN      TXT    "MS-5F57EA05461D4B11AEF4C3F8ED0F52F53DB08A4B"
upc.edu.        3600   IN      TXT    "google-site-verification=5KL5bpexmYTeAvf7kLvGgeUQtgMLovdgYamQVNsnHGo"
upc.edu.        3600   IN      TXT    "1687262495881"
upc.edu.        3600   IN      TXT    "1657696626453"
upc.edu.        3600   IN      TXT    "voje86dnmac5sh6b3s8fgdf179"
upc.edu.        172800  IN      NAPTR  30 0 "s" "SIP+D2U" "" _sip._udp.upc.edu.
upc.edu.        3600   IN      MX    1 aspmx.l.google.com.
upc.edu.        3600   IN      MX    5 alt2.aspmx.l.google.com.
upc.edu.        3600   IN      MX    10 alt3.aspmx.l.google.com.
upc.edu.        3600   IN      MX    5 alt1.aspmx.l.google.com.
upc.edu.        3600   IN      MX    10 alt4.aspmx.l.google.com.
upc.edu.        172800  IN      SOA   ns.upc.edu. hostmaster.upcnet.es. 2019134015 1800 900 1814400 7200

;; Query time: 7 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 12:58:35 CEST 2024
;; MSG SIZE rcvd: 788
```

On podem observar que:

Capçalera de resposta

- Status: NOERROR: La consulta ha estat completada sense errors.
- ANSWER: 19: Es retornen 19 registres DNS en la secció de resposta.

Secció de resposta (ANSWER SECTION)

- Registre A: 147.83.2.135: Aquesta és l'adreça IP associada amb el domini upc.edu, és a dir, l'adreça del servidor on està allotjat el lloc web principal.
- Registre NAPTR: 30 0 "s" "SIP+D2U" "" \_sip.\_udp.upc.edu.: Especifica una política de reescritura URI per al servei SIP sobre UDP, útil per a encaminar trucades de VoIP. Amb una prioritat de 30 i un pes de 0. La consulta DNS ha estat processada per un servidor DNS local.

Els resultats de la consulta de **dig -t SRV \_sip.\_udp.upc.edu** connectat a una wifi de la universitat són:

```
c6890730@aul-1927:~/home-ubilwan-c6890730/XAMU$ dig -t SRV _sip._udp.upc.edu
; <>> DiG 9.16.48-Ubuntu <>> -t SRV _sip._udp.upc.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NXDOMAIN, id: 25714
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;_t.                      IN      SRV

;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 27286
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;_sip._udp.upc.edu.        IN      A

;; Query time: 4 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 12:59:16 CEST 2024
;; MSG SIZE rcvd: 38

;; Got answer:
;; ->>HEADER<- opcode: QUERY, status: NOERROR, id: 27286
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;_sip._udp.upc.edu.        IN      A

;; Query time: 43 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 12:59:16 CEST 2024
;; MSG SIZE rcvd: 46
```

On podem observar que:

#### Anàlisi de la Primera Consulta

- Status: NXDOMAIN: Aquest status indica que el domini consultat no existeix.
- Query: La secció de pregunta mostra -t, que s'interpreta com un nom de domini en aquest context a causa d'un error en el tipus o format de la comanda.
- Resultat: No es retorna cap resposta (ANSWER: 0) i no hi ha autoritat o informació addicional que indique quin servidor podria tenir informació pertanyent, llevat de l'EDNS que indica la capacitat de paquet.

#### Anàlisi de la Segona Consulta:

- Status: NOERROR: Encara que no hi ha errors, no es retorna informació.
- Query: La pregunta per \_sip.\_udp.upc.edu amb tipus de registre A també és incorrecta ja que estàs buscant un registre A per un servei que requeriria un registre SRV. El registre A és per adreces IP de host, no per especificacions de serveis.
- Resultat: Com abans, no es retorna cap resposta (ANSWER: 0).

Però des de un ordinador que no és de la aula el dig retorna:

```
mariona@mariona-laptop:~$ dig -t SRV _sip._udp.upc.edu
; <>> DiG 9.18.18-0ubuntu0.22.04.2-Ubuntu <>> -t SRV _sip._udp.upc.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18235
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
_sip._udp.upc.edu.          IN      SRV

;; ANSWER SECTION:
_sip._udp.upc.edu.      3600    IN      SRV      0 0 5060 cursa-v.upc.es.

;; Query time: 4 msec
;; SERVER: 127.0.0.53#53(127.0.0.53) (UDP)
;; WHEN: Mon May 13 13:06:50 CEST 2024
;; MSG SIZE  rcvd: 80
```

On podem observar que:

Capçalera de resposta

- Status: La consulta ha estat exitosa (NOERROR), indicant que s'ha trobat informació pertinent.
- Consulta: Demana el registre SRV per \_sip.\_udp.upc.edu, específic per al protocol SIP sobre UDP.

Secció de resposta

- Registre SRV: El registre es mostra en la secció de resposta: \_sip.\_udp.upc.edu. 3415 IN SRV 0 0 5060 cursa-v.upc.es.
  - Prioritat: 0 : La prioritat més baixa, que en aquest context significa la màxima prioritat (els números més baixos tenen prioritat més alta).
  - Pes: 0 : Utilitzat per triar entre múltiples registres amb la mateixa prioritat; aquí sembla no aplicar-se doncs només hi ha un servidor.
  - Port: 5060 : El port típic per a SIP.
  - Nom del servidor: cursa-v.upc.es :El servidor designat per manejar les peticions SIP.

**b) Torneu a fer-ho, preguntant pel domini upf.edu, uc3m.es, i uoc.edu. Proveu també amb sip.us, i comenteu la seva configuració en quant a prioritat i pes.**

Els resultats de la consulta de **dig -t ANY upf.edu** connectat a una wifi de la universitat són:

```
c6890730@aul-1927:~/home-ubiwan-c6890730/XAMU$ dig -t ANY upf.edu
; <>> DiG 9.16.48-Ubuntu <>> -t ANY upf.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54065
;; flags: qr rd ra; QUERY: 1, ANSWER: 13, AUTHORITY: 0, ADDITIONAL: 1
;;
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;upf.edu.           IN      ANY
;;
;; ANSWER SECTION:
upf.edu.        27865   IN      A       34.160.111.234
upf.edu.        34628   IN      NS      ns4.upf.edu.
upf.edu.        34628   IN      NS      ns3.upf.edu.
upf.edu.        34628   IN      NS      ns6.upf.edu.
upf.edu.        34628   IN      NS      ns1.upf.edu.
upf.edu.        34628   IN      NS      ns5.upf.edu.
upf.edu.        34628   IN      NS      ns2.upf.edu.
upf.edu.        22619   IN      MX      1 aspmx.l.google.com.
upf.edu.        22619   IN      MX      5 alt1.aspmx.l.google.com.
upf.edu.        22619   IN      MX      5 alt2.aspmx.l.google.com.
upf.edu.        22619   IN      MX      10 aspmx3.googlemail.com.
upf.edu.        22619   IN      MX      10 aspmx2.googlemail.com.
upf.edu.        21665   IN      NAPTR   30 0 "s" "SIP+D2U" "" _sip._udp.upf.edu.

;; Query time: 75 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 13:01:02 CEST 2024
;; MSG SIZE  rcvd: 332
```

On podem observar en la secció de resposta els registres obtinguts són:

- Registre A: 34.160.111.234 és l'adreça IP assignada a upf.edu.
- Registre NAPTR: Especificació per a serveis de SIP sobre UDP (\_sip.\_udp.upf.edu). Amb una Prioritat de 30 i un Pes de 0.

Els resultats de la consulta de **dig -t ANY uc3m.es** connectat a una wifi de la universitat són:

```
c6890730@aul-1927:~/home-ubikan-c6890730/XAMU$ dig -t ANY uc3m.es
; <>> DiG 9.16.48-Ubuntu <>> -t ANY uc3m.es
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52035
;; flags: qr rd ra; QUERY: 1, ANSWER: 16, AUTHORITY: 0, ADDITIONAL: 1
;;
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;uc3m.es.          IN      ANY
;;
;; ANSWER SECTION:
uc3m.es.        300    IN      A      176.58.10.138
uc3m.es.        86400   IN      NS     saruman.uc3m.es.
uc3m.es.        86400   IN      NS     chico.rediris.es.
uc3m.es.        86400   IN      NS     sun.rediris.es.
uc3m.es.        86400   IN      NS     vortex.uc3m.es.
uc3m.es.        86400   IN      SOA    vortex.uc3m.es. netmaster.uc3m.es. 2024051301 86400 7200 2592000 172800
uc3m.es.        300    IN      MX    1 aspmx.l.google.com.
uc3m.es.        300    IN      MX    5 alt1.aspmx.l.google.com.
uc3m.es.        300    IN      MX    10 alt4.aspmx.l.google.com.
uc3m.es.        300    IN      MX    10 alt3.aspmx.l.google.com.
uc3m.es.        300    IN      MX    5 alt2.aspmx.l.google.com.
uc3m.es.        300    IN      TXT   "v=spf1 redirect=_spf.uc3m.es"
uc3m.es.        300    IN      AAAA  2a0a:7dc0:101:340:176:58:10:138
uc3m.es.        86400   IN      CAA   0 issue "digicert.com"
uc3m.es.        86400   IN      CAA   0 iodef "mailto:cert@uc3m.es"
uc3m.es.        86400   IN      CAA   0 issue "sectigo.com"
;;
;; Query time: 59 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 13:02:04 CEST 2024
;; MSG SIZE rcvd: 473
```

On podem observar en la secció de resposta els registres obtinguts són:

- Registre A i AAAA: Adreces IP IPv4 (176.58.10.138) i IPv6 per a uc3m.es.
- Registres CAA: Polítiques de certificat que especificuen quins emissors de certificats estan autoritzats per uc3m.es.

Però cap més registre de registres sip com NAPTR o SRV.

Els resultats de la consulta de **dig -t ANY uoc.edu** connectat a una wifi de la universitat són:

```
c6890730@aul-1927:~/home-ubikan-c6890730/XAMU$ dig -t ANY uoc.edu
; <>> DiG 9.16.48-Ubuntu <>> -t ANY uoc.edu
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 45669
;; flags: qr rd ra; QUERY: 1, ANSWER: 8, AUTHORITY: 0, ADDITIONAL: 1
;;
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;uoc.edu.          IN      ANY
;;
;; ANSWER SECTION:
uoc.edu.        153    IN      A      75.2.63.131
uoc.edu.        153    IN      A      99.83.131.89
uoc.edu.        237    IN      NS     ns-142.awsdns-17.com.
uoc.edu.        237    IN      NS     ns-1381.awsdns-44.org.
uoc.edu.        237    IN      NS     ns-1588.awsdns-06.co.uk.
uoc.edu.        237    IN      NS     ns-620.awsdns-13.net.
uoc.edu.        151    IN      TXT   "v=spf1 a:legacy.uoc.edu include:_spf.google.com include:in1.uoc.edu include:ex1.uoc.edu ~all"
uoc.edu.        62     IN      NAPTR  0 30 "S" "SIP+D2U" "" _sip._udp.uoc.edu.
;;
;; Query time: 15 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 13:02:32 CEST 2024
;; MSG SIZE rcvd: 352
```

On podem observar en la secció de resposta els registres obtinguts són:

- Registres A: Dues adreces IP diferents, 75.2.63.131 i 99.83.131.89.
- Registre NAPTR: Per a serveis de SIP sobre UDP (\_sip.\_udp.uoc.edu) Amb una prioritat de 0 i un pes de 30.

Els resultats de la consulta de **dig -t ANY sip.us** connectat a una wifi de la universitat són:

```
c6890730@aul-1927:~/home-ubikan-c6890730/XAMU$ dig -t ANY sip.us
; <>> DLG 9.16.48-Ubuntu <>> -t ANY sip.us
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 48802
;; flags: qr rd ra; QUERY: 1, ANSWER: 7, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;sip.us.                      IN      ANY

;; ANSWER SECTION:
sip.us.          86400   IN      SOA     ns10.dnsmadeeasy.com. dns.dnsmadeeasy.com. 2009011536 43200 3600 1209
600 180
sip.us.          7200    IN      NS       ns11.dnsmadeeasy.com.
sip.us.          7200    IN      NS       ns13.dnsmadeeasy.com.
sip.us.          7200    IN      NS       ns12.dnsmadeeasy.com.
sip.us.          7200    IN      NS       ns10.dnsmadeeasy.com.
sip.us.          7200    IN      NS       ns15.dnsmadeeasy.com.
sip.us.          7200    IN      NS       ns14.dnsmadeeasy.com.

;; Query time: 43 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon May 13 13:31:06 CEST 2024
;; MSG SIZE rcvd: 204
```

On podem observar en la secció de resposta els registres obtinguts són:

- Registre SOA: és un tipus de registre DNS que proporciona informació sobre la zona DNS i els paràmetres de gestió d'aquesta zona, en aquest cas dona varis dns.
- Els registres NS no tenen una prioritat explícita com els registres MX, però tots els servidors de noms són igualment vàlids per a la resolució DNS.

Però cap més registre de registres sip com NAPTR o SRV.

Les 4 consultes anteriors, retornen un estatus de NOERROR, indicant que les consultes DNS han sigut exitoses i recuperen les dades corresponents sense cap error. Dins de la consulta s'ha demanat tots els tipus de registres disponibles (ANY) per cada domini, reflectint una àmplia gamma de configuracions DNS per a cada institució.

La seva configuració per la prioritat i el seu pes són:

**UPF.EDU:** Registres NAPTR: upf.edu inclou un registre NAPTR que especifica encaminament per SIP sobre UDP amb una prioritat de 30 i un pes de 0. El valor de prioritat és relativament alt (prioritats més baixes tenen preferència), i un pes de 0 suggerint que no hi ha altres opcions amb la mateixa prioritat per balançar.

**UC3M.ES:** Registres SRV: No es mostren registres SRV en la resposta. Per tant, no tenim dades sobre la configuració de prioritat i pes per serveis com SIP o altres.

**UOC.EDU:** Registres NAPTR: Similar a upf.edu, uoc.edu té un registre NAPTR que apunta a \_sip.\_udp.uoc.edu amb una prioritat de 0 i un pes de 30. Aquesta configuració indica prioritat màxima (el nombre més baix) i un

pes que podria usar-se per balançar càrregues si hi hagués múltiples registres amb la mateixa prioritat.

**SIP.US:** Registres SRV/NAPTR: No hi ha registres SRV o NAPTR mostrats per a sip.us en aquesta consulta, cosa que significa que no podem analitzar la configuració de prioritat i pes per a SIP o altres serveis directament d'aquesta sortida.

## Exercici previ 7

Obriu amb el Wireshark la captura **SIP\_RTP\_complet.pcap** que teniu a l'Atenea, i analitzeu-la amb Wireshark tant a nivell del diàleg (veure Figura 1) com entrant en el detall dels missatges, quan calgui.

- a) Identifiqueu quantes màquines estan involucrades en el diàleg i quines adreces IP tenen. És una trucada directa o passeeu per un proxy?**

Hi ha 2 màquines involucrades, que passen per un proxy(en el camp via ens indica que hi ha un proxy pel que pasa una trucada)

19 30.845116	192.168.1.128	147.83.194.150	SIP	513 Status: 200 OK
20 30.847574	192.168.1.128	147.83.194.150	SIP	513 Status: 200 OK
21 33.052782	192.168.1.128	147.83.194.150	SIP/SDP	1012 Request: INVITE sip:user01.sai@upc.edu
22 33.105372	147.83.194.150	192.168.1.128	SIP	414 Status: 100 Giving a try
23 33.105378	147.83.194.150	192.168.1.128	SIP	576 Status: 407 Proxy Authentication Required
24 33.106360	192.168.1.128	147.83.194.150	SIP	408 Request: ACK sip:user01.sai@upc.edu
25 33.109617	192.168.1.128	147.83.194.150	SIP/SDP	1223 Request: INVITE sip:user01.sai@upc.edu
26 33.165380	147.83.194.150	192.168.1.128	SIP	414 Status: 100 Giving a try
27 33.296385	147.83.194.150	192.168.1.128	SIP	722 Status: 180 Ringing
28 46.812132	147.83.194.150	192.168.1.128	SIP/SDP	1323 Status: 200 OK

Message Header

```
> Via: SIP/2.0/UDP 192.168.1.128:13620;branch=z9hG4bK-d8754z-a6c2fce0b44665fd-1---d8754z;-rport
> Max-Forwards: 70
> Contact: <sip:user02.sai@95.17.60.222:13620;rinstance=4d28cb4479592173>
> To: "user02.sai"<sip:user02.sai@upc.edu>
> From: "user02.sai"<sip:user02.sai@upc.edu>;tag=d420616d
> Call-ID: YzU5ZWU0YmI0ZTQxYj1hYzBhYtgzZmUzMWViMDUwYTA.
> [Generated Call-ID: YzU5ZWU0YmI0ZTQxYj1hYzBhYtgzZmUzMWViMDUwYTA.]
> CSeq: 5 REGISTER
> Expires: 3600
```

Per tant amb aquesta captura, que mostra el diàleg Sip, podem afirmar que hi ha 2 màquines, user02.sai 192.168.1.128 i user01.sai 147.83.194.150:

No.	Time	Source	Destination	Protocol	Length	Info	Delta
1	0.000000	192.168.1.128	147.83.194.150	SIP	596	Request: REGISTER sip:upc.edu (1 binding)	
2	0.050154	147.83.194.150	192.168.1.128	SIP	571	Status: 401 Unauthorized	
3	0.053685	192.168.1.128	147.83.194.150	SIP	790	Request: REGISTER sip:upc.edu (1 binding)	
4	0.105041	147.83.194.150	192.168.1.128	SIP	548	Status: 200 OK (REGISTER) (1 binding)	
5	0.108595	192.168.1.128	147.83.194.150	SIP	785	Request: REGISTER sip:upc.edu (remove 1 binding)	
6	0.160540	147.83.194.150	192.168.1.128	SIP	571	Status: 401 Unauthorized	
7	0.163434	192.168.1.128	147.83.194.150	SIP	785	Request: REGISTER sip:upc.edu (remove 1 binding)	
8	0.215710	147.83.194.150	192.168.1.128	SIP	461	Status: 200 OK (REGISTER) (0 bindings)	
9	0.218761	192.168.1.128	147.83.194.150	SIP	789	Request: REGISTER sip:upc.edu (1 binding)	

També cal recalcar que en el diàleg RTP apareix la maquina 147.83.194.156

30 47.421954	192.168.1.128	147.83.194.156	RTP	58 PT=DynamicRTP-Type-126, SSRC=0x139F9A82, Seq=6261, ..
31 47.438185	147.83.194.156	192.168.1.128	RTP	134 PT=BV32, SSRC=0x97EDEAC2, Seq=7601, Time=2009700
32 47.459186	147.83.194.156	192.168.1.128	RTP	134 PT=BV32, SSRC=0x97EDEAC2, Seq=7602, Time=2010020

En resum aquí hi ha una captura que mostra les 3 màquines:

Intervalo	192.168.1.128	147.83.194.150	Comentario
33.052782	13620 INVITE SDP (BV32 g711U g711A telephone-event)	5062	SIP INVITE From: "user02.sai" <sip:user02.sai@upc.edu>
33.105372	13620 100 Giving a try	5062	SIP Status 100 Giving a try
33.105378	13620 407 Proxy Authentication Required	5062	SIP Status 407 Proxy Authentication Required
33.106361	13620 ACK	5062	SIP ACK From: "user02.sai" <sip:user02.sai@upc.edu>
33.109617	13620 INVITE SDP (BV32 g711U g711A telephone-event)	5062	SIP INVITE From: "user02.sai" <sip:user02.sai@upc.edu>
33.165380	13620 100 Giving a try	5062	SIP Status 100 Giving a try
33.296385	13620 180 Ringing	5062	SIP Status 180 Ringing
46.812152	13620 200 OK SDP (BV32 g711U g711A telephone-event)	5062	SIP Status 200 OK
46.847290	13620 ACK	5062	SIP Request INVITE ACK 200 CSeq:2
47.421954	57120 RTP (RTPType-126)	4	RTP, 3 packets. Duration: 20.12s SSRC: 0x139F9A82
47.438185	57120 RTP (BV32)	4	RTP, 1136 packets. Duration: 22.74s SSRC: 0x97ED..
70.316499	13620 BYE	5062	SIP Request BYE CSeq:2
70.345733	13620 200 OK	5062	SIP Status 200 OK

- b) Identifiqueu els usuaris involucrats (la seva SIP URI) i el seu paper (qui truca/qui és trucat). Esbrineu si algun dels elements involucrats a la trucada està darrera d'un NAT, i en aquest cas, què s'està fent als missatges SIP per poder saltar-lo.**

Amb la captura podem afirmar que la maquina 192.168.1.128 es qui truca al 147.83.194.150

```
> To: <sip:user01.sai@upc.edu>
> From: "user02.sai" <sip:user02.sai@upc.edu>;tag=1b658bd0
```

En aquesta captura com podem veure en el paquet 'request' en el camp de via hi ha rport, per tant, hi ha Nat al darrere:

Frame 29: 915 bytes on wire (7320 bits), 915 bytes captured (7320 bits)
Ethernet II, Src: Wistron_f6:f8:fe (00:1f:16:f6:f8:fe), Dst: AskeyCom_fb:f2:80 (e8:39:df:fb:f2:80)
Internet Protocol Version 4, Src: 192.168.1.128, Dst: 147.83.194.150
User Datagram Protocol, Src Port: 13620, Dst Port: 5062
Session Initiation Protocol (ACK)
> Request-Line: ACK sip:u13-1545@147.83.194.150:5062 SIP/2.0
Message Header
> Via: SIP/2.0/UDP 192.168.1.128:13620;branch=z9hG4bK-d8754z-2c7ddcb446519208-1---d8754z;rport
Max-Forwards: 70
> Route: <sip:uuser@147.83.194.150:5062;lr;ftag=1b658bd0>
> Route: <sip:147.83.194.150:5070;lr-on;ftag=1b658bd0>
> Route: <sip:147.83.194.150:5070;lr-on;ftag=1b658bd0>

- c) Comenteu, com a mínim (però no us limiteu a això)**

**a. Procés de registre**

Al principi hi ha molts intents per registrar upc.edu, pero es denega (unauthorized 401), fins que s'accepta (200 ok), i després es registra correctament

**b. Hi ha missatges de la sèrie 1xx? Què indiquen?**

1xx equival a INVITE, per tant sí que hi ha, i indiquen qui usuari truca a l'altre i el número de seqüència. En aquest cas el 192.168.1.128 truca a 147.83.194.150, i el número de seqüència es 1.

```

Message Header
> Via: SIP/2.0/UDP 192.168.1.128:13620;branch=z9hG4bK-d8754z-983ce08897c0acd8-1---d8754z-;rport
> Max-Forwards: 70
> Contact: <sip:user02.sai@95.17.60.222:13620>
> To: <sip:user01.sai@upc.edu>
> From: "user02.sai"<sip:user02.sai@upc.edu>;tag=1b658bd0
> Call-ID: NDC2NmJ3NzhkZj3ZGlxN2FkQW0YzODNhMjNhN2Y.
> [Generated Call-ID: NDC2NmJ3NzhkZj3ZGlxN2FkQW0YzODNhMjNhN2Y.]
> CSeq: 1 INVITE
> Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO
Content-Type: application/sdp
Supported: replaces
User-Agent: X-Lite 4 release 4.1 stamp 63214
Content-Length: 410
> Message Body

```

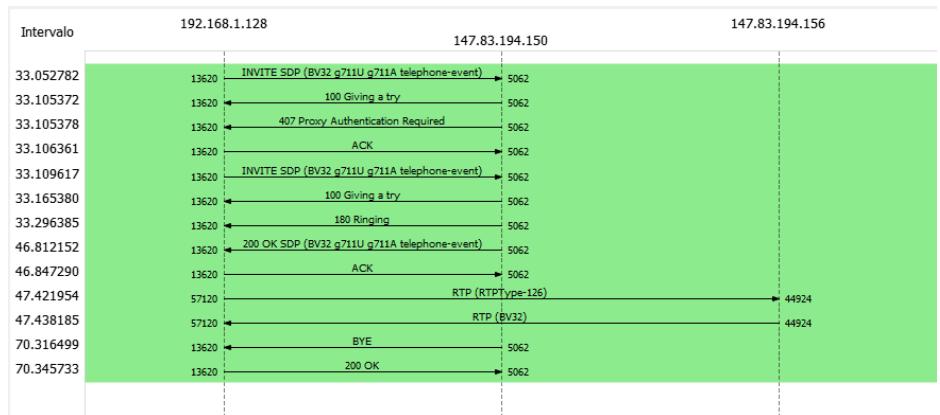
20 30.847574	192.168.1.128	147.83.194.150	SIP	513 Status: 200 OK (OPTI...
21 33.052782	192.168.1.128	147.83.194.150	SIP/SDP	1012 Request: INVITE sip:...
22 33.105372	147.83.194.150	192.168.1.128	SIP	414 Status: 100 Giving a...
23 33.105378	147.83.194.150	192.168.1.128	SIP	576 Status: 407 Proxy Au...
24 33.106361	192.168.1.128	147.83.194.150	STP	409 Request: ACK sip:use...

### c. Comenteu el procés de negociació dels ports i els còdecs:

#### i. Quants fluxos audiovisuals (unidireccionals) s'estableixen? A quins ports RTP?

Hi han dos fluxos, amb els ports 5062 i 13620 respectivament

#### ii. Quins còdecs proposa cada costat, i qui s'escull finalment?



Del primer flux (192.168.1.128 -> 147.83.194.150) el payload es 970 per tant un codec de G711U

Del segon flux (147.83.194.150 -> 192.168.1.128) proposa varis.

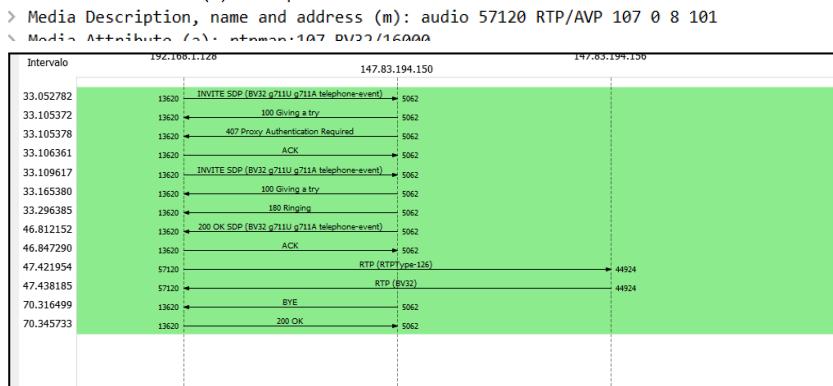
Pero finalment s'escull el 970 = BV32 -> codec=G711U G711A, ja que en el procés d'elecció de còdecs, cada terminal indica als SDP una llista prioritizada de còdecs, i s'escull el primer que aparegui en les dues llistes i que estigui a la posició més prioritària pels dos terminals

#### iii. Comproveu les vostres respostes visualitzant els primers paquets RTP.

En la primera captura veiem que proposa els diferents codecs, i en la segona captura confirmem que han triat el primer codec en el camp Payload:

```
> Media Description, name and address (m): audio 57120 RTP/AVP 107 0 8 101
Media Attribute (a): a=rtpmap:107 BV32/16000
0... .... = Marker: False
Payload type: BV32 (107)
```

En la tercera captura també ho podem confirmar, que utilitza els mateixos codecs que hem dit anteriorment



#### d. Comenteu com finalitza la trucada.

el 147.83.194.150 finalitza la trucada amb un BYE Request, i el 192.168.1.128 respon amb un 200 OK

#### d) Filtreu per veure els fluxos RTP (i estudieu-los també amb l'analitzador RTP de Wireshark)

a. Calculeu la diferència de *timestamps* entre paquets consecutius. Tenint en compte la freqüència de rellotge associada al *payload type* (on la podeu trobar? Repasseu el diàleg SIP) convertiu la diferència de *timestamps* a temps d'àudio que transporta el paquet RTP.

El Payload Type el trobem en la info, per tant com que PT=BV32 la freqüència es de 16khz

Index	Timestamp	Source IP	Destination IP	Protocol	Timestamp	Source IP	Destination IP	Protocol
31	47.438185	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7601	147.83.194.150	RTP
32	47.459188	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7602	147.83.194.150	RTP
33	47.482186	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7603	147.83.194.150	RTP
34	47.499187	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7604	147.83.194.150	RTP
35	47.518188	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7605	147.83.194.150	RTP
36	47.538237	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7606	147.83.194.150	RTP
37	47.558192	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7607	147.83.194.150	RTP
38	47.570202	147.83.194.156	192.168.1.128	RTP	134	PT=BV32, SSRC=0x97ED0AC2, Seq=7608	147.83.194.150	RTP

Timestamps : 2010340-2010020=320 ticks

Temps audio=320ticks/16000ticks/s=20ms cada paquet

#### b. Amb l'analitzador RTP, visualitzeu la separació

**entre paquets consecutius. Comenteu si el jitter us sembla alt o petit. Amb quin valor l'heu de comparar?**

Packet	Sequence	Delta(ms)	Filtered Jitter(ms)	Skew(ms)	IP BW(kbps)	Marker	Status
31	7601	0.00	0.00	0.00	0.96	[ Ok ]	
32	7602	21.00	0.06	-1.00	1.92	[ Ok ]	
33	7603	23.00	0.25	-4.00	2.88	[ Ok ]	
34	7604	17.00	0.42	-1.00	3.84	[ Ok ]	
35	7605	19.00	0.45	-0.00	4.80	[ Ok ]	
36	7606	20.05	0.43	-0.05	5.76	[ Ok ]	
37	7607	19.95	0.41	-0.01	6.72	[ Ok ]	
38	7608	21.00	0.44	-1.01	7.68	[ Ok ]	
39	7609	20.00	0.41	-1.01	8.64	[ Ok ]	
40	7610	20.00	0.39	-1.01	9.60	[ Ok ]	
41	7611	21.01	0.43	-2.02	10.56	[ Ok ]	
42	7612	17.99	0.53	-0.01	11.52	[ Ok ]	
43	7613	20.00	0.49	-0.01	12.48	[ Ok ]	
44	7614	21.00	0.53	-1.01	13.44	[ Ok ]	
45	7615	19.01	0.56	-0.02	14.40	[ Ok ]	
46	7616	20.00	0.52	-0.02	15.36	[ Ok ]	
47	7617	22.00	0.61	-2.02	16.32	[ Ok ]	
48	7618	18.00	0.70	-0.02	17.28	[ Ok ]	
49	7619	22.00	0.78	-2.02	18.24	[ Ok ]	
50	7620	19.00	0.79	-1.02	19.20	[ Ok ]	
51	7621	20.00	0.74	-1.02	20.16	[ Ok ]	

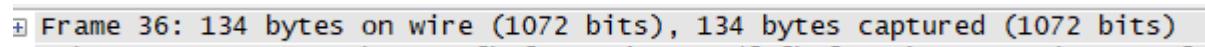
Si comparem el jitter amb el delt ens surt que és més petit, per tant és bo.

- e) Sabent que el còdec BV32 genera una taxa de 32 Kbit/s, i utilitzant el temps real d'àudio calculat abans, calculeu quants bytes d'àudio transporta cada paquet RTP.**

Bytes cada paquet=  $32000 * 0.02 = 640$  bits /8=80 bytes

- f) A partir del resultat anterior, i de les mides de les capçaleres Ethernet, IP, UDP i RTP, justifiqueu numèricament la mida de les trames Ethernet que transporten paquets RTP.**

Ethernet/IP/UDP/RTP(14/20/8/12) bytes= 54 bytes  
80bytes+54 bytes =134 bytes



- g) Calculeu, teòricament, el bitrate generat a nivell IP pel flux d'àudio. Compareu amb el que reporta l'analitzador expert RTP de Wireshark i discutiu possibles discrepàncies.**

IP/UDP/RTP(20/8/12) bytes= 40 bytes

80bytes+54 bytes =120 bytes

120 bytes \* 8 bits/(0.02s)=48000 bit/s

L'ample de banda màxim es 50.88 bps pero la mitja es de 48 bps

Paquete	Sequence	Delta (ms)	Jitter (ms)	Skew	Ancho de banda	Marker	Estado
778	8343	17.999000	1.627728	-44.852000	50.88	✓	
777	8342	25.999000	1.602843	-46.853000	49.92	✓	
865	8430	18.999000	1.158029	-42.952000	49.92	✓	
1151	8715	2.011000	3.902268	-41.289000	49.92	✓	
1155	8719	16.001000	3.600610	-41.282000	49.92	✓	
83	7653	18.005000	1.238061	-0.062000	48.96	✓	
84	7654	20.005000	1.160994	-0.067000	48.96	✓	
87	7657	16.984000	1.263643	0.938000	48.96	✓	
89	7659	18.008000	1.411022	-0.072000	48.96	✓	
91	7661	17.003000	1.602957	-0.070000	48.96	✓	

```

` Internet Protocol Version 4, Src: 147.83.194.156, Dst: 192.168.1.128
  0100 .... = Version: 4
  .... 0101 = Header Length: 20 bytes (5)
> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
  Total Length: 120
  Identification: 0x0000 (0)
> 010. .... = Flags: 0x2, Don't fragment

```

## Exercici previ 8

**Coordineu-vos per fer una trucada entre dos terminals (idealment, un Linphone i un MicroSIP), configurats amb els comptes UPCnet, on almenys un dels dos terminals és a casa vostra (és important, per assegurar que hi ha un NAT involucrat) registreu un terminal SIP al proxy UPC.**

El escenari que tenim serà: una màquina Linux amb l'aplicació de Wireshark capturant els diferents paquets i la aplicació Linphone.

web per la desarrega:

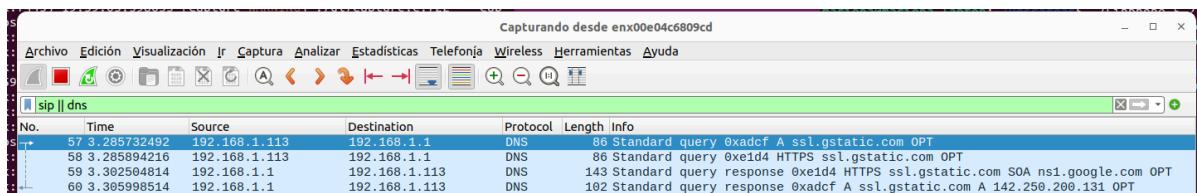
<https://www.linphone.org/category-product/gnu-linux>

La segona màquina serà un ordinador Windows que tindrà la aplicació de MicroSIP que es específica per aquest sistema operatiu.

### a) Abans de configurar el compte, inicie la captura de Wireshark.

Filtreu per capturar els protocols SIP i DNS.

Abans de començar aquest exercici, posar el Wireshark a capturar i filtrarem per protocols SIP i DNS, per aixi només mostrar els paquets que ens importen per aquest protocol:



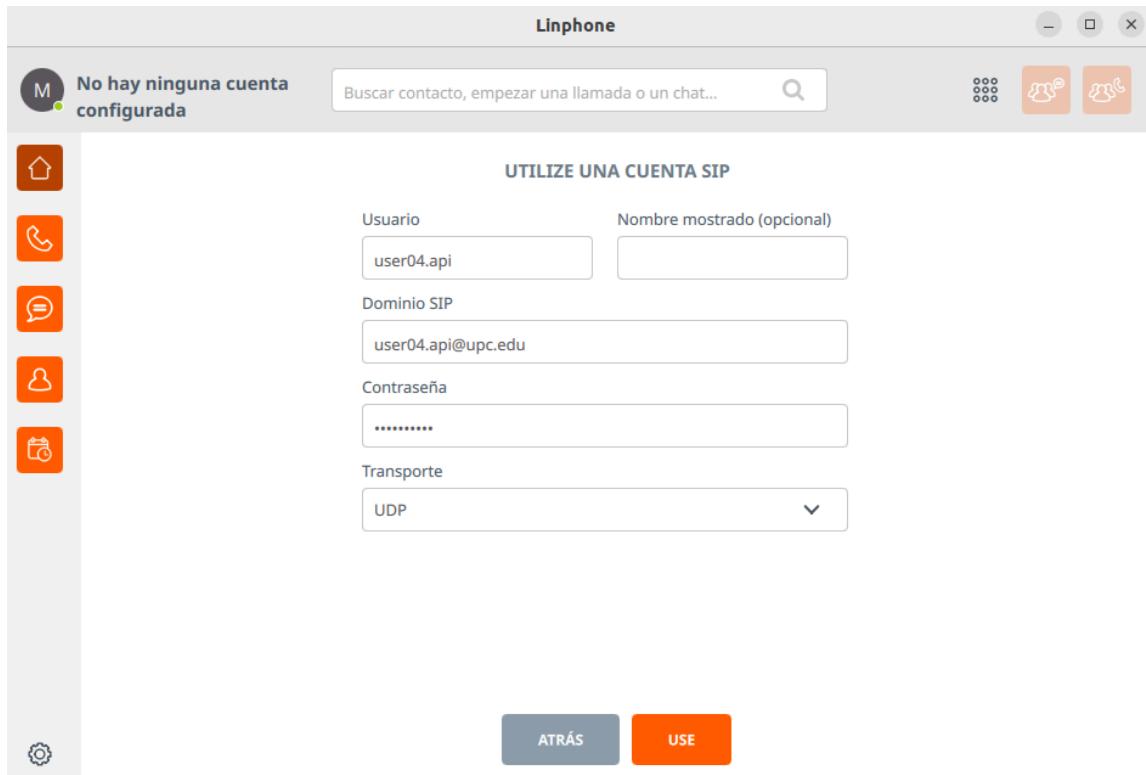
### b) Configureu el client amb un dels comptes SIP de la UPC.

Amb l'aplicació de MicroSIP configurar l'usuari donat:

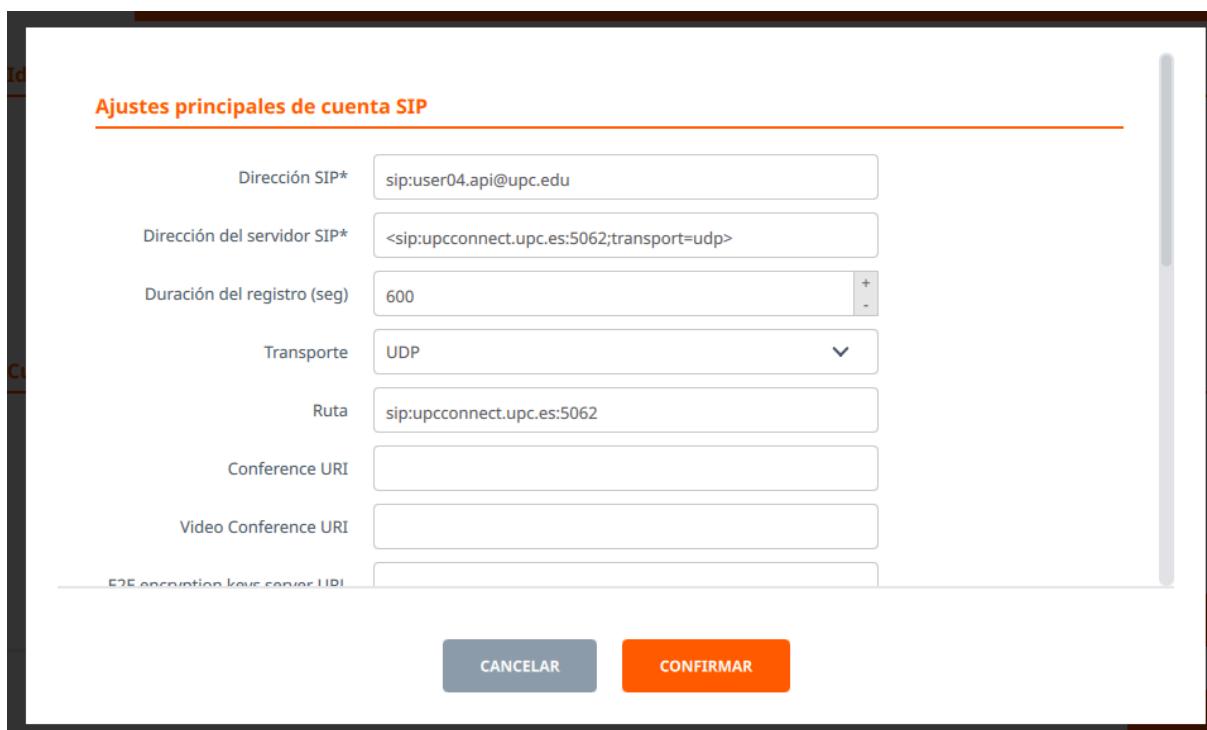
Ara podrem veure que tenim el MicroSIP ven configurat si ens surt el missatge de connectat abaix de la pantalla:  
Una vegada entrat es poden fer trucades des de aquest usuari(usuari04)

Ara des de l'aplicació de Linphone crear un usuari amb els comptes SIP donats per aquest laboratori.

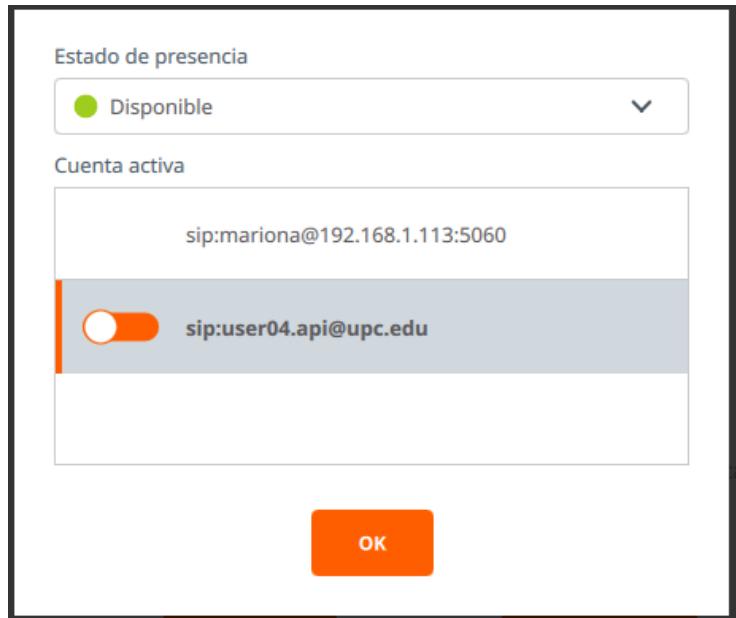
A dins de l'aplicació anar a: Assistente de cuenta → Usar una cuenta SIP i entrar les dades inicials.



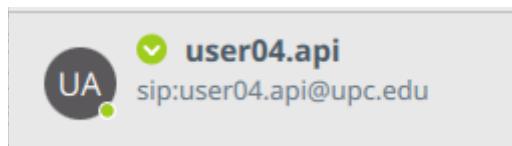
Després s'han de modificar els en paràmetres per la direcció del servidor que sigui la UPC:



Una vegada creat el usuari podem veure els diferents usuaris de l'aplicació i que el nou usuari 04 ha estat correctament configurat:



Ara pot sortir que haguem d'entrar una altre vegada la contrasenya, però una vegada fet, ja tindrem el compte correctament configurat.



### c) Analitzeu el diàleg DNS, relacionant-ho amb el que heu vist a teoria i exercicis previs.

El que podem veure en aquesta captura de paquets es lo següent:

Timestamp	Source IP	Destination IP	Protocol	Message
396 16.060337792	147.83.194.150	192.168.1.15	SIP	547 Request: OPTIONS sip:user04.api@84.76.199.200:20004;transport=udp
391 16.073194744	192.168.1.15	147.83.194.150	DNS	416 Status: 200 OK (OPTIONS)
404 16.073194744	192.168.1.15	192.168.1.1	DNS	86 Standard query 0x0ed2 A play.google.com OPT
486 16.160987899	192.168.1.15	192.168.1.1	DNS	86 Standard query 0x0ed2 HTTPS play.google.com OPT
407 17.1623914544	192.168.1.1	192.168.1.15	DNS	102 Standard query response 0x0e42 A play.google.com A 142.250.200.142 OPT
413 17.179591802	192.168.1.1	192.168.1.15	DNS	136 Standard query response 0x0a6 HTTPS play.google.com SOA ns1.google.com OPT
583 18.596012835	192.168.1.15	192.168.1.1	DNS	98 Standard query 0xe9e3 A clients4.google.com OPT
584 18.596191401	192.168.1.15	192.168.1.1	DNS	98 Standard query 0xcdbd HTTPS clients4.google.com OPT
585 18.599188659	192.168.1.1	192.168.1.15	DNS	149 Standard query response 0xe9e3 A clients4.google.com CNAME clients.l.google.com A 142.250.184.14 OPT
588 18.613724158	192.168.1.1	192.168.1.15	DNS	168 Standard query response 0xcedb HTTPS clients4.google.com CNAME clients.l.google.com SOA ns1.google.com OPT
589 18.614579054	192.168.1.15	192.168.1.1	DNS	91 Standard query 0x5dff HTTPS clients.l.google.com OPT
598 18.634769727	192.168.1.1	192.168.1.15	DNS	141 Standard query response 0x5dff HTTPS clients.l.google.com SOA ns1.google.com OPT
618 18.638841763	192.168.1.15	147.83.194.150	SIP	93 Request: REGISTER sip:upc.edu (1 binding)
619 18.6388328104	147.83.194.150	192.168.1.15	SIP	558 Status: 200 OK (REGISTER) (2 bindings)
742 25.987890464	147.83.194.150	192.168.1.15	SIP	547 Request: OPTIONS sip:user04.api@84.76.199.200:20004;transport=udp
643 25.987890464	192.168.1.15	147.83.194.150	SIP	416 Status: 200 OK (OPTIONS)
686 35.564418352	192.168.1.15	147.83.194.150	SIP	935 Request: REGISTER sip:upc.edu (1 binding)
867 35.573807410	147.83.194.150	192.168.1.15	SIP	404 Status: 401 Unauthorized
868 35.597961044	192.168.1.15	147.83.194.150	SIP	935 Request: REGISTER sip:upc.edu (1 binding)
869 35.6119898279	147.83.194.150	192.168.1.15	SIP	553 Status: 200 OK (REGISTER) (2 bindings)
874 35.897247929	147.83.194.150	192.168.1.15	SIP	547 Request: OPTIONS sip:user04.api@84.76.199.200:20004;transport=udp
875 35.995513655	192.168.1.15	147.83.194.150	SIP	416 Status: 200 OK (OPTIONS)
976 44.159881046	192.168.1.15	192.168.1.1	DNS	80 Standard query 0x0ed A ssl.gstatic.com OPT
977 44.159892357	192.168.1.15	192.168.1.1	DNS	80 Standard query 0xadx9e HTTPS ssl.gstatic.com SOA ns1.google.com OPT
978 44.168798675	192.168.1.15	192.168.1.1	DNS	143 Standard query response 0xad9e HTTPS ssl.gstatic.com A 142.250.201.67 OPT
979 44.1723580436	192.168.1.1	192.168.1.15	DNS	102 Standard query response 0x0ed A ssl.gstatic.com A 142.250.201.67 OPT
1014 45.799263382	147.83.194.150	192.168.1.15	SIP	547 Request: OPTIONS sip:user04.api@84.76.199.200:20004;transport=udp
1015 45.811333042	192.168.1.15	147.83.194.150	SIP	416 Status: 200 OK (OPTIONS)
1027 47.049263239	192.168.1.15	147.83.194.150	SIP	935 Request: REGISTER sip:upc.edu (1 binding)
1027 47.049263249	147.83.194.150	192.168.1.15	SIP	494 Status: 401 Unauthorized
1030 47.064937776	192.168.1.15	147.83.194.150	SIP	935 Request: REGISTER sip:upc.edu (1 binding)
1031 47.074896534	147.83.194.150	192.168.1.15	SIP	553 Status: 200 OK (REGISTER) (2 bindings)

Podem veure que entre els diferents paquets hi ha alguns de protocol DNS, on hi ha dos tipus:

- **Standard query:** Aquest és un missatge de consulta enviat pel client a un servidor DNS per resoldre un nom de domini.
- **Standard query response:** Aquest és un missatge de resposta enviat pel servidor DNS al client amb la informació sol·licitada.

En aquesta captura, podem veure diferents interaccions DNS entre les IPs 192.168.1.15 (aquesta mateixa màquina linux) i el 192.168.1.1 (la màquina on hi ha el MicroSIP configurat).

Podem fitxar-nos en diferents paquets com:

→ 405 17.160796685 192.168.1.15	192.168.1.1	DNS	86 Standard query 0x0e42 A play.google.com OPT
406 17.160988709 192.168.1.15	192.168.1.1	DNS	86 Standard query 0x10a6 HTTPS play.google.com OPT
↓ 407 17.162914544 192.168.1.1	192.168.1.15	DNS	102 Standard query response 0xe42 A play.google.com A 142.250.200.142 OPT

On el paquet 405 es la query i el 407 la resposta d'aquesta query  
 405: Standard query de 192.168.1.15 a 192.168.1.1 per al domini play.google.com.

407: Standard query response de 192.168.1.1 a 192.168.1.15 amb l'adreça IP 142.250.200.142 per play.google.com.

↑ 583 18.596012835 192.168.1.15	192.168.1.1	DNS	90 Standard query 0xe9e3 A clients4.google.com OPT
584 18.596191401 192.168.1.15	192.168.1.1	DNS	90 Standard query 0xcd8d HTTPS clients4.google.com OPT
↓ 585 18.599108650 192.168.1.1	192.168.1.15	DNS	148 Standard query response 0xe9e3 A clients4.google.com CNAME clients.l.google.com A 142.250.184.14 OPT

On el paquet 583 es la query i el 585 la resposta d'aquesta query  
 583: Standard query de 192.168.1.15 a 192.168.1.1 per al domini clients4.google.com.

585: Standard query response de 192.168.1.1 a 192.168.1.15 amb diverses respostes, incloent informació de CNAME i A.

**d) Analitzeu el diàleg de registre SIP en detall. Comenteu el valor dels paràmetres relacionats amb el registre: From, To, Contact, Expires.**

Ara si mirem els paquets de protocol SIP, podem veure diferents paquets com:

- Paquets 70-71: Són una petició i una resposta d'OPCIONS SIP.  
 Aquesta és una manera que utilitza SIP per a descobrir les característiques i la disponibilitat del servidor.

Dins d'aquests podem veure més informació de les opcions SIP:

- Paquet 70:

The screenshot shows a Wireshark capture of a SIP OPTIONS request. The selected frame is labeled "Frame 70: 547 bytes on wire (4376 bits), 547 bytes captured (4376 bits) on interface wlx5091e3c3b759, id 0". The "Session Initiation Protocol (OPTIONS)" section is expanded, showing the Request-Line: "OPTIONS sip:user04.api@84.76.199.200:20004;transport=udp SIP/2.0", Method: "OPTIONS", Request-URI: "sip:user04.api@84.76.199.200:20004;transport=udp", and various message headers including Record-Route, Via, From, To, Call-ID, CSeq, Max-Forwards, and Content-Length.

Podem extreure la informació següent:

- **From**: sip:pinger@upcnet.es;tag=c3179b86

Aquest camp identifica l'usuari que envia el missatge. En aquest cas, l'usuari pinger del domini upcnet.es. El tag c3179b86 s'utilitza per identificar de manera única la sessió o diàleg SIP.

- **To** sip:user04.api@84.76.199.200:20004;transport=udp

Aquest camp identifica el destinatari del missatge. En aquest cas, l'usuari user04.api a l'adreça 84.76.199.200 al port 20004 utilitzant el protocol UDP.

- **Contact** sip:pinger@upcnet.es

Aquest camp proporciona una adreça de contacte per a respostes a aquest missatge. En aquest cas, l'adreça de contacte és pinger@upcnet.es.

- **Expires**

Aquest camp no apareix en el paquet 70. És típicament utilitzat en missatges de registre per especificar la durada durant la qual la informació de registre és vàlida.

I El paquet 71 sent la resposta del 70:

```
- Session Initiation Protocol (200)
  - Status-Line: SIP/2.0 200 Ok
    Status-Code: 200
    [Resent Packet: False]
    [Request Frame: 70]
    [Response Time (ms): 11]
  - Message Header
    - Via: SIP/2.0/UDP 147.83.194.150:5062;branch=z9hG4bK947a.413d32c42b0fc58f394dbe9154e87439.0
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5062
      Branch: z9hG4bK947a.413d32c42b0fc58f394dbe9154e87439.0
    - Via: SIP/2.0/UDP 147.83.194.150:5070;branch=z9hG4bK240db264
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5070
      Branch: z9hG4bK240db264
    - From: <sip:pinger@upcnet.es>;tag=c3179b86
      - SIP from address: sip:pinger@upcnet.es
        SIP from address User Part: pinger
        SIP from address Host Part: upcnet.es
        SIP from tag: c3179b86
    - To: <sip:user04.api@84.76.199.200:20004>;transport=udp;alias=84.76.199.200-20004-1;tag=V6VgV
      - SIP to address: sip:user04.api@84.76.199.200:20004
        SIP to address User Part: user04.api
        SIP to address Host Part: 84.76.199.200
        SIP to address Host Port: 20004
        SIP to tag: V6VgV
      Call-ID: b26dbdb4-240db264-f5201@147.83.194.150
      [Generated Call-ID: b26dbdb4-240db264-f5201@147.83.194.150]
    - CSeq: 1 OPTIONS
      Sequence Number: 1
      Method: OPTIONS
```

Podem extreure la informació següent:

- **From** sip:pinger@upcnet.es;tag=c3179b86

Aquest camp identifica l'usuari que envia el missatge de resposta. És el mateix que en el missatge d'OPCIONS original.

- **To** sip:user04.api@84.76.199.200:20004;transport=udp;  
alias=84.76.199.200-20004-1;tag=V06yV

Aquest camp identifica el destinatari de la resposta, que és el mateix usuari destinatari del missatge inicial, però amb un afegit alias=84.76.199.200-20004-1 (serà l'usuari 04.apo) que indica informació adicional per a la gestió de la connexió. El tag V06yV s'utilitza per identificar de manera única la sessió o diàleg SIP.

- **Contact**

Aquest camp no apareix en el paquet 71.

- **Expires**

Aquest camp no apareix en el paquet 71.

- e) **Busqueu si al registre SIP, com es pot veure l'adreça pública i el port públic del NAT a través del qual esteu sortint (i que és l'adreça i port amb els quals el proxy UPC us veu).**

Per determinar l'adreça pública i el port públic del NAT a través del qual esteu sortint, hem de buscar la informació específica en els paquets SIP que mostren les adreces IP i ports tal com els veu el proxy UPC.

Com l'exercici anterior, mirarem els paquets 70 i 71:

El paquet 70 ens dona la següent informació:

```
Session Initiation Protocol (OPTIONS)
  - Request-Line: OPTIONS sip:user04.api@84.76.199.200:20004;transport=udp SIP/2.0
    Method: OPTIONS
  - Request-URI: sip:user04.api@84.76.199.200:20004;transport=udp
    Request-URI User Part: user04.api
    Request-URI Host Part: 84.76.199.200
    Request-URI Host Port: 20004
    [Resent Packet: False]
  - Message Header
    - Record-Route: <sip:147.83.194.150:5062;lr;ftag=c3179b86>
      - Record-Route URI: sip:147.83.194.150:5062;lr;ftag=c3179b86
        Record-Route Host Part: 147.83.194.150
        Record-Route Host Port: 5062
        Record-Route URI parameter: lr
        Record-Route URI parameter: ftag=c3179b86
    - Via: SIP/2.0/UDP 147.83.194.150:5062;branch=z9hG4bK947a.413d32c42b0fc58f394dbe9154e87439.0
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5062
      Branch: z9hG4bK947a.413d32c42b0fc58f394dbe9154e87439.0
    - Via: SIP/2.0/UDP 147.83.194.150:5070;branch=z9hG4bK240db264
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5070
      Branch: z9hG4bK240db264
    - From: sip:pinger@upcnet.es;tag=c3179b86
      - SIP from address: sip:pinger@upcnet.es
        SIP from address User Part: pinger
        SIP from address Host Part: upcnet.es
        SIP from tag: c3179b86
    - To: sip:user04.api@84.76.199.200:20004;transport=udp;alias=84.76.199.200-20004~1
      - SIP to address: sip:user04.api@84.76.199.200:20004
        SIP to address User Part: user04.api
        SIP to address Host Part: 84.76.199.200
        SIP to address Host Port: 20004
      Call-ID: b26dbdb4-240db264-f5201@147.83.194.150
      [Generated Call-ID: b26dbdb4-240db264-f5201@147.83.194.150]
  - CSeq: 1 OPTIONS
    Sequence Number: 1
    Method: OPTIONS
    Max-Forwards: 69
    Content-Length: 0
```

- **Request-Line:** sip:user04.api@84.76.199.200:20004;transport=udp

Aquesta línia mostra la destinació de la sol·licitud OPTIONS, indicant que el missatge es dirigeix a l'usuari user04.api a l'adreça 84.76.199.200 i el port 20004 utilitzant el transport UDP.

- **Via Header:** Via: SIP/2.0/UDP

147.83.194.150:5062;branch=z9hG4bK471a.413d32c42b0fc58f394dbe9154e87439.0

Aquest camp mostra l'adreça IP i el port des d'on s'origina el missatge abans de la traducció NAT, en aquest cas, 147.83.194.150:5062.

- **Contact Header:** Contact: <sip:pinger@upcnet.es>

Aquest camp proporciona una adreça de contacte per a la resposta a aquest missatge.

I Pel paquet 71 de resposta ens dona:

```

- Session Initiation Protocol (200)
  - Status-Line: SIP/2.0 200 Ok
    Status-Code: 200
    [Resent Packet: False]
    [Request Frame: 70]
    [Response Time (ms): 11]
  - Message Header
    - Via: SIP/2.0/UDP 147.83.194.150:5062;branch=z9hG4bK947a.413d32c42b0fc58f394dbe9154e87439.0
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5062
      Branch: z9hG4bK947a.413d32c42b0fc58f394dbe9154e87439.0
    - Via: SIP/2.0/UDP 147.83.194.150:5070;branch=z9hG4bK240db264
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5070
      Branch: z9hG4bK240db264
    - From: <sip:pinger@upcnet.es>;tag=c3179b86
      SIP from address: sip:pinger@upcnet.es
      SIP from address User Part: pinger
      SIP from address Host Part: upcnet.es
      SIP from tag: c3179b86
    - To: <sip:user04.api@84.76.199.200:20004>;transport=udp;alias=84.76.199.200~20004~1;tag=V6VgV
      SIP to address: sip:user04.api@84.76.199.200:20004
      SIP to address User Part: user04.api
      SIP to address Host Part: 84.76.199.200
      SIP to address Host Port: 20004
      SIP to tag: V6VgV
      Call-ID: b26dbdb4-240db264-f5201@147.83.194.150
      [Generated Call-ID: b26dbdb4-240db264-f5201@147.83.194.150]
    - CSeq: 1 OPTIONS
      Sequence Number: 1
      Method: OPTIONS

```

- **Via Header:** Via: SIP/2.0/UDP  
147.83.194.150:5062;branch=z9hG4bK471a.413d32c42b0fc58f394dbe9154e87439.0

Novament, es mostra l'adreça IP interna i el port del client 147.83.194.150:5062.

- **To Header:** To:  
<sip:api@84.76.199.200:20004>;transport=udp;alias=84.76.199.200-20004-1;tag=V06yV

Aquest camp és crucial perquè inclou l'alias:  
alias=84.76.199.200-20004-1

Això indica l'adreça IP pública 84.76.199.200 i el port públic 20004 assignats pel NAT.

En analitzar els registres SIP dels paquets 70 i 71, hem pogut determinar l'adreça pública i el port públic del NAT a través dels quals esteu sortint. Aquesta informació es pot trobar en el camp To de la resposta SIP 200 OK (Paquet 71):

Adreça IP pública: 84.76.199.200  
Port públic: 20004

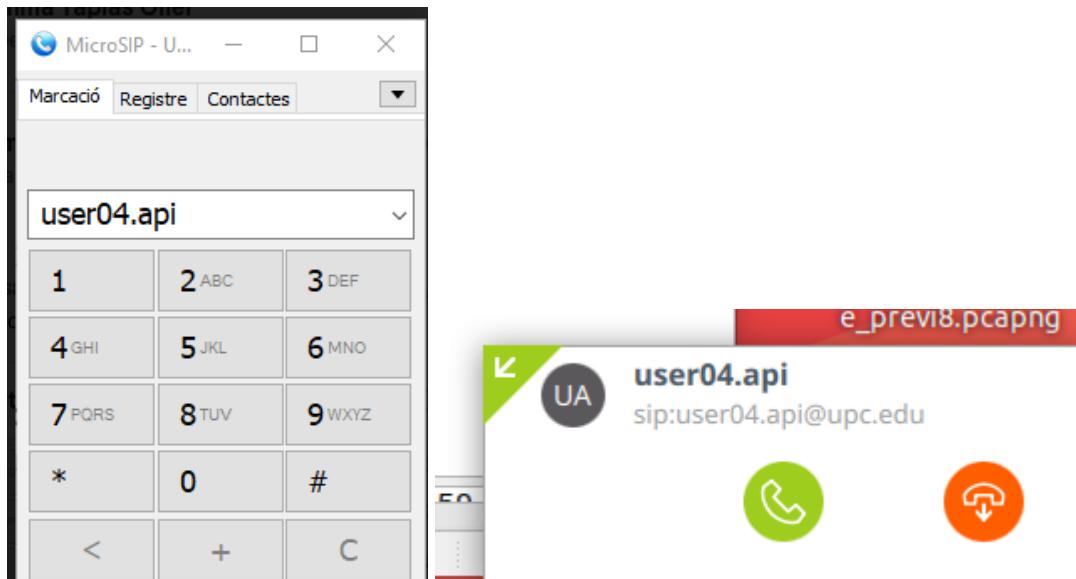
Aquestes dades confirmen com l'adreça IP interna 147.83.194.150 i el port 5062 són traduïts pel NAT. Així és com el proxy UPC us veu des de fora de la xarxa interna.

- f) Feu una trucada de prova, per comprovar que tot funciona correctament. Obteniu de Wireshark el diagrama del diàleg SIP entre cada terminal i els servidors d'UPCnet. Correleu els dos diàlegs per obtenir la visió completa de l'escenari, i dibuixeu una figura (similar a la Fig. 10, però no necessàriament igual) on es vegin tots els nodes involucrats (terminals, servidors, possibles NATs, etc).**

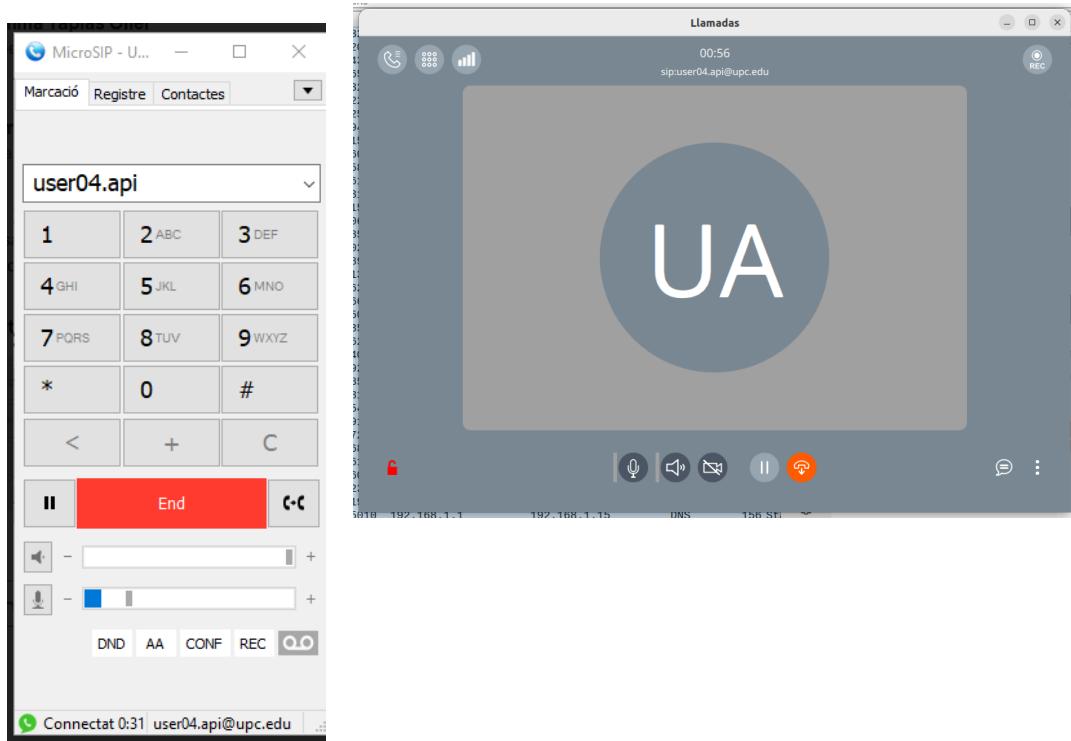
Des del Wireshark en l'apartat de trucades o Call Flow es poden veure els intercanvis de paquets entre les trucades.

Ara per fer una trucada de prova, s'ha de anar al MicroSIP i trucar al usuari04.api, aquest al estar configurat a l'altre màquina amb el Linphone, li hauria de sortir la trucada entrant i hauria de ser capaç de agafar-la:

Trucant des de MicroSIP i com es veu des del Linphone:



Amb la trucada agafada, des del MicroSIP i Linphone:



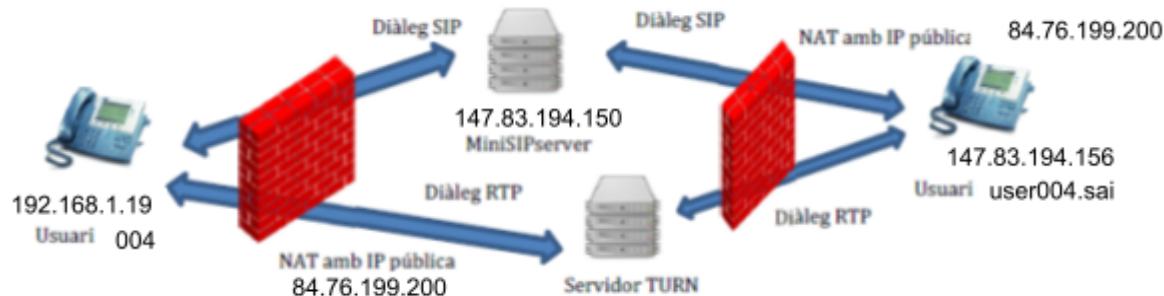
Captura des de la Màniga Linux amb Linphone:

Time	192.168.1.19	147.83.194.150	Comment
9.850511	59473 → INVITE SDP (g711A g711U telephone-event)	5062 ←	SIP INVITE From: "user04.api" <sip:user04.api@upc.edu> To: ...
9.855524	59473 ← 100 trying -- your call is important to us	5062 →	SIP Status 100 trying -- your call is important to us
9.855993	59473 ← 407 Proxy Authentication Required	5062 →	SIP Status 407 Proxy Authentication Required
9.856203	59473 → ACK	5062 ←	SIP ACK From: "user04.api" <sip:user04.api@upc.edu> To: ...
9.856451	59473 → INVITE SDP (g711A g711U telephone-event)	5062 ←	SIP INVITE From: "user04.api" <sip:user04.api@upc.edu> To: ...
9.861575	59473 ← 100 trying -- your call is important to us	5062 →	SIP Status 100 trying -- your call is important to us
9.871440	59473 ← INVITE SDP (g711A g711U telephone-event)	5062 →	SIP INVITE From: "user04.api" <sip:user04.api@upc.edu> To: ...
9.886235	59473 → 100 Trying	5062 ←	SIP Status 100 Trying
9.886544	59473 → 486 Call already exists	5062 ←	SIP Status 486 Call already exists
9.894447	59473 ← ACK	5062 →	SIP ACK From: "user04.api" <sip:user04.api@upc.edu> To: ...
10.264928	59473 ← 180 Ringing	5062 →	SIP Status 180 Ringing
13.562804	59473 ← 200 Ok SDP (g711A g711U telephone-event)	5062 →	SIP Status 200 Ok
13.598200	59473 → ACK	5062 ←	SIP Request INVITE ACK 200 CSeq:8630
13.598561	59473 → UPDATE SDP (g711A telephone-event)	5062 ←	SIP UPDATE From: "user04.api" <sip:user04.api@upc.edu> ...
13.761910	59473 ← 200 Ok SDP (g711A telephone-event)	5062 →	SIP Status 200 Ok
19.145677	59473 → BYE	5062 ←	SIP Request BYE CSeq:111
19.146090	59473 ← 200 OK	5062 →	SIP Status 200 OK

## Captura des de la Màquina Windows amb MicroSIP:

Intervalo	Origen	Destinació	Comentari
9.850511	59473	192.168.1.19	SIP INVITE From: "user04.api" <sip:user04.api@upc.e...
9.855524	59473	147.83.194.150	SIP Status 100 trying – your call is important to us
9.855993	59473	147.83.194.156	SIP Status 407 Proxy Authentication Required
9.856203	59473	5062	SIP ACK From: "user04.api" <sip:user04.api@upc.edu...
9.856451	59473	192.168.1.19	SIP INVITE From: "user04.api" <sip:user04.api@upc.e...
9.861575	59473	147.83.194.150	SIP Status 100 trying – your call is important to us
9.871440	59473	147.83.194.156	SIP INVITE From: "user04.api" <sip:user04.api@upc.e...
9.886235	59473	5062	SIP Status 100 Trying
9.886544	59473	192.168.1.19	SIP Status 486 Call already exists
9.894447	59473	147.83.194.150	SIP ACK From: "user04.api" <sip:user04.api@upc.edu...
10.264928	59473	5062	SIP Status 180 Ringing
13.562804	59473	192.168.1.19	SIP Status 200 Ok
13.572188	4008	147.83.194.150	RTP (g711A)
13.598200	59473	5062	SIP Request INVITE ACK 200 CSeq:8630
13.598561	59473	192.168.1.19	SIP UPDATE From: "user04.api" <sip:user04.api@upc.e...
13.674130	4008	147.83.194.150	RTP (g711A)
13.761910	59473	5062	SIP Status 200 Ok
13.772109	4008	147.83.194.150	RTP (g711A)
19.145677	59473	5062	SIP Request BYE CSeq:111
19.146090	59473	192.168.1.19	SIP Status 200 OK

El dibuix de l'esquema final seria:

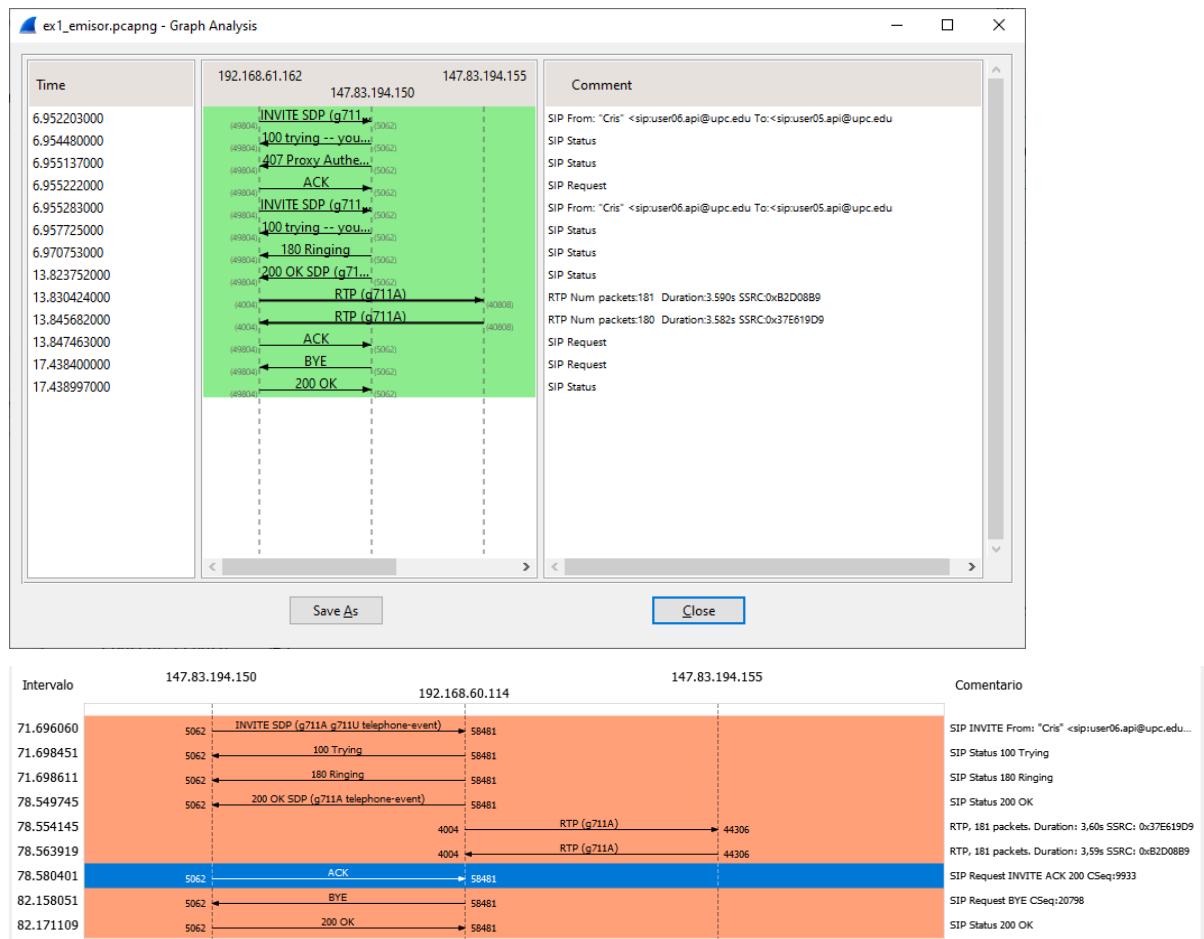


## EXERCICIS LABORATORI

### Exercici 1

**Comproveu als menús dels terminals quins còdecs d'àudio i vídeo estan disponibles. Invalideu els còdecs de vídeo. Truqueu des d'un dels terminals cap a l'altre, mantenint la trucada uns pocs segons i pengeu (des del terminal que no ha iniciat la trucada).**

- a) Feu una anàlisi automàtica (VoIP calls) de les dues trucades SIP (des del punt de vista de cada terminal) amb Wireshark. Captureu el diagrama de fletxes dels missatges intercanviats.



**a. S'està quedant el proxy SIP al mig de la trucada, inspeccionant i reenviant tots els missatges SIP? Per què? Com ho fa? (pista: analitzeu les capçaleres Contact enviades en el 200 OK del trucat i en l'ACK del que truca, i com el proxy les reescriu abans de reenviar els missatges cap a l'altre costat.**

192.168.60.114 (receptor) 192.168.61.162 (emisor)

Al principi, els missatges per el proxy i un cop comença la trucada els missatges passen per una cuarta 'máquina' (147.83.194.155).

Ho fa per temes de voler grabar la trucada, o per temes NAT.

Al principi: Tenim dos maquines la 192.168.60.114 i 192.168.61.162.

Quan el 192.168.61.162 truca al 192.168.60.114, els paquets van de 192.168.61.162 al proxy i del proxy al 192.168.60.114 (i viceversa)

Missatges RTP: tenim dues màquines la 192.168.60.114 i 192.168.61.162. Quan el 192.168.61.162 truca al 192.168.60.114, els paquets van de 147.83.194.155 a la quarta maquina i de la quarta maquina al 192.168.60.114 (i viceversa)

Per tant, el Proxy SIP s'està quedant al mig de la trucada, inspeccionant i reenviant tots els missatges SIP. Ho fa reescrivint les capçaleres Contact en els missatges 200 OK i ACK, de manera que els missatges semblen provenir del Proxy en lloc de directament entre els clients. Això assegura que la comunicació es mantingui a través del Proxy, permetent-li gestionar i monitoritzar la sessió SIP.

El Proxy força a que els fluxos RTP passin pel RTP-proxy reescrivint les capçaleres dels missatges SDP de l'INVITE i 200 OK durant l'establiment de la trucada.

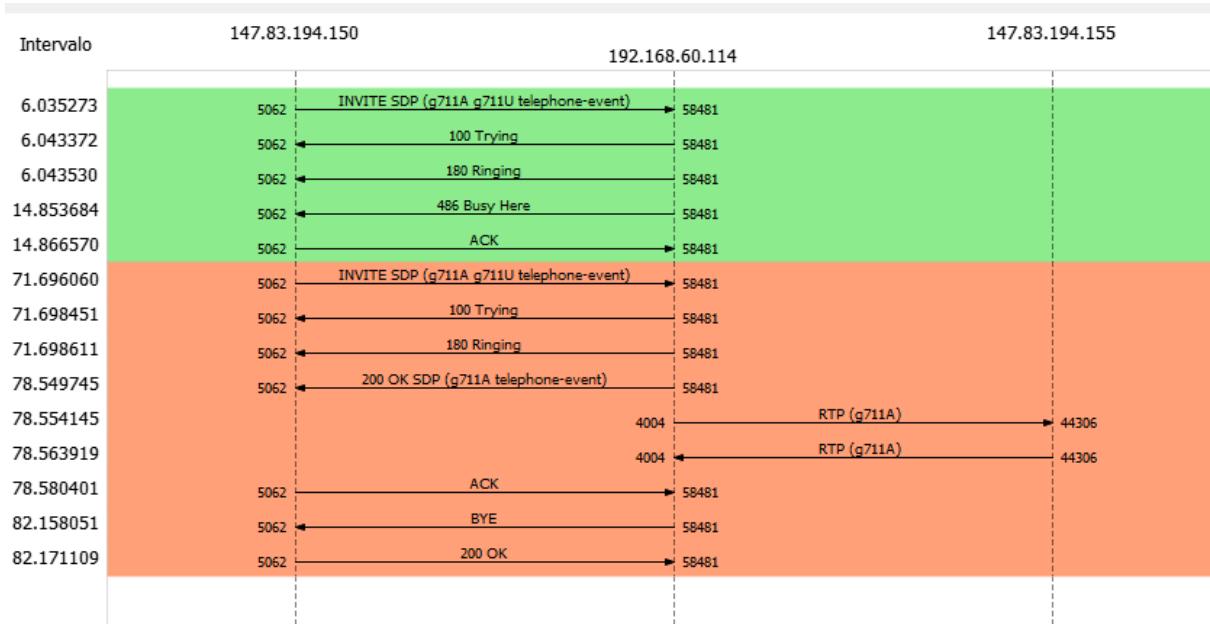
### b. Els fluxos RTP s'envien directament entre els terminals? De quina manera s'indica això als protocols de senyalització?

El fluxos RTP no s'envien directament ja que quan veiem el diagrama de flux veiem que passa per una quarta máquina (147.83.194.155), el proxy.

```
> Frame 1264: 233 bytes on wire (1864 bits), 233 bytes captured (1864 bits) on interface \Device\NPF_{32416FB2-CC44-475A-86D2-3E042FB0AC96}, id 0
> Ethernet II, Src: Fortinet_a7:b3:32 (00:09:0f:a7:b3:32), Dst: Micro-St_99:4f:c2 (00:d8:61:99:4f:c2)
> Internet Protocol Version 4, Src: 147.83.194.150, Dst: 192.168.60.114
> User Datagram Protocol, Src Port: 5062, Dst Port: 58481
< Session Initiation Protocol (INVITE)
  > Request-Line: INVITE sip:user05.api@192.168.60.114:58481;ob SIP/2.0
  > Message Header
  < Message Body
    < Session Description Protocol
      Session Description Protocol Version (v): 0
      > Owner/Creator, Session Id (o): - 3924673172 3924673172 IN IP4 147.83.194.155
      Session Name (s): pjmedia
      > Bandwidth Information (b): AS:84
      > Time Description, active time (t): 0 0
      > Session Attribute (a): X-nat:0
      > Media Description, name and address (m): audio 44306 RTP/AVP 8 0 101
      > Connection Information (c): IN IP4 147.83.194.155
      > Bandwidth Information (b): TIAS:64000
      > Media Attribute (a): rtpc:44307
      Media Attribute (a): sendrecv
```

### b) Analitzeu detalladament tot el diàleg SIP, fent especial èmfasi en:

#### a. Establiment de la trucada SIP: INVITE, 200 OK, ACK. Hi ha algun missatge inesperat?

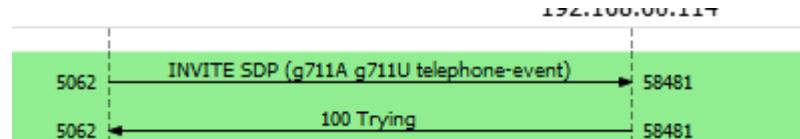


Apareix el missatge 486 Busy Here.

El missatge "486 Busy Here" és una resposta del Protocol d'Inici de Sessió que indica que el destí al qual es va intentar trucar està ocupat i no pot acceptar la trucada en aquell moment.

### b. Analitzeu els SDP intercanviats (als attachments de l'INVITE i del 200 OK):

#### i. Quins còdecs s'ofereixen? Quins paràmetres addicionals tenen (i per què)?



Els codecs són G711A, G711U i telephone-event.

El paràmetre adicional que tienen és el 8000 que indica la freqüència de rellotge que està associat al còdec.

A més el fmtp:101 0-15, són importants per especificar el rang d'esdeveniments DTMF que es poden enviar.

- ▼ Media Description, name and address (m): audio 40808 RTP/AVP 8 101
  - Media Type: audio
  - Media Port: 40808
  - Media Protocol: RTP/AVP
  - Media Format: ITU-T G.711 PCMA
  - Media Format: DynamicRTP-Type-101
- > Connection Information (c): IN IP4 147.83.194.155
- > Bandwidth Information (b): TIAS:64000
- > Media Attribute (a): rtcp:40809
- > Media Attribute (a): sendrecv
- > Media Attribute (a): rtpmap:8 PCMA/8000
- > Media Attribute (a): rtpmap:101 telephone-event/8000

## **ii. Com es posen d'acord els dos terminals en el còdec?**

Els dos terminals es posen d'acord en el còdec seguint aquest procés:

El Terminal A envia una llista de còdecs suportats en l'SDP del missatge INVITE.

El Terminal B selecciona un dels còdecs compatibles i envia aquesta selecció en l'SDP del missatge 200 OK.

El Terminal A confirma la selecció amb un missatge ACK, completant així la negociació del còdec.

## **iii. Confirmeu que els paràmetres dels SDP es reflecteixen als fluxos RTP: còdec escollit, port, altres paràmetres.**

```
User Datagram Protocol, Src Port: 4004, Dst Port: 40808
Real-Time Transport Protocol
  > [Stream setup by SDP (frame 142)]
    10... .... = Version: RFC 1889 Version (2)
    ..0. .... = Padding: False
    ...0 .... = Extension: False
    .... 0000 = Contributing source identifiers count: 0
    1.... .... = Marker: True
    Payload type: ITU-T G.711 PCMA (8)

  ▼ Media Description, name and address (m): audio 40808 RTP/AVP 8 101
    Media Type: audio
    Media Port: 40808
    Media Protocol: RTP/AVP
    Media Format: ITU-T G.711 PCMA
    Media Format: DynamicRTP-Type-101
  > Connection Information (c): IN IP4 147.83.194.155
  > Bandwidth Information (b): TIAS:64000
  > Media Attribute (a): rtcp:40809
    Media Attribute (a): sendrecv
  > Media Attribute (a): rtpmap:8 PCMA/8000
  > Media Attribute (a): rtpmap:101 telephone-event/8000
  > Media Attribute (a): fmtp:101 0-16
  > Media Attribute (a): ssrc:937826777 cname:50644d5439ce3bb1
  > " " " " "
```

Com podem veure tant en la captura del RTP com la del SDP podem confirmar el codec escollit (payload) es el mateix:

PCMA = ITU-T G.711 i amb una freqüència de mostreig de 8000 Hz i el port (40809)

## **c. Missatges “informacionals” de la sèrie 1xx. Quins apareixen, quina tasca fan?**

Tenim el 100 Trying i el 180 Rigning.

El missatge "100 Trying" indica que el servidor ha rebut la sol·licitud de trucada i la processa.

El missatge "180 Ringing" indica que el servidor ha començat a alertar l'usuari destí sobre una trucada entrant.

**d. Els paper de Cseq i del from: i el to:. Ha de quedar clar quins són els parells petició/resposta durant la trucada, i com es relacionen a través d'aquests paràmetres.**

Cseq: número de seqüència del paquet i el mètode SIP(Invite, ACK)

From: host origen de la trucada

To: el host destinatari de la trucada

El from y el to no es modifiquen en cap moment de la trucada. El Cseq canvia després de Invite, Bye o ACK.

**e. Finalització de la trucada: BYE i 200 OK. tenint en compte que ha penjat el que NO ha iniciat la trucada, què passa amb el Cseq, comparat amb els missatges anteriors?**

El Cseq no es incremental, comparant amb els missatges anteriors observem que es un Cseq diferent.

**c) Compareu el missatge INVITE que va del terminal que truca cap al proxy, i l'INVITE que el proxy reenvia cap al terminal trucat.**

**Específicament, mireu què passa amb la capçalera Contact, i amb els SDP (adreça IP i còdecs). Repetiu l'anàlisi comparant els 200 OK que van en sentit contrari.**

**Comenteu les vostres conclusions sobre què està fent el proxy, i per què.**

Contact:

- › Contact: "Cris" <sip:user06.api@192.168.61.162:49804;ob;alias=147.83.13.68~49804~1>  
Call-ID: c8eefe7af1d04fed956cec0719aaa3d1  
[Generated Call-ID: c8eefe7af1d04fed956cec0719aaa3d1]
- › Contact: "Cris" <sip:user06.api@192.168.61.162:49804;ob;alias=147.83.13.68~49804~1>  
Call-ID: 8b72999f3e154e72a3952cb75f9fa8ba  
[Generated Call-ID: 8b72999f3e154e72a3952cb75f9fa8ba]

El contact es el mateix, pero en el cas del Owner canvia:

Owner/Creator, Session Id (o): - 3924673172 3924673172 IN IP4 147.83.194.155

Owner Username: -

Session ID: 3924673172

Session Version: 3924673172

Owner Network Type: IN

Owner Address Type: IP4

Owner Address: 147.83.194.155

```
✓ Owner/Creator, Session Id (o): - 3924673106 3924673106 IN IP4 147.83.194.156
  Owner Username: -
  Session ID: 3924673106
  Session Version: 3924673106
  Owner Network Type: IN
  Owner Address Type: IP4
  Owner Address: 147.83.194.156
```

200 OK:

```
> Contact: "jiale.chen" <sip:user05.api@192.168.60.114:58481;ob>
  Supported: replaces, 100rel, timer, norefersub
  Content-Type: application/sdp
  Content-Length: 321
  ✓ Message Body
    ✓ Session Description Protocol
      Session Description Protocol Version (v): 0
      ✓ Owner/Creator, Session Id (o): - 3924673172 3924673173 IN IP4 192.168.60.114
        Owner Username: -
        Session ID: 3924673172
        Session Version: 3924673173
        Owner Network Type: IN
        Owner Address Type: IP4
        Owner Address: 192.168.60.114
        Session Name (s): pjmedia
```

Els missatges s'envien al proxy per a que aquest pugui administrar la trucada.

**d) Hi ha algun indicí que us indiqui si hi ha algun NAT entre el client i el Proxy? Si és així, comenteu-lo i identifiqueu amb quina IP pública i quin port us veu el client.**

```
✓ Session Initiation Protocol (200)
  ▶ Status-Line: SIP/2.0 200 OK
  ✓ Message Header
    ▶ Via: SIP/2.0/UDP 147.83.194.150:5062;received=147.83.194.150
      Transport: UDP
      Sent-by Address: 147.83.194.150
      Sent-by port: 5062
      Received: 147.83.194.150
```

Com podem veure aquí hi ha un indicí de NAT amb la IP pública 147.83.194.150 i el port 5062.

**e) Analitzeu el paper dels missatges OPTIONS. Són periòdics? Per a què poden servir?**

Els missatges OPTIONS mantenen obert el port del NAT, que es tanca automàticament després d'un temps d'inactivitat.

f) Feu un esquema global on es vegi totes les entitats (telèfons, servidors, NATs etc) involucrades en la trucada. Identifiqueu adreces IP, usuaris registrats, qui parla SIP amb qui, qui envia flux RTP cap a qui, etc. Exemple (modifiqueu-lo, no presuposeu que l'escenari és així):

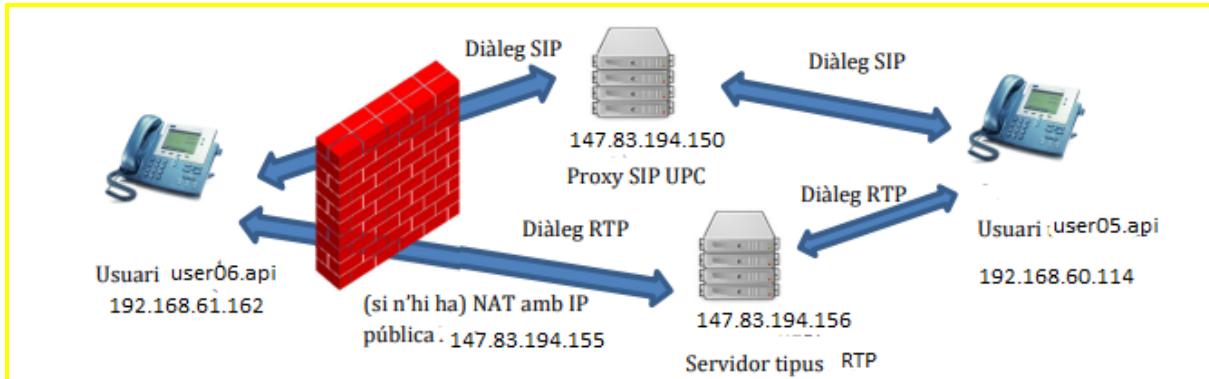


Figura 10. Plantilla (no necessàriament és així) de l'escenari de l'exercici 1.

## Exercici 2

**Forceu (canviant l'ordre o eliminant còdecs) almenys dos còdecs diferents als G.711 PCMA i PCMU i almenys un amb rellotge RTP different a 8 KHz (per exemple: AMR-WB a 16 KHz, Opus, Speex a 16 KHz, etc) i analitzeu els fluxos RTP.**

**a) Per a cadascun dels fluxos RTP, a partir de la diferència de timestamps entre dos paquets consecutius, calculeu el temps d'àudio que transporta cada paquet RTP.**

Captura SILK\_8:

Amb un Timestamps de 160 i amb 8000 de freqüència de rellotge.

Temps audio =  $160 / 8000 = 20 \text{ ms}$

Per tant, cada paquet conté 20 ms d'àudio.

Captura AMR-WB\_16:

Amb un Timestamps de 640 i un codec AMR-WB amb 16 KHz

Temps audio =  $640 \text{ ticks} * 1\text{s}/16000 \text{ ticks} = 0.04 \text{ s}$

Per tant, cada paquet conté 0.04 s d'àudio.

**b) Relacioneu i justifiqueu numèricament:**

- 1) el temps d'àudio transportat a cada paquet RTP,**
- 2) la mida del paquet RTP en bytes,**
- 3) el bitrate del còdec**

Captura SILK\_8:

El temps transportat es de 20 ms(8khz);

La mida del paquet RTP es de 214 bytes(8khz)

El bitrate = 64 Kbit/s(8khz);

Per tant,  $64\text{kbit/s} \times 20\text{ms} = 1280 \text{ bits} / 8 = 160 \text{ bytes d'àudio}$

160 bytes + 12 bytes RTP + 8 bytes UDP + 20

bytes IP + 14 bytes Ethernet = 214 bytes per cada trama Ethernet.

Captura AMR-WB\_16:

El temps transportat es de 40 ms(16khz)

La mida del paquet RTP es de 137 bytes(16khz)

El bitrate = 16.6 Kbit/s(16khz)

Per tant,  $16.6 \text{ Kbit/s} \times 40\text{ms} = 664 \text{ bits} / 8 = 83 \text{ bytes d'àudio}$

83 bytes + 12 bytes RTP + 8 bytes UDP + 20

bytes IP + 14 bytes Ethernet = 137 bytes per cada trama Ethernet.

**c) Analitzeu amb Wireshark (figura 3) la qualitat de servei (pèrdues, jitter) de cada flux RTP i comenteu-la.**

Stream	Paquete	Sequence	Delta (ms)	Jitter (ms)	Skew	Ancho de banda	Marker	Estado
147.83.194.155:26258 →	191	16828	0.000000	0.000000	0.000000	0.38	*	✓
192.168.60.114:4000	193	16829	20.186000	0.011625	-0.186000	0.76		✓
	204	16830	20.311000	0.030336	-0.497000	1.32		✓
<b>SSRC</b> 0x729b45a7	214	16831	20.439000	0.055877	-0.936000	1.93		✓
<b>Max Delta</b> 36.486000 ms @ 529	216	16832	20.381000	0.076198	-1.317000	2.52		✓
<b>Max Jitter</b> 7.615881 ms	219	16833	20.384000	0.095435	-1.701000	3.11		✓
<b>Mean Jitter</b> 3.798280 ms	221	16834	20.460000	0.118221	-2.161000	3.72		✓
<b>Max Skew</b> -18.125000 ms	222	16835	20.593000	0.147894	-2.754000	4.34		✓
<b>RTP Packets</b> 457	224	16836	10.191000	0.751713	7.055000	4.94		✓
<b>Expected</b> 457	228	16837	31.216000	1.405731	-4.161000	5.54		✓
<b>Lost</b> 0 (0.00 %)	231	16838	9.994000	1.943248	5.845000	6.11		✓
<b>Seq Errs</b> 0	233	16839	15.626000	2.095170	10.219000	6.69		✓
<b>Start at</b> 6.439434 s @ 191	236	16840	30.562000	2.624347	-0.343000	7.27		✓
<b>Duration</b> 9.12 s	237	16841	20.739000	2.506513	-1.082000	7.88		✓
<b>Clock Drift</b> 0 ms	244	16842	20.732000	2.395606	-1.814000	8.48		✓
<b>Freq Drift</b> 0 Hz (0.00 %)	245	16843	20.538000	2.279505	-2.352000	9.06		✓
	248	16844	20.508000	2.168786	-2.860000	9.64		✓
	249	16845	20.258000	2.049362	-3.118000	10.17		✓

No s'observen perdudes. La variació del delta time es bastant gran passa de 10 a 30 ms, però la mitjana és de 20 ms aproximadament. El valor del Jitter es molt mes petit que el Delta per tant té una bona qualitat de servei, i en el apartat de Lost veiem que hi ha una pèrdua del 0%.

## Exercici 3

**A les captures anteriors, analitzeu els missatges RTCP, si n'hi ha.**

**a) De quin tipus són? En quin sentit s'envien?**

Missatges son Sender Report i Receiver Report.

El Sender Report va de l'emisor 192.168.61.162 al proxy 147.83.194.156, i també a l'inrevés, és a dir del proxy a l'emisor i també passa el mateix pel cas del Receiver Report.

El Sender Report informa sobre la transmissió dels paquets RTP i el Receiver Report informa sobre la recepció dels paquets RTP.

Aquest dos missatges són importants per gestionar i controlar la qualitat de la transmissió en les comunicacions RTP

No.	rtcp	me	Source	Destination	Protocol	Length	Info	Delta
168	8.645633		192.168.61.162	147.83.194.156	RTCP	102	Receiver Report	So...
169	8.645733		192.168.61.162	147.83.194.156	RTCP	102	Receiver Report	So...
222	9.061654		192.168.61.162	147.83.194.156	RTCP	122	Sender Report	Sour...
224	9.076729		147.83.194.156	192.168.61.162	RTCP	122	Sender Report	Sour...
533	13.267874		192.168.61.162	147.83.194.156	RTCP	122	Sender Report	Sour...
550	13.501751		147.83.194.156	192.168.61.162	RTCP	122	Sender Report	Sour...
656	15.075036		147.83.194.156	192.168.61.162	RTCP	130	Sender Report	Sour...
657	15.075037		147.83.194.156	192.168.61.162	RTCP	110	Receiver Report	So...
660	15.081515		192.168.61.162	147.83.194.156	RTCP	130	Sender Report	Sour...

**b) Analitzeu els paràmetres relacionats amb la qualitat de servei.**

Pel cas del Receiver veiem que no hi ha Padding. Pel cas del jitter durant els primers 13 s veiem que era casi nul, pero a partir del segon 15 el jitter incrementa cap a 90 . En quan el delay ronda cap 1797 ms. Per tant hi ha bona qualitat de servei

```
Real-time Transport Control Protocol (Receiver Report)
> [Stream setup by SDP (frame 59)]
10... .... = Version: RFC 1889 Version (2)
...0. .... = Padding: False
...0 0001 = Reception report count: 1
Packet type: Receiver Report (201)
Length: 7 (32 bytes)
Sender SSRC: 0x783d1982 (2017270146)
> Source 1
```

Pel cas Sender tampoc hi ha padding. Pel cas del jitter durant els primers 13 s veiem que era casi nul, pero a partir del segon 13 el jitter incrementa cap a 80 (mitjana). En quan el delay ronda cap 14987 ms, pero varia bastant. Per tant hi ha bona qualitat de servei.

```
Sender's packet count: 204
Sender's octet count: 12418
- Source 1
  Identifier: 0x729b45a7 (1922778535)
  - SSRC contents
    Fraction lost: 0 / 256
    Cumulative number of packets lost: 0
  - Extended highest sequence number received: 17061
    Sequence number cycles count: 0
    Highest sequence number received: 17061
  Interarrival jitter: 38
  Last SR timestamp: 3322583253 (0xc60a98d5)
  Delay since last SR timestamp: 4748 (72 milliseconds)
```

---

En la anterior imatge podem observar dades com per exemple els paquets perduts o el jitter.

## Exercici 4

**Ara veurem alguns dels missatges que gestionen accions (o inaccions) dels usuaris.**

**a) Torneu a fer la trucada, però ara refuseu-la al receptor.**

**a. Quin missatge us ho indica? De quina sèrie és? Quins paràmetres té?**

El paquet que indica que hem refusat es el següent:

253	12.169879	192.168.60.114	147.83.194.150	SIP	1185 Status: 486 Busy Here
-----	-----------	----------------	----------------	-----	----------------------------

Amb el missatge Busy Here i la sèrie 486.

Si observem les trucades VoIP del receptor podem veure que ha sigut refusada.

Start Time	Stop Time	Interlocutor inicial	Desde	A	Protocolo	Duración	Paquetes	Estado	Comentarios
4.242385	12.171549	147.83.194.150	"Cris" <sip:user06.api@upc.edu>	<sip:user05.api@upc.edu>	SIP	00:00:07	5	REJECTED	INVITE 486

**b) Torneu a fer la trucada, però no despengeu i espereu a que expiri el temporitzador.**

**a. Quin missatge us ho indica? De quina sèrie és? Quins paràmetres té?**

En la captura del receptor observem que s'ha fet diversos 200 OK (OPTIONS) i després fa un Request: CANCEL seguida de un 200 OK (CANCEL) i finalment termina amb un 487 Request Terminated acabant així la trucada.

528	23.596930	147.83.194.150	192.168.60.114	SIP	437 Request: CANCEL sip:user05.api@192.168.60.114:63254;ob
529	23.597111	192.168.60.114	147.83.194.150	SIP	402 Status: 200 OK (CANCEL)
530	23.597150	192.168.60.114	147.83.194.150	SIP	1194 Status: 487 Request Terminated

Si anem a les trucades VoIP del receptor s'observa que s'ha cancel·lat.

Start Time	Stop Time	Interlocutor inicial	Desde	A	Protocolo	Duración	Paquetes	Estado	Comentarios
3.298122	23.604259	147.83.194.150	"Cris" <sip:user06.api@upc.edu>	<sip:user05.api@upc.edu>	SIP	00:00:20	7	CANCELLED	INVITE 200 487

**b. Quant ha trigat en expirar? La durada apareix en algun paràmetre? Qui la decideix?**

La durada de la trucada la decideix el servidor que es el que controla la trucada.

El temps que ha trigat a expirar la podem veure en la captura del receptor en trucades VoIP.

Start Time	Stop Time	Interlocutor inicial	Desde	A	Protocolo	Duración	Paquetes	Estado	Comentarios
3.298122	23.604259	147.83.194.150	"Cris" <sip:user06.api@upc.edu>	<sip:user05.api@upc.edu>	SIP	00:00:20	7	CANCELLED	INVITE 200 487

El temps que triga a expirar es de 20 segons.

**c) Torneu a fer la trucada, però abans de despenjar, cancel·leu-la per part del que truca.**

**a. Quin missatge us ho indica?**

**b. De quina sèrie és? Quins paràmetres té?**

En aquest cas es el emisor el que envia un request per cancelar la trucada, després continua amb un 200 canceling (CANCEL) i finalment s'acaba amb un 487 Request Terminated acabant així la trucada.

120 7.522633	192.168.61.162	147.83.194.150	SIP	453 Request: CANCEL sip:user05.api@upc.edu
121 7.525497	147.83.194.150	192.168.61.162	SIP	460 Status: 200 canceling (CANCEL)
122 7.531334	147.83.194.150	192.168.61.162	SIP	850 Status: 487 Request Terminated

I en la captura del receptor podem observar que accepta el request de cancelar la trucada amb un 200 OK (CANCEL).

139 8.118190	147.83.194.150	192.168.60.114	SIP	445 Request: CANCEL sip:user05.api@192.168.60.114:63254;ob
140 8.118551	192.168.60.114	147.83.194.150	SIP	402 Status: 200 OK (CANCEL)
141 8.118661	192.168.60.114	147.83.194.150	SIP	1194 Status: 487 Request Terminated

Finalment, acabem de confirmar la cancelació mirant les trucades VoIP de l'emissor i del receptor respectivament.

Start Time	Stop Time	Interlocutor inicial	Desde	A	Protocolo	Duración	Paquetes	Estado	Comentarios
3.237290	7.531487	192.168.61.162	"Cris" <sip:user06.api@upc.edu> <sip:user05.api@upc.edu>	SIP		00:00:04	11	CANCELLED	INVITE 407 200 487
3.848285	8.136291	147.83.194.150	"Cris" <sip:user06.api@upc.edu> <sip:user05.api@upc.edu>	SIP		00:00:04	7	CANCELLED	INVITE 200 487

**d) Durant una trucada, genereu tons DTMF (tecles numèriques) i analitzeu els paquets RTP. Veieu els RTP EVENT? Analitzeu-los i descriviu els seus paràmetres, específicament el primer i els últims RTP EVENT de cada cop que piqueu un número. Per què es repeteix idènticament tres vegades l'últim paquet de cada event?**

Se'ns ha perdut la captura d'aquest apartat per tant no podem afegir captures, pero sabem que el que succeeix és el següent:

Al teclejar rebem i enviem diversos events RTP que son identificats per el número de secuencia.

En el primer event si analitzem els paràmetres el booleà de final de event será False, pero en canvi el últim paquet será un True i estarà repetit 3 vegades.

S'envien tres paquets per assegurar que arriba el missatge al destinatari, evitant així problemes si es perd algun paquet per al camí.

## Exercici 5

**Configureu cadascun dels dos terminals de manera que els còdecs d'àudio que accepten siguin completament diferents. Què succeeix?**

Hem configurat el codec del receptor amb un SILK de 8 khz i el codec del emisor amb AMR-WB de 16 khz.

En el moment que es fa la trucada ens mostra un pop up amb un missatge indicant que no s'ha acceptat, dos codecs diferents.

En la captura del receptor s'observa un paquet amb el missatge 488 Not acceptable here.

98 5.936677 192.168.60.114 147.83.194.150 SIP 1210 Status: 488 Not Acceptable Here |

I si indaguem una mica en el paquet podem veure un missatge que indica que no hi ha ningú códec adequat per la trucada.

▼ CSeq: 17953 INVITE  
Sequence Number: 17953  
Method: INVITE  
Warning: 399 AUL-1976 "No suitable codec for remote offer (PJMEDIA\_SDPNEG\_NOANS\_CODEC)"

L'estat del receptor i del emisor es de rejected com podem observar:

Start Time	Stop Time	Interlocutor inicial	Desde	A	Protocolo	Duración	Paquetes	Estado	Comentarios
5.935270	5.938651	147.83.194.150	"Cris" <sip:user06.api@upc.edu> <sip:user05.api@upc.edu>	SIP		00:00:00	3	REJECTED	INVITE 488

Start Time	Stop Time	Interlocutor inicial	Desde	A	Protocolo	Duración	Paquetes	Estado	Comentarios
5.032207	5.054294	192.168.61.162	"Cris" <sip:user06.api@upc.edu> <sip:user05.api@upc.edu>	SIP		00:00:00	8	REJECTED	INVITE 407 486

## Exercici 6

**Torneu a fer la trucada, però connectant prèviament les webcams, i configurant els còdecs d'àudio i vídeo de manera que els dos terminals siguin compatibles.**

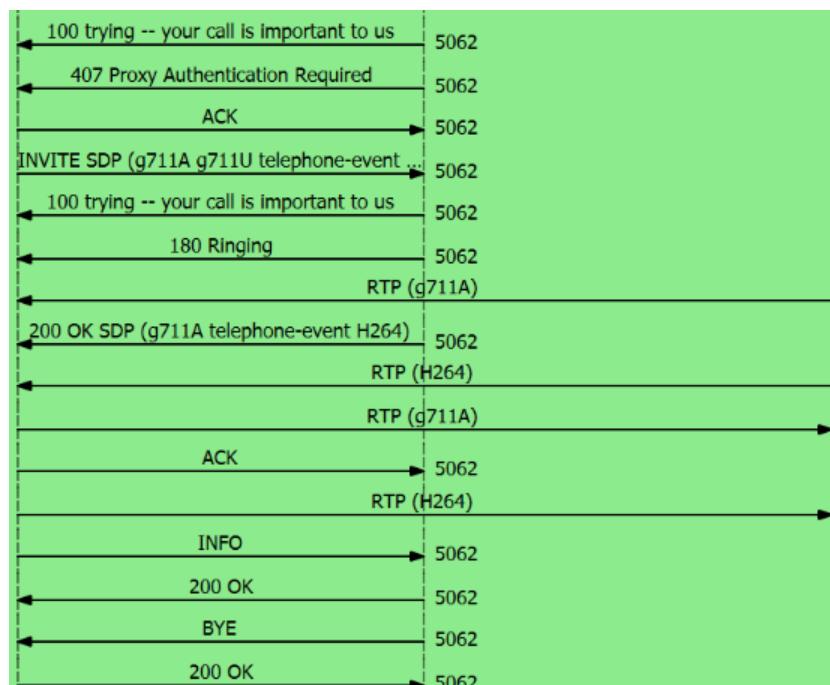
Per a utilitzar la càmera s'ha de configurar en el microsip.



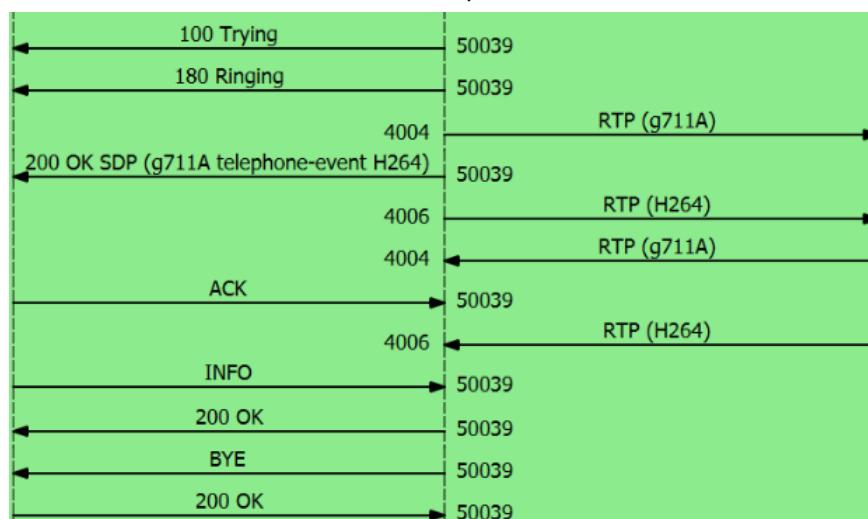
**a) Com canvia la negociació de còdecs SDP? Quins còdecs s'escullen, i com es reflecteix a SDP?**

Aquesta és la comunicació entre l'emissor i el receptor.

Flux de secuencia de l'Emissor:



Flux de secuencia de el Receptor:



Podem observar en els fluxos que s'envien dos tipus de paquets RTP per a vídeo (H264) i per a àudio (g711A).

També sabem que els codecs que s'utilitzen per àudio són el G.711 PCMA i el G.711 PCMU. I per a vídeo s'utilitzen H264, H263, VP8 i VP9.

**b) Quants fluxos RTP s'estableixen? Justifiqueu-ho.**

S'estableixen dos fluxos RTP, un per a vídeo i altre per a àudio.

En una videotrucada es fan servir fluxos separats per al vídeo i l'àudio perquè el vídeo necessita més ample de banda que l'àudio i separar-los permet optimitzar la compressió i l'ús de codecs específics, també es poden aplicar tècniques de sincronització més avançades i prioritzar l'àudio per reduir la latència i separar els fluxos facilita la compatibilitat amb diversos dispositius i escalabilitat.

## Exercici 7

**Configureu almenys dos terminals amb el mateix usuari SIP (diguem-li usuari A), i un tercer terminal amb l'altre usuari SIP (usuari B).**

**a) Truqueu des de l'usuari B cap a l'A. Què passa en els terminals configurats amb l'usuari A?**

Quan el usuari B feia la trucada, als usuaris A els sonava el telèfon fins que un respongui o fins que acabi el temps de uns 20 segons.

Si ningú dels dos usuaris B respon l'estat dels dos es CANCELLED, però en el cas de que un respongui canvia l'estat a COMPLETED i l'altre a REJECTED.

**b) Despenseu només un dels terminals configurats amb l'usuari A. Què passa en els altres? Féu un diagrama dels missatges intercanviats entre tots els actors (terminals A, terminal B, proxy).**

Si despenguem un dels terminals del usuari A l'altre es despenja automàticament.

En la següent captura observem com el terminal de l'usuari A que despenja envia un request BYE i seguidament passa a Status: 200 OK (BYE), confirmant la finalització de la trucada.

1112 17.149577	192.168.61.162	147.83.194.150	SIP	792 Request: BYE sip:user05.api@192.168.60.114:63254;ob;alias=147.83.13.68~63254~1
1114 17.160245	147.83.194.150	192.168.61.162	SIP	410 Status: 200 OK (BYE)

No tenim la captura de l'usuari A que finaliza automàticament després de que l'usuari A ha despenjat, però tindria que sortir un missatge de Request: CANCEL seguida de un Status: 200 OK (CANCEL).

Per fer el diagrama s'ha de tenir en compte que les IP's són els següents:

IP de Usuari A que despenja: 192.168.61.162

IP de Usuari A que no despenja: 192.168.61.163

IP de Usuari B: 192.168.60.114

IP del servidor: 147.83.194.150

Diagrama en text:

192.168.61.162 envia un request BYE a 147.83.194.150 i 147.83.194.150 li retorna un OK (BYE) a 192.168.61.162

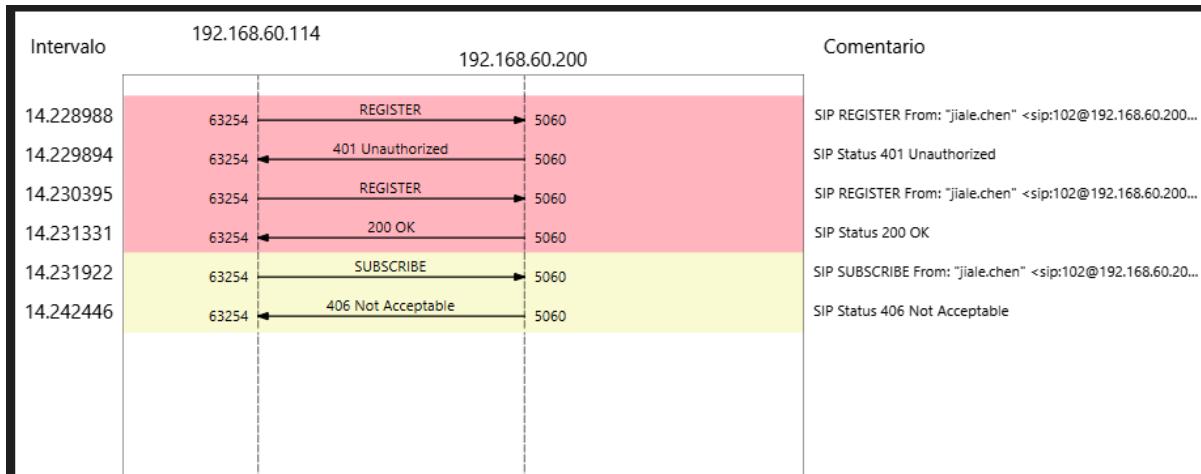
147.83.194.150 envia un request CANCEL a 192.168.61.163 i li retorna un OK (CANCEL) a 147.83.194.150

147.83.194.150 envia un request BYE a 192.168.60.114 i 192.168.60.114 li retorna un OK (BYE) a 147.83.194.150

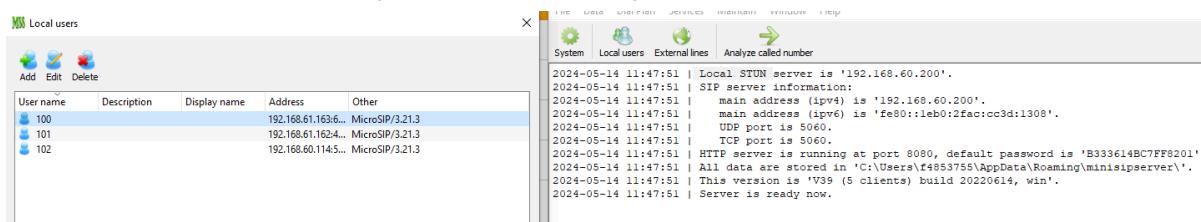
## Exercici 8

**Assegureu-vos que a un dels PC no corri cap client SIP (feu-lo acabar de manera ordenada, i si és necessari, mateu el procés). Arrenqueu el servidor MyVoIPapp MiniSIPserver.**

**a) Configureu els tres terminals per a que es registrin cap al MiniSIPserver, en comptes del Proxy UPC (en comptes de domini, heu d'apuntar contra l'adreça IP del servidor), i analitzeu el registre.**



Aquí podem veure l'intent de registre al servidor per part de la màquina 192.168.60.114. Els missatges van desde la màquina cap al servidor, que té l'IP 192.168.60.200, com podem veure aquí:



Amb aquesta imatge també podem veure, a la part dreta, que la màquina que ha intentat fer el registre, té l'usuari 102.

La seqüència que veiem en la primera imatge vol dir el següent:

- El client fa una petició al servidor per a registrar-se amb un missatge REGISTER
- El servidor contesta, rebutjant la sol·licitud de registre inicial, per que requereix d'autenticació, amb un missatge Status: 401 Unauthorized
- El client torna a fer una petició de registre al servidor amb una altre missatge de REGISTER.
- El servidor accepta el registre i envia un missatge informant del èxit del registre, enviant un missatge Status: 200 OK (REGISTER).

- El següent missatge del client cap al servidor és un SUBSCRIBE, on el client intenta subscriures a events o notificacions del servidor.
- El servidor contesta amb un missatge Status: 406 Not Acceptable, que indica que la sol·licitud de subscripció ha sigut rebutjada. Aquesta sol·licitud és rebutjada perquè el servidor no ha estat configurat per a acceptar notificacions o events.

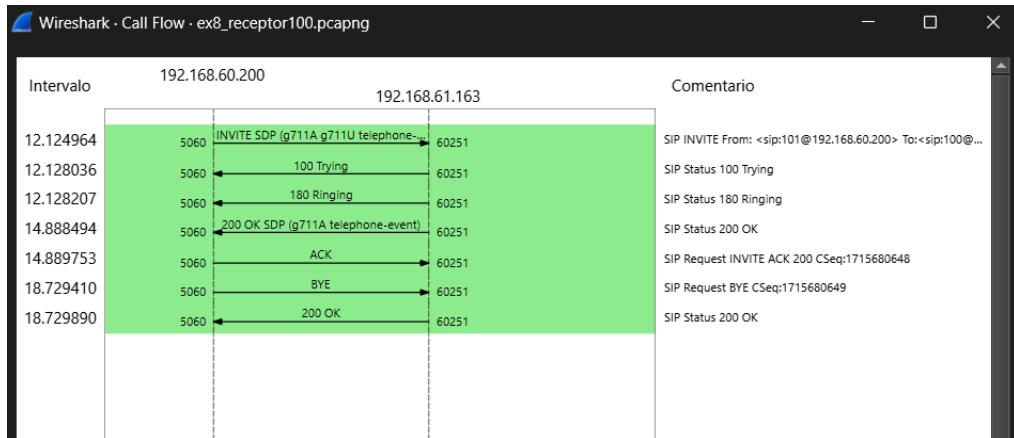
Aquesta és la configuració del client que s'ha connectat al servidor:

Cuenta X

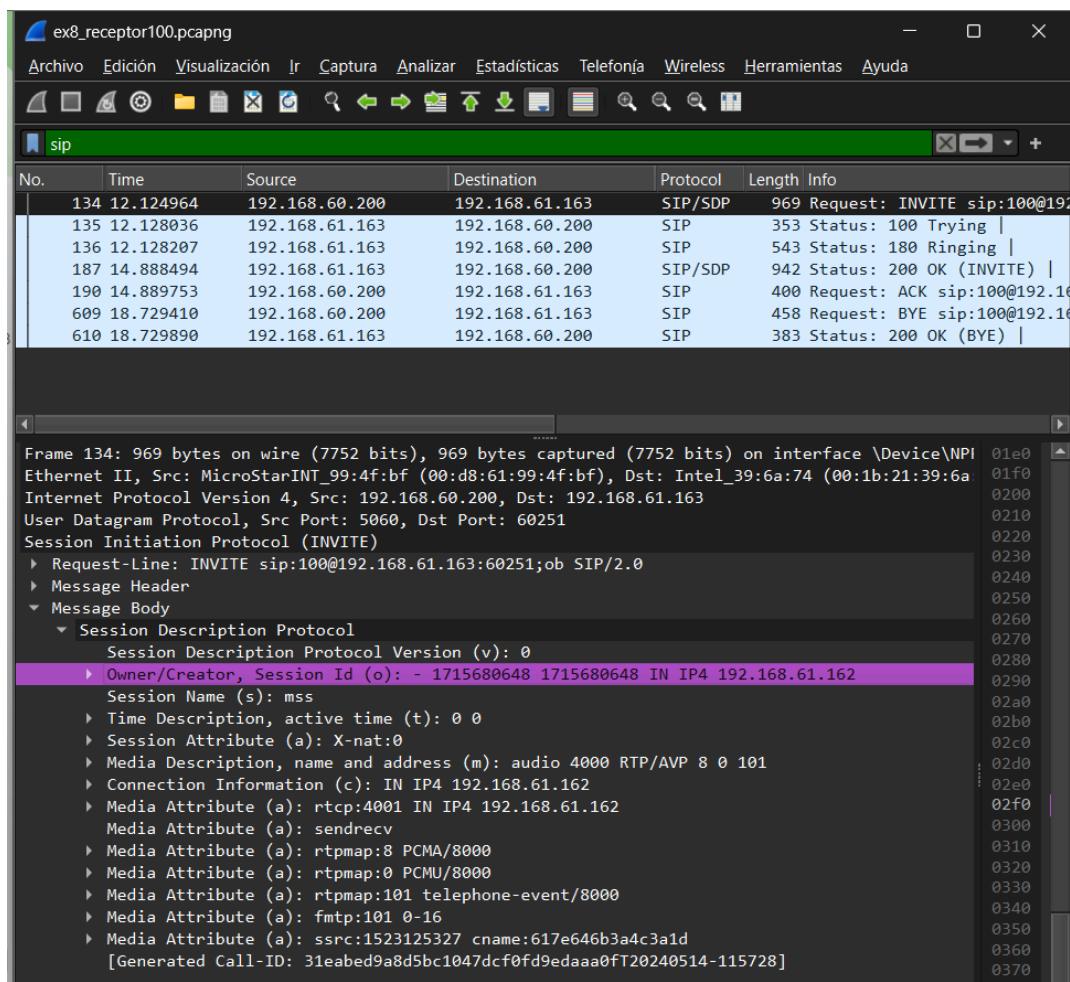
Nombre de cuenta	MiniSIPserver	<a href="#">?</a>
Servidor SIP	192.168.60.200	<a href="#">?</a>
Proxy SIP	192.168.60.200:5060	<a href="#">?</a>
Usuario *	102	<a href="#">?</a>
Dominio *	192.168.60.200	<a href="#">?</a>
Iniciar sesión	102	<a href="#">?</a>
Contraseña	***	<a href="#">?</a>
<a href="#">Mostrar contraseña</a>		
Nombre para mostrar	jiale.chen	<a href="#">?</a>
Núm. buzón de voz	*86	<a href="#">?</a>
Prefijo de Marcación		<a href="#">?</a>
Plan de marcado		<a href="#">?</a>
<input type="checkbox"/> Hide Caller ID		<a href="#">?</a>
Comunicación cifrada	Desactivado	<a href="#">?</a>
Transporte	UDP	<a href="#">?</a>
Dirección pública	Automático	<a href="#">?</a>
Refresco de Registro	300	Mantener Conexión <input type="text" value="15"/>
<input type="checkbox"/> Publicar presencia <a href="#">?</a> <input type="checkbox"/> Permitir reescritura IP <a href="#">?</a> <input type="checkbox"/> ICE <a href="#">?</a> <input type="checkbox"/> Desactivar temporiz. de sesión <a href="#">?</a>		
<a href="#">x</a>	<a href="#">Guardar</a>	<a href="#">Cancelar</a>

**b) Feu trucades entre ells, analitzeu els missatges, etc. Compareu els resultats amb els obtinguts amb el servidor UPCnet. Específicament, captureu la trucada tant al servidor com a un dels clients.**

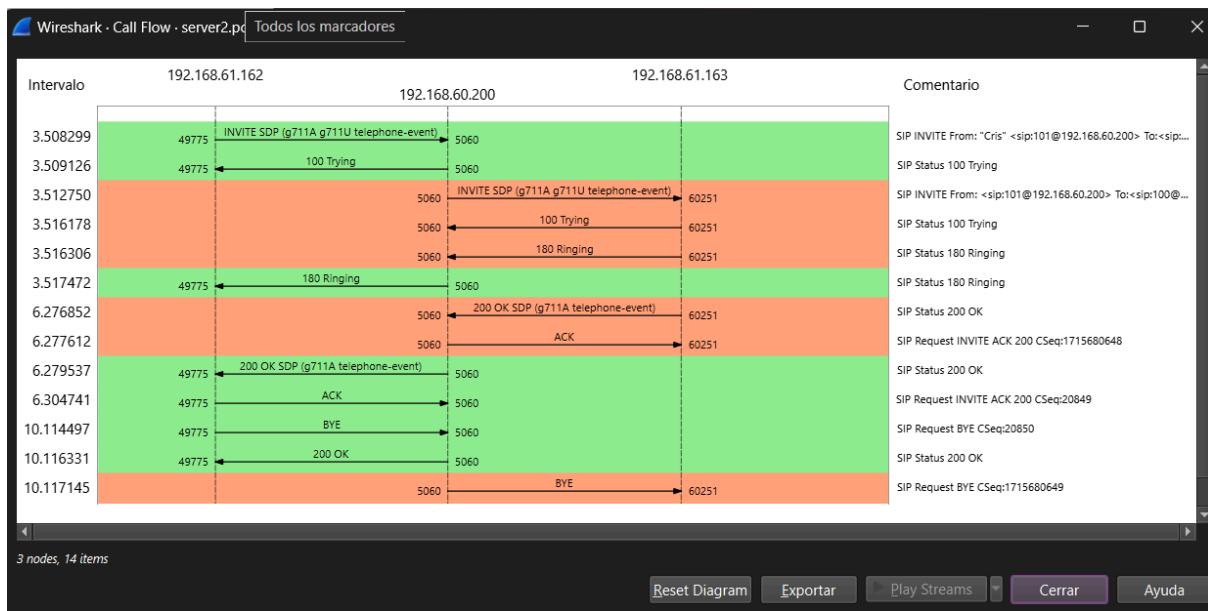
Aquí veiem el flux de la trucada desde el receptor d'aquesta:



Aquest és el contingut del paquet INVITE que arriba desde el servidor al receptor per a convidar al receptor a la trucada que fa l'emisor:



I aquest és el flux de la trucada desde el servidor:



Podem veure que en aquest cas, el receptor, pot veure la IP real del client emissor de la trucada. Ja que el servidor no està enmascarant les IP, utilitzant la 'quarta màquina'. En aquest cas, els fluxos RTP s'envien directament entre els terminals (passan't per el servidor).

## Exercici 10

**Amb netem, introduïu (un a un) problemes de QoS durant una trucada: pèrdues (en un rang entre 1% i 20%), retards absoluts (0.5s – 1s – què passa amb la interactivitat?), i jitter (100 ms +/- 50 ms, de manera que els paquets es reordenin).**

*Introducir pèrdues entre el 1% y el 20%:*

*sudo tc qdisc add dev enp3s0 root netem loss 1% 20%*

*IGUAL*

*Introducir un retardo absoluto de 0.5 segundos:*

*sudo tc qdisc change dev enp3s0 root netem delay 0.5s*

*IGUAL*

*Introducir un retardo absoluto de 1 segundo:*

*sudo tc qdisc change dev enp3s0 root netem delay 1s*

*IGUAL*

*Introducir jitter de 100ms +/- 50ms:*

*sudo tc qdisc change dev enp3s0 root netem delay 100ms 50ms*

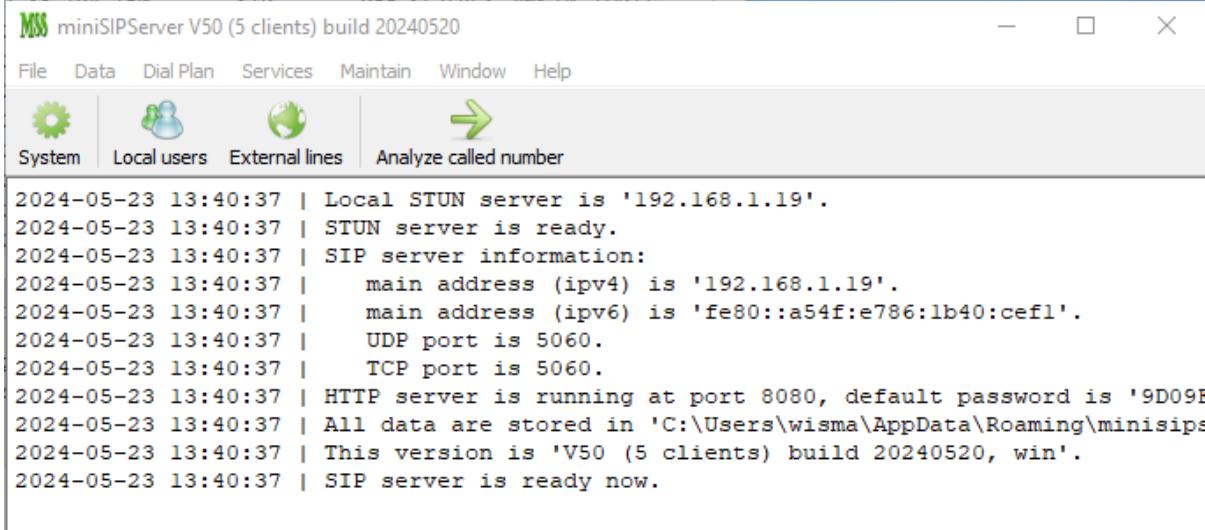
*MODO 0.5seg RETARD*

En aquest escenari tindrem el servidor MiniSIPServer en una màquina Windows i dos clients per l'aplicació Linphone en màquines Linux.

Configuració dels diferents servidors i clients:

### MINI SIP SERVER:

Al activar-se ens haurà de dir que el servidor SIP està activat per connectar usuaris:



The screenshot shows the main interface of the miniSIPServer V50 application. At the top, there's a menu bar with File, Data, Dial Plan, Services, Maintain, Window, and Help. Below the menu is a toolbar with four icons: System (green gear), Local users (blue people), External lines (green globe), and Analyze called number (green arrow). The main area is a text log window displaying the server's startup messages:

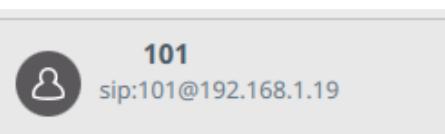
```
2024-05-23 13:40:37 | Local STUN server is '192.168.1.19'.
2024-05-23 13:40:37 | STUN server is ready.
2024-05-23 13:40:37 | SIP server information:
2024-05-23 13:40:37 |     main address (ipv4) is '192.168.1.19'.
2024-05-23 13:40:37 |     main address (ipv6) is 'fe80::a54f:e786:1b40:cef1'.
2024-05-23 13:40:37 |     UDP port is 5060.
2024-05-23 13:40:37 |     TCP port is 5060.
2024-05-23 13:40:37 | HTTP server is running at port 8080, default password is '9D09E'.
2024-05-23 13:40:37 | All data are stored in 'C:\Users\wisma\AppData\Roaming\minisips'.
2024-05-23 13:40:37 | This version is 'V50 (5 clients) build 20240520, win'.
2024-05-23 13:40:37 | SIP server is ready now.
```

Els usuaris locals donats i configurats per el servidor MiniSIP: ( en la captura ja estan connectats)

Local users				
User name	Description	Display name	Address	Other
100				
101			192.168.1.24:5060 Linphone-Desktop/5.2.4 (marionas-laptop) ubuntu...	
102			192.168.1.20:5060 Linphone-Desktop/5.2.4 (mariona-laptop) ubuntu/...	

### Usuari 101:

Client Sip al Linphone utilitzant un usuari local (101) amb la ip del servidor MiniSIP (192.168.1.19)

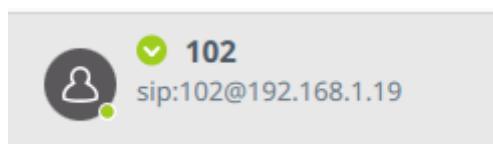


#### Ajustes principales de cuenta SIP

Dirección SIP*	sip:101@192.168.1.19
Dirección del servidor SIP*	<sip:192.168.1.19:5060;transport=udp>
Duración del registro (seg)	3600
Transporte	UDP
Ruta	sip:192.168.1.19:5060

### Usuari 102:

Client Sip al Linphone utilitzant un usuari local (101) amb la ip del servidor MiniSIP (192.168.1.19)



#### Ajustes principales de cuenta SIP

Dirección SIP*	sip:102@192.168.1.19
Dirección del servidor SIP*	<sip:192.168.1.19:5060;transport=udp>
Duración del registro (seg)	600
Transporte	UDP
Ruta	sip:192.168.1.19
Conference URI	sip:conference-factory@sip.linphone.org

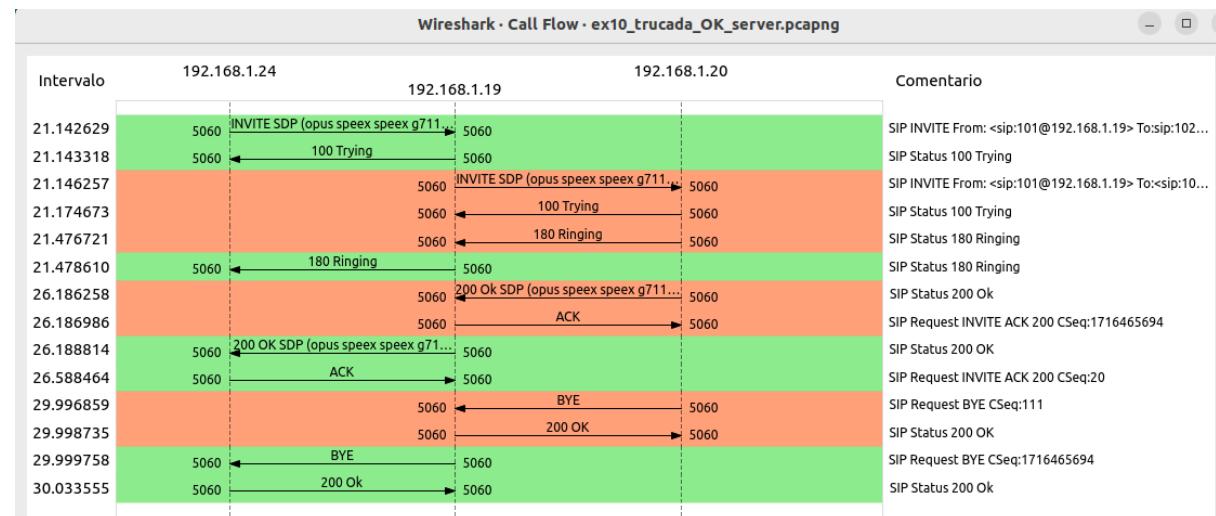
**a) Comenteu la qualitat subjectiva en cada cas, i compareu-la amb les mesures de qualitat que us donen els clients com Linphone o Zoiper. Compareu també amb les mesures de QoS de l'analitzador RTP de Wireshark.**

Per poder comentar la qualitat subjectiva en cada cas donat anteriorment, mirarem primer per una trucada normal i sense filtres que introduceixen problemes de QoS perquè sigui l'estàndard de la trucada.

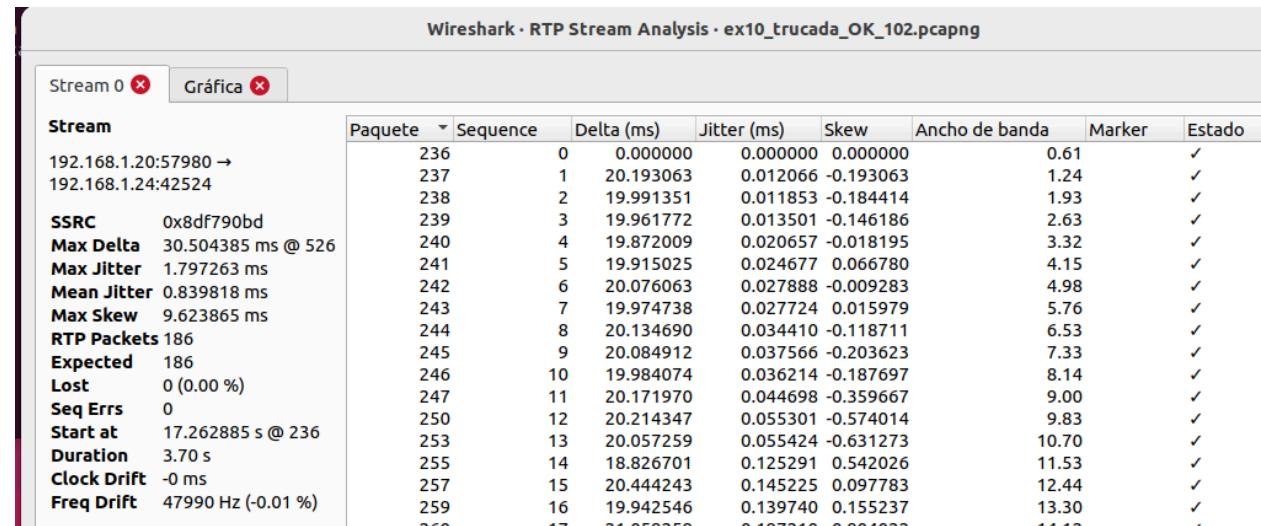
Per mirar les mesures de qualitat dels clients, mirarem les estadístiques del Linphone que va donant quan hi ha una trucada activa:

### Trucada OK:

Per aquesta trucada podem veure que el client 101 vol trucar al 102 i es acceptada. Això ho podem veure amb la seqüència del servidor MiniSIP:



Les estadístiques de RTP d'un dels clients són:



Source Address	Source Port	Destination Address	Destination Port	SSRC	Start Time	Duration	Payload	Packets	Lost	Min Delta (ms)	Mean Delta (ms)	Max Delta (ms)	Min Jitter	Mean Jitter	Max Jitter	Status
192.168.1.20	57980	192.168.1.24	42524	0x8df790bd	17.262885	3.70	opus	186	0 (0.0%)	9.644621	20.004201	30.504385	0.011853	0.839818	1.797263	
192.168.1.24	42524	192.168.1.20	57980	0x55268d0b	17.538787	3.45	opus	173	0 (0.0%)	0.028333	20.038334	46.058779	0.068149	3.392864	7.307524	

La qualitat és estàndar i bona, sense cap tipus de pèrdues de la qualitat i no hi ha cap retard de comunicació perceptible. Aquesta experiència de trucada clara i fluïda es deu a una xarxa estable i a una configuració adequada del sistema.

### Amb pèrdues:

Mesures de qualitat abans de les pèrdues:



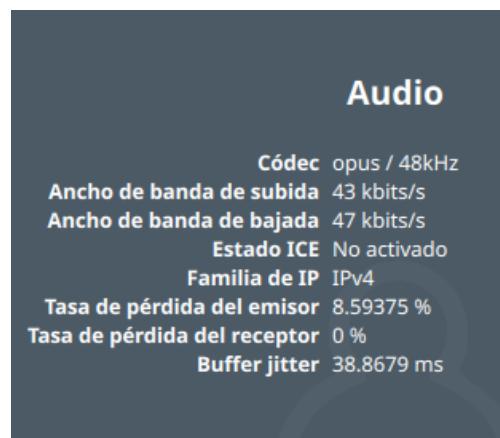
Aplicar les pèrdues amb la comanda:

```
sudo tc qdisc add dev wlo1 root netem loss 10% 10%
```

Seguidament podem veure que s'han aplicat bé:

```
usuario@marionas-laptop:~/Escritorio$ tc qdisc show dev wlo1
qdisc netem 8001: root refcnt 2 limit 1000 loss 10% 10%
```

Les mesures de qualitat després de les pèrdues:



Les estadístiques de RTP d'un dels clients són:

Wireshark - RTP Stream Analysis - ex10\_trucada\_perduda10\_102.pcapng

Stream 0		Gráfica						
Stream	Paquete	Sequence	Delta (ms)	Jitter (ms)	Skew	Ancho de banda	Marker	Estado
192.168.1.20:41944 →	105	0	0.000000	0.000000	0.000000	0.63	✓	
192.168.1.24:55996	106	1	20.224067	0.014004	-0.224067	1.30	✓	
	107	2	21.038975	0.078065	-1.263042	1.96	✓	
<b>SSRC</b> 0xa40e5fb5	108	3	19.833864	0.083569	-1.096906	2.66	✓	
<b>Max Delta</b> 40.855658 ms @ 3300	109	4	20.876825	0.133148	-1.973731	3.53	✓	
<b>Max Jitter</b> 3.922514 ms	110	5	19.371339	0.164117	-1.345070	4.33	✓	
<b>Mean Jitter</b> 1.097330 ms	113	6	20.462703	0.182779	-1.807773	5.10	✓	
<b>Max Skew</b> -32.715760 ms	115	7	19.913389	0.176768	-1.721162	5.90	✓	
<b>RTP Packets</b> 1664	117	8	29.831687	0.780201	-11.552849	6.71	✓	
<b>Expected</b> 1664	118	9	20.171157	0.742136	-11.724006	7.55	✓	
<b>Lost</b> 0 (0.00 %)	119	10	20.434767	0.722925	-12.158773	8.45	✓	
<b>Seq Errs</b> 0	120	11	19.802101	0.690111	-11.960874	9.34	✓	
<b>Start at</b> 13.241102 s @ 105	121	12	19.409165	0.683906	-11.370039	10.19	✓	
<b>Duration</b> 33.27 s	123	13	20.200462	0.653691	-11.570501	11.06	✓	
<b>Clock Drift</b> 1 ms	127	14	20.218430	0.626487	-11.788931	11.94	✓	
<b>Freq Drift</b> 48002 Hz (0.00 %)	128	15	20.241106	0.602401	-12.030037	12.83	✓	
	130	16	19.295899	0.608757	-11.325936	13.69	✓	

Source Address	Source Port	Destination Address	Destination Port	SSRC	Start Time	Duration	Payload	Packets	Lost	Min Delta (ms)	Mean Delta (ms)	Max Delta (ms)	Min Jitter	Mean Jitter	Max Jitter	Status
192.168.1.20	41944	192.168.1.24	55996	0xa40e5fb5	13.241102	33.27	opus	1664	0 (0.0%)	0.054711	20.007135	40.855658	0.014004	1.097330	3.922514	
192.168.1.24	55996	192.168.1.20	41944	0x7ec8d85b	13.563401	32.94	opus	1600	48 (2.9%)	0.004273	20.599344	75.834759	0.077721	3.782285	10.113740	*

Després d'aplicar les pèrdues, mesures de qualitat de la trucada mostren un augment significatiu en la taxa de pèrdua de paquets de l'emissor (fins a 8.59375%), mentre que no hi ha cap pèrdua de paquets del receptor. L'ample de banda de pujada augmenta lleugerament i l'ample de banda de baixada disminueix una mica. El buffer jitter també ha disminuït, cosa que pot ser degut a la configuració de pèrdues aplicada.

Les estadístiques RTP mostren que hi ha pèrdua de paquets, però molt petit per notar-ho (2,9%) i un jitter relativament baix en el flux capturat. Això és coherent amb les estadístiques de qualitat de trucada abans d'aplicar les pèrdues. Després d'aplicar les pèrdues, s'observa un augment significatiu en la taxa de pèrdua de l'emissor en les estadístiques de qualitat de trucada.

Pel que fa a la qualitat subjectiva de la trucada, no s'ha trobat gaire diferència ni en la qualitat ni en la percepció de retard o interrupcions. Tot i les estadístiques que mostren pèrdues i variabilitat en el jitter, no hem percebut canvis significatius en l'experiència de la trucada.

Finalment borrar el filtre de pèrdues:  
 sudo tc qdisc del dev wlo1 root netem

## RETARD ABSOLUT 0,5:

Mesures de qualitat abans del retard de 0,5 seg:



Aplicar el retard de 0,5 segons amb la comanda:

```
sudo tc qdisc add dev wlo1 root netem delay 500ms
```

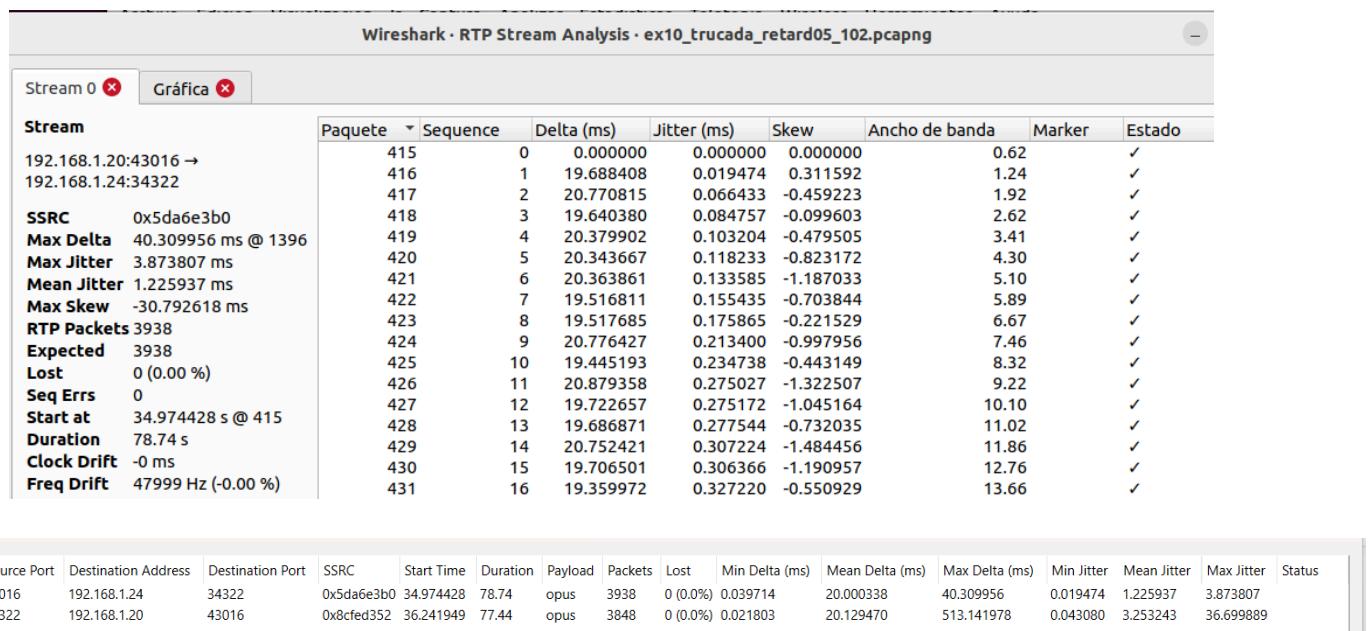
Seguidament podem veure que s'han aplicat bé:

```
usuario@marionas-laptop:~/Escritorio$ tc qdisc show dev wlo1
qdisc netem 8002: root refcnt 2 limit 1000 delay 500ms
```

Les mesures de qualitat després del retard de 0,5 seg :



Les estadístiques de RTP d'un dels clients són:



Després d'aplicar un retard de 0,5 segons, es pot observar que el buffer jitter ha augmentat de manera considerable (de 56.5 ms a 304.25 ms), mentre que la taxa de pèrdua de paquets s'ha mantingut en 0% tant per l'emissor com pel receptor. Aquest augment del jitter pot afectar la qualitat de l'àudio percebuda, tot i que la taxa de pèrdua de paquets no hagi canviat.

Les estadístiques RTP mostren un jitter màxim de 3.873007 ms i un jitter mitjà de 1.225937 ms, molt inferiors al buffer jitter observat en les estadístiques de qualitat de trucada (304.25 ms), que podria indicar que el retard de 0,5 segons està causant una major variabilitat en la latència dels paquets a nivell de l'aplicació, però els paquets RTP es reben de manera més consistent., el buffer jitter a nivell de l'aplicació és significativament més alt després d'aplicar el retard, cosa que podria afectar la percepció de qualitat de la trucada pels usuaris

Pel que fa a la qualitat subjectiva de la trucada, tampoc ja s'ha trobat gaire diferència ni en la qualitat ni en la percepció de retard o interrupcions.

finalment borrar el filtre del retard:  
 sudo tc qdisc del dev wlo1 root netem

## RETARD ABSOLUT 1:

Mesures de qualitat abans del retard de 1 segon:



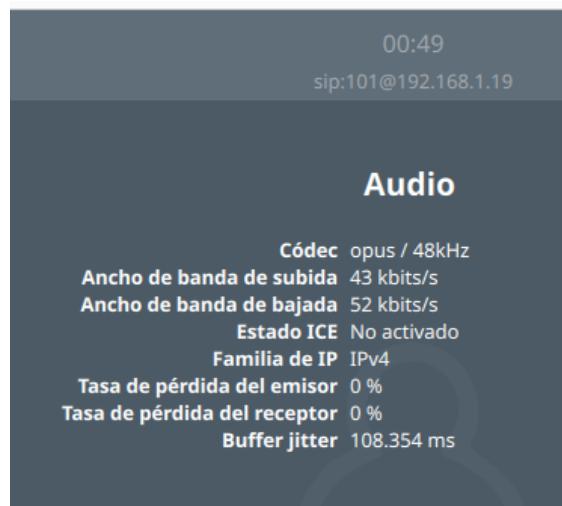
Aplicar el retard de 1 segon amb la comanda:

```
sudo tc qdisc add dev wlo1 root netem delay 1000ms
```

Seguidament podem veure que s'han aplicat bé:

```
usuario@marionas-laptop:~/Escritorio$ tc qdisc show dev wlo1
qdisc netem 8003: root refcnt 2 limit 1000 delay 1s
```

Les mesures de qualitat després del retard de 1 seg :



Les estadístiques de RTP d'un dels clients són:

Stream 0									Gráfica
Stream	Paquete	Sequence	Delta (ms)	Jitter (ms)	Skew	Ancho de banda	Marker	Estado	
192.168.1.20:36811 → 192.168.1.24:47869	205	0	0.000000	0.000000	0.000000	0.63	✓		
<b>SSRC</b> 0x45334a75	212	1	20.108094	0.006756 -0.108094	1.27	✓			
<b>Max Delta</b> 41.336055 ms @ 1427	214	2	19.781012	0.020020 0.110894	1.94	✓			
<b>Max Jitter</b> 4.140276 ms	215	3	19.948973	0.021958 0.161921	2.62	✓			
<b>Mean Jitter</b> 1.137413 ms	216	4	29.827555	0.634808 -9.665634	3.37	✓			
<b>Max Skew</b> -31.719749 ms	217	5	20.439879	0.622625 -10.105513	4.26	✓			
<b>RTP Packets</b> 3921	219	6	19.804726	0.595916 -9.910239	5.10	✓			
<b>Expected</b> 3921	220	7	20.129799	0.566783 -10.040038	5.93	✓			
<b>Lost</b> 0 (0.00 %)	221	8	20.076026	0.536111 -10.116064	6.74	✓			
<b>Seq Errs</b> 0	222	9	20.323823	0.522843 -10.439887	7.56	✓			
<b>Start at</b> 19.303443 s @ 205	225	10	20.206050	0.503043 -10.645937	8.39	✓			
<b>Duration</b> 78.41 s	228	11	20.199775	0.484089 -10.845712	9.29	✓			
<b>Clock Drift</b> -0 ms	230	12	19.201656	0.503730 -10.047368	10.17	✓			
<b>Freq Drift</b> 47999 Hz (-0.00 %)	231	13	20.105151	0.478819 -10.152519	11.06	✓			
	234	14	20.044970	0.451703 -10.197489	11.90	✓			
	236	15	19.816418	0.434946 -10.013907	12.80	✓			
	237	16	19.852106	0.417005 -9.866013	13.70	✓			
	240	17	20.303707	0.409924 -10.169720	14.56	✓			
	241	18	20.314731	0.403974 -10.484451	15.38	✓			
	243	19	19.918916	0.383794 -10.403367	16.30	✓			
	246	20	20.422241	0.386197 -10.825608	17.21	✓			
	248	21	18.971115	0.426365 -9.796723	18.13	✓			
	249	22	20.152843	0.409270 -9.949566	19.01	✓			

Wireshark - RTP Streams - ex10_trucada_retard1_102.pcapng																
Source Address	Source Port	Destination Address	Destination Port	SSRC	Start Time	Duration	Payload	Packets	Lost	Min Delta (ms)	Mean Delta (ms)	Max Delta (ms)	Min Jitter	Mean Jitter	Max Jitter	Status
192.168.1.20	36811	192.168.1.24	47869	0x45334a75	19.303443	78.41	opus	3921	0 (0.0%)	0.046369	20.002796	41.336055	0.006756	1.137413	4.140276	
192.168.1.24	47869	192.168.1.20	36811	0xb4ad4a51	19.550397	78.15	opus	3858	0 (0.0%)	0.004038	20.260912	1010.056905	0.244476	3.441566	68.896767	

Després d'aplicar un retard de 1 segon, es pot observar que el buffer jitter ha augmentat de manera considerable (de 62.2535 ms a 108.354 ms), mentre que la taxa de pèrdua de paquets s'ha mantingut en 0% tant per l'emissor com pel receptor. Aquest augment del jitter pot afectar la qualitat de l'àudio percebuda, tot i que la taxa de pèrdua de paquets no hagi canviat. A més, el retard perceptible de 1 segon fa que la comunicació sigui menys fluida i interfereixi amb la interactivitat de la conversa.

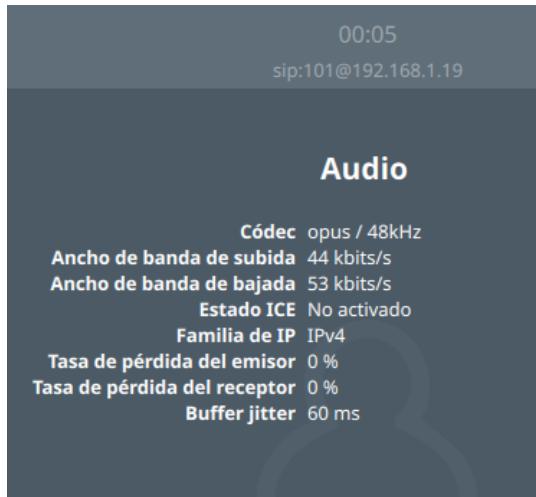
Les estadístiques RTP mostren un jitter màxim de 4.140276 ms i un jitter mitjà de 1.137413 ms, molt inferiors al buffer jitter observat en les estadístiques de qualitat de trucada (108.354 ms), que podria indicar que el retard de 1 segon està causant una major variabilitat en la latència dels paquets, però aquests es van reben de manera més consistent. El buffer jitter a nivell de l'aplicació és significativament més alt després d'aplicar el retard, cosa que podria afectar la percepció de qualitat de la trucada pels usuaris.

Pel que fa a la qualitat subjectiva de la trucada, s'ha trobat diferència, no en la qualitat sinó en el temps de la parla i quan s'escoltava realment el missatge, ja que semblava retardat i trigava més en escoltar el missatge transmès per la trucada., hem percebut canvis prou significatius en l'experiència de la trucada pel que fa al retard perceptible en la interacció.

Finalment borrar el filtre del retard:  
 sudo tc qdisc del dev wlo1 root netem

## JITTER:

Mesures de qualitat abans del jitter:



Aplicar el jitter amb la comanda:

```
sudo tc qdisc add dev wlo1 root netem delay 500ms 100ms 50ms
distribution normal
```

Seguidament podem veure que s'han aplicat bé:

```
usuario@marionas-laptop:~/Escritorio$ tc qdisc show dev wlo1
qdisc netem 8004: root refcnt 2 limit 1000 delay 100ms 50ms
```

Les mesures de qualitat després d'aplicar el jitter:



Les estadístiques de RTP d'un dels clients són:

Wireshark - RTP Stream Analysis - ex10_trucada_jitter_102.pcapng								
Stream 0	Gráfica							
Stream	Paquete	Sequence	Delta (ms)	Jitter (ms)	Skew	Ancho de banda	Marker	Est.
192.168.1.20:46850 →	146	0	0.000000	0.000000	0.000000	0.64	✓	
192.168.1.24:41646	147	1	19.697311	0.018918	0.302689	1.27	✓	
	148	2	20.575349	0.053695	-0.272660	1.98	✓	
<b>SSRC</b> 0xb16bfc27	149	3	20.379245	0.074042	-0.651905	2.69	✓	
<b>Max Delta</b> 31.652161 ms @ 6838	150	4	20.451769	0.097650	-1.103674	3.48	✓	
<b>Max Jitter</b> 3.936576 ms	151	5	19.564519	0.118764	-0.668193	4.29	✓	
<b>Mean Jitter</b> 1.148388 ms	152	6	20.404459	0.136620	-1.072652	5.05	✓	
<b>Max Skew</b> -21.767016 ms	153	7	19.164501	0.180300	-0.237153	5.85	✓	
<b>RTP Packets</b> 7328	155	8	20.461727	0.197889	-0.698880	6.67	✓	
<b>Expected</b> 7328	157	9	20.321969	0.205644	-1.020849	7.54	✓	
<b>Lost</b> 0 (0.00 %)	158	10	19.541233	0.221464	-0.562082	8.42	✓	
<b>Seq Errs</b> 0	159	11	20.641486	0.247716	-1.203568	9.32	✓	
<b>Start at</b> 19.184518 s @ 146	162	12	20.351513	0.254203	-1.555081	10.18	✓	
<b>Duration</b> 146.55 s	165	13	29.833779	0.852927	-11.388860	11.02	✓	
<b>Clock Drift</b> 1 ms	167	14	19.850905	0.808937	-11.239765	11.91	✓	
<b>Freq Drift</b> 48000 Hz (0.00 %)	169	15	18.822196	0.831991	-10.061961	12.79	✓	
	171	16	20.697618	0.823593	-10.759579	13.71	✓	
	173	17	20.139514	0.780838	-10.899093	14.59	✓	
	175	18	19.924972	0.736725	-10.824065	15.50	✓	
	179	19	19.286961	0.735245	-10.111026	16.33	✓	
	181	20	20.600035	0.726794	-10.711061	17.18	✓	
	183	21	19.867517	0.689650	-10.578578	18.07	✓	
	185	22	19.625225	0.669970	-10.203803	18.99	✓	
	187	23	19.793402	0.641009	-9.997205	19.88	✓	
	190	24	20.092772	0.606744	-10.089977	20.80	✓	
	192	25	20.033293	0.570904	-10.123270	21.71	✓	

Source Address	Source Port	Destination Address	Destination Port	SSRC	Start Time	Duration	Payload	Paquets	Lost	Min Delta (ms)	Mean Delta (ms)	Max Delta (ms)	Min Jitter	Mean Jitter	Max Jitter	Status
192.168.1.20	46850	192.168.1.24	41646	0xb16bfc27	19.184518	146.55	opus	7328	0 (0.0%)	8.379622	20.001496	31.652161	0.018918	1.148388	3.936576	
192.168.1.24	41646	192.168.1.20	46850	0xf1a9c2dd	19.471088	146.21	opus	7308	-1 (-0.0%)	0.002891	20.009829	185.725804	0.020363	15.011299	56.446693	*

Després d'aplicar un retard de 500 ms amb una variabilitat de 100 ms, s'observa un augment significatiu en el buffer jitter (de 60 ms a 115.758 ms). Tot i que la taxa de pèrdua de paquets es manté en 0% tant per l'emissor com pel receptor, aquest increment del jitter afecta a la qualitat de l'àudio percebuda, causant interrupcions o distorsions en la reproducció. A més, es nota una diferència en la qualitat subjectiva de la trucada, no en la qualitat de l'àudio, sinó en el temps de la parla i quan s'escoltava realment el missatge.

Les estadístiques RTP mostren una major variabilitat en la transmissió, especialment amb un max delta de 185.725804 ms i un mean jitter de 15.011299 ms, molt superiors als valors observats en la transmissió inversa. Aquest augment en la variabilitat i el jitter és el causant d'interrupcions o distorsions en l'àudio percebut pels usuaris. Tot i així, la pèrdua de paquets es manté en 0%, indicant que no hi ha pèrdua de dades durant la transmissió.

Pel que fa a la qualitat subjectiva de la trucada, s'ha trobat diferència, ja que semblava retardat i trigava més en escoltar el missatge transmès per la trucada. La qualitat de l'àudio ha estat prou terrible ja que costava entendre algunes frases, fent l'experiència de la trucada molt menys fluida i comprensible.

Finalment borrar el filtre del jitter:

```
sudo tc qdisc del dev wlo1 root netem
```

**b) Intenteu trobar el límit de correcció de jitter del buffer de reproducció que té cada client.**

Donat les estadístiques de l'anterior apartat, podem extreure vàries dades sobre el jitter en el

Transmissió de 192.168.1.20 a 192.168.1.24 amb Jitter Baix:

Max Jitter: 3.936576 ms

Mean Jitter: 1.148388 ms

Qualitat de l'àudio sense problemes significatius.

Transmissió de 192.168.1.24 a 192.168.1.20 amb Jitter Alt:

Max Jitter: 56.446693 ms

Mean Jitter: 15.011299 ms

Qualitat de l'àudio degradada, difícil entendre algunes frases.

Límits Estimats de Correcció de Jitter

Max Jitter: Aproximadament 4 ms

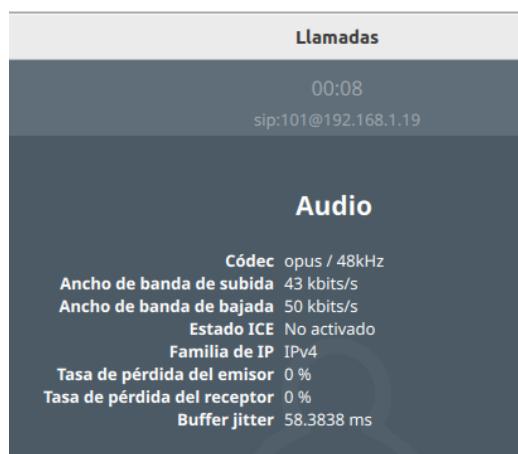
Mean Jitter: Aproximadament 1.5 ms

El límit de correcció de jitter del buffer de reproducció dels clients sembla estar al voltant d'un jitter màxim de 4 ms i un jitter mitjà de 1.5 ms. Quan el jitter supera aquests valors, com es va observar amb un jitter màxim de 56.446693 ms i un jitter mitjà de 15.011299 ms, la qualitat de l'àudio es degrada significativament, fent que sigui difícil entendre algunes frases. Això suggereix que el buffer de reproducció no pot gestionar de manera efectiva un jitter tan alt, resultant en una experiència d'àudio dolenta.

**c) Amb un nivell extrem de pèrdues (50% o més), intenteu obtenir una situació on els missatges SIP es perdin i veieu com es retransmeten.  
Quin és el valor del temporitzador de retransmissió?**

PÈRDUES DEL 50%:

Mesures de qualitat abans grans pèrdues:



Aplicar el grans pèrdues amb la comanda:

```
sudo tc qdisc add dev wlo1 root netem loss 50%
```

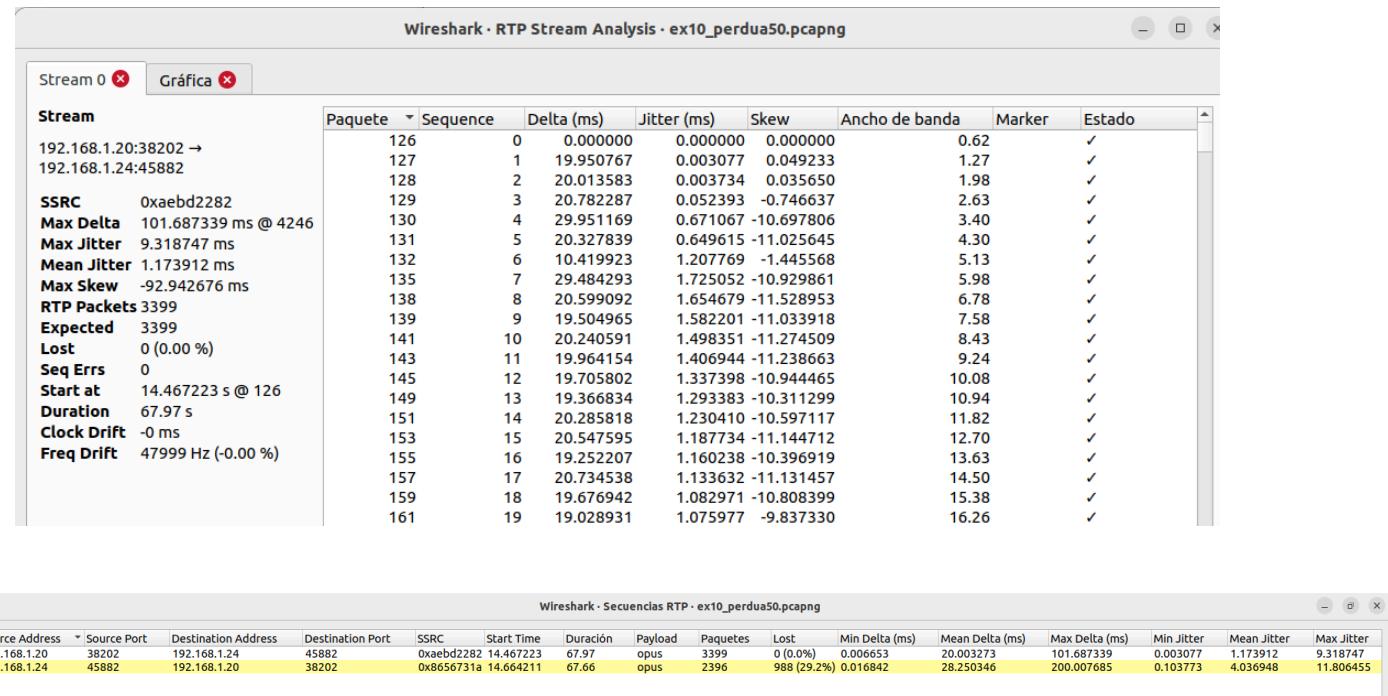
Seguidament podem veure que s'han aplicat bé:

```
usuario@marionas-laptop:~/Escritorio$ tc qdisc show dev wlo1
qdisc netem 8005: root refcnt 2 limit 1000 loss 50%
```

Les mesures de qualitat després de les grans pèrdues :



Les estadístiques de RTP d'un dels clients són:



Quan els missatges SIP es perden en grans quantitats, s'activa un mecanisme de retransmissió per assegurar que la comunicació es

mantingui. Per defecte, el temporitzador T1 s'utilitza com a base per les retransmissions.

Temporitzador T1: Generalment, 500 ms (0.5 segons)

Retransmissions: Normalment es retransmeten de manera exponencial fins a un màxim de T2 (generalment 4 segons).

Per exemple si el valor T1 és de 500 ms, les retransmissions seguiran aquest patró:

Primera retransmissió: 500 ms després de l'enviament inicial.

Segona retransmissió: 1 segon després (500 ms \* 2).

Tercera retransmissió: 2 segons després (1 segon \* 2).

Quarta retransmissió: 4 segons després (2 segons \* 2).

Aquest patró continuará fins a un límit predefinit o fins que es rebi una resposta.

Amb un nivell extrem de pèrdues (50% o més), la qualitat de l'àudio es degrada significativament, fent difícil entendre les frases.

Els missatges SIP es retransmeten seguint un temporitzador de retransmissió basat en T1, inicialment configurat a 500 ms, amb un augment exponencial fins a T2 o fins que es rebi una resposta.

Amb un nivell extrem de pèrdues (50% o més), la qualitat de l'àudio es degrada significativament, fent difícil entendre les frases. La transmissió de 192.168.1.24 a 192.168.1.20 mostra una pèrdua del 29.2% dels paquets RTP, afectant greument l'experiència de l'usuari.

amb una variabilitat en la Latència i el Jitter de:

Max Delta: 200.007685 ms

Mean Jitter: 4.036948 ms

Max Jitter: 11.806455 ms

Aquestes variacions en la latència i el jitter provoquen interrupcions i distorsions en l'àudio, afectant la comunicació fluida i entendible.

I com s'ha explicat anteriorment, els missatges SIP es retransmeten seguint un temporitzador basat en T1 (500 ms), augmentant exponencialment fins a T2 (4 segons) per recuperar missatges perduts, però això introduceix un retard addicional.

finalment borrar el filtre de les pèrdues:

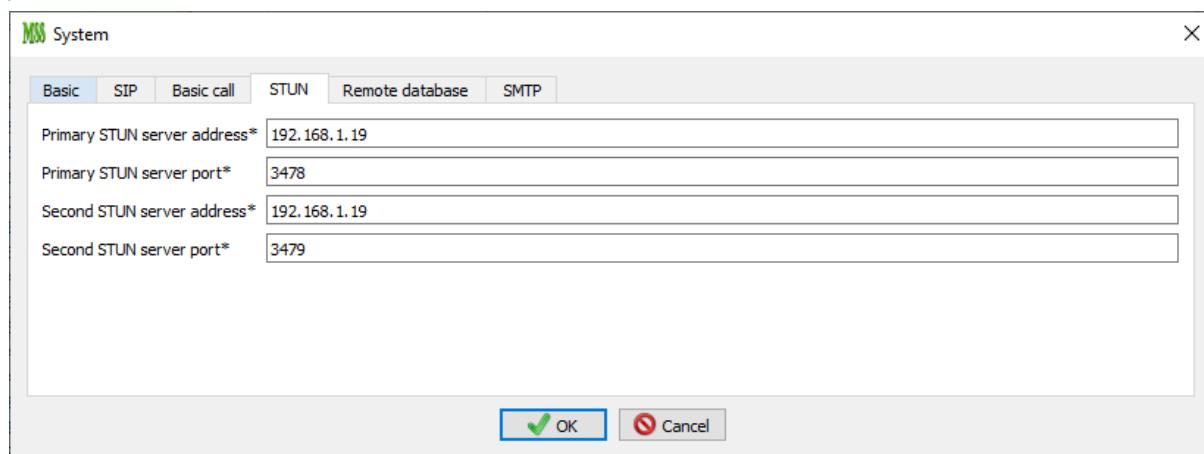
```
sudo tc qdisc del dev wlo1 root netem
```

## Exercici 11

**A partir de l'escenari de l'exercici 8 (o l'exercici 9, però obviant la part de SIP trunk i fent trucades només internes), almenys amb dos clients SIP associats al MiniSIPserver:**

- a) Configureu un client amb STUN apuntant cap al servidor STUN del MiniSIPserver. Analitzeu el diàleg STUN i comproveu com es pot veure que no hi ha cap NAT entre el client i el proxy.**

Tenint el Servidor MiniSIPServer configurat, mirar quina és la seva adreça pel STUN:



Després modificar un dels clients SIP perquè es connecti al servidor STUN. Configurarem el Linphone per utilitzar l'usuari 100 i les ip i el port del STUN del servidor MiniSIPsERVER:

### Ajustes principales de cuenta SIP

Dirección SIP*	sip:100@192.168.1.19
Dirección del servidor SIP*	<sip:192.168.1.19:3478;transport=udp>
Duración del registro (seg)	3600
Transporte	UDP
Ruta	sip:192.168.1.19:3478

### NAT y cortafuegos

Activar ICE

Servidor STUN/TURN

192.168.1.19:3789

Al activar aquest usuari, des de wireshark podem veure aquests paquets:

No.	Time	Source	Destination	Protocol	Length	Info
2165	102.877631995	192.168.1.20	192.168.1.19	STUN	62	Binding Request
2166	102.881318109	192.168.1.19	192.168.1.20	STUN	106	Binding Success Response MAPPED-ADDRESS: 192.168.1.20:60733
2167	102.952942667	192.168.1.20	192.168.1.19	STUN	62	Binding Request
2168	102.955734783	192.168.1.19	192.168.1.20	STUN	106	Binding Success Response MAPPED-ADDRESS: 192.168.1.20:60198

El diàleg stun consisteix en:

- **Binding Request:** El client STUN envia una sol·licitud de vincle (Binding Request) al servidor STUN per conèixer la seva adreça IP pública i el port a través del qual es veu des de fora de la xarxa NAT. Un Binding Request s'envia des del client (192.168.1.20) al servidor STUN (192.168.1.19) per descobrir la seva adreça pública i port.
- **Binding Success Response:** El servidor STUN respon amb una resposta d'èxit de vincle (Binding Success Response), proporcionant l'adreça IP pública i el port mapejat (MAPPED-ADDRESS) del client. El servidor STUN (192.168.1.19) respon al client (192.168.1.20) amb una Binding Success Response, indicant que l'adreça mapejada del client és 192.168.1.20 i el port és 60733.

Si extenem el paquet de Binding Success Response podem veure la següent informació:

```
Session Traversal Utilities for NAT
[Request In: 2167]
[Time: 0.002792116 seconds]
- Message Type: 0x0101 (Binding Success Response)
  .... ...0 .... = Message Class: 0x10 Success Response (2)
  ..00 000. 0001 = Message Method: 0x0001 Binding (0x001)
  ..0. .... .... = Message Method Assignment: IETF Review (0x0)
  Message Length: 44
  Message Cookie: 2112a442
  Message Transaction ID: c3b6585fdee7e07f7fd4ad41
  [STUN Network Version: RFC-5389/8489 (3)]
- Attributes
  - MAPPED-ADDRESS: 192.168.1.20:60198
    - Attribute Type: MAPPED-ADDRESS
      0... .... .... = Attribute Type Comprehension: Required (0x0)
      .0... .... .... = Attribute Type Assignment: IETF Review (0x0)
      Attribute Length: 8
      Reserved: 00
      Protocol Family: IPv4 (0x01)
      Port: 60198
      IP: 192.168.1.20
    - SOURCE_ADDRESS (Deprecated): 192.168.1.19:3478
      - Attribute Type: SOURCE_ADDRESS
        0... .... .... = Attribute Type Comprehension: Required (0x0)
        .0... .... .... = Attribute Type Assignment: IETF Review (0x0)
        Attribute Length: 8
        Reserved: 00
        Protocol Family: IPv4 (0x01)
        Port: 3478
        IP: 192.168.1.19
    - CHANGED_ADDRESS (Deprecated): 192.168.1.19:3479
      - Attribute Type: CHANGED_ADDRESS
        0... .... .... = Attribute Type Comprehension: Required (0x0)
        .0... .... .... = Attribute Type Assignment: IETF Review (0x0)
        Attribute Length: 8
        Reserved: 00
        Protocol Family: IPv4 (0x01)
        Port: 3479
        IP: 192.168.1.19
    - FINGERPRINT
      - Attribute Type: FINGERPRINT
        1... .... .... .... = Attribute Type Comprehension: Optional (0x1)
        .0... .... .... .... = Attribute Type Assignment: IETF Review (0x0)
        Attribute Length: 4
        CRC-32: 0x4ef31ddf [correct]
        [CRC-32 Status: Good]
```

En aquest paquet les ips que podem trobar són :

1. **Adreça Mapejada (MAPPED-ADDRESS):**
  - o IP: 192.168.1.20 Port: 60198
2. **Adreça d'Origen (SOURCE\_ADDRESS):**
  - o IP: 192.168.1.19 Port: 3478
3. **Adreça Canviada (CHANGED\_ADDRESS):**
  - o IP: 192.168.1.19 Port: 3479

Per comprovar si no hi ha cap NAT entre el client i el proxy, s'ha de revisar l'adreça IP i el port en la resposta exitosa (Binding Success Response):

- **MAPPED-ADDRESS:** Aquest atribut mostra l'adreça IP i el port que el servidor STUN veu per al client.
- **SOURCE\_ADDRESS i CHANGED\_ADDRESS:** Encara que aquests atributs estan obsolets, poden proporcionar informació addicional sobre la configuració de la xarxa.

L'adreça IP i el port en el camp MAPPED-ADDRESS són els mateixos que els del client (en aquest cas, 192.168.1.20), això indica que no hi ha cap NAT entre el client i el servidor STUN, ja que el client està directament connectat al proxy sense cap traducció d'adreça de xarxa intermediària.

**b) Configureu un client amb STUN apuntant cap a un servidor STUN públic (en particular, el que ofereix MicroSIP és una possibilitat). Analitzeu el diàleg STUN i comproveu com és capaç d'identificar la IP pública del NAT de la UPC. En acabar, desconfigureu l'STUN del client.**

Ara si ho connectem a un servidor públic, el nostre cas utilitzarem el de google, podem observar varíes diferències en els paquets:

#### Ajustes principales de cuenta SIP

---

Dirección SIP*	<input type="text" value="sip:100@stun.l.google.com"/>
Dirección del servidor SIP*	<input type="text" value="&lt;sip:stun.l.google.com:19302;transport=udp&gt;"/>
Duración del registro (seg)	<input type="text" value="3600"/> <span style="border: 1px solid #ccc; padding: 2px;">+ -</span>
Transporte	<input type="text" value="UDP"/> <span style="border: 1px solid #ccc; padding: 2px;">▼</span>
Ruta	<input type="text" value="sip:stun.l.google.com:19302"/>

## NAT y cortafuegos

Activar ICE	<input checked="" type="checkbox"/>	Servidor STUN/TURN	stun.l.google.com:19302
Activar TURN	<input type="checkbox"/>	Usuario TURN	100
		Contraseña TURN	

En aquest escenari els paquets capturats són els següents:

No.	Time	Source	Destination	Protocol	Length	Info
11954	80.848765014	192.168.1.20	74.125.250.129	STUN	62	Binding Request
11955	80.879987039	74.125.250.129	192.168.1.20	STUN	74	Binding Success Response XOR-MAPPED-ADDRESS: 84.76.199.200:34713
11956	80.931662033	192.168.1.20	74.125.250.129	STUN	62	Binding Request
11957	80.944283977	74.125.250.129	192.168.1.20	STUN	74	Binding Success Response XOR-MAPPED-ADDRESS: 84.76.199.200:55974

Si ara extenem el paquet de Binding Success Response podem veure la següent informació:

Session Traversal Utilities for NAT	
[Request In: 11956]	[Time: 0.012621944 seconds]
Message Type: 0x0101 (Binding Success Response)	.... .1 ...0 .... = Message Class: 0x10 Success Response (2) .00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001) .0. .... .... .... = Message Method Assignment: IETF Review (0x0)
Message Length: 12	
Message Cookie: 2112a442	
Message Transaction ID: 0ef6a05db106d1f31d051f72	
[STUN Network Version: RFC-5389/8489 (3)]	
Attributes	
XOR-MAPPED-ADDRESS: 84.76.199.200:55974	
Attribute Type: XOR-MAPPED-ADDRESS	0.... .... .... .... = Attribute Type Comprehension: Required (0x0) .0.... .... .... .... = Attribute Type Assignment: IETF Review (0x0)
Attribute Length: 8	
Reserved: 00	
Protocol Family: IPv4 (0x01)	
Port (XOR-d): fbb4	
[Port: 55974]	
IP (XOR-d): 755e638a	
[IP: 84.76.199.200]	

En aquest paquet les ips que podem trobar són :

### 1. Adreça Mapejada XOR (XOR-MAPPED-ADDRESS):

- IP: 84.76.199.200 Port: 55974

En aquest cas, l'adreça mapejada (XOR-MAPPED-ADDRESS) del client és 84.76.199.200:55974. Aquesta adreça IP és una adreça pública, la qual cosa indica que el client està darrere d'una NAT. Això es pot deduir perquè l'adreça IP pública (84.76.199.200) és diferent de l'adreça IP privada del client, que hauria de ser la 192.168.1.20.

Així doncs, el Binding Success Response mostra que l'adreça i el port que veu el servidor STUN per al client són diferents dels que el client té a la seva pròpia xarxa privada. Això confirma que hi ha una NAT entre el client i el

servidor STUN, ja que la NAT està canviant l'adreça IP i el port del client en sortir cap a Internet i anant a un servidor STUN públic.

Cuenta X

Nombre de cuenta	upcconnect		
Servidor SIP	upc.edu		
Proxy SIP	upcconnect.upc.es:5062		
Usuario *	user05.api		
Dominio *	upc.edu		
Iniciar sesión	user05.api		
Contraseña	*****		
Nombre para mostrar	jiale.chen		
Núm. buzón de voz	*86		
Prefijo de Marcación			
Plan de marcado			
<input type="checkbox"/> Hide Caller ID			
Comunicación cifrada	Desactivado		
Transporte	UDP		
Dirección pública	Automático		
Refresco de Registro	300	Mantener Conexión	15
<input type="checkbox"/> Publicar presencia			
<input type="checkbox"/> Permitir reescritura IP			
<input type="checkbox"/> ICE			
<input type="checkbox"/> Desactivar temporiz. de sesión			

x Guardar Cancelar

## **Conclusió**

Al llarg d'aquesta pràctica de laboratori, hem abordat de manera detallada els fonaments i aplicacions pràctiques de la telefonía IP. Hem configurat i utilitzat diferents clients i servidors SIP, analitzat diàlegs i fluxos RTP amb Wireshark, i experimentat amb diversos escenaris per avaluar la qualitat del servei. Aquests exercicis no només han aprofundit la nostra comprensió tècnica de VoIP, sinó que també han demostrat la importància de la correcta configuració i gestió dels recursos de xarxa per garantir comunicacions efectives i d'alta qualitat.