



ESCUELA SUPERIOR DE INGENIERÍA

INGENIERO TÉCNICO INFORMÁTICA

SISTEMAS

DISEÑO E IMPLEMENTACIÓN DE UN SISTEMA VoIP BASADO EN CISCO Y ASTERISK

DEPARTAMENTO: INGENIERÍA EN AUTOMÁTICA,
ELECTRÓNICA, ARQUITECTURA Y REDES DE COMPUTADORES

DIRECTOR DEL PROYECTO : FERNANDO PÉREZ PEÑA

AUTOR DEL PROYECTO : DANIEL SÁNCHEZ BENÍTEZ

Cádiz, Julio 2017

Fdo: Daniel Sánchez Benítez

Agradecimientos

Este proyecto no se podría haber llevado a cabo sin la colaboración y ayuda de un gran número de personas.

En primer lugar quiero dar las gracias a mi director de proyecto Fernando Pérez Peña, por darme la oportunidad de poder contar con su ayuda, guiándome y aconsejándome durante estos meses.

También quiero dar las gracias a la Universidad de Cádiz y al departamento de ingeniería en automática electrónica, arquitectura y redes de computadores por cederme el laboratorio para llevar a cabo la parte práctica del proyecto.

Gracias a toda mi familia y amigos por darme su apoyo durante toda la carrera.

No quiero acabar estas líneas sin antes agradecer a la persona que más me ha ayudado, tanto en los buenos y sobre todo en los malos momentos. Gracias María.

Índice de contenidos

1	Introducción	1
1.1	Objetivos	2
1.2	Alcance	2
2	Historia	3
2.1	Nacimiento VoIP	3
2.1.1	1975	3
2.1.2	1995	4
2.1.3	1996	4
2.1.4	1998	4
2.1.5	1999	4
2.2	Evolución del VoIP	4
2.2.1	2000	4
2.2.2	2003	5
2.2.3	2005	5
2.2.4	2006	5
2.2.5	2009	5
2.2.6	2010	5
2.2.7	2011	5
2.3	Actualidad del VoIP	5
2.3.1	2012	5
2.3.2	2017	6
3	¿Qué es y por que usar VoIP?	7
3.1	Telefonía IP vs Telefonía Convencional	8
4	VoIP y el modelo OSI	11
4.1	Protocolos de control en las transmisiones H.323	13
4.2	Protocolos de control en las transmisiones SIP	14
4.3	Protocolos de datos en las transmisiones H.323/SIP	15
5	Protocolos que intervienen VoIP	17
5.1	H.323	19
5.1.1	Arquitectura del protocolo H.323	20
5.1.2	Funcionamiento del protocolo H.323	21
5.1.3	Ventajas	24
5.1.4	Desventajas	24

5.2	SIP (Session Initiation Protocol)	25
5.2.1	Arquitectura SIP	25
5.2.2	Funcionamiento SIP	26
5.2.3	Ventajas del SIP	27
5.2.4	Desventajas del SIP	27
5.3	IAX2 (Inter-Asterisk eXchange)	28
5.3.1	Arquitectura del protocolo IAX2	28
5.3.2	Funcionamiento IAX2	28
5.3.3	Ventajas del IAX2	29
5.3.4	Desventajas del IAX2	29
5.4	SCCP (Skinny Client Control Protocol)	30
5.4.1	Arquitectura del SCCP	30
5.4.2	Funcionamiento del SCCP	32
5.4.3	Ventajas del SCCP	35
5.4.4	Desventajas del SCCP	35
6	Descripción de los materiales a usar	36
7	Cisco CCNA Voice	38
7.1	Configuración Cisco	38
7.1.1	Configuración básica de puertos del router	39
7.1.2	Configurando DHCP para VoIP y Datos	39
7.1.3	Configuración del Servicio VoIP	40
7.1.4	Creando directorio de números	40
7.1.5	Configuración terminales	40
7.2	Configuración Switch	41
7.2.1	Configurando las Vlans rama a	41
7.2.2	Configurando las Vlans rama b	41
8	Asterisk FreePBX	42
8.1	Creación de usuarios desde Terminal	42
8.1.1	Configuración general (codecs audio/video, Quality of service)	43
8.1.2	Creación de usuarios	44
8.1.3	Configuración de extensiones	46
8.1.4	Configuración QOS (Calidad de servicio)	47
9	Configuración Router con firmware dd-wrt	48
10	Monitorización de codecs	50
10.1	Codec g711-u (pcmu - ulaw)	52
10.2	Codec g711-a (pcma - alaw)	53
10.3	Codec g722	54
10.4	Codec GSM (RPE-LPC)	55
10.5	Codec iLBC	56
10.6	Codec Speex8	57
10.7	Codec Speex16	58
10.8	Codec Speex32	59
11	Conclusión	60
Anexo I:	Cisco CCNA Voice	62

Anexo II: Asterisk FreePBX	77
Anexo III: Router Neutro (firmware dd-wrt)	105
Anexo IV: Monitorización protocolo SIP Wireshark	110
Anexo V: Monitorización protocolo SCCP Wireshark	153
Anexo Conclusión	166
Bibliografía	171

Índice de imágenes

3.1 Red telefónica conmutada	8
4.1 Pila de control en protocolo H.323	13
4.2 Pila de control en protocolo SIP	14
4.3 Pila de datos en protocolo H.323/SIP	15
5.1 Protocolo h.323	19
5.2 Funcionamiento del protocolo H.323	23
5.3 Funcionamiento del protocolo SIP	26
5.4 Funcionamiento del protocolo IAX2	29
5.5 Funcionamiento del protocolo SCCP	32
7.1 Red Cisco	38
10.1 Monitorización ancho de banda codec g711-u	52
10.2 Monitorización ancho de banda codec g711-a	53
10.3 Monitorización ancho de banda codec g722	54
10.4 Monitorización ancho de banda codec GSM	55
10.5 Monitorización ancho de banda codec iLBC	56
10.6 Monitorización ancho de banda codec Speex8	57
10.7 Monitorización ancho de banda codec Speex16	58
10.8 Monitorización ancho de banda codec Speex32	59

Índice de tablas

3.1	Pros y Contras telefonía IP	9
3.2	Pros y Contras telefonía tradicional	10
4.1	Modelo Osi aplicado VOIP	11
4.2	Codecs y características	16
6.1	Materiales de laboratorio	37
8.1	Quality of service: Tos	43
8.2	Quality of service: Cos	44
11.1	Tabla de resultados.	60
11.4	Configuración del servicio VoIP	69

Capítulo 1

Introducción

Índice

1.1	Objetivos	2
1.2	Alcance	2

La telefonía IP [8]¹ fue creada para la integración de voz y video dentro de una misma red IP (redes convergentes²).

La proliferación de esta tecnología es debido a diferentes factores. Cabe destacar la necesidad de eliminar infraestructuras obsoletas. Otro punto fundamental a tener en cuenta, es el crecimiento de las velocidades en las comunicaciones. Sumado a la aparición de los protocolos de control y transporte que han hecho posible la implementación de voz a través de la infraestructura de datos. Por lo tanto, se reducen considerablemente los costes en arquitectura física, ya que van integradas en una misma red. De esta manera, se "abaratara" el servicio para los clientes finales (Empresa o cliente).

La telefonía IP está formada por:

- Terminales clientes IP. Son los encargados de generar las llamadas de voz. Cabe destacar las siguientes plataformas:
 - Móvil. Usan software específico (Softphone) para realizar llamadas a través del wifi o de la conexión de datos (3g o 4g).
 - Ordenador. Usan software específico (Softphone) para realizar llamadas a través del wifi.
 - Tablets. Usan software específico (Softphone) para realizar llamadas a través del wifi o del la conexión de datos (3g o 4g).
 - Teléfonos IP. Teléfonos que transmiten voz a través de la red IP.
 - Teléfonos analógicos adaptados (ATAs). Teléfonos tradicionales adaptados para su uso a través de la red IP.
- Gateway IP (Puertas de enlace). Provee interconexión entre terminales (analógica o digital)
- Centralita IP (Servidor). Función principal es gestionar las llamadas internas y externas.

¹IP: Internet Protocol o Protocolo de internet.

²Redes convergentes: Son redes que integran en una misma infraestructura los servicios de datos, voz y video. Eliminando la necesidad de diferentes plataformas físicas para usos específicos.

1.1 Objetivos

El proyecto, titulado "Diseño e implementación de un sistema VoIP basado en Cisco y Asterisk", tiene como objetivo el diseño, implementación y configuración de dos redes piloto VoIP (Voz sobre IP). Ambas redes serán similares y su carácter piloto es debido a que son redes pequeñas y experimentales mediante las que se persigue una comparativa entre ambas y la demostración de una validez operativa.

La primera de las redes estará basada en electrónica de red proporcionada por la empresa Cisco (disponible en la Escuela Superior de Ingeniería). Por tanto, se tratará de una red que explotará características específicas de los equipos Cisco; siendo estos equipos los que hagan todas las tareas necesarias para una comunicación satisfactoria sobre una red IP. Previo al despliegue de la red en el laboratorio y una vez diseñada, se empleará el software Packet Tracer para hacer una simulación de su funcionamiento.

La segunda red, mantendrá la electrónica de red de Cisco, pero en este caso, incluirá un servidor Asterisk que será el encargado de realizar las tareas de centralita para las comunicaciones de voz. Este servidor es gratuito y de acceso libre (Free and Open Source framework).

1.2 Alcance

El alcance del proyecto incluye el diseño a nivel de tecnología y electrónica de red (estudiando los requisitos de calidad de la red en términos de ancho de banda, calidad de servicio, etc.), la configuración de todos los equipos empleados, la simulación del comportamiento en términos de tráfico y, por último, el despliegue en el laboratorio de redes de computadores (Escuela Superior de Ingeniería) de la red y el análisis de su comportamiento. Como conclusión del proyecto, se realizará una comparativa entre ambas redes mediante medidas del rendimiento.

Capítulo 2

Historia

Índice

2.1	Nacimiento VoIP	3
2.1.1	1975	3
2.1.2	1995	4
2.1.3	1996	4
2.1.4	1998	4
2.1.5	1999	4
2.2	Evolución del VoIP	4
2.2.1	2000	4
2.2.2	2003	5
2.2.3	2005	5
2.2.4	2006	5
2.2.5	2009	5
2.2.6	2010	5
2.2.7	2011	5
2.3	Actualidad del VoIP	5
2.3.1	2012	5
2.3.2	2017	6

2.1 Nacimiento VoIP

2.1.1 1975

[9, 10] Se crea el protocolo Sistema de Señalización por canal común número 7 (SS7).

Es un conjunto de protocolos de control y señalización telefónica, desarrollado por la empresa AT&T en 1975, convirtiéndose en un estándar para las telecomunicaciones en 1981. El propósito de este estándar es el establecimiento y la finalización de las llamadas. Cabe destacar, la importancia de los enlaces de las redes digitales (tráfico VoIP) y las redes públicas (PSTN).

Usa un sistema de señalización fuera de linea (fuera de banda). Es decir, se usan 2 canales diferentes. Uno se dedica a la señalización y el otro a la comunicación. Con ello, evita problemas de seguridad, ya que el usuario no puede acceder a ese canal.

2.1.2 1995

La empresa VocalTec Communications Inc. pionera en el lanzamiento del primer Softphone¹. Este programa comprimía los datos de voz, los fragmentaba en paquetes más pequeños y los enviaba por internet. Esta comunicación solo era posible cuando las 2 máquinas eran idénticas.

2.1.3 1996

Mark Handley y Eve Schooler presentan el borrador del SIPv1².

Se incorporó el uso de la tecnología VoIP en las redes PSTN³ (red telefónica convencional).

Nace el protocolo H.323, diseñado y desarrollado por la ITU-T⁴.

2.1.4 1998

Fabricación de los primeros ATA/gateways para poder hacer llamadas entre los PC y teléfonos convencionales y/o teléfonos convencionales entre sí a partir de la colocación de convertidores (ATAs) en los extremos.

Se crea la página web WhichVoIP.com, creada por un empleado de VocalTec. Permitía conectar PCs y teléfonos convencionales.

Se crea la empresa Peoplecall, permitía la gratuidad del servicio a cambio de la instalación de un programa.

Se crea la empresa LlamadaIP, primera empresa de lengua no sajona (Argentina) en dedicarse a las llamadas VoIP.

2.1.5 1999

Se crea el estándar MGCP⁵, también conocido como H.248 o Megaco. Es un protocolo del tipo cliente - servidor, el cual, define los mecanismos de control necesarios para gestionar la señalización y las sesiones de voz/fax entre PSTN-IP o IP-IP.

2.2 Evolución del VoIP

2.2.1 2000

Nace Asterisk, como primera central telefónica (PC) basada en GNU/linux bajo licencia libre GPL. En el siguiente año nacería la empresa Digium.

Nace Yahoo Instant Messenger (YIM), ahora llamado Yahoo! MEssenger. Cuenta con la posibilidad de hacer llamadas en grupo, intercambiar mensajes de texto y realizar videollamadas.

¹Softphone: Internet Phone Software.

²SIP: Session Initiation Protocol a la IETF.

³PSTN: Public Switched Telephone Network.

⁴ITU-T: International Telecommunication Union.

⁵MGCP: Media Gateway Control Protocol - RFC 3435.

2.2.2 2003

Nace Skype (Microsoft), se creó como un softphone gratuito. A día de hoy, permite las comunicaciones tanto de voz, texto como videollamadas.

2.2.3 2005

Se crea la empresa Rebtel (Suecia), que ofrecía llamadas nacionales (sin coste) e internacionales (bajo coste) a partir de una aplicación para móviles.

Skype es comprada por ebay 8.500 millones de dólares (5.920 millones de euros).

2.2.4 2006

Se crearon las empresas de VoIP; Talkety, Lowratevoip.com y Fring.com como plataformas de llamadas a bajo coste.

2.2.5 2009

Se crea la aplicación multiplataforma Tango, que permitía llamadas a través de la web proporcionando cierta privacidad.

2.2.6 2010

Se crea la aplicación Google Voice que en sus primeras versiones estaba integrado en Gmail. En la actualidad, se puede acceder al servicio por medio de la aplicación Android, IOS o por la misma Web.

Se crea la aplicación multiplataforma Viber. Inicialmente solamente fue dirigido a los iPhones. Posteriormente fueron extendiéndose a diferentes plataformas Android(2011), Blackberry(2012), Windows Phone(2012) Windows(2013) y Mac(2013). Alcanzando 200 millones de usuarios en sus 3 años iniciales.

2.2.7 2011

Se crean las empresas como vox.io (no funcional), GNU Free Call o WeChat.

2.3 Actualidad del VoIP

2.3.1 2012

Gracias al uso masivo de móviles, las aplicaciones fueron obteniendo mayor relevancia (MessageTalk.com, Yulop, imo.im Line, ...)

2.3.2 2017

El nicho de mercado está bastante fraccionado. Cabe destacar Skype (Microsoft), Hangout (Google), Whatsapp (facebook).

Capítulo 3

¿Qué es y por que usar VoIP?

Índice

3.1 Telefonía IP vs Telefonía Convencional	8
--	---

VoIP (Voz sobre protocolo de Internet) es la tecnología que proporciona la posibilidad de transmitir voz a través de las redes IP. El proceso consiste en dividir los datos de video y audio en fragmentos y transmitirlos por las redes IP, estos fragmentos se reensamblan en el destino. Esta integración en las redes IP facilita los procesos de transmisión de voz que antes se realizaban con las redes tradicionales y analógicas (redes de voz PSTN¹).

Cabe destacar algunas de las ventajas más importantes como consecuencia de la implantación de este protocolo:

- **Reducción en costes:** Es una de las principales ventajas que presenta esta tecnología.
- **Servicios suplementarios:** Transferencia de llamadas, identificación. Son fácilmente implementables sobre esta tecnología (VoIP), ya que vienen de forma predeterminada. Solamente hace falta su configuración.
- **Integración de SmartPhones:** Gracias a las aplicaciones para móviles es bastante fácil integrar estos dispositivos dentro de las redes VoIP.
- **Libertad en la elección de Equipos:** Esta tecnología no está diseñada para su uso con una marca en concreto. Es un tecnología que se ha convertido en un estándar en la comunicación y que poco a poco está desbancando a la telefonía tradicional en su nicho de mercado.
- **Aprovechamiento del equipamiento existente:** No hace falta disponer de una red muy avanzada para implementar esta tecnología, incluso sería fácil su implementación a nivel de hogar. Después de la reducción de costes, esta es otra de las ventajas mas claras. Poder tener una única red que se encarga de toda la comunicación (voz, datos, video o cualquier otro tipo de información).
- **Fácil crecimiento:** Son fácilmente escalable sin tener que hacer modificaciones de hardware. Solamente habría que hacer modificaciones de configuración. Agregando y configurando nuevas cuentas de usuarios.
- **Alta calidad en llamadas telefónicas:** Gracias al avance de las conexiones a internet (velocidad) y algoritmos de compresión (G.711, G.726, G.729, GSM, iLBC, Speex) que

¹PSTN: Public Switched Telephone Network o Red telefónica conmutada.

pueden comprimir las llamadas hasta 8 kbps.

3.1 Telefonía IP vs Telefonía Convencional

La telefonía analógica convencional RTB² está de camino hacia su obsolescencia (siendo la más usada en los hogares), ya que utiliza tecnología RTC³ ineficiente. Este tipo de tecnología ha sido usada por las operadoras de telefonía desde hace más de 100 años.

No obstante, existen operadoras (Telefónica España S.A.) que está empezando a implementar VoIP sobre FTTH (Fibra Óptica).⁴

Este tipo de tecnología (telefonía analógica convencional) es llamada "circuito" debido a que se hacía una conexión punto a punto de manera bidireccional, siendo este el fundamento de la comunicación telefónica ordinaria o convencional. Pero la aparición del RDSI⁵, facilitó las conexiones digitales comprendidas entre dos extremos para proporcionar una gran gama de servicios. De los cuales, cabe destacar la transmisión de voz y de datos.

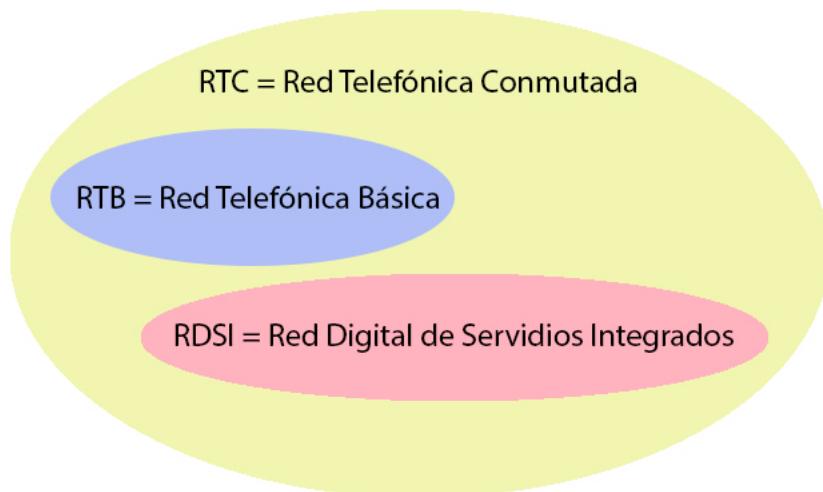


Imagen 3.1: Red telefónica conmutada

Esta tecnología se implantó en los años 90, pero no ha proliferado hasta la estandarización de los sistemas de control de calidad (QOS) y la llegada masiva de internet a los hogares.

²RTB: Red telefónica básica

³RTC: Red telefónica conmutada

⁴FTTH: Fiber To The Home.

⁵RDSI: Red digital de servicios integrados

Las características más destacables de la **Telefonía IP** son las siguientes:

PROS	CONTRAS
<ul style="list-style-type: none"> • Abaratamiento del servicio. Debido a la inclusión de este servicio sobre una misma red. • Conjunto de elementos integrados que suministran un servicio de telefonía. • Está compuesto por diferentes elementos (centralita IP, Gateway IP y teléfonos IP). • Simplifica la infraestructura de las comunicaciones en las empresas. • Gestión centralizada (llamadas internas, plan de numeración integrado, optimización de las comunicaciones, movilidad y acceso a buzón de voz, IVR, ACD, CTI, etc). 	<ul style="list-style-type: none"> • Conexiones de banda ancha. Proliferan hoy en día las conexiones (ADSL y RDSI). • Deslocalización del servicio VoIP. Implica problemas con llamadas a servicio de urgencias, ya que no existe asociación de dirección ip con área geográfica determinada. • No se da aún la calidad de servicio (QOS). Los datos derivados de las llamadas VoIP se ven afectados por los demás datos que viajan por el mismo canal. Dando como resultado, problemas de latencia o perdidas de paquetes. • Problemas de seguridad (Hackeo).

Tabla 3.1: Pros y Contras telefonía IP

Las características más destacables de la **Telefonía analógica Convencional** son las siguientes:

PROS	CONTRAS
<ul style="list-style-type: none"> • Funcionamiento a partir de una señal analógica. Este tipo de señal es el resultado de una fenómeno electromagnético, que dispone de una variación eléctrica entre una situación de signo positivo y otra de signo negativo, todo ello en intervalos de medio ciclo. Esto genera una gráfica matemática continua de carácter sinusoidal. • Commutación humana (telefonista) VS Commutación electromecánica. Antiguamente se creó esta figura laboral que se encargaba de dirigir las llamadas hacia su destino. Esta figura fue desapareciendo conforme se iba extendiendo el uso del teléfono modernizando sus centralitas. • Cada central o nodo, atiende las líneas de sus abonados. • Cada nodo está unido a diferentes nodos, formando el sistema telefónico nacional, que a su vez está unido al sistema telefónico internacional. 	<ul style="list-style-type: none"> • Pérdidas de calidad (ruido) y caídas de conexión. Nos referimos a ruido cuando hablamos de cualquier señal que pueda interferir directa o indirectamente en una llamada. • Cada teléfono tiene una dirección telefónica. Haciendo más difícil la deslocalización física de un número. • Están construidas por dos hilos (par de cobre). Quedando totalmente en desuso en la actualidad.

Tabla 3.2: Pros y Contras telefonía tradicional

Capítulo 4

VoIP y el modelo OSI

Índice

4.1	Protocolos de control en las transmisiones H.323	13
4.2	Protocolos de control en las transmisiones SIP	14
4.3	Protocolos de datos en las transmisiones H.323/SIP	15

El modelo OSI (Open System Interconnection) es una referencia, formada por 7 capas para los protocolos de red. Se caracteriza por estandarizar las funciones de los sistemas de comunicaciones.

Se desarrolló en 1980 por la International Organization for Standardization (ISO), la publicación de su estándar no fue hasta 1984.

Este estándar crea las normas básicas para la intercomunicación en Internet, dada la gran masificación de tecnologías, fabricantes y compañías.

Cuando se realiza una llamada IP, se implementan diferentes protocolos a través de las capas OSI:

Núm capa	Modelo OSI	Protocolos VoIP
7	Capa de Aplicación	Programas y aplicaciones VoIP, SDP
6	Capa de Presentación	Codecs (G.711, G.722, GSM, G.729, Speex)
5	Capa de Sesión	H.323, SIP, IAX
4	Capa de Transporte	RTP, RTCP, TCP, UDP, SCCP
3	Capa de Red	IP (IPv4, IPv6), Tos
2	Capa de Enlace de datos	MAC, 802.1Q (VLANs), Cos
1	Capa Física	802.11 (WLAN), 802.3 (Ethernet)

Tabla 4.1: Modelo Osi aplicado VOIP

VoIP es una tecnología que está formada por protocolos que funcionan en la capa de red y utiliza varios protocolos la capa de enlace de datos. Esta tecnología permite a los routers, switches y servidores acceder a los multiservicios para enviar y recibir voz, datos y video a través de la red IP.

- **La capa de Aplicación.** Es la encargada de proporcionar los servicios que son soportados por las aplicaciones (softphones, aplicaciones para mensajería, aplicaciones para videoconferencias).

- **La capa de Presentación.** Se encarga de cifrar y comprimir los datos que se van a usar posteriormente en la capa de aplicaciones. Los paquetes de voz son encriptados y comprimidos según el formato que sea (Video o voz).
- **Capa de Sesión.** La labor de esta capa es la de establecer los enlaces de comunicaciones entre los dispositivos (emisor y receptor), también gestiona la sesión que se establece entre los dos dispositivos.

Al activarse la sesión entre los dispositivos, esta capa ofrece en cierta medida la tolerancia a fallos mientras que esté abierta la sesión. Si se pierde la comunicación, los datos se apilarán esperando a restablecer la comunicación y enviar de nuevo los datos que faltan. De esta manera, se evita mandar todos los paquetes de nuevo.

Existen dos tipos de comunicaciones en esta capa:

- Comunicaciones orientadas a la conexión.
- Comunicaciones sin conexión.

La que nos interesa es la orientada a la conexión, ya que los protocolos H.323, SIP y MGCP funcionan de esta manera.

- **Capa de Transporte.** Esta capa se encarga de controlar el flujo de datos, también evalúa el tamaño de los paquetes. Las comunicaciones VoIP usan el protocolo RTP y RTCP para la señalización y protocolo de control.

Características y funciones de los protocolos RTP y RTCP:

- Pueden usar el modo Unicast (punto a punto) y multicast (multipunto).
- Identifican el tipo de información en cada momento (voz, video o datos).
- Regular el flujo de datos, cuantificar retardos y evitar fluctuaciones.
- Secuenciación y numeración de paquetes para detectar perdidas.
- Usa por defecto los puertos 5004 UDP y 5005 UDP.

- **Capa de Red.** Proporciona enrutamiento y direccionamiento de los paquetes. Esta capa es la encargada de convertir las direcciones lógicas (direcciones IP) en direcciones físicas (MAC).
- La **Capa de Enlace de datos.** Su principal función es la de corrección de errores que provienen de la capa física. La tecnología VoIP no presenta funciones en la capa física. Sobre la capa de enlace la función que cabe destacar es la Calidad de servicio (QOS).
- La **Capa Física.** Es la encargada de dar soporte físico necesario para establecer las transmisiones de datos a través de la red.

4.1 Protocolos de control en las transmisiones H.323

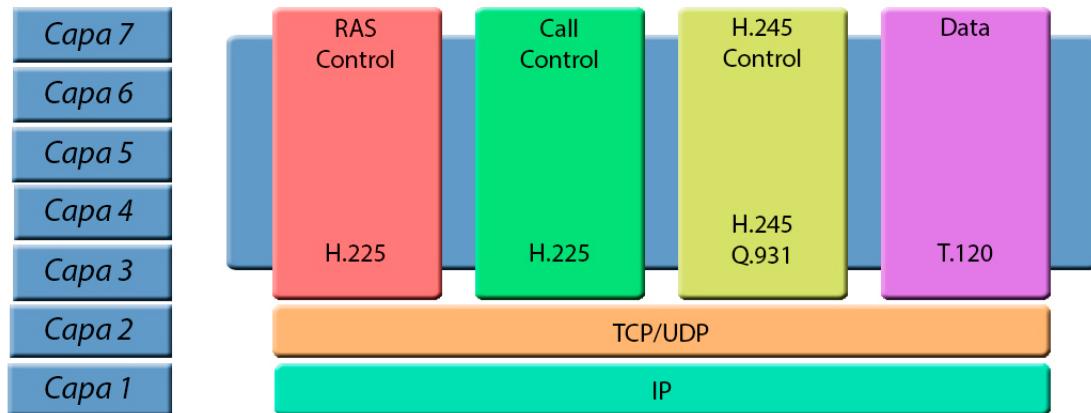


Imagen 4.1: Pila de control en protocolo H.323

- **H.225** - Este protocolo es el encargado:
 - RAS¹: Registro, Admisión y Estado.
 - Señalización de la llamada.
 - Anexo G.²
- **H.245** - Protocolo de control multimedia. Sus funciones son las siguientes:
 - Intercambios de capacidades.
 - Apertura y cierre de canales lógicos.
 - Peticiones de preferencias.
 - Mensajes de control de flujo.

Por último, se transportan los datos multimedia por RTP.

- **Q.931** - Protocolo de control RDSI³. Es una especificación de capa 3 (capa de red) para el control básico de llamadas.
- **T.120** - Protocolo que da soporte en tiempo real y comunicación multipunto.

¹RAS: Registration Administration Status o Registro, Admisión y Estado

²Anexo G: Comunicaciones entre dominios.

³RDSI: Red Digital de Servicios Integrados

4.2 Protocolos de control en las transmisiones SIP

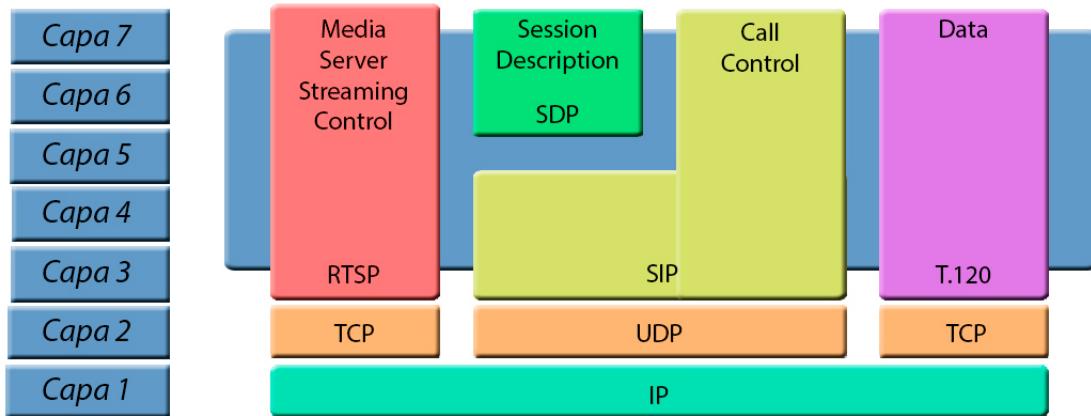


Imagen 4.2: Pila de control en protocolo SIP

- **RTSP⁴** - Se encarga de controlar el flujo de datos (audio y video). Permite el control remoto del servidor de streaming.
- **SDP⁵** - Es el protocolo encargado de negociar el ancho de banda y codecs involucrados entre los terminales.
- **SIP⁶** - Sus funciones como protocolo, son las siguientes:
 - Señalización.
 - Establecimiento.
 - Control de sesión.
 - Terminación de sesión.

⁴RTSP: Real Time Streaming Protocol

⁵SDP: Session Description Protocol

⁶SIP: Session Initiation Protocol

4.3 Protocolos de datos en las transmisiones H.323/SIP

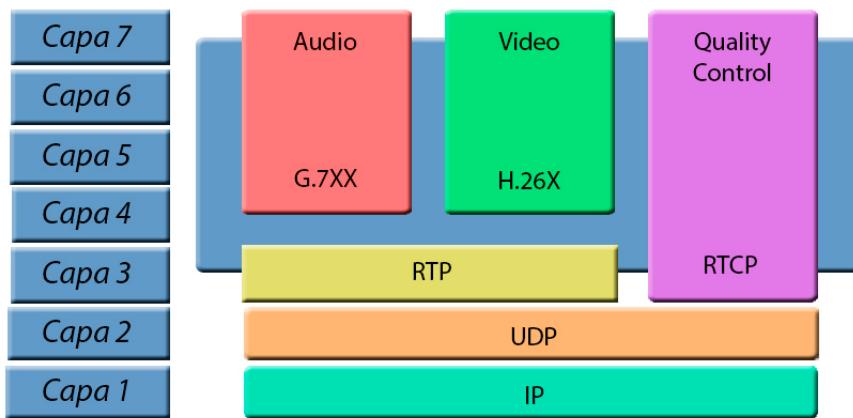


Imagen 4.3: Pila de datos en protocolo H.323/SIP

- **G.711** - Modulación por código de impulsos (PCM) de frecuencias de voz. Conversión analógico - digital, son realizados gracias a 3 pasos:
 - Muestreo: Tomar señales analógicas a intervalos de tiempo con ritmo uniforme.
 - Cuantificación: Asignación de valores matemáticos discretos a las muestras obtenidas en el muestreo.
 - Codificación: Existen 2 tipos:
 - * Continuas: Intervalos ordenados con señales de nivel bajo y niveles alto
 - * Segmentos: Intervalos de grupos. Dentro de cada grupo existen intervalos iguales, lo que los distingue de otros grupos.
- **G.723.1** - Codecs de voz de comunicaciones multimedia que se transmiten a 5.3 y 6.3 kbit/s.
- **G.729** - Codificación de la voz a 8kbit/s mediante predicción lineal con excitación por código algebraico de estructura conjugada.
- **H.264** - Codificación de video avanzada para servicios audiovisuales genéricos.
- **H.263** - Estándar para la codificación de video de baja velocidad.
- **H.261** - Estándar para la codificación de video a 64kbit/s.
- **RTP** - Real-Time Protocol o Protocolo a tiempo real. Este estándar es usado para la transmisión de voz y video a través de internet. Las funciones más importantes son:
 - Seguridad e identificar la información.
 - Detección de pérdidas.
 - Creado para su uso multicast y unicast.⁷ ⁸
 - Añadir marcadores temporales.

⁷Multicast: Envío de información a múltiples redes simultáneamente.

⁸Unicast: De un emisor hacia un único receptor.

- Control de llegada de los paquetes.
- **RTCP** - Real-Time Control Protocol o Protocolo de control del RTP. Su labor es la de controlar el flujo RTP, garantizar este flujo y la calidad del servicio.

Tipo de códec	Nombre	Ancho de banda	Retardo
G.711 u - a GSM	PCM: Pulse Code Modulation. RPE-LTP: Regular Pulse Excitation LongTerm Predictor.	56 - 64 kbit/s 13 kbit/s	1 ms 22.5 ms
iLBC	Internet Low Bitrate Codec	15.2 kbit/s - 13.33 kbit/s	20ms - 30 ms
SPEEX SPEEX16 SPEEX32	Narrowband (8 KHZ) WideBand (16 KHZ) Ultra-wideband (32 KHZ)	24.6 kbit/s 42.2 kbit/s 44.0 kbit/s	30 ms 34 ms 34 ms

Tabla 4.2: Codecs y características

Capítulo 5

Protocolos que intervienen VoIP

Índice

5.1 H.323	19
5.1.1 Arquitectura del protocolo H.323	20
5.1.2 Funcionamiento del protocolo H.323	21
5.1.3 Ventajas	24
5.1.4 Desventajas	24
5.2 SIP (Session Initiation Protocol)	25
5.2.1 Arquitectura SIP	25
5.2.2 Funcionamiento SIP	26
5.2.3 Ventajas del SIP	27
5.2.4 Desventajas del SIP	27
5.3 IAX2 (Inter-Asterisk eXchange)	28
5.3.1 Arquitectura del protocolo IAX2	28
5.3.2 Funcionamiento IAX2	28
5.3.3 Ventajas del IAX2	29
5.3.4 Desventajas del IAX2	29
5.4 SCCP (Skinny Client Control Protocol)	30
5.4.1 Arquitectura del SCCP	30
5.4.2 Funcionamiento del SCCP	32
5.4.3 Ventajas del SCCP	35
5.4.4 Desventajas del SCCP	35

Existen varios protocolos usados para realizar llamadas VOIP, estos protocolos se detallarán seguidamente por orden de antigüedad:

- **H.323**. Estándar para la comunicación multimedia. Definido por la ITU-T ¹
- **SIP** (Session Initiation Protocol o Protocolo de Inicio de Sesiones). Definido por la IETF. ²
- **Megaco (H.248) + MGCP** (Media Gateway Control Protocol o protocolo de control). Este protocolo permite la comunicación de voz, fax y multimedia entre redes PSTN y las redes IP más avanzadas.

¹ITU-T = International Telecommunication Union

²IETF = Internet Engineering Task Force

- **UNIStim.** Propiedad de Nortel (Avaya).
- **SCCP** (Skinny Client Control Protocol). Propiedad de Cisco.
- **MiNet.** Propiedad de Mitel.
- **CorNet-IP.** Propiedad de Siemens.
- **IAX.** Protocolo original entre comunicaciones PBX Asterisk (estandar de comunicaciones).
- **Skype.** propiedad de Microsoft. Usa el protocolo p2p o red punto a punto.
- **IAX2** reemplazó IAX. Protocolo estándar para las comunicaciones PBX Asterisk.
- **Jingle.** Código abierto, se usa en tecnologías XMPP.
- **weSIP.** Licencia gratuita para uso no comercial.

5.1 H.323

Es un protocolo diseñado y desarrollado por la ITU-T (International Telecommunication Union) en 1996. En el se describen el conjunto de especificaciones (estándares) para el transporte de datos (paquetes) para ofrecer servicios multimedia. Las principales características que ofrece el protocolo H.323 [11 - 14], son los siguientes:

- Interoperabilidad entre distintos fabricantes e independencia de la red.
- Independencia de la plataforma y de la aplicación.
- Soporte para multiconferencias.
- Gestión del ancho de banda, gestión de direccionamientos y uso de DNS para resolución de direcciones.
- Transmisión en multicast.
- Seguridad entre comunicaciones.
- Llamadas rápidas (Fast Call).
- Establecer calidad de servicio (QOS) y su monitorización.
- Control de problemas en la red (redundancias cíclicas, caídas de red, ...)
- Otros servicios o suplementarios.
 - Desvío de llamadas.
 - Llamada en espera.
 - Recuperación de llamada (On hold).
 - Identificador de llamadas.
 - Prioridad de llamadas.
- Gestión de movilidad de los perfiles.

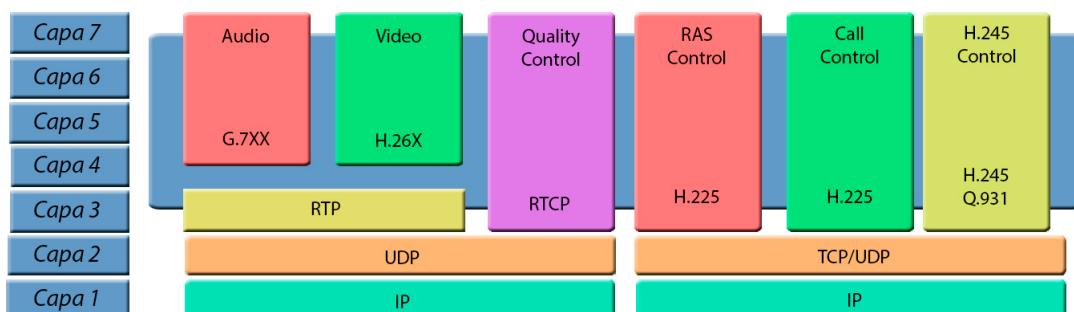


Imagen 5.1: Protocolo h.323

Este protocolo está compuesto por varios protocolos. Los cuales, tienen una función específica.

- **V.150** - Modem sobre redes de protocolo internet: Procedimientos para la conexión de extremo a extremo de los equipos de terminación del circuito de datos de la serie V.
- **T.120** - Protocolo que da soporte en tiempo real y comunicación multipunto.

- **T.38** - Procedimientos para la comunicación facsimil en tiempo real entre terminales facsimil del grupo 3 por redes con protocolo internet.

Control de Multimedia (Codecs específicos de audio).

- **G.711** - Modulación por código de implusos (PCM) de frecuencias de voz.
- **G.723.1** - Codecs de voz de comunicaciones multimedia que se transmiten a 5.3 y 6.3 kbit/s.
- **G.729** - Codificación de la voz a 8kbit/s mediante predicción lineal con excitación por código algebraico de estructura conjugada.
- **RTP** - Real-Time Protocol o Protocolo a tiempo real.
- **RTCP** - RTP Control Protocol o Protocolo de control del RTP.

Control y gestión de terminales.

- **H.225** - Contiene tres protocolos: RAS ³, señalización de llamadas y Anexo G.
- **H.245** - Protocolo de control multimedia.
- **H.235** - Protocolo referido a la seguridad en redes que conectan sistemas de control H.245.
- **H.450** - Servicios Suplementarios.
- **H.460** - Extensiones del H.323.
- **H.281** - Protocolo que describe el control de cámara para movimientos lejanos PTZ. ⁴

5.1.1 Arquitectura del protocolo H.323

El protocolo estándar H.323 está compuesto por 4 partes:

- **Gateway.** Es un sistema que se encarga de dar permisos a los equipos que tengan el protocolo H.323 puedan comunicarse con otras redes. Entre sus funciones más importantes, cabe destacar:
 - Señalización.
 - Información de control.
 - Información de los usuarios.
 - codificación de audio y video.

Todo esto consigue que haya una gran interoperabilidad entre las redes, entre los terminales y entre los servicios, optimizando los servicios aun siendo diferentes plataformas (PSTN o redes IP).

- **GateKeeper.** Hace la función entre otras, de traducir las direcciones y controlar el acceso a la red por parte de los terminales, gateways y MCU.

Las funciones que realiza el GateKeeper, son las siguientes:

- Controlar la señalización.

³RAS: Registration Administration Status o Registro, Admisión y Estado

⁴PTZ: Pan Tilt Zoom

- Controlar el acceso y administrar los recursos y autorizar las llamadas.
- Traducir direcciones IP.
- Gestionar el ancho de banda.
- Gestionar las llamadas.

Para realizar estas funciones, el GateKeeper emplea el protocolo RAS sobre UDP 1719⁵.

- **Terminales.** Está en los extremos de la red, son los encargados de la comunicación bidireccional. Los terminales se comunican no solo con los terminales, sino que también se comunican con los gateway y unidad de control multipunto (MCU). Esta comunicación consta de señales de control, audio, video y datos.

Las funciones que realizan los terminales son los siguientes:

- Negociar el canal, protocolo H.245.
- Señalizar y controlar la llamada, protocolo H.225 (RAS).
- Comunicación con el gatekeeper, protocolo H.225 (RAS).

- **Unidad de control multipunto o multipoint control unit (MCU).** Es el encargado de controlar las capacidades de procesamiento de audio, video y el control de la multidifusión.

Las señales de audio se digitalizan y se comprimen. Esta compresión se realiza a partir de uno de los algoritmos (G.711 o G.723).

Las señales de video en una comunicación pueden ser opcionales, pero cuando se efectúan se rigen por el protocolo H.261, H.263 o H.264. Estas señales son gestionadas a partir del estándar T.120 para poder realizar las conferencias multipunto o punto - punto.

Para garantizar un buen funcionamiento se apoya en un buffer de recepción, que amortigua el posible jitter que pueda aparecer en los flujos de información en las comunicaciones de audio y video.

5.1.2 Funcionamiento del protocolo H.323

Para realizar una llamada con este protocolo, se realizan 4 fases:

1. Establecimiento.

- El Terminal A, envía una petición(**ARQ**⁶) al GateKeeper. El GateKeeper admite o rechaza la petición, mandando los siguientes mensajes **ACF**⁷ o **ARJ**⁸ que están contenidos en el protocolo RAS (Registration Administration Status).
- El terminal A, envía un mensaje de **SETUP** hacia el terminal B, usando el protocolo h.225 (Call signaling).
- El terminal B responde al anterior mensaje con un mensaje **CALL PROCEEDING** (procedimiento de llamada) que recibe el terminal A.

⁵UDP 1719: Registro del gatekeeper.

⁶ARQ: Admision Request

⁷ACF: Admision Confirm

⁸ARJ: Admision Reject

- El terminal B procede a enviar una petición **ARQ** al GateKeeper, este último responde a esta petición, con un mensaje de confirmación **ACF** o de rechazo **ARJ**.
- Se activa el terminal B y envía un mensaje **ALERTING**, con el que comienza la fase de generación de tono.
- El mismo terminal envía un último mensaje **CONNECT** al terminal A para activar la conexión.

2. Señalización de Control.

- A continuación, los terminales (A y B) comienzan una negociación e intercambio de mensajes para establecer quien gestiona la conexión (master o slave). Esta gestión la realizar el **protocolo h.245**. También se gestionan los codecs de audio y video.

3. Audio.

- El terminal A, envía una petición de apertura del canal de audio. El terminal B lo recibe, lo habilita y devuelve una la misma petición con un **acuse de recibo (ACK)**.
- El terminal B, realiza la misma petición al terminal A, y este terminal recibe y habilita el canal de audio enviando un **acuse de recibo (ACK)** hacia el terminal B.

4. Desconexión.

- El proceso de desconexión puede ser realizado por cualquiera de los dos terminales, se llevará a cabo a partir de los mensajes **CLOSE LOGICAL CHANNEL** y **END SESSION COMMAND**, que está contenido en el protocolo h.245.
- Finalmente el protocolo h.225 cierra la conexión a través del mensaje **RELEASE**.

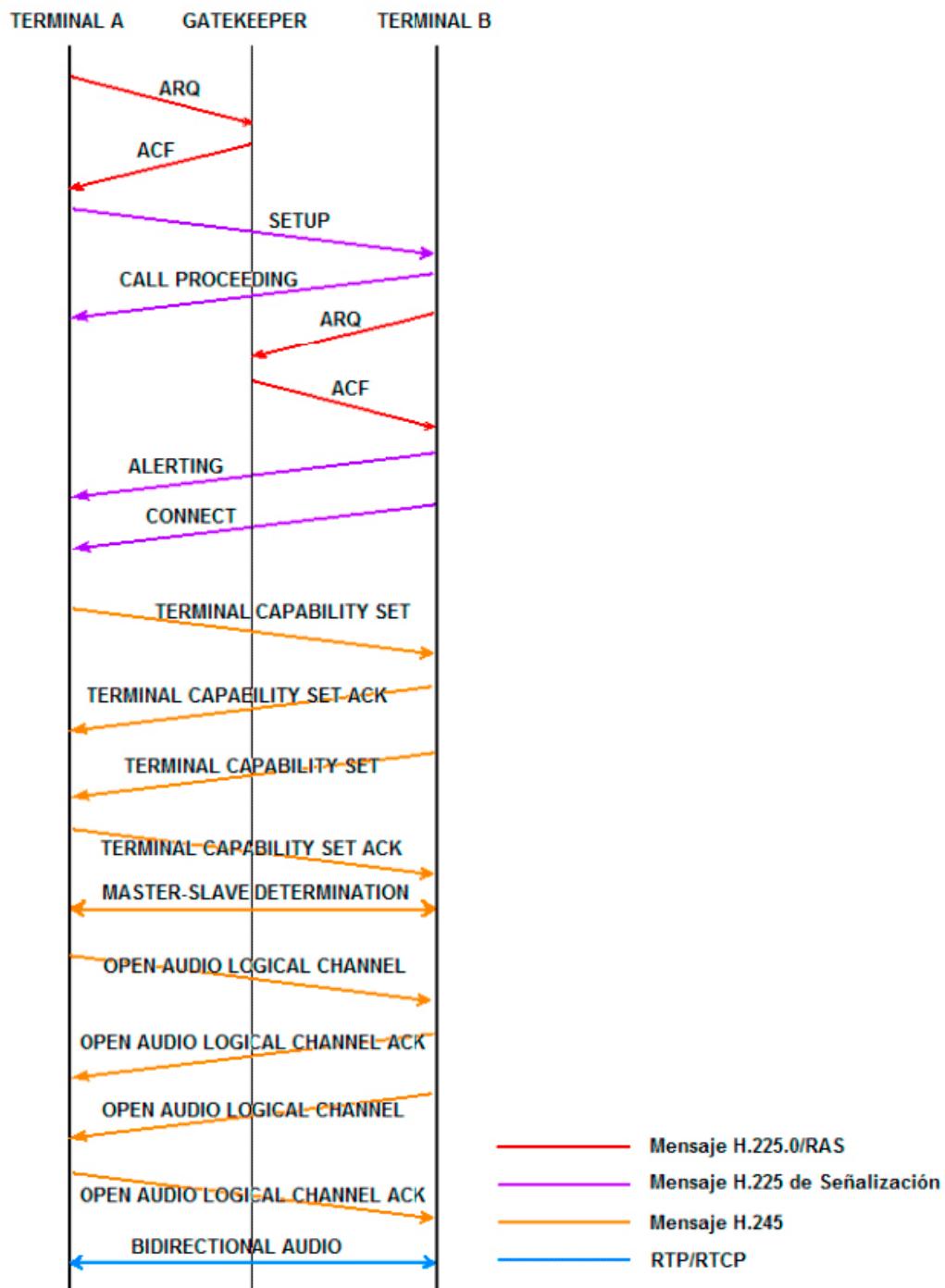


Imagen 5.2: Funcionamiento del protocolo H.323

⁹[http://www.grc.upv.es/docencia/tdm/trabajos2007/Abel_H.323%20vs%20SIP%20\(1\).pdf-figura2](http://www.grc.upv.es/docencia/tdm/trabajos2007/Abel_H.323%20vs%20SIP%20(1).pdf-figura2)

5.1.3 Ventajas

- **Distinción de canales lógicos.** Distingue la cantidad de datos (optimización del tamaño del PDU) que se puede enviar y recibir, y el tipo de dato que es enviado.
- **Controlar las conferencias.** Control independiente a partir del protocolo H.323.
- **Codificación del canal.** Se utiliza una codificación binaria.
- **Gatekeeper.** Este actúa como guardián de la comunicación.

5.1.4 Desventajas

- **Complejo.** Los servicios y los productos del H.323 son complejos y por ello más costosos de adquirir.
- **Fallas de seguridad.** Son debidos a su arquitectura que no se enfoca a cliente/servidor.
- **Firewalls y Proxies.** Problemas y complejidad de configuración a través de los Firewalls y Proxies.

5.2 SIP (Session Initiation Protocol)

El protocolo SIP [15 - 22] (Session Initiation Protocol). Es un protocolo dedicado a la señalización (establecimiento y terminación de sesiones), control en los sistemas de telefonía IP. Este protocolo fue diseñado y desarrollado por el *IETF*¹⁰. Se complementa con otros dos protocolos; *SDP*¹¹ (Session Description Protocol) y *RTP*¹²(Real-Time Transport Protocol). Los puertos funcionales de este protocolo son **5060 UDP**¹³ y **TCP**¹⁴, y el **puerto 5061** para la señal encriptada. Las funciones principales del protocolo:

- Determinar la ubicación de los usuarios.
- Señalización (establecer y terminar) y modificar las sesiones.

5.2.1 Arquitectura SIP

El protocolo está definido por varios componentes, y estos poseen varias formas de implementar el sistema de control:

- **Usuarios SIP.** Un usuario puede ser varios dispositivos o software que sea compatible con el protocolo SIP (softphone, teléfono IP, ...). Tiene que tener la capacidad de registrarse en el servidor de registros SIP. Los usuarios reciben una dirección URI que está formada por "usuario@dominio". Siendo el campo del dominio el servidor SIP.
- **Servidor SIP.** Es un dispositivo o software encargado de crear y gestionar cuentas de usuarios SIP. Almacena las cuentas, registrando las direcciones IP de cada usuario.
- **Proxy y redirección.** Es una aplicación que permite la interconexión entre usuarios SIP. Esta aplicación envía los paquetes necesarios para establecer la comunicación bidireccional entre los usuarios. Una vez establecida la comunicación, los paquetes streaming de audio y video se envían entre los dominios registrados.
- **Servidores de Localización.** Es un dispositivo que recopila información de la situación física en la que se encuentra el usuario SIP.

¹⁰IETF: Internet Engineering Task Force.

¹¹SDP: Es un protocolo que describe los parámetros de inicialización de los flujos multimedia. RFC-4566 IETF.

¹²RTP: Estandar para la transmisión de voz y video RFC-3550 IETF.

¹³UDP: User Datagram Protocol.

¹⁴TCP: Transmission Control Protocol.

5.2.2 Funcionamiento SIP

A continuación se detallarán las fases (*frames*) que componen una llamada según el protocolo SIP: El esquema anterior, nos muestra como el teléfono A (izquierda) realiza una llamada al

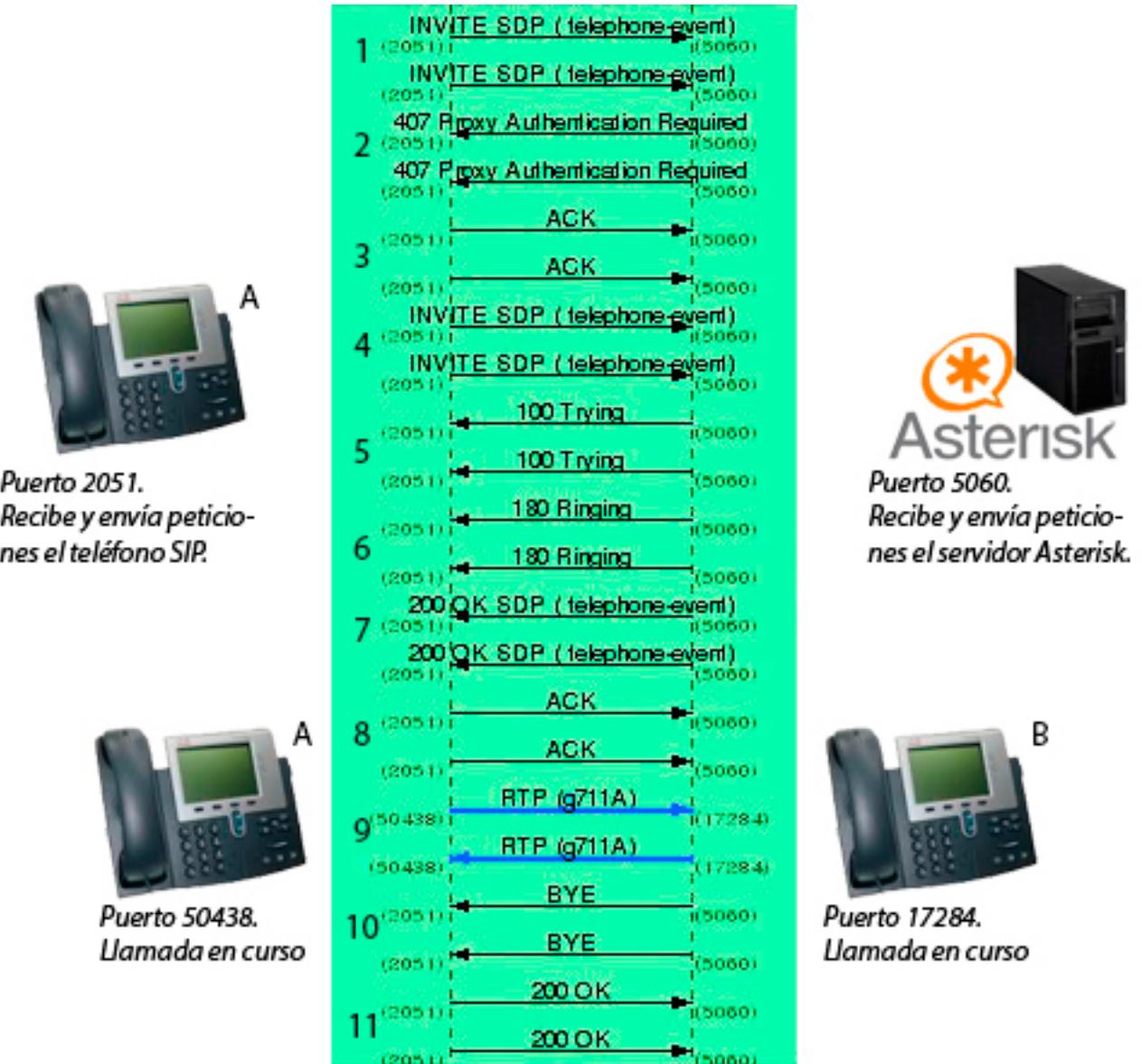


Imagen 5.3: Funcionamiento del protocolo SIP

teléfono B (derecha). Se detallan todas las situaciones desde la realización de la llamada hasta la finalización de la misma:

1. El **teléfono A** al realizar la llamada, envía un *frame INVITE* al servidor Asterisk.
2. El **servidor Asterisk solicita** (407 Autenticación Proxy) autenticación para realizar la llamada.
3. El **teléfono A** envía un mensaje **ACK** (Acuse de recibo).
4. El **servidor Asterisk envía** una respuesta **100 Trying** (teléfono tratando de hacer conexión).

5. El **servidor Asterisk** tras haber confirmado el último punto, *envía* un mensaje **180 Ringing** (teléfono sonando).
6. El **servidor Asterisk** *envía* un mensaje **200 OK SDP** (Llamada realizándose).
7. El **teléfono A** *envía* un **ACK** al servidor Asterisk.
8. El **teléfono A** *envía* un **RTP** con la encriptación del audio (g.711a).
9. El **teléfono B** *envía* un **RTP** con la encriptación del audio (g.711a).
10. Cuando el **teléfono B** (al que se llama) cuelga el teléfono, el servidor Asterisk *envía* un mensaje **BYE** al teléfono A para concluir la llamada.
11. El **teléfono A** *responde* al anterior estado con un mensaje **200 OK**.

5.2.3 Ventajas del SIP

- Control de llamadas sin estado (stateless).
- Necesita menos ciclos de CPU para generar mensajes de señalización.
- Es independiente a la existencia de una conexión en la capa de transporte.
- Autenticación de llamada mediante HTTP.
- Autenticación y encriptación son soportados (SSL/TSL).

5.2.4 Desventajas del SIP

- Calidad de llamada. Es levemente inferior a la calidad telefónica. Los datos están empaquetados, por lo cual, puede haber perdida de información y/o demora en la transmisión.
- Latencia. Se deben priorizar los datos (QOS) para poder conseguir una transmisión de calidad.
- Robo de datos. Existe la posibilidad de hackear el servidor VoIP y tener acceso a los datos almacenados y al servicio telefónico.
- Virus. Si se viera algún equipo del servidor afectado por un virus, el servicio se vería comprometido pudiendo quedar interrumpido o afectado de diversas maneras.

5.3 IAX2 (Inter-Asterisk eXchange)

El protocolo IAX2 [23 - 25] es el usado por Asterisk. Usa como central telefónica privada PBX¹⁵. Esta central, comunica las extensiones internas y a la vez conecta con la red pública (PSTN ó RTC). Su función es la de enrutar las llamadas.

El protocolo crea sesiones internas, estas pueden utilizar cualquier códec configurado en el servidor que transmita voz y/o video. También controla el flujo de datos multimedia.

Se diseñó para compatibilizarlo con varios estándares de transmisión de datos (SIP¹⁶, MGCP¹⁷ y RTP¹⁸), reducir el ancho de banda usado en la transmisión de datos (voz y video), control de las llamadas de voz.

5.3.1 Arquitectura del protocolo IAX2

Este protocolo VoIP fue diseñado para su uso entre servidores Asterisk, aunque también sirve para servidores que soporten este protocolo.

El objetivo principal de este protocolo es la minimización del ancho de banda a la hora de transmitir datos de voz y de video a partir de la red IP. Prestando especial atención al control y a la calidad de las llamadas de voz, dando soporte a partir de un puerto UDP (4569) para evitar problemas con el NAT.

Es un protocolo binario y su estructura se fundamenta en la multiplexación de la señalización y la canalización o flujo de datos por el puerto UDP antes citado, dispuestos entre los dos sistemas.

5.3.2 Funcionamiento IAX2

Consta de 3 fases:

- Establecimiento de llamada. El terminal A envía un mensaje "new" al terminal B. Este, responde con un mensaje "accept". A lo que el terminal A envía un "ACK" de confirmación. En este punto, el terminal B da señales de "ringing" y el terminal A contesta con un "ACK" de confirmación. Para finalizar, el terminal B activa la llamada con un "answer" y el terminal A envía un "ACK" de confirmación.
- Flujo de datos (audio). Se envían frames M y F en sentido bidireccional. Los frames M contienen solamente una cabecera de 4 bytes para reducir el ancho de banda. Los frames F están formados por la información de sincronización. Cabe destacar una peculiaridad del protocolo IAX. Este protocolo usa un puerto de UDP (4569) para los mensajes de señalización, evitando los problemas de NAT.
- Desconexión. El terminal A envía un mensaje de "hangup" o de desconexión al terminal B. Este último responde al anterior mensaje con una confirmación (ACK).

¹⁵PBX: Private Branch Exchange

¹⁶SIP: Session Initiation Protocol

¹⁷MGCP: Media Gateway Control Protocol

¹⁸RTP: Real-time Transport Protocol

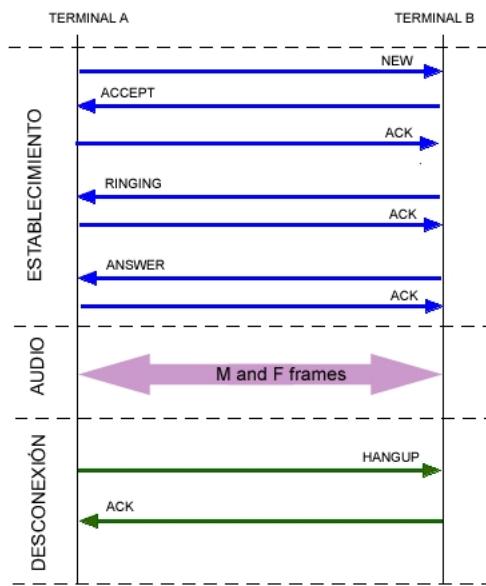


Imagen 5.4: Funcionamiento del protocolo IAX2

19

5.3.3 Ventajas del IAX2

- La comunicación entre dos terminales, se basa en la multiplexación de los datos a través del mismo puerto UDP.
- Solamente necesita una cabecera para la señalización. Por lo cual, reduce considerablemente el tamaño de los paquetes y también la latencia en la comunicación.
- Toda la información fluye por el mismo canal y por el mismo puerto UDP evitando problemas con los cortafuegos y con el NAT.
- Este protocolo incorpora características específicas de seguridad. Soportando autenticación del tipo PKI²⁰ con claves RSA²¹.

5.3.4 Desventajas del IAX2

Existen dos desventajas derivados del protocolo IAX2.

- Los ataques de denegación de servicio (DDoS). Por ello, se ideó el uso token de autenticación para evitar sufrir ataques DDoS por máquinas sin autenticar.
- El constante avance y adaptación a las necesidades específicas que se vayan requiriendo según futuras necesidades.

¹⁹<http://elastixtech.com/protocolo-iax/>

²⁰PKI: Public-key infrastructure

²¹RSA: Rivest, Shamir y Adelman. Sistema criptográfico de clave pública.

5.4 SCCP (Skinny Client Control Protocol)

Este protocolo [26 - 27] es propiedad de Cisco System (originalmente de la empresa Selsuis corporation), y se usa para la comunicación entre terminales Cisco UCM (Cisco Unified Communications Manager o CallManager)

Tratamos con un protocolo cliente servidor que se activa mediante eventos que son enviados mediante mensajes al Cisco UCM o CallManager. Este envía instrucciones específicas a los terminales con las tareas a realizar en cada evento.

Existe poca información sobre el funcionamiento y arquitectura sobre este protocolo de carácter privativo.

5.4.1 Arquitectura del SCCP

Los terminales SCCP usan para comunicarse entre si y con el Cisco UCM o CallManager el protocolo TCP/IP.

- **Cisco Unified Communications Managers (CUCM).** Es un sistema de comunicaciones basado en la integración de video, voz, datos y aplicaciones. Posee más efectividad a la hora de realizar la activación de los dispositivos y cuenta con seguridad en las comunicaciones.

Está compuesto por diferentes capas. Estas capas son las siguientes:

- Capa de infraestructuras: Está compuesta por Routers, switches y gateways de voz. Esta capa transportan datos, voz, video entre todos los dispositivos y aplicaciones de la red. Provee una alta disponibilidad, gestión, calidad de servicio (QOS) y seguridad en la red.
- Capa de Control de sistemas: Suministra el procesamiento de la llamada, control de los dispositivos y la administración del Dialplan.
- Capa de aplicaciones: Son independientes a los sistemas de control y diferentes infraestructuras (voz, video y datos).
- Capa de usuarios finales: Pueden ser un software específico para comunicaciones VoIP, teléfonos IP, video terminales o algún otro cliente de comunicaciones. Estos terminales deben poseer protocolos de comunicación (SIP, SCCP, MGCP, H.323).

Este sistema de comunicaciones tiene como funciones las siguientes:

- Control de llamadas: Realización, enrutamiento y finalización de las llamadas.
- Señalización y control de los dispositivos: Realiza las llamadas entre los terminales finales.
- Administración del Dialplan: Control de un listado de enrutamiento de llamadas.
- Administración de las características de los teléfonos: Transferencias, conferencias, marcación rápida, remarcar, llamada en espera, y muchas otras opciones de los teléfonos ip y gateways.

- Directorio de servicios: Posee una base de datos propia para guardar la información de los usuarios (login). La sincronización del directorio permite la gestión centralizada de los usuarios.
 - Interfaz de programación para aplicaciones externas: Proporciona una interfaz de programación para aplicaciones externas para gestionar productos VoIp de Cisco.
 - Herramientas para hacer backup y restauración: Dispone de una opción para realizar backup y restauraciones de la base de datos de configuración, información de los usuarios, detalles de las llamadas.
- **Gateways.** Dispositivo encargado de convertir las llamadas en datos de una red IP.
 - **Teléfonos IP Cisco.**

5.4.2 Funcionamiento del SCCP



Imagen 5.5: Funcionamiento del protocolo SCCP

El esquema anterior, nos muestra como el teléfono A (izquierda) realiza una llamada al teléfono B (derecha). Se detallan todas las situaciones desde la realización de la llamada hasta la finalización de la misma:

1. **CallState (Off Hook):** El CallManager envía un frame para pedir información sobre el estado del teléfono IP/aplicación voip. El usuario ha descolgado el auricular del teléfono.
2. **ClearPromptStatus:** El CallManager envía un frame para borrar el estado de ambos dispositivos (teléfono IP/aplicación voip).
3. **SelectSoftKeys (NewCall):** El CallMANager envía un frame que activa las opciones disponibles en los dispositivos (teléfono IP/aplicación VoIP).
4. **DisplayPromptStatusV2:** El CallManager envía un frame para actualizar el estado del teléfono IP/aplicación VoIP y muestra mensaje. El estado inicial del teléfono IP/aplicación voip a la hora del descuelgue es "Enter number" (Introduzca número).
5. **StartTone:** El CallManager envía un frame para activar el tono continuo de dial.
6. **KeypadButton:** Se dispone a pulsar números en el teclado del teléfono IP/aplicación voip.
7. **KeypadButton:** Se pulsa el siguiente número.
8. **StopTone:** El CallManager envía un frame para parar el tono continuo cuando se haya pulsado el primer número.
9. **KeypadButton:** Se pulsa el siguiente número.
10. **KeypadButton:** Se pulsa el siguiente número.
11. **DialedNumber:** El CallManager envía un frame para conectar los dos números e iniciar la llamada.
12. **CallState (Proceed):** El CallManager envía un frame de actualización del estado de los dispositivos (teléfono IP/aplicación voip), al estado de Proceed (procediendo la llamada).
13. **DisplayPromptStatusV2 (RingOut):** El CallManager envía un frame para actualizar el estado de los dispositivos (teléfono IP/aplicación VoIP), al estado de RingOut (Llamando).
14. **SelectSoftKeys (RingOut):** El CallMANager envía un frame que activa las opciones disponibles en el teléfono IP/aplicación voip. Se activa la opción RingOut (Llamando)
15. **CallInfoV2:** El CallManager envía un frame especificando la información sobre la llamada (Nombre, número, linea, tipo de llamada (interna/externa)).
16. **StartTone (Alerting-Tone):** El CallManager envía un frame para activar el tono de llamada a los dispositivos (teléfono IP/aplicación voip).
17. **CallInfoV2:** El CallManager envía un frame especificando la información sobre la llamada (Nombre, número, linea, tipo de llamada (interna/externa)).
18. **Callstate:** El CallMANager envía un frame de actualización del estado de los dispositivos (teléfono IP/aplicación voip).
19. **DisplayPromptStatusV2 (Connected):** El CallManager envía un frame para actualizar el estado de los dispositivos (teléfono IP/aplicación VoIP), cambiando al estado de Connected (Conectados).
20. **SelectSoftKeys (Connected):** El CallMANager envía un frame que activa las opciones disponibles en los dispositivos (teléfono IP/aplicación voip). Se activa la opción Conected (Conectados).

21. **StopTone:** El CallManager envía un frame para parar el tono de llamada en ambos dispositivos (teléfono IP/aplicación voip).
22. **ConnectionStaticsReq (clearStats):** El CallMANager envía un frame para actualizar el estado de la información estadística de la llamada. Pone a cero el resumen estadístico de ambos dispositivos (teléfono IP/aplicación voip).
23. **OpenRecieveChannel:** El CallManager envía un frame a ambos dispositivos (teléfono IP/aplicación voip) abriendo el canal para poder enviar y recibir datos de audio durante la llamada.
24. **OpenRecieveChannelAck:** Los dispositivos envían un frame de vuelta al CallManager avisando de la recepción del frame anterior.
25. **CallInfoV2:** El CallManager envía un frame actualizando la información y especificando la existencia de una llamada en curso.
26. **StopTone:** El CallManager envía un frame a ambos dispositivos (teléfono IP/aplicación voip) de parada de tono.
27. **StartMediaTransmission:** El CallManager envía un frame a ambos dispositivos (teléfono IP/aplicación voip) para comenzar a compartir archivos de audio.
28. **StartMediaTransmissionAck:** Los dispositivos envían un frame de vuelta al CallManager avisando de la recepción del frame anterior.
29. **ConnectionStaticsReq (doNotClearStats):** El CallManager envía un frame de recopilación de información estadística de la llamada en curso entre los dispositivos (teléfono IP/aplicación voip).
30. **ConnectionStaticsRes (doNotClearStats):** El CallManager envía un frame mostrando información recopilada en el anterior frame.
31. **SoftkeyEvent:** El CallMANager envía un frame que activa las opciones disponibles en la teléfono IP/aplicación voip.
32. **ClearPromptStatus:** El CallManager envía un frame para borrar el estado de ambos dispositivos (teléfono IP/aplicación voip).
33. **DisplayPromptStatusV2:** El CallManager envía un frame para actualizar el estado de ambos dispositivos (teléfono IP/aplicación VoIP), quedando a la espera.
34. **ConnectionStaticsReq:** El CallManager envía un frame de recopilación de información estadística de la llamada en curso entre los dispositivos (teléfono IP/aplicación voip).
35. **ConnectionStaticsRes:** El CallManager envía un frame mostrando información recopilada en el anterior frame.
36. **CloseReceiveChannel (CLOSE_PORT):** El CallManager envía un frame para cerrar el canal, por el cual, se envían los archivos de audio.
37. **StopMediaTransmission (CLOSE_PORT):** El CallMANager envía un frame de parada a ambos dispositivos (teléfono IP/aplicación voip) de la transmisión de datos de audio.
38. **ConnectionStaticsReq (clearStats):** El CallManager envía un frame para borrar la información estadística de la llamada entre los dispositivos (teléfono IP/aplicación voip).

39. **CallState (On Hook):** El CallManager envía un frame actualizando el estado de la llamada de los dispositivos (teléfono IP/aplicación voip). El estado actual es On Hook (Colgado).
40. **SelectSoftKeys (On Hook):** El CallManager envía un frame actualizando el estado de ambos dispositivos (teléfono IP/aplicación voip) posicionandolos en el estado de On Hook (Colgado).
41. **StopTone:** El CallMAnager envía un frame para mantener desactivado el estado de tono.
42. **ConnectionStaticsRes:** El CallManager envía un frame con el resultado estadístico de la anterior llamada.
43. **SelectSoftKeys:** El CallManager envía un frame actualizando el estado de los dispositivos.

5.4.3 Ventajas del SCCP

- Seguridad. Al ser un producto privado con un código fuente privativo las brechas de seguridad son mínimas.
- Fácil mantenimiento. El programa (CUCM) gestiona a nivel GUI todas las configuraciones.
- Registra las MACs. Este protocolo no registra los datos de los usuarios (usuario y contraseña) sino que registra y administra las MACs de los terminales.
- Señalización y transporte. Usa el puerto TCP 2000 para la señalización y el protocolo UDP y RTP para el transporte de los datos en tiempo real (multimedia).
- Protocolo liviano. Es un protocolo que posee una estructura muy simple.

5.4.4 Desventajas del SCCP

- Sin Compatibilidad. Los productos Cisco (VoIP) son incompatibles con las demás plataformas. Es decir, las redes Asterisk no pueden administrar teléfonos con el protocolo SCCP. Sería necesario la modificación del firmware (firmware adaptado SIP) para su uso en las redes Asterisk.
- Producto privado. Elevado coste del producto que conlleva a las empresas (pequeñas y medianas) a buscar otras opciones más baratas.

Capítulo 6

Descripción de los materiales a usar

Características
<p>Router Cisco 2801.</p>  <ul style="list-style-type: none">• Ethernet LAN(RJ-45): 2 puertos.• Ethernet LAN, velocidad de transferencia de datos: 10, 100Mbit/s.• Algoritmos de seguridad soportados: 3DES, DES, WPA-AES.• Bidireccional completo (Full duplex): Si.• Calidad de servicio (QoS) soporte: Si.• Estándares de red: IEEE 802.3.• Protocolos de red compatibles: 802.11a/b/g.
<p>Switch Catalyst 2960 24tt-l.</p>  <ul style="list-style-type: none">• Administración: RS-232.• Ethernet host(RJ-45): 2 puertos.• Ethernet 10Base-T/100Base-TX/1000Base-T(RJ-45): 24 puertos.• Ethernet 10Base-T/100Base-TX(RJ-45): 2 puertos.
<p>MSI CX61.</p>  <ul style="list-style-type: none">• Procesador: Intel® Core, i7-4712MQ (2.3 GHz, 6 MB).• Memoria RAM: 8GB DDR3 SODIMM (1x8GB).

Características	
	Teléfono VoIP Cisco C7941G. <ul style="list-style-type: none"> • Autoalimentación PoE • 2 puertos Ethernet (1GigE). • Pantalla LCD. • Gestión de 4 líneas. • Protocolo SIP, MGCP, SCCP. • Navegador XML
	Nexus 4 E960. <ul style="list-style-type: none"> • Sistema Operativo: Cyanogen. • Procesador: Quad-core 1.5GHZ. • Pantalla: 4.7". • Cámara: 8MP.
	Jiayu S3 Advance. <ul style="list-style-type: none"> • Sistema Operativo: AOSP N. • Procesador: ARM Cortex-A53 1.7GHZ. • Pantalla: 5.5". • Cámara: 12MP.
	Router TP- LINK N600. <ul style="list-style-type: none"> • Conexiones: 300Mbps - 2.4 GHz y 300Mbps - 5 GHz. • Puertos ethernet: Gigabit. • Velocidad de transferencia: WAN y LAN 800 Mbps.
	Linphone. Es una aplicación softphone SIP, disponible para varias plataformas (Windows, linux, OsMAC, Android, Iphone). Distribuido bajo la licencia GNU/GPL. <ul style="list-style-type: none"> • Soporte de codecs de audio (G.711, GSM, iLBC). • Soporte de codecs de video (H.263, MPG-4, H.264, QCIF). • Completamente escrito en C. • Nat transversal y soporte STUN. • Muestra información. • Llamada en espera. • Desvío de llamadas. • http://www.linphone.org/

Tabla 6.1: Materiales de laboratorio

Capítulo 7

Cisco CCNA Voice

Índice

7.1 Configuración Cisco	38
7.1.1 Configuración básica de puertos del router	39
7.1.2 Configurando DHCP para VoIP y Datos	39
7.1.3 Configuración del Servicio VoIP	40
7.1.4 Creando directorio de números	40
7.1.5 Configuración terminales	40
7.2 Configuración Switch	41
7.2.1 Configurando las Vlans rama a	41
7.2.2 Configurando las Vlans rama b	41

7.1 Configuración Cisco

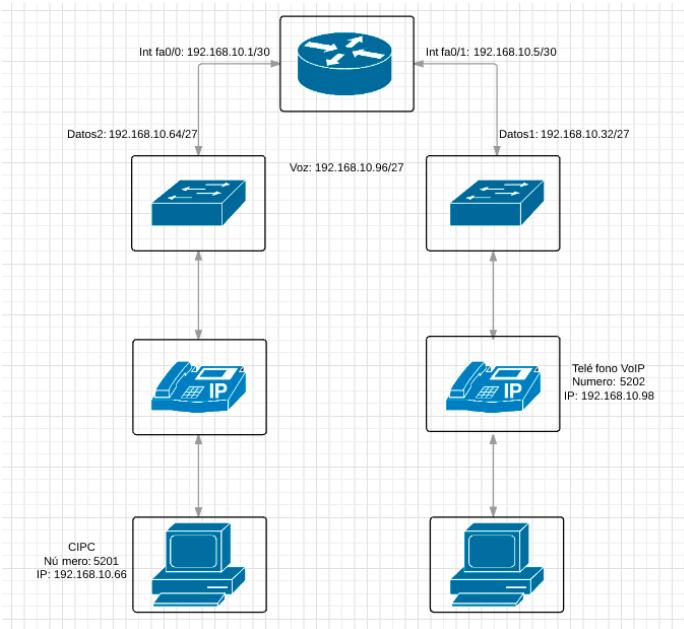


Imagen 7.1: Red Cisco

A continuación se explicará como realizar la configuración de Cisco [28 - 34].

7.1.1 Configuración básica de puertos del router

- Comenzamos la configuración del Router, tecleando los siguientes comandos **enable**, pulsaremos **intro**. Después introduciremos **configure terminal**, lo que nos dará acceso a la parte de configuración del Router. Primeramente, daremos un nombre al terminal, que en este caso será **Alpha**. Seguidamente, ejecutamos el comando **interface fastEthernet 0/0** (**int fa 0/0**), que corresponde a la entrada física ethernet 0 del router. Introduciremos el comando **no ip address** indicando que el puerto en sí no llevará una dirección ip.
- Crearemos y activaremos la **subinterfaz 0/0.1** del mismo puerto (**interface fastethernet 0/0**). Introduciremos el comando **encapsulation dot1q 1 native**, que convertirá la subinterfaz en vlan de administración o vlan nativa. Configuraremos una ip y una mascara de subred, que en este caso será **192.168.10.1** (ip) y **255.255.255.252** (netmask o mascara de subred), y levantaremos el puerto si aún está apagado, con el comando **no shutdown**.
- De la misma manera, crearemos y configuraremos 2 subinterfaces en este puerto, pero los datos a encapsular serán totalmente diferentes, uno de ellos irá dirigido a Voz y el otro a Datos. Cada una de ellas está condicionada con la numeración de la vlan. En este caso, la encapsulación de la vlan 10 en el switch, corresponde a la vlan de Voz. En el caso de la vlan 20 en el switch, corresponde a la vlan de Datos.
- Se les administrará diferentes ip's de clase c, aunque podrían ser de otra clase, es una situación muy modificable según las necesidades.
- La subinterface que controlará la **vlan 10 (Voz)**, tendrá la ip **192.168.10.97** con una máscara de subred **255.255.255.224**.
- La siguiente subinterface que controlara la **vlan 20 (Datos)**, tendrá la ip **192.168.10.33** con un máscara de subred **255.255.255.224**.
- De igual manera, ejecutamos el comando **interface fastethernet 0/1** (**int fa 0/1**), que corresponde con la entrada física ethernet 1 del router. No Se configurará ninguna IP a este puerto.
- Como en el anterior puerto levantaremos 2 subinterfaces, estas las configuraremos de la siguiente manera. La primera, irá dirigida a la **vlan 1 de administración** o vlan nativa con la siguiente ip **192.168.10.5** y una mascara de subred **255.255.255.252** y levantaremos el puerto con el comando **no shutdown**. La segunda, irá dirigida a la **vlan 20** o vlan dirigida a lo Datos y llevará la siguiente ip **192.168.10.65** y una máscara de subred **255.255.255.224**¹.

7.1.2 Configurando DHCP para VoIP y Datos

- Seguidamente, configuraremos los dhcp's, en este caso son 3, 2 de datos y 1 de voz. Cada pool de datos pertenece a una rama de la red, y coinciden con las ip's de las vlan antes administradas en las subinterfaces de los puertos ethernet del router.

¹Anexo I: Cisco CCNA Voice / Configuración básica de puertos del router

- Cada dhcp pool, está compuesta por la ip de la red y su rango que está delimitada por su mascara de red. Esto se realiza a partir del siguiente código, tecleando **network seguido de la ip y la mascara de subred**.
- Se configura la puerta de enlace, a partir del siguiente código. Tecleamos **default-router seguido de la ip**.
- Configuramos la dirección del servidor TFTP, donde se alojará el archivo de configuración del teléfono. Esto se realiza a partir del siguiente comando, **option 150 ip seguido de la ip de la puerta de enlace**².

7.1.3 Configuración del Servicio VoIP

- Activamos el servicio de telefonía de cisco. configuramos 10 terminales ephone como máximo, con el siguiente comando max-ephone. Administramos las 10 lineas virtuales max-dn.
- Configuraremos el puerto 2000 (TCP-Protocolo SCCP)la ip 192.168.10.97 que corresponde a la vlan 10 del puerto ethternet 0 del router.
- Se automatizará la asignación de numeración con el comando auto assign 1 to 10³.

7.1.4 Creando directorio de números

- En esta parte del código se registrarán y configurarán cada uno de los terminales en el Cisco Unified CallManager Express.
- A partir de aquí se establecen ephones-dn's individuales, siendo los mismos una linea virtual. Cada uno de los teléfonos físico debe configurarse en el Cisco CallManager Express de manera manual para que sea reconocido por la red ⁴.

7.1.5 Configuración terminales

- Como ultimo paso en la configuración del router, hay que introducir las mac's de los dispositivos involucrados, asignándoles un ephone en concreto. Ejecutamos el comando **mac-address** seguido de la **mac** correspondiente.
- El siguiente punto es aclarar que tipo de dispositivo esta asociado a esa mac, para ello, usaremos el comando **type** seguido del modelo. En este caso, hay varias posibilidades (7941, CIPC, ata). Elegiremos la opción **CIPC** en uno de los dispositivos y 7941 para el teléfono VoIP que está conectado en el otro extremo. Esto indica que se va a usar un dispositivo (portátil) que lleva instalado un programa (softphone) y por otro lado se va a usar un teléfono IP.
- Asociamos el número del botón con las características de la extensión de la linea. Para ello introducimos el comando button 1:1 en el caso del softphone, y button 1:2 para el caso del teléfono IP Cisco 7941G.

²Anexo I: Cisco CCNA Voice / Configurando DHCP para VoIP y Datos

³Anexo I: Cisco CCNA Voice / Configuración del Servicio VoIP

⁴Anexo I: Cisco CCNA Voice / Creando directorio de números

El primer número (1) indica llamadas normales. El segundo número (1/2) indica el directorio al que pertenece ⁵.

7.2 Configuración Switch

7.2.1 Configurando las Vlans rama a

- Crearemos 2 redes de área local virtual (VLAN). Separaremos el tráfico de voz (VLAN 10) y el tráfico de datos (VLAN 20). Para ello, crearemos 2 subredes lógicas que conforman dominios de difusión múltiples. De esta manera, evitamos la posibilidad de que haya bucles en la red.

7.2.2 Configurando las Vlans rama b

- De la misma manera crearemos 2 redes de área local virtual (VLAN). Separaremos el tráfico de voz (VLAN 10) y el tráfico de datos (VLAN 20). Para ello, crearemos 2 subredes lógicas que conforman dominios de difusión múltiples. De esta manera, evitamos la posibilidad de que haya bucles en la red. Haremos lo mismo que en el párrafo anterior, ya que es la otra parte de la configuración del switch⁶.

⁵Anexo I: Cisco CCNA Voice / Configuración terminales

⁶Anexo I: Cisco CCNA Voice / Configuración Switch

Capítulo 8

Asterisk FreePBX

Índice

8.1 Creación de usuarios desde Terminal	42
8.1.1 Configuración general (codecs audio/video, Quality of service)	43
8.1.2 Creación de usuarios	44
8.1.3 Configuración de extensiones	46
8.1.4 Configuración QOS (Calidad de servicio)	47

Para poder realizar un configuración óptima de Asterisk FreePBX [35 - 39], es necesario que se lleven a cabo diferentes puntos antes de la introducción de usuarios¹.

8.1 Creación de usuarios desde Terminal

Crearemos 5 usuarios. Les daremos la siguiente numeración y contraseña a cada uno.

- Usuario: "5201" Contraseña "d5201s".
- Usuario: "5202" Contraseña "m5202r".
- Usuario: "5203" Contraseña "f5203p".
- Usuario: "5204" Contraseña "p5204v".
- Usuario: "5205" Contraseña "i5205r".

Se explicará los pasos a tomar para la creación de los usuarios y la configuración de las extensiones.

Habiendo introducido el localhost login; "root" y el Password; "*password introducido en la configuración inicial*". introduciremos los siguientes comando en el terminal:

1. "**service asterisk start**", que nos devolverá una mensaje "*Asterisk is already running*".
2. "**asterisk -r**", accederemos a la configuración por terminal.
3. "**sip show peers**", para visualizar los usuarios SIP que están registrados en el programa, que en nuestro caso no habrá ninguno.
4. "**core stop now**", detendremos la aplicación Asterisk.

¹Anexo II: Asterisk FreePBX / Configuración Asterisk

5. Nos dirigiremos a la carpeta donde se configuran los usuarios en Asterisk. "**cd etc/asterisk/**". Pulsaremos **intro** y haremos que muestre todos los archivos con el comando "**ls**".

8.1.1 Configuración general (codecs audio/video, Quality of service)

6. Accederemos al archivo "**sip.conf**", lo haremos introduciendo el comando "**nano sip.conf**".
7. Buscaremos la sección "**[general]**", donde se llevará a cabo la configuración de los usuarios.

Los apartados de esta sección se explicarán a continuación:

- **udpbindaddr**: Especifica la IP y el puerto por el cual se escucharán las peticiones de entrada. En este caso, será 0.0.0.0 (para que se escuchen todas las peticiones) y se usará el puerto por defecto 5060.
- **context**. Está ligado directamente al dialplan² Se define por defecto para todas las conexiones SIP.
- **language**: Parámetro de lenguaje. Lo pondremos en español.
- **disallow**: Deniega una acción, IP o codec.
- **allow**: Permite una acción, IP o codec.
- **videosupport**: Parámetro de soporte de video.
- **maxcallbitrate**: Ancho de banda máximo para llamadas.
- **tos** (Type of service): Dispone de 6 bits en el protocolo IPV4 para indicar al router que paquetes tienen prioridad de paso. Forma parte de la capa 3 del modelo OSI.

Nombre DSCP	Prioridad IP
CS0	0 - Mejor esfuerzo
CS1, AF11 - 13	1 - Prioritario
CS2, AF21 - 23	2 - Inmediato
CS3, AF31 - 33	3 - Relámpago (Señal de voz)
CS4, AF41 - 43	4 - Relámpago sobrecargado (Comunicación de video RTP)
CS5, EF	5 - Crítico (Comunicación de voz RTP)
CS6	Internet
CS7	Red

Tabla 8.1: Quality of service: Tos

- **Cos**: Dispone de 3 bits (PCP) en el protocolo VLAN 802.1Q para indicar que paquetes tienen prioridad de paso. Forma parte de la capa 2 del modelo OSI.

²dialplan: Llamadas entrantes y llamadas salientes

<i>PCP</i>	<i>Prioridad</i>	<i>Tipos de tráfico</i>
1	0 (Más bajo)	En segundo plano
0	1	Mejor esfuerzo
2	2	Excelente esfuerzo
3	3	Aplicaciones críticas
4	4	Video
5	5	Voz
6	6	Internet
7	7	Red

Tabla 8.2: Quality of service: Cos

```
[general]

udpbindaddr=0.0.0.0:5060
context=default
language=es
disallow=all

; codec de audio (Llamadas)
allow=alaw
allow=ulaw
allow=gsm
allow=g722
allow=ilbc
allow=speex
allow=speex16
allow=speex32

; codec de video (Videollamadas)
allow=h264
allow=h263p
allow=h263
videosupport=yes
maxcall bitrate=512

; QoS sip, audio y video
tos_sip=cs3
tos_audio=ef
tos_video=af41
cos_sip=3
cos_audio=5
cos_video=4
```

- Nos dirigiremos al final del archivo e introduciremos los usuarios que anteriormente hemos indicado.

8.1.2 Creación de usuarios

Vamos a explicar primeramente que significa cada punto.

Englobaremos los patrones comunes, los usaremos en cada usuario. De esta manera ahorraremos tiempo a la hora de configuración.

- **type:** Se refiere al tipo de cliente. Puede ser de 3 tipos:
 - **peer.** Autentica llamadas y envía llamadas (solo salientes).
 - **user.** Autentica llamadas y recibe llamadas (solo entrantes).
 - **friend.** Autentica llamadas, envía y recibe llamadas (entrantes y salientes).
- **dtmfmode**³: Señalización (tonos de llamadas) de los sistemas de voz (rfc2833/info). Asignaremos "rfc2833".
- **host:** Indica la dirección IP o el nombre del host del cliente. Si es dinámico por DHCP, se pondrá *dynamic*.
- **context:** Dialplan que se asocia al usuario. En este caso se asociarán al apartado de "extensiones-internas".
- **canreinvite:** Este parámetro especifica si se fuerza que el streaming de audio pase por Asterisk. Asignándole "no" se obligará a que pase por Asterisk.
- **qualify:** Monitorización del estado de la extensión.
- **username:** Nombre del usuario.
- **secret:** Contraseña para autenticarse en el sistema.
- **callerid:** Nombre del usuario y su extensión.
- **mailbox:** Buzón de voz de la extensión.

```
[comunes] (!)
type=friend
dtmfmode=rfc2833
host=dynamic
context=extensiones-internas
canreinvite=no
qualify=yes

[5201] (comunes)

username=5201
secret=d5201s
callerid="Daniel Sanchez" <5201>
mailbox=5201@default

[5202] (comunes)

username=5202
secret=m5202r
callerid="Maria Ramirez" <5202>
mailbox=5202@default

[5203] (comunes)

username=5203
secret=f5203p
callerid="Fernando Perez" <5203>
```

³DTMF: Multifrecuencia de doble tono

```

mailbox=5203@default
_____
[5204] (comunes)

username=5204
secret=p5204v
callerid="Pablo Valdivia" <5204>
mailbox=5204@default
_____

[5205] (comunes)

username=5205
secret=i5205r
callerid="Irene Ramirez" <5205>
mailbox=5205@default
_____
```

Cerramos el archivo pulsando "**control + x**", nos indicará que salvaremos el archivo pulsando "**y**", o por el contrario pulsando "**n**" destruiremos los cambios. Pulsaremos "**y**" y seguidamente le daremos a la tecla "**intro**" para indicar que vamos a sobrescribir el archivo.

8.1.3 Configuración de extensiones

9. Tecleamos en el terminal "**nano extensions.conf**" para configurar las extensiones.
10. Buscaremos el contexto "[default]", e introduciremos lo siguiente:

```

[default]

exten => _X,1,Hangup(21) "Recibe todos los context".
exten => s,1,Hangup(21) "Rechaza todo por seguridad".
```

Este contexto [default] se usará cuando el usuario no este definido en ningún otro contexto. Esta configuración se realiza por seguridad y colgará todas las llamadas bajo este contexto evitando que un usuario sin identificar pueda hacer llamadas.

También se creará una cabecera llamada "[extensiones-internas]", la que especificamos en el archivo sip.conf en los apartados de configuración de usuarios "context=extensiones-internas".

exten => patrón,índice,acción(parámetro)

- **exten**. Es una palabra reservada que especifica las líneas de dialplan.
- **patrón**. Todas las extensiones de 4 cifras que comiencen por 52 y le sigan 2 números que están comprendidas entre el 0 y el 9.
- **índice**. Orden secuencial de las acciones a tomar en la extensión.
- **acción(parámetros)**. Acción que se debe ejecutar.
- **_52XX**: Extensión es de 4 dígitos que comiencen por 52.

- 1: Numeración de la acción del dialplan.
- Dial(SIP/*EXTEN*) : Llamada a la extensión que se indique. *Dial()*, en modo llamada SIP, es el protocolo.
- Hangup (16): Normal call clearing. Cuelga la llamada si no se puede realizar la comunicación (no disponible, no descuelga, rechaza la llamada).

```
[extensiones-internas]
exten => _52XX,1,NoOp(Llamadas a terminal interno)
exten => _52XX,2,Dial(SIP/${EXTEN})
exten => _52XX,3,Hangup(16)
```

Al finalizar pulsaremos de nuevo "**control + x**" e "**intro**" para salvar los datos que se han modificado en el archivo.

8.1.4 Configuración QOS (Calidad de servicio)

11. Tecleamos en el terminal "**nano iax.conf**" para configurar la calidad de servicio en este archivo.

```
tos=ef
```

12. Escribiremos en el terminal "**service asterisk start**", pulsaremos "**intro**", y seguidamente, introduciremos el comando "**asterisk -r**". De esta manera, podremos ver que los datos introducidos de los clientes están en todos funcionales.
13. Actualizaremos los archivos "sip.conf" y "extensions.conf", a través de los comandos "**dialplan reload**", seguidamente pulsaremos "**intro**". Y para finalizar, introduciremos el comando "**sip reload**" y de nuevo pulsaremos "**intro**".

De esta manera, quedarían configurados los clientes⁴.

⁴Anexo II: Asterisk FreePBX / Creación de usuarios

Capítulo 9

Configuración Router con firmware dd-wrt

Configuraremos el router de la siguiente manera:

1. Setup -> Basic Setup

- WAN Setup -> Connection type.
 - *Conection type* -> Disable
- WAN Setup -> Optional Setting.
 - *STP* -> Disable
- Network Setup -> Router IP (DHCP).
 - *Local IP Address* -> 192.168.10.62
 - *Subnet mask* -> 255.255.255.224
 - *Gateway* -> 192.168.10.33
- Network Setup -> Network Address Server Settings (DHCP).
 - *DHCP Server* -> Disable

2. Wireless -> Basic Setup

- Wireless Physical Interface ath0 (2.4 GHz).
 - *Wireless mode* -> Ap
 - *Wireless Network Mode* -> Mixed
 - *Channel Width* -> Full (20 MHz)
 - *Wireless Channel* -> Auto
 - *Wireless Network Name (SSID)* -> "NOMBRE DE LA RED"
 - *Wireless SSID Broadcast* -> Enable

3. Wireless -> Wireless Security

- Wireless Security ath0.
 - *Security mode* -> WPA2 Personal

- *WPA Algorithm* -> AES
- *WPA Shared Key* -> "PASSWORD"

4. Security -> Firewall

- Security -> Firewall Protection.
 - *SPI Firewall* -> Disable

5. NAT/QoS -> QoS

- Quality of Service (QoS) -> QoS Settings.
 - *Start QoS* -> Enable
 - *Port* -> LAN/WAN
- Quality of Service (QoS) -> TCP-Packet Priority.
 - *Sip* -> Premium
 - *rtp* -> Premium

Capítulo 10

Monitorización de codecs

Índice

10.1 Codec g711-u (pcmu - ulaw)	52
10.2 Codec g711-a (pcma - alaw)	53
10.3 Codec g722	54
10.4 Codec GSM (RPE-LPC)	55
10.5 Codec iLBC	56
10.6 Codec Speex8	57
10.7 Codec Speex16	58
10.8 Codec Speex32	59

En este apartado se va a realizar un análisis del ancho de banda consumido por cada *codec* [40 - 42]. Estos resultados se han obtenido a partir del router con el firmware modificado (dd-wrt).

- **G.711.** Este codec es un estándar usado en telefonía liberado en 1972. Es utilizado para la codificación digital de audio de 8 bits. Posee una tasa de muestreo de 8000hz y su flujo de datos es de 64kbit/s (8 bits * 8 khz). La señal es comprimida antes de ser transportada. Existen dos tipos de compresión:
 - G.711 u-law -> Estándar usado en EEUU y Japón.
 - G.711 a-law -> Estándar usado en el resto del mundo.
- **G.722.** Este códec es un estándar aprobado por la ITU en 1988. Es utilizado para la codificación digital de audio de 14 bits. Posee diferentes calidades (48kbit/s, 56kbit/s y 64kbit/s), con una tasa de muestreo de 7000hz.
- **GSM (RPE - LPC).** Este códec es un estándar diseñado a principio de los años 90. Es utilizado para la codificación digital de audio de 10 bits (narrowband) y 16 bits (wideband). Posee una tasa de muestreo de 8000hz y su flujo de datos es de 13kbit/s.
- **iLBC.** Este códec es un estándar presentado en 2002 y publicado en 2004. Es utilizado para la codificación digital de audio de 16 bits. Posee una tasa de muestreo de 8000hz y su flujo de datos es de 13.33 kbit/s para 30 ms y 15.2 kbit/s para 20 ms.
- **Speex8.** Este códec es un estándar presentado en 2002 y publicado en 2003. Posee una tasa de muestreo de 8000hz (Narrowband) y su flujo de datos transcurre entre 2.15 kbit/s y 24 kbit/s.

- **Speex16.** Este códec es un estándar presentado en 2002 y publicado en 2003. Posee una tasa de muestreo de 16000hz (Wideband) y su flujo de datos transcurre entre 3.95 kbit/s y 42.2 kbit/s.
- **Speex32.** Este códec es un estándar presentado en 2002 y publicado en 2003. Posee una tasa de muestreo de 32000hz (Ultra - Wideband) y su flujo de datos transcurre entre 5.75 kbit/s y 44 kbit/s.

10.1 Codec g711-u (pcmu - ulaw)

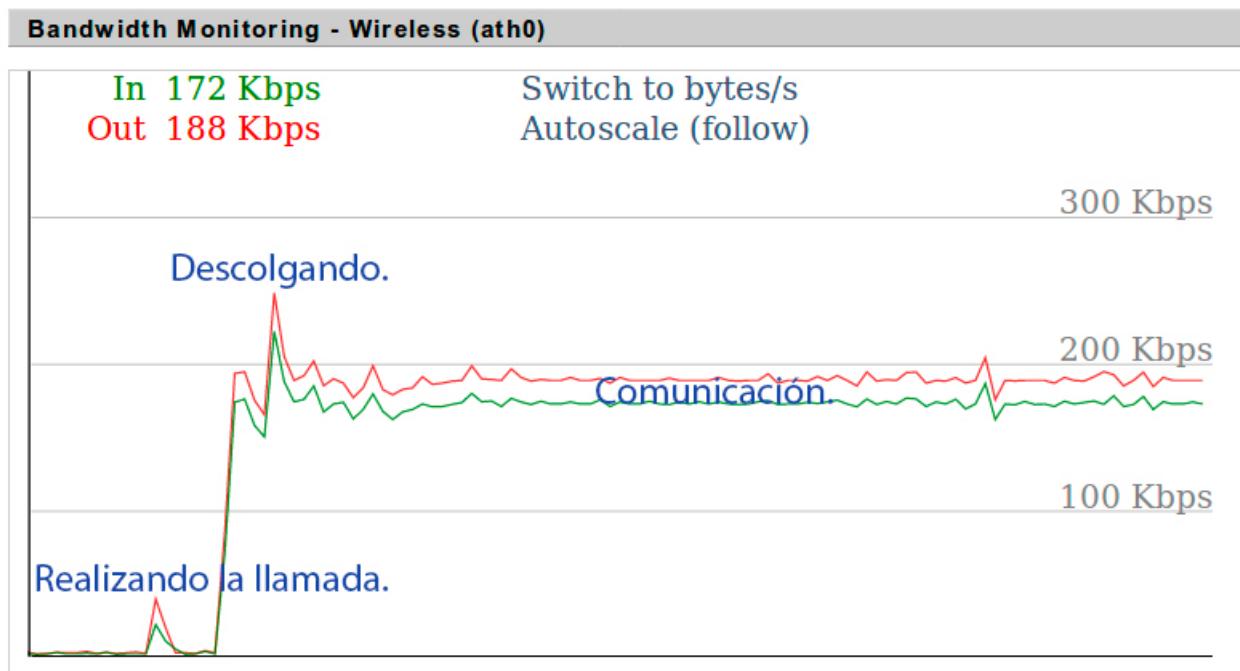


Imagen 10.1: Monitorización ancho de banda codec g711-u

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - **In** (0kbit/s - 30kbit/s).
 - **Out** (0kbit/s - 50kbit/s).
- **Descolgando** el auricular.
 - **In** (150kbit/s - 230kbit/s).
 - **Out** (175kbit/s - 250kbit/s).
- **Comunicación.**
 - **In** (165kbit/s - 175kbit/s).
 - **Out** (180kbit/s - 205kbit/s).

10.2 Codec g711-a (pcma - alaw)

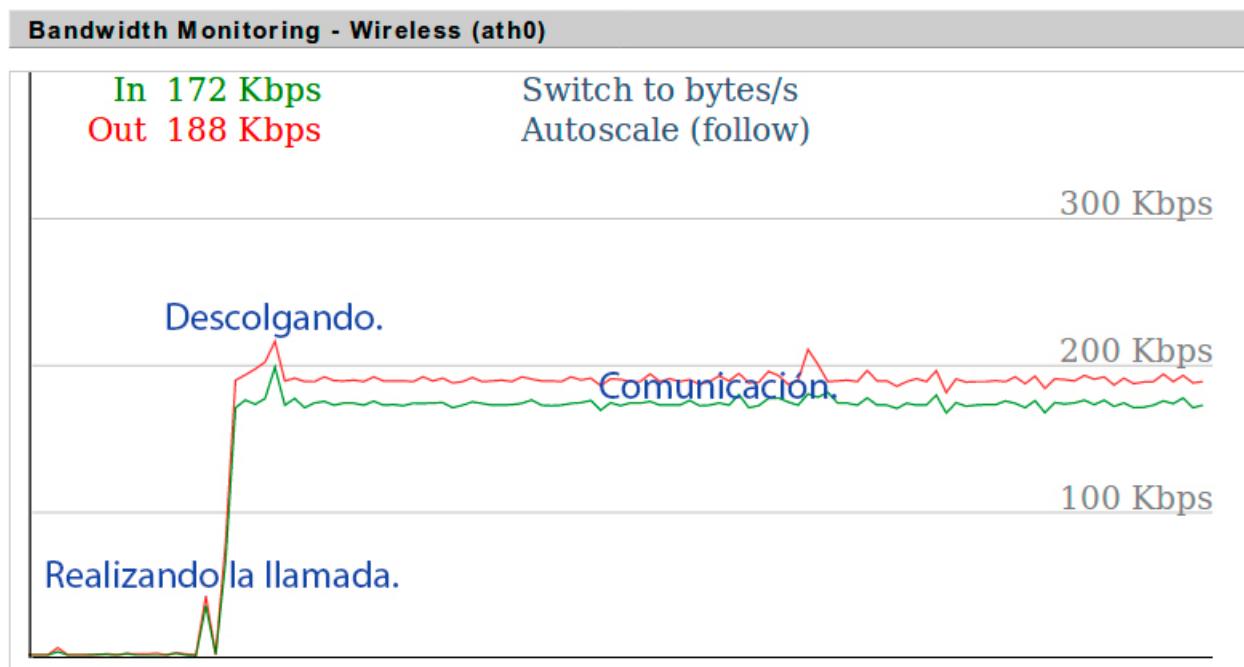


Imagen 10.2: Monitorización ancho de banda codec g711-a

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - **In** (0kbit/s - 40kbit/s).
 - **Out** (0kbit/s - 40kbit/s).
- **Descolgando** el auricular.
 - **In** (175kbit/s - 200kbit/s).
 - **Out** (190kbit/s - 210kbit/s).
- **Comunicación.**
 - **In** (170kbit/s - 185kbit/s).
 - **Out** (185kbit/s - 205kbit/s).

10.3 Codec g722

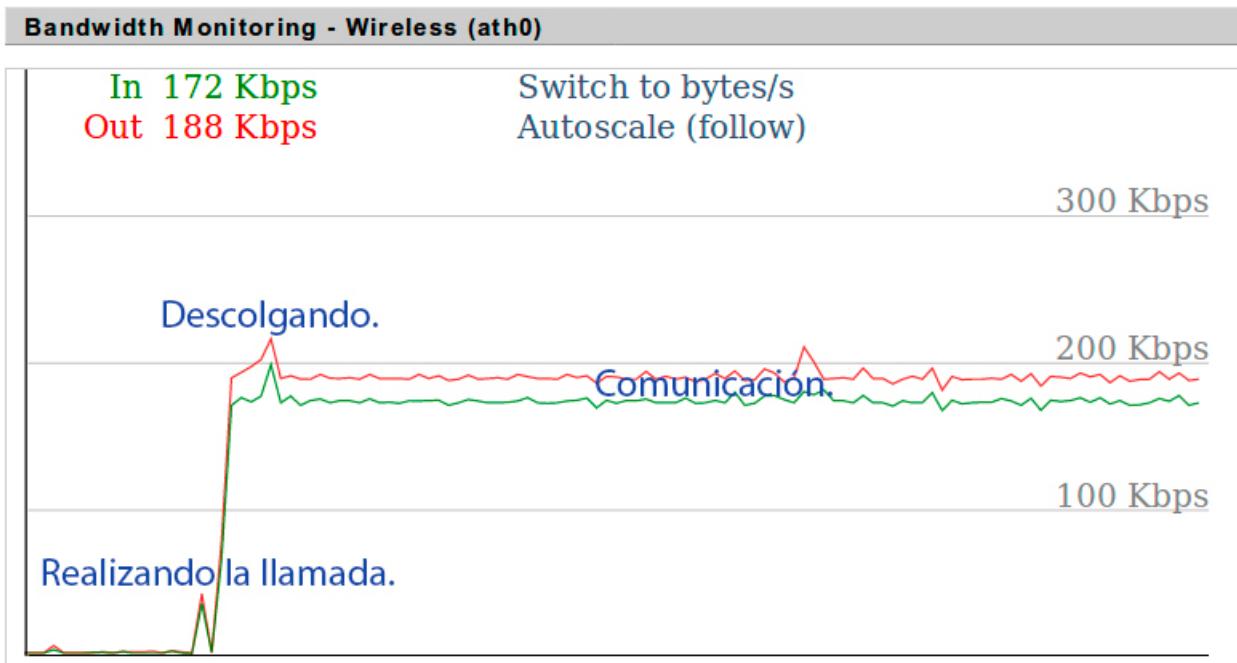


Imagen 10.3: Monitorización ancho de banda codec g722

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**

- **In** (0kbit/s - 40kbit/s).
- **Out** (0kbit/s - 40kbit/s).

- **Descolgando** el auricular.

- **In** (175kbit/s - 200kbit/s).
- **Out** (190kbit/s - 210kbit/s).

- **Comunicación.**

- **In** (170kbit/s - 185kbit/s).
- **Out** (185kbit/s - 205kbit/s).

In (170kbit/s - 200kbit/s). Out (185kbit/s - 225kbit/s).

10.4 Codec GSM (RPE-LPC)

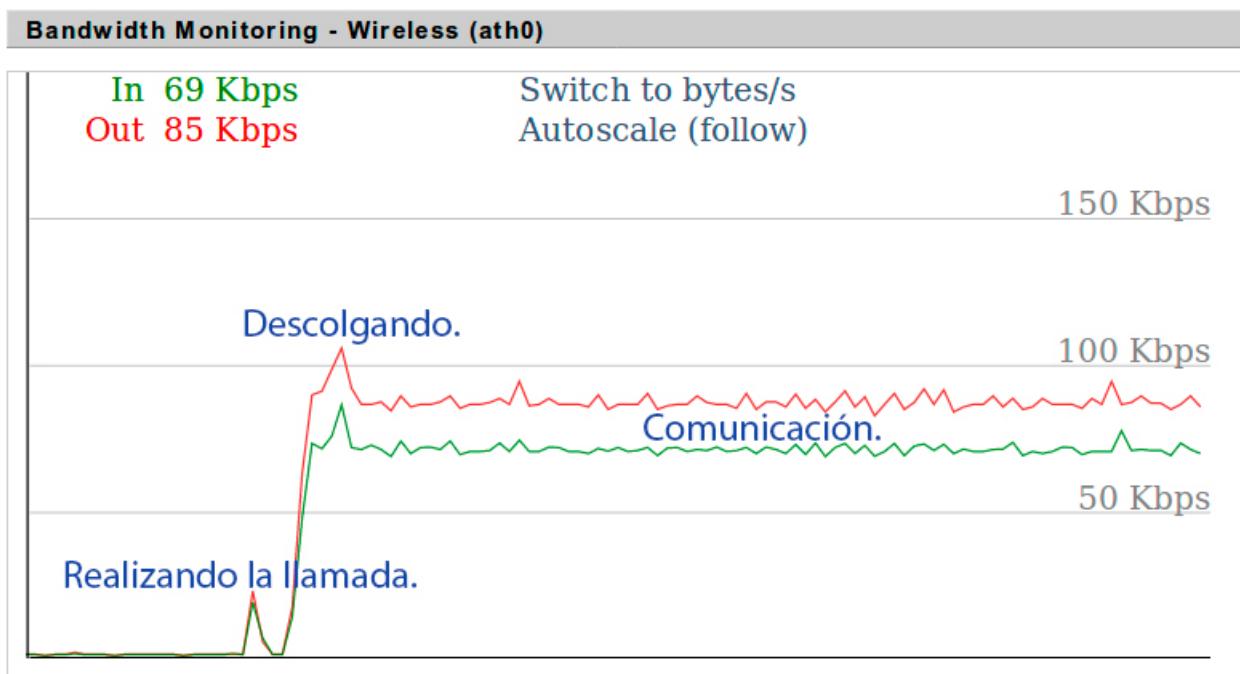


Imagen 10.4: Monitorización ancho de banda codec GSM

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - **In** (0kbit/s - 25kbit/s).
 - **Out** (0kbit/s - 25kbit/s).
- **Descolgando** el auricular.
 - **In** (75kbit/s - 85kbit/s).
 - **Out** (90kbit/s - 110kbit/s).
- **Comunicación.**
 - **In** (65kbit/s - 80kbit/s).
 - **Out** (80kbit/s - 95kbit/s).

10.5 Codec iLBC

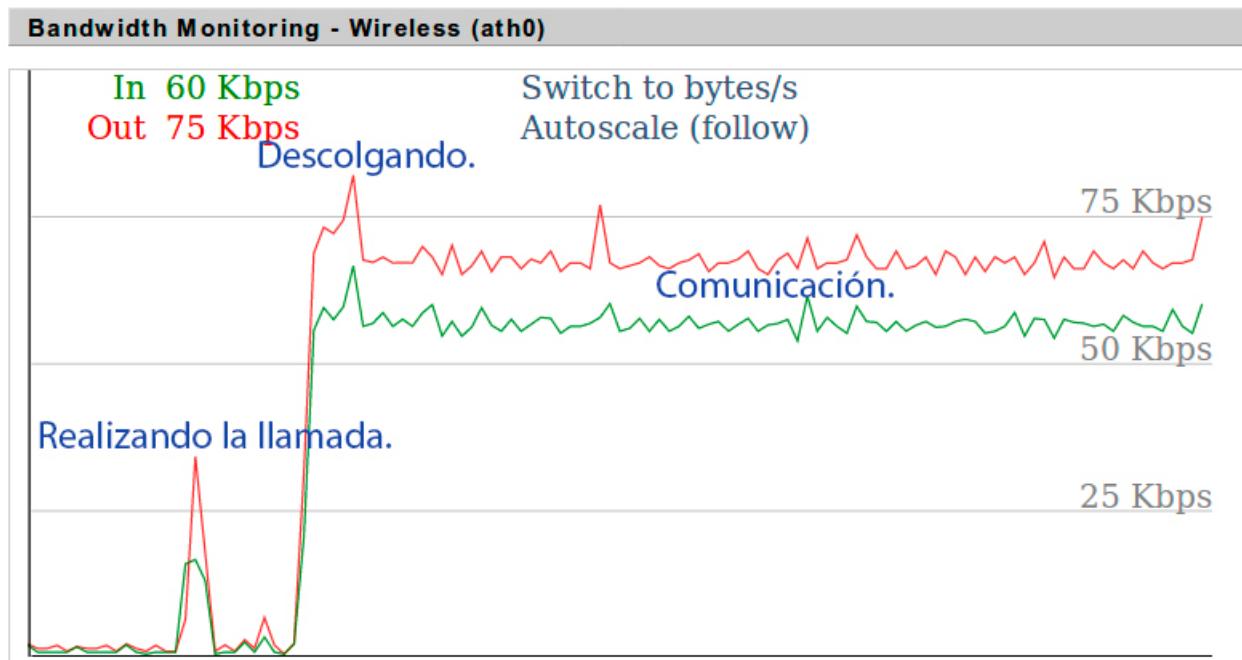


Imagen 10.5: Monitorización ancho de banda codec iLBC

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - **In** (0kbit/s - 15kbit/s).
 - **Out** (0kbit/s - 30kbit/s).
- **Descolgando** el auricular.
 - **In** (55kbit/s - 65kbit/s).
 - **Out** (65kbit/s - 80kbit/s).
- **Comunicación.**
 - **In** (55kbit/s - 65kbit/s).
 - **Out** (70kbit/s - 77kbit/s).

10.6 Codec Speex8

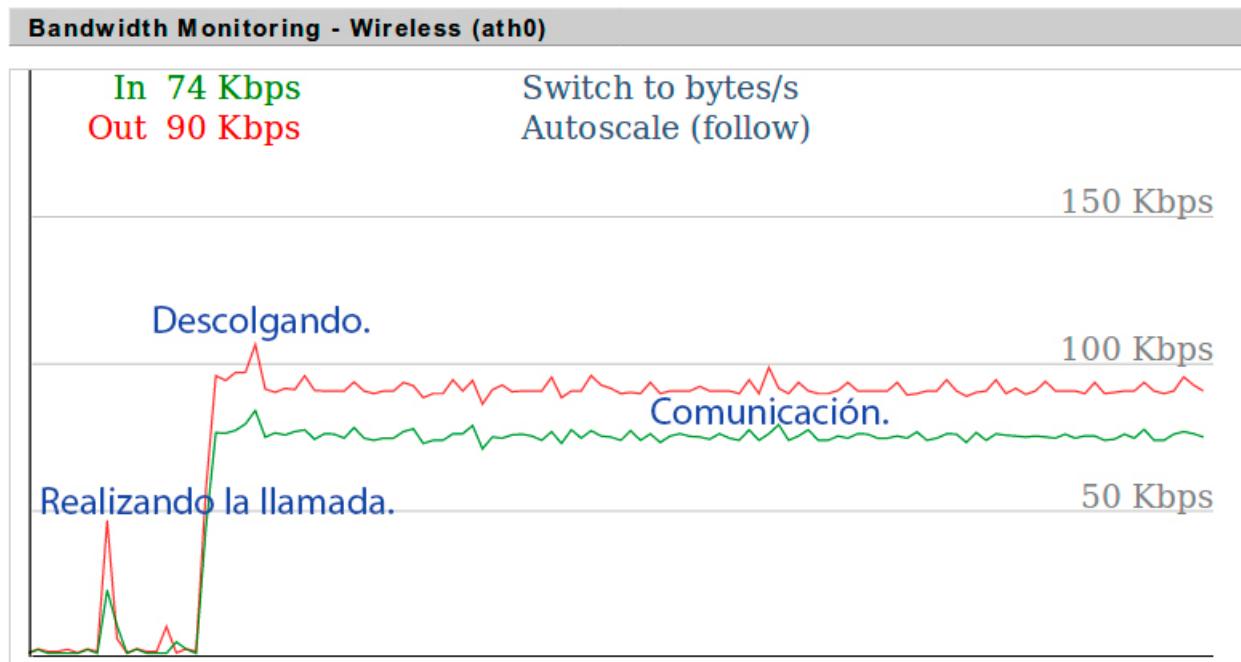


Imagen 10.6: Monitorización ancho de banda codec Speex8

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - **In** (0kbit/s - 25kbit/s).
 - **Out** (0kbit/s - 50kbit/s).
- **Descolgando** el auricular.
 - **In** (75kbit/s - 85kbit/s).
 - **Out** (85kbit/s - 110kbit/s).
- **Comunicación.**
 - **In** (70kbit/s - 80kbit/s).
 - **Out** (90kbit/s - 95kbit/s).

10.7 Codec Speex16

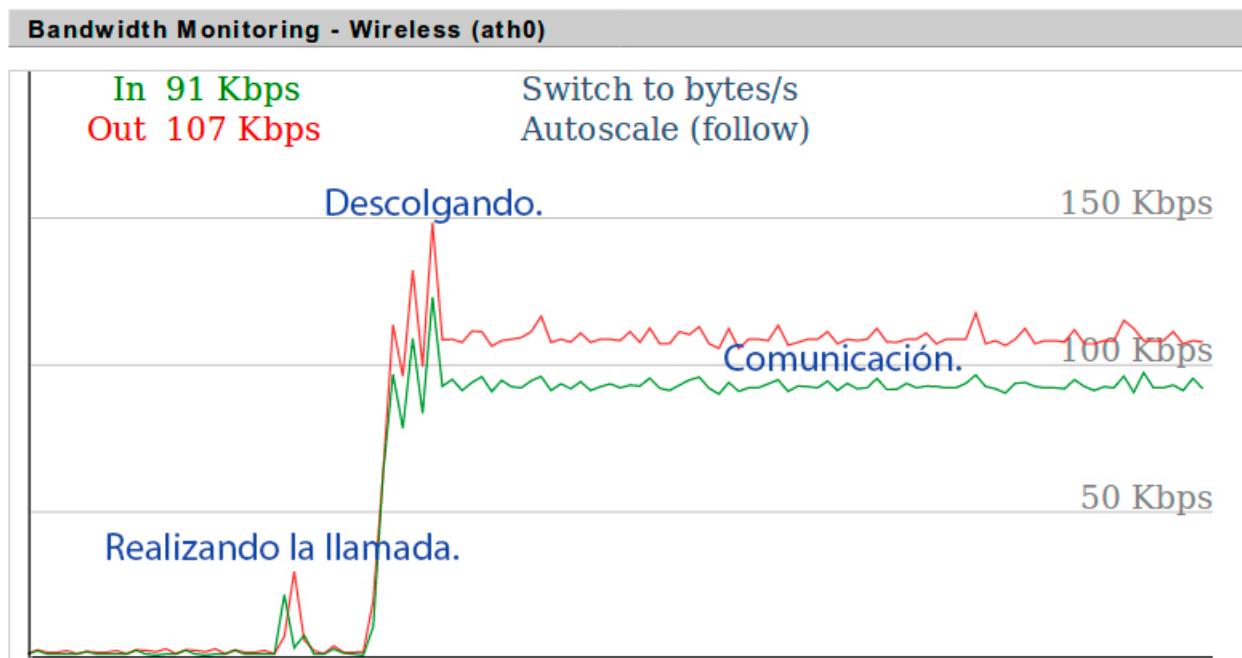


Imagen 10.7: Monitorización ancho de banda codec Speex16

En la figura anterior, cabe destacar la entrada (**verde**) y salida (**rojo**) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - **In** (0kbit/s - 25kbit/s).
 - **Out** (0kbit/s - 30kbit/s).
- **Descolgando** el auricular.
 - **In** (75kbit/s - 125kbit/s).
 - **Out** (95kbit/s - 150kbit/s).
- **Comunicación.**
 - **In** (90kbit/s - 100kbit/s).
 - **Out** (105kbit/s - 115kbit/s).

10.8 Codec Speex32

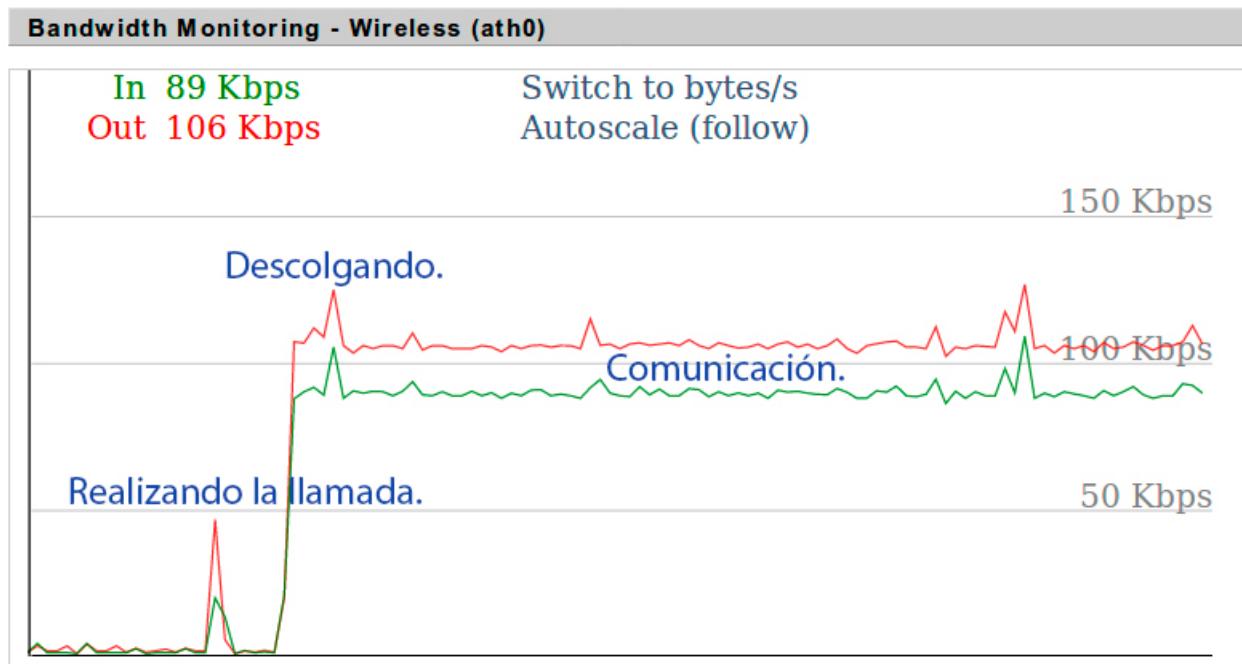


Imagen 10.8: Monitorización ancho de banda codec Speex32

En la figura anterior, cabe destacar la entrada (verde) y salida (rojo) de información en el router. Monitorizando la red se han obtenido los resultados fraccionados, en los siguientes apartados:

- **Realizando la llamada.**
 - In (0kbit/s - 25kbit/s).
 - Out (0kbit/s - 50kbit/s).
- **Descolgando** el auricular.
 - In (75kbit/s - 110kbit/s).
 - Out (110kbit/s - 125kbit/s).
- **Comunicación.**
 - In (85kbit/s - 115kbit/s).
 - Out (125kbit/s - 110kbit/s).

Capítulo 11

Conclusión

La siguiente tabla refleja la cantidad de *frames* que utilizan ambos protocolos cuando se realiza una llamada (realización de la llamada, descuelgue, comunicación y cuelgue).

Tabla 11.1: Tabla de resultados.

		SIP	SCCP
1	Estableciendo la llamada	49 frames	44 frames
	Finalización de la llamada	5 frames	28 frames
2	Estableciendo la llamada	15944 bytes (127552 bits)	3634 bytes (29072 bits)
	Finalización de la llamada	2485 bytes (20080 bits)	2408 bytes (19264 bits)
3	Estableciendo la llamada	0.41799549 seg	1.463633 seg
	Finalización de la llamada	0.077101084 seg	0.03926 seg

Según los datos expuestos en la tabla, los siguientes resultados:

Si realizamos una comparativa de los *frames*, podemos observar:

1. En el protocolo SCCP la cantidad de frames utilizados es en proporción mucho mayor que en el protocolo SIP.
2. Los *frames* del protocolo SCCP poseen un peso menor (bytes) optimizando la función del servidor de llamadas encargado de gestionar los *frames* y encaminar la comunicación entre los dos dispositivos.
3. El tiempo (segundos) de respuesta en el protocolo SCCP sea mayor a la hora de establecer la llamada, presenta un menor tiempo en la finalización de la misma.

Por ello, podemos concluir, que el protocolo SCCP presenta una mayor optimización a nivel técnico que el protocolo SIP. Según los datos expuestos, se denota lo siguiente:

Si realizamos una comparativa de los frames, podemos observar que en el protocolo SCCP la cantidad de frames utilizados es en proporción mucho mayor que en el protocolo SIP, pero por el contrario, estos frames (protocolo SCCP) poseen un peso menor¹ (bytes) optimizando la función del servidor de llamadas encargado de gestionar los frames y encaminar la comunicación entre los dos dispositivos y aunque el tiempo² (segundos) de respuesta sea mayor a la hora de establecer la llamada, presenta un tiempo menor en la finalización de la misma.

¹Anexo Conclusión / Tamaño de los frames

²Anexo Conclusión / Tiempos de respuesta

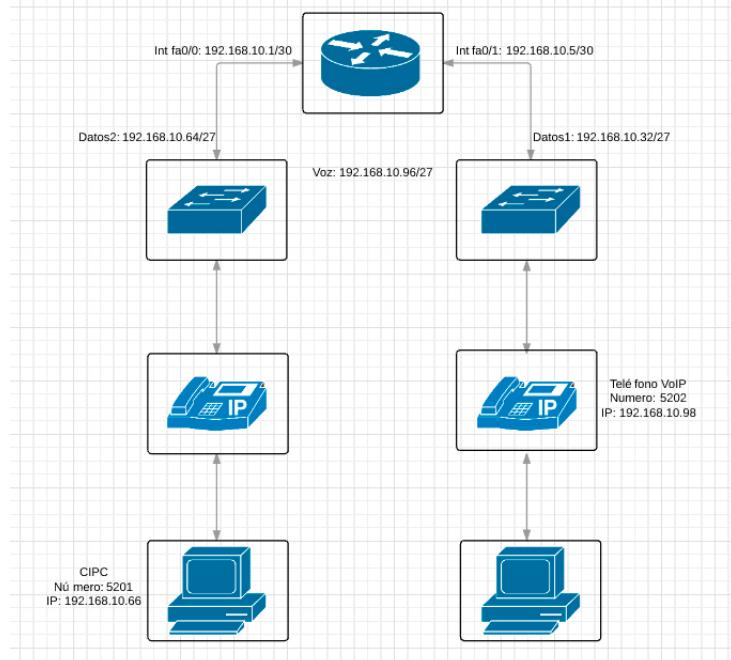
Por ello, podemos concluir, que el protocolo SCCP presenta una mayor optimización a nivel técnico que el protocolo SIP.

Si realizamos la comparativa entre los protocolos SIP y SCCP a nivel general, podemos destacar.

- Las ventajas de utilizar el primero de ellos (protocolo SIP), que es open source (código abierto), posibilitando la descarga e instalación sin permiso ni licencias establecidas por el fabricante, por el contrario el protocolo SCCP es un software de carácter privativo enfocado a la infraestructura de red, a instalación y configuración de la telefonía IP que posee dicho protocolo en contraposición a la instalación y configuración de la PBX que caracteriza al protocolo SIP.
- La flexibilidad de configuración que posee el protocolo SIP frente a una mayor rigidez de la misma en el protocolo SCCP.
- Ambos protocolos poseen un buen soporte y documentación.
- Existen grandes diferencias a la hora de realizar la autenticación de los usuarios. El protocolo SIP muestra en los frames información relevante como son el usuario y contraseña a la hora de registrar el dispositivo en el servidor. Por el contrario, el protocolo SCCP únicamente registra las MACs de los dispositivos.
- Con respecto a los frames generados, son más simples y ligeros en el protocolo SCCP, mientras que en el protocolo SIP contienen una mayor información adicional, quedando expuesta en términos de seguridad la información (usuario, contraseña) de los usuarios.
- Haciendo referencia al envío y recepción de frames, ambos protocolos emplean la encapsulación de paquetes UDP.
- Cabe destacar también, mediante el uso del protocolo SIP no es necesario la renovación temporal de los certificados oficiales, y además es mucho más barato debido a que se puede aprovechar la infraestructura (red) existente, ventajas que no posee el protocolo SCCP, ya que solamente es funcional con productos de Cisco.
- El protocolo SCCP posee una mayor facilidad de implementación y mantenimiento, alejándose de la complejidad que suponen para el personal no técnico el uso del protocolo SIP.

Anexo I: Cisco CCNA Voice

Configuración Cisco



Packet tracer: Red Cisco

Configuración de Router

Comenzaremos a programar los comandos necesarios en el router Alpha. Tendremos que programar las siguientes partes en el router:

- Hostname: Nombre del Router.
- Redes: 2 subredes.
- Dhcp pool: Configuración del protocolo DHCP.
- Ip NAT: Mapeo múltiples direcciones ip privadas.
- Access-list: configuración de listas de acceso IP.

- Telephony-service: Configuración del Cisco Unified CallManager express.
- ephone-dn: Configuración manual de los terminales virtuales telefónicos.

Configuración básica de puertos del router

```

Router>enable
Router#configure terminal
Router(config)#hostname Alpha
Alpha(config)#interface FastEthernet0/0
Alpha(config-if)#no ip address
Alpha(config-if)#exit

Alpha(config)#interface fastEthernet 0/0.1
Alpha(config-subif)#encapsulation dot1q 1 native
Alpha(config-subif)#ip address 192.168.10.1 255.255.255.252
Alpha(config-if)#no shutdown
Alpha(config-if)#exit

Alpha(config)#interface fastEthernet 0/0.2
Alpha(config-subif)#encapsulation dot1q 10
Alpha(config-subif)#ip address 192.168.10.97 255.255.255.224
Alpha(config-if)#no shutdown
Alpha(config-subif)#exit

Alpha(config)#interface fastEthernet 0/0.3
Alpha(config-subif)#encapsulation dot1q 20
Alpha(config-subif)#ip address 192.168.10.33 255.255.255.224
Alpha(config-if)#no shutdown
Alpha(config-subif)#exit

Alpha(config)#interface FastEthernet0/1
Alpha(config-if)#no ip address
Alpha(config-if)#exit

Alpha(config)#interface fastEthernet 0/1.1
Alpha(config-subif)#encapsulation dot1q 1 native
Alpha(config-subif)#ip address 192.168.10.5 255.255.255.252
Alpha(config-if)#no shutdown
Alpha(config-if)#exit

Alpha(config)#interface fastEthernet 0/1.3
Alpha(config-subif)#encapsulation dot1q 20
Alpha(config-subif)#ip address 192.168.10.65 255.255.255.224
Alpha(config-subif)#no shutdown
Alpha(config-subif)#exit

```

	Comandos	Explicación
Paso 1	< enable > Router>enable	Activa el modo EXEC privilegiado.
Paso 2	< configure terminal > Router#configure terminal	Acceso al modo de configuración global.
Paso 3	hostname < name > Router(config)#hostname Alpha	Administra nombre al Router.

	Comandos	Explicación
Paso 4	interface FastEthernet < port > Alpha(config)#interface FastEthernet0/0	Acceso al modo de configuración de la interfaz.
Paso 5	< no ip address > Alpha(config-subif)#no ip address	No se administra IP.
Paso 6	no shutdown Alpha(config-if)#no shutdown	Activa la interfaz.
Paso 7	exit Alpha(config-if)#exit	Salir de la configuración de la interfaz.
Paso 8	interface FastEthernet < port > Alpha(config)#interface fastEthernet 0/0.1	Acceso al modo de configuración de la subinterfaz.
Paso 9	encapsulation < dot1q/ISL > < id.num > < native > Alpha(config-subif)#encapsulation dot1q 1 native	Habilitar conexión troncal y nativa.
Paso 10	ip address < ip-address > < mask > Alpha(config-if)#ip address 192.168.10.1 255.255.255.252	Asigna una dirección válida a la interfaz.
Paso 11	< no shutdown > Alpha(config-if)#no shutdown	Se activa la interfaz.
Paso 12	< exit > Alpha(config-if)#exit	Salir de la configuración de la interfaz.
Paso 13	interface FastEthernet < port > Alpha(config)#interface fastEthernet 0/0.2	Acceso al modo de configuración de la subinterfaz.
Paso 14	encapsulation < dot1q/ISL > < id.num > Alpha(config-subif)#encapsulation dot1q 10	Habilitar la conexión troncal.
Paso 15	ip address < ip-address > < mask > Alpha(config-subif)#ip address 192.168.10.97 255.255.255.224	Asignar una dirección ip válida a la subinterfaz de voz.
Paso 16	< no shutdown > Alpha(config-if)#no shutdown	Se activa la interfaz.

	Comandos	Explicación
Paso 17	< exit > Alpha(config-subif)#exit	Salir de la configuración de la interfaz.
Paso 18	interface FastEthernet < port > Alpha(config)#interface fastEthernet 0/0.3	Acceso al modo de configuración de la subinterfaz.
Paso 19	encapsulation < dot1q/ISL > < id.num > Alpha(config-subif)#encapsulation dot1q 20	Habilitar la conexión troncal.
Paso 20	ip address < ip-address > < mask > Alpha(config-subif)#ip address 192.168.10.33 255.255.255.224	Asignar una dirección ip válida a la subinterfaz de Datos.
Paso 21	< no shutdown > Alpha(config-if)#no shutdown	Se activa la interfaz.
Paso 22	< exit > Alpha(config-subif)#exit	Salir de la configuración de la interfaz.
Paso 23	interface FastEthernet < port > Alpha(config)#interface FastEther- net0/1	Acceso al modo de configuración de la interfaz.
Paso 24	ip address < ip-address > < mask > Alpha(config-if)# no ip address	No se administra IP.
Paso 25	< no shutdown > Alpha(config-if)#no shutdown	Se activa la interfaz.
Paso 26	< exit > Alpha(config-if)#exit	Salir de la configuración de la interfaz.
Paso 27	interface FastEthernet < port > Alpha(config)#interface fastEthernet 0/1.1	Acceso al modo de configuración de la subinterfaz.
Paso 28	encapsulation < dot1q/ISL > < id.num > < native > Alpha(config-subif)#encapsulation dot1q 1 native	Habilitar conexión troncal y nativa.
Paso 29	< no ip address > Alpha(config-subif)#ip address 192.168.10.5 255.255.255.252	Asignar una dirección ip válida a la interfaz .

	Comandos	Explicación
Paso 30	< <i>no shutdown</i> > Alpha(config-if)#no shutdown	Se activa la interfaz.
Paso 31	< <i>exit</i> > Alpha(config-if)#exit	Salir de la configuración de la interfaz.
Paso 32	interface FastEthernet < <i>port</i> > Alpha(config)#interface fastEthernet 0/1.3	Acceso al modo de configuración de la subinterfaz.
Paso 33	encapsulation < <i>dot1q/ISL</i> > < <i>id.num</i> > Alpha(config-subif)#encapsulation dot1q 20	Habilitar la conexión troncal.
Paso 34	ip address < <i>ip-address</i> > < <i>mask</i> > Alpha(config-subif)#ip address 192.168.10.65 255.255.255.224	Asignar una dirección ip válida a la subinterfaz de datos.
Paso 35	< <i>no shutdown</i> > Alpha(config-if)#no shutdown	Se activa la interfaz.
Paso 36	< <i>exit</i> > Alpha(config-subif)#exit	Salir de la configuración de la interfaz.

Configuración puertos Router Cisco

Configurando DHCP para VoIP y Datos

```

Alpha(config)#ip dhcp pool Voz
Alpha(dhcp-config)#network 192.168.10.96 255.255.255.224
Alpha(dhcp-config)#default-router 192.168.10.97
Alpha(dhcp-config)#option 150 ip 192.168.10.97
Alpha(dhcp-config)#exit

Alpha(config)#ip dhcp pool Datos_1
Alpha(dhcp-config)#network 192.168.10.32 255.255.255.224
Alpha(dhcp-config)#default-router 192.168.10.33
Alpha(dhcp-config)#option 150 ip 192.168.10.33
Alpha(dhcp-config)#exit

Alpha(config)#ip dhcp pool Datos_2
Alpha(dhcp-config)#network 192.168.10.64 255.255.255.224
Alpha(dhcp-config)#default-router 192.168.10.65
Alpha(dhcp-config)#option 150 ip 192.168.10.65
Alpha(dhcp-config)#exit

```

	Comandos	Explicación
Paso 1	ip dhcp pool < pool-name > Alpha(config)#ip dhcp pool Voz	Crea nombre para servidor DHCP y accede al modo de configuración del servidor .
Paso 2	network < ip-address > < mask > Alpha(dhcp-config)#network 192.168.10.96 255.255.255.224	Especifica la dirección ip del servidor DCHP y su máscara (opcional).
Paso 3	default-router < ip-address > Alpha(dhcp-config)#default-router 192.168.10.97	Especifica al router que los teléfonos IP utilizarán esta dirección ip para enviar y recibir tráfico de información.
Paso 4	option 150 ip < ip-address > Alpha(dhcp-config)#option 150 ip 192.168.10.97	Especifica la dirección del servidor TFTP. De la cual, se descargarán los teléfonos ip el archivo de configuración
Paso 5	< exit > Alpha(config-if)#exit	Salir de la configuración de la interfaz.
Paso 6	ip dhcp pool < pool-name > Alpha(config)#ip dhcp pool Datos_1	Crea nombre para servidor DHCP y accede al modo de configuración del servidor .
Paso 7	network < ip-address > < mask > Alpha(dhcp-config)#network 192.168.10.32 255.255.255.224	Especifica la dirección ip del servidor DCHP y su máscara (opcional).
Paso 8	default-router < ip-address > Alpha(dhcp-config)#default-router 192.168.10.33	Especifica al router que los teléfonos IP utilizarán esta dirección ip para enviar y recibir tráfico de información.
Paso 9	option 150 ip < ip-address > Alpha(dhcp-config)#option 150 ip 192.168.10.33	Especifica la dirección del servidor TFTP. De la cual, se descargarán los teléfonos ip el archivo de configuración
Paso 10	< exit > Alpha(config-if)#exit	Salir de la configuración de la interfaz.
Paso 11	ip dhcp pool < pool-name > Alpha(config)#ip dhcp pool Datos_2	Crea nombre para servidor DHCP y accede al modo de configuración del servidor .
Paso 12	network < ip-address > < mask > Alpha(dhcp-config)#network 192.168.10.64 255.255.255.224	Especifica la dirección ip del servidor DCHP y su máscara (opcional).

	Comandos	Explicación
Paso 13	default-router < <i>ip-address</i> > Alpha(dhcp-config)#default-router 192.168.10.65	Especifica al router que los teléfonos IP utilizarán esta dirección ip para enviar y recibir tráfico de información.
Paso 14	option 150 ip < <i>ip-address</i> > Alpha(dhcp-config)#option 150 ip 192.168.10.65	Especifica la dirección del servidor TFTP. De la cual, se descargarán los teléfonos ip el archivo de configuración
Paso 15	< <i>exit</i> > Alpha(config-if)#exit	Salir de la configuración de la interfaz.

Configuración DHCP(VoIP y Datos)

Configuración del Servicio VoIP

```
Alpha(config) #telephony-service
Alpha(config-telephony) #max-ephones 10
Alpha(config-telephony) #max-dn 10
Alpha(config-telephony) #ip source-address 192.168.10.97 port 2000
Alpha(config-telephony) #auto assign 1 to 10
Alpha(config-telephony) #exit
```

	Comandos	Explicación
Paso 1	< <i>telephony-service</i> > Alpha(config)#telephony-service	Acceder al modo configuración de VoIP .
Paso 2	max-ephones < <i>num-ephones</i> > Alpha(config-telephony)#max-ephones 10	Especifica la cantidad máxima de ephones(teléfonos ip) que se pueden conectar.
Paso 3	max-dn < <i>num-extensions</i> > Alpha(config-telephony)#max-dn 10	Especifica el número de extensiones posibles y configurables en el router.
Paso 4	ip source-address < <i>ip-address</i> >port< <i>num-port</i> > Alpha(config-telephony)#ip source-address 192.168.10.97 port 2000	Especifica la dirección ip y el puerto de registro de teléfonos (puerto 2000).
Paso 5	auto assign < <i>num</i> >to < <i>num</i> > Alpha(config-telephony)#auto assign 1 to 10	Asigna automáticamente los la numeración de los botones .
Paso 6	< <i>exit</i> > Alpha(config-telephony)#exit	Salir del modo configuración de teléfonos IP.

Comandos	Explicación
----------	-------------

Tabla 11.4: Configuración del servicio VoIP

Creando directorio de números

```

Alpha(config)#ephone-dn 1
Alpha(config-ephone-dn)#number 5201
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 2
Alpha(config-ephone-dn)#number 5202
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 3
Alpha(config-ephone-dn)#number 5203
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 4
Alpha(config-ephone-dn)#number 5204
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 5
Alpha(config-ephone-dn)#number 5205
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 6
Alpha(config-ephone-dn)#number 5206
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 7
Alpha(config-ephone-dn)#number 5207
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 8
Alpha(config-ephone-dn)#number 5208
Alpha(config-ephone-dn)#exit

Alpha(config)#ephone-dn 9
Alpha(config-ephone-dn)#number 5209
Alpha(config-ephone-dn)#exit

```

Comandos	Explicación
Paso 1 ephone-dn< num-directory > Alpha(config)#ephone-dn 1	Acceder al modo configuración del directorio 1.
Paso 2 number < num-phone > Alpha(config-ephone-dn)#number 5201	Asigna número de teléfono.
Paso 3 < exit > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 4 ephone-dn< num-directory > Alpha(config)#ephone-dn 2	Acceder al modo configuración del directorio 2.

	Comandos	Explicación
Paso 5	number < num-phone > Alpha(config-ephone-dn)#number 5202	Asigna número de teléfono.
Paso 6	< exit > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 7	ephone-dn< num-directory > Alpha(config)#ephone-dn 3	Acceder al modo configuración del directorio 3.
Paso 8	number < num-phone > Alpha(config-ephone-dn)#number 5203	Asigna número de teléfono.
Paso 9	< exit > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 10	ephone-dn< num-directory > Alpha(config)#ephone-dn 4	Acceder al modo configuración del directorio 4.
Paso 11	number < num-phone > Alpha(config-ephone-dn)#number 5204	Asigna número de teléfono.
Paso 12	< exit > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 13	ephone-dn< num-directory > Alpha(config)#ephone-dn 5	Acceder al modo configuración del directorio 5.
Paso 14	number < num-phone > Alpha(config-ephone-dn)#number 5205	Asigna número de teléfono.
Paso 15	< exit > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 16	ephone-dn< num-directory > Alpha(config)#ephone-dn 6	Acceder al modo configuración del directorio 6.
Paso 17	number < num-phone > Alpha(config-ephone-dn)#number 5206	Asigna número de teléfono.
Paso 18	< exit > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.

	Comandos	Explicación
Paso 19	ephone-dn< <i>num-directory</i> > Alpha(config)#ephone-dn 7	Acceder al modo configuración del directorio 7.
Paso 20	number < <i>num-phone</i> > Alpha(config-ephone-dn)#number 5207	Asigna número de teléfono.
Paso 21	< <i>exit</i> > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 22	ephone-dn< <i>num-directory</i> > Alpha(config)#ephone-dn 8	Acceder al modo configuración del directorio 8.
Paso 23	number < <i>num-phone</i> > Alpha(config-ephone-dn)#number 5208	Asigna número de teléfono.
Paso 24	< <i>exit</i> > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.
Paso 25	ephone-dn< <i>num-directory</i> > Alpha(config)#ephone-dn 9	Acceder al modo configuración del directorio 9.
Paso 26	number < <i>num-phone</i> > Alpha(config-ephone-dn)#number 5209	Asigna número de teléfono.
Paso 27	< <i>exit</i> > Alpha(config-ephone-dn)#exit	Salir del modo configuración de teléfonos IP.

Creando directorio de números

Configuración terminales

```

Alpha(config) #ephone 1
Alpha(config-ephone) #mac-address 448A.5BEE.BFD1
Alpha(config-ephone) #username "ordenador"
Alpha(config-ephone) #type CIPC
Alpha(config-ephone) #button 1:1
Alpha(config-ephone) #end

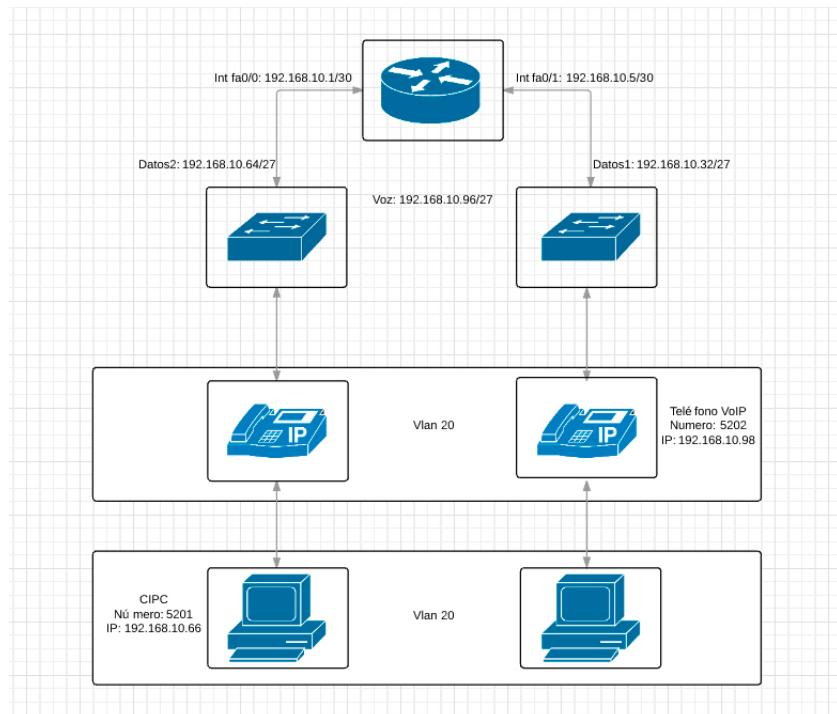
Alpha(config) #ephone 2
Alpha(config-ephone) #mac-address 001F.9EAD.C8E3
Alpha(config-ephone) #username "telefono_cisco"
Alpha(config-ephone) #type 7941
Alpha(config-ephone) #button 1:2
Alpha(config-ephone) #end

```

	Comandos	Explicación
Paso 1	ephone < <i>num</i> > Alpha(config)#ephone 1	Acceder al modo configuración del directorio 1.
Paso 2	mac < <i>xxxx.xxxx.xxxx</i> > Alpha(config-ephone)#mac-address 448A.5BEE.BFD1	Asigna MAC del dispositivo SCCP.
Paso 3	username < <i>name</i> > Alpha(config-ephone)#username "ordenador"	Nombre del ephone.
Paso 4	type < CIPC/7941/... > Alpha(config-ephone)#type CIPC	Tipo de dispositivo.
Paso 5	button < <i>num:num</i> > Alpha(config-ephone)#button 1:1	Asigna el primer botón en el teléfono en el directorio número 1.
Paso 6	< <i>exit</i> > Alpha(config-ephone)#end	Salir del modo configuración de teléfonos IP.
Paso 7	ephone < <i>num</i> > Alpha(config)#ephone 2	Acceder al modo configuración del directorio 2.
Paso 8	mac < <i>xxxx.xxxx.xxxx</i> > Alpha(config-ephone)#mac-address 001F.9EAD.C8E3	Asigna MAC del dispositivo SCCP.
Paso 9	username < <i>name</i> > Alpha(config-ephone)#username "telefono_cisco"	Nombre del ephone.
Paso 10	type < CIPC/7941/... > Alpha(config-ephone)#type 7941	Tipo de dispositivo.
Paso 11	button < <i>num:num</i> > Alpha(config-ephone)#button 1:2	Asigna el primer botón en el teléfono en el directorio número 2.
Paso 12	< <i>exit</i> > Alpha(config-ephone)#end	Salir del modo configuración de teléfonos IP.

[Configuración terminales](#)

Configuración Switch



Packet tracer: Red Cisco + Vlans

Configurando las VLANs rama a

```

switch>enable
switch#configure terminal
switch(config)hostname Switch_a_1
Switch_a_1(config)#vlan 10
Switch_a_1(config-vlan)#name Voz
Switch_a_1(config-vlan)#exit
Switch_a_1(config)#vlan 20
Switch_a_1(config-vlan)#name Datos
Switch_a_1(config-vlan)#exit
Switch_a_1(config)#int fa 0/24
Switch_a_1(config-if)#switchport mode trunk
Switch_a_1(config-if)#exit
Switch_a_1(config)#int range fa0/1 - 10
Switch_a_1(config-if-range)#switchport mode access
Switch_a_1(config-if-range)#switchport acces vlan 20
Switch_a_1(config-if-range)#switchport voice vlan 10
Switch_a_1(config-if-range)#end
Switch_a_1#

```

	Comandos	Explicación
Paso 1	<enable> switch>enable	Acceder al modo EXEC privilegiado.

	Comandos	Explicación
Paso 2	< configure-terminal > switch#configure terminal	Accede al modo de configuración global.
Paso 3	hostname< name > switch(config)#hostname Switch_a_1	Dar nombre al Switch.
Paso 4	vlan < num-vlan > Switch_a_1(config)#vlan 10	Asigna número a la Vlan.
Paso 5	name< name-vlan > Switch_a_1(config-vlan)#name Voz	Asigna nombre a la Vlan.
Paso 6	< exit > Switch_a_1(config-vlan)#exit	Salir del modo configuración de la Vlan.
Paso 7	vlan < num-vlan > Switch_a_1(config)#vlan 20	Asigna número a la Vlan.
Paso 8	name< name-vlan > Switch_a_1(config-vlan)#name Datos	Asigna nombre a la Vlan.
Paso 9	< exit > Switch_a_1(config-vlan)#exit	Salir del modo de configuración de la vlan.
Paso 10	int fa < number >/< number > Switch_a_1(config)#int fa 0/24	Acceder al modo de configuración del interfaz 24.
Paso 11	switchport mode< mode/acces/voice > Switch_a_1(config-if)#switchport mode trunk	Habilitar conexión troncal.
Paso 12	< exit > Switch_a_1(config-if)#exit	Salir del modo de configuración de la interfaz 0/24.
Paso 13	int range fa < number >-< number > Switch_a_1(config)#int range fa0/1 - 10	Acceder al modo de configuración del rango de interfaces.
Paso 14	switchport mode < acces/voice > Switch_a_1(config-if- range)#switchport mode access	Habilita el modo acceso a Datos.
Paso 15	switchport acces vlan< num-vlan > Switch_a_1(config-if- range)#switchport voice vlan 10	Habilita el modo acceso de voz en vlan concreta.

	Comandos	Explicación
Paso 17	switchport voice vlan< <i>num-vlan</i> > Switch_a_1(config-if-range)#switchport acces vlan 20	Habilita el modo de datos sobre vlan concreta.
Paso 18	<end> Switch_a_1(config-if-range)#end	Salir del modo configuración del rango de interfaces.

Configuración Vlans rama a

Configurando las Vlans rama b

```

switch>enable
switch#configure terminal
switch(config)hostname Switch_b_1
Switch_b_1(config)#vlan 10
Switch_b_1(config-vlan)#name Voz
Switch_b_1(config-vlan)#exit
Switch_b_1(config)#vlan 20
Switch_b_1(config-vlan)#name Datos
Switch_b_1(config-vlan)#exit
Switch_b_1(config)#int fa 0/24
Switch_b_1(config-if)#switchport mode trunk
Switch_b_1(config-if)#exit
Switch_b_1(config)#int range fa0/1 - 10
Switch_b_1(config-if-range)#switchport mode access
Switch_b_1(config-if-range)#switchport acces vlan 20
Switch_b_1(config-if-range)#switchport voice vlan 10
Switch_b_1(config-if-range)#end
Switch_b_1#

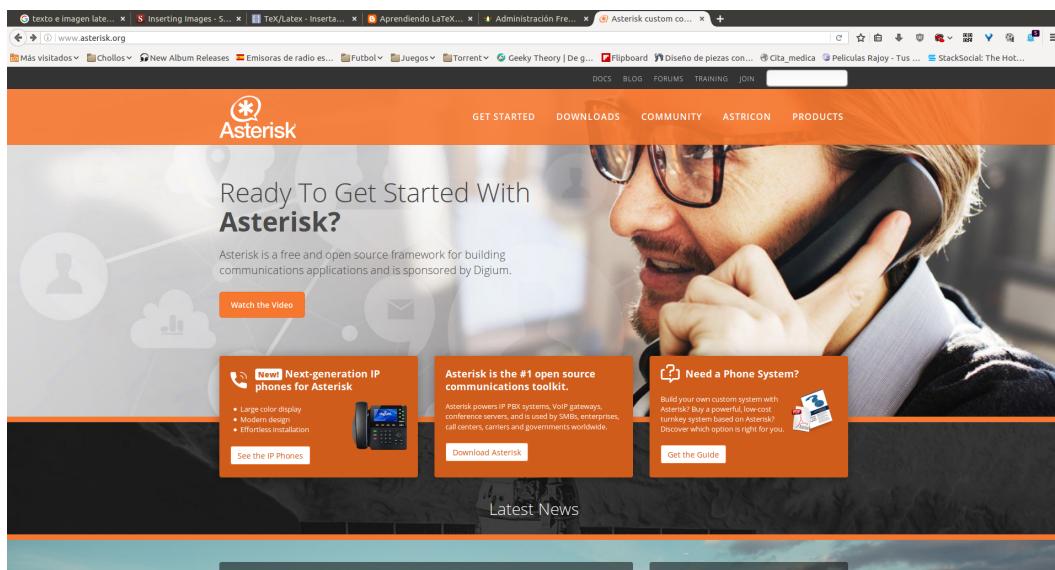
```

	Comandos	Explicación
Paso 1	< <i>enable</i> > switch>enable	Acceder al modo EXEC privilegiado.
Paso 2	< <i>configure-terminal</i> > switch#configure terminal	Accede al modo de configuración global.
Paso 3	hostname< <i>name</i> > switch(config)#hostname Switch_b_1	Dar nombre al Switch.
Paso 4	vlan < <i>num-vlan</i> > Switch_b_1(config)#vlan 10	Asigna número a la Vlan.
Paso 5	name< <i>name-vlan</i> > Switch_b_1(config-vlan)#name Voz	Asigna nombre a la Vlan.

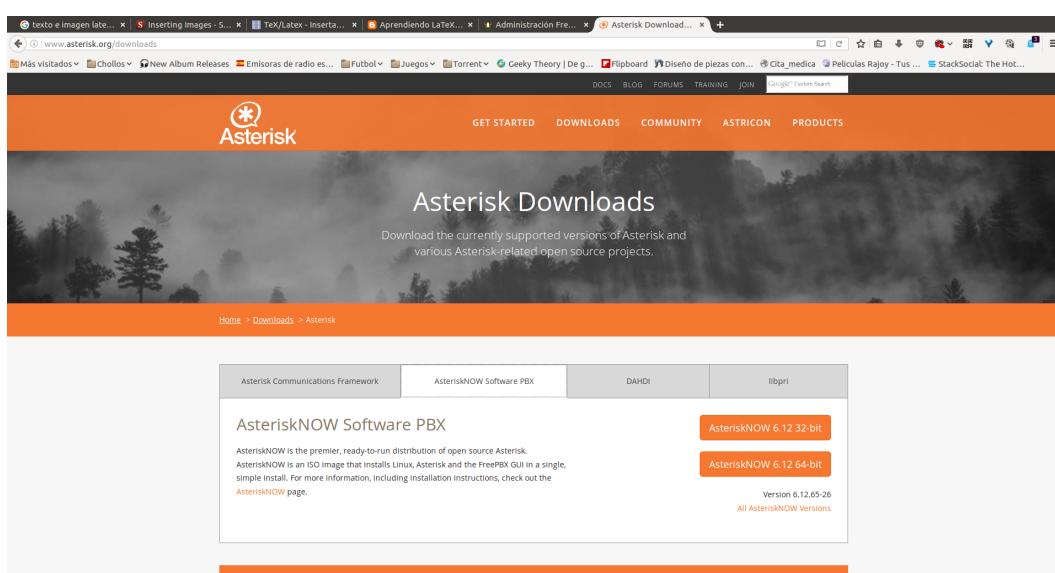
	Comandos	Explicación
Paso 6	<exit> Switch_b_1(config-vlan)#exit	Salir del modo configuración de la Vlan.
Paso 7	vlan < num-vlan > Switch_b_1(config)#vlan 20	Asigna número a la Vlan.
Paso 8	name< name-vlan > Switch_b_1(config-vlan)#name Datos	Asigna nombre a la Vlan.
Paso 9	<exit> Switch_b_1(config-vlan)#exit	Salir del modo de configuración de la vlan.
Paso 10	int fa < number >/< number > Switch_b_1(config)#int fa 0/24	Acceder al modo de configuración del interfaz 24.
Paso 11	switchport mode< mode/acces/voice > Switch_b_1(config-if)#switchport mode trunk	Habilitar conexión troncal.
Paso 12	<exit> Switch_b_1(config-if)#exit	Salir del modo de configuración de la interfaz 0/24.
Paso 13	int range fa < number >-< number > Switch_b_1(config)#int range fa0/1 - 10	Acceder al modo de configuración del rango de interfaces.
Paso 14	switchport mode < acces/voice > Switch_b_1(config-if-range)#switchport mode access	Habilita el modo acceso a Datos.
Paso 15	switchport acces vlan< num-vlan > Switch_b_1(config-if-range)#switchport acces vlan 20	Habilita el modo acceso a datos en vlan concreta.
Paso 17	witchport voice vlan< num-vlan > Switch_b_1(config-if-range)#switchport voice vlan 10	Habilita el modo de Voz sobre vlan concreta.
Paso 18	< end > Switch_b_1(config-if-range)#end	Salir del modo configuración del rango de interfaces.

[Configuración Vlans rama b](#)

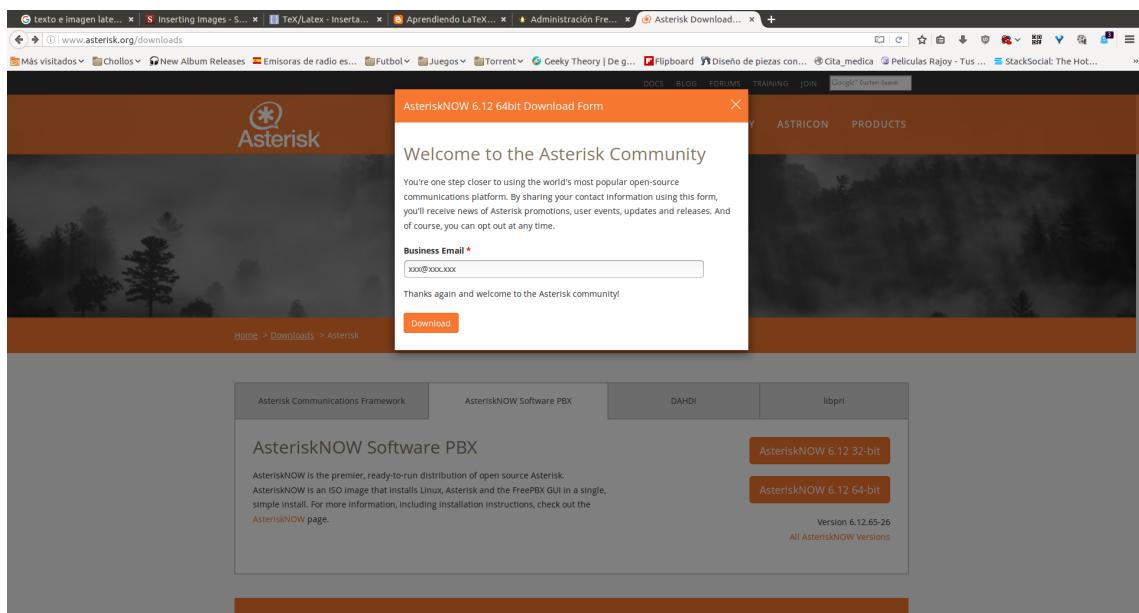
Anexo II: Asterisk FreePBX



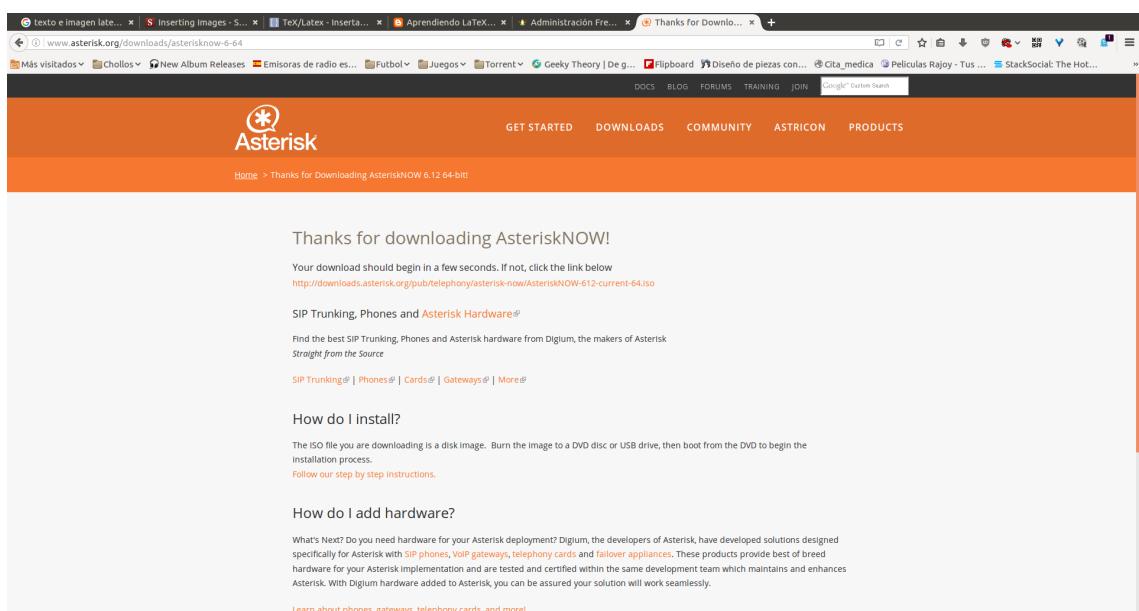
www.asterisk.org



Agregar Usuario FreePBX



Gestor de Usuario FreePBX



Agregar Usuario FreePBX

Descarga

1. Nos dirigiremos a la página oficial de Asterisk³. Pulsaremos **Download** y aparecerá la página oficial de descargas del archivo.
2. Pulsaremos la opción **AsteriskNow Software PBX**. Nos aparecerán en naranja 2 opciones, una de 32 bits y otra de 64 bits.

³<http://www.asterisk.org/>

3. Elegiremos la que más se adecúe a nuestro sistema. Si no queremos poner el email para la base de datos con poner xxx@xxx.xxx nos dará acceso directo a la descarga.
4. Si no se activara la descarga, se puede pinchar directamente en el archivo iso (1.1 GB).

Al finalizar la descarga, tocará instalar la ISO, que está basada en dos sistemas operativos con núcleo kernel linux (Fedora y CentOS). Esto nos ofrece estabilidad y compatibilidad a la hora de desarrollar e instalar aplicaciones al sistema.



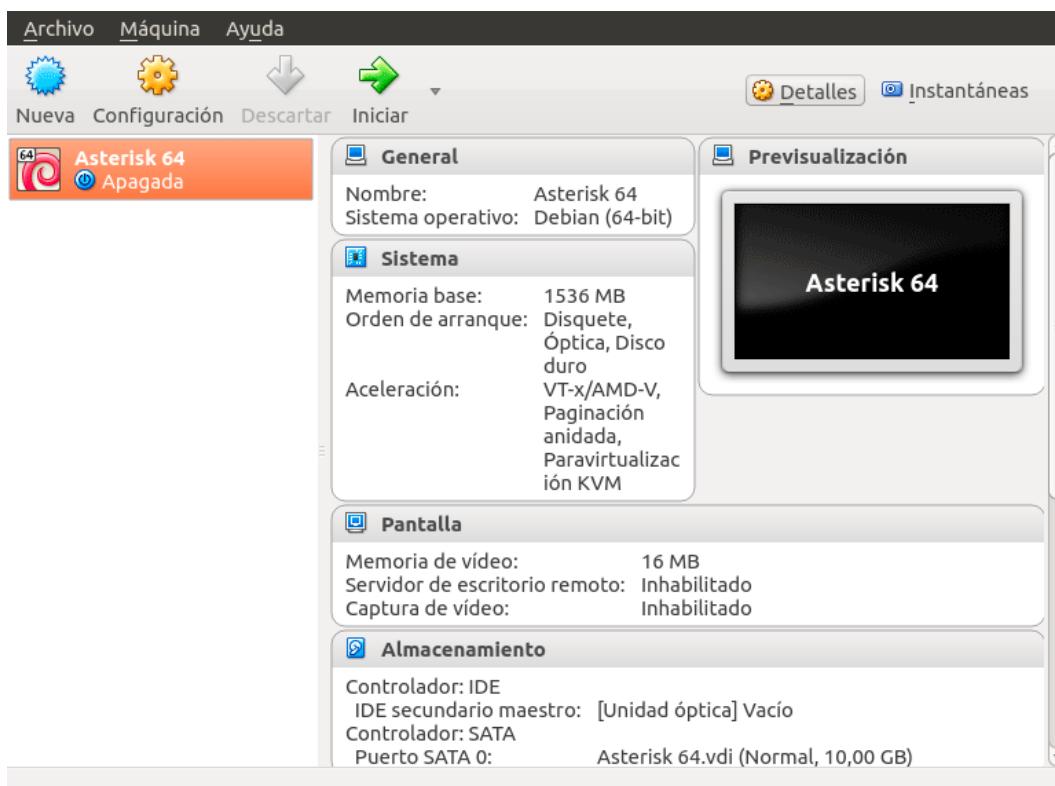
Creación Asterisk Virtual



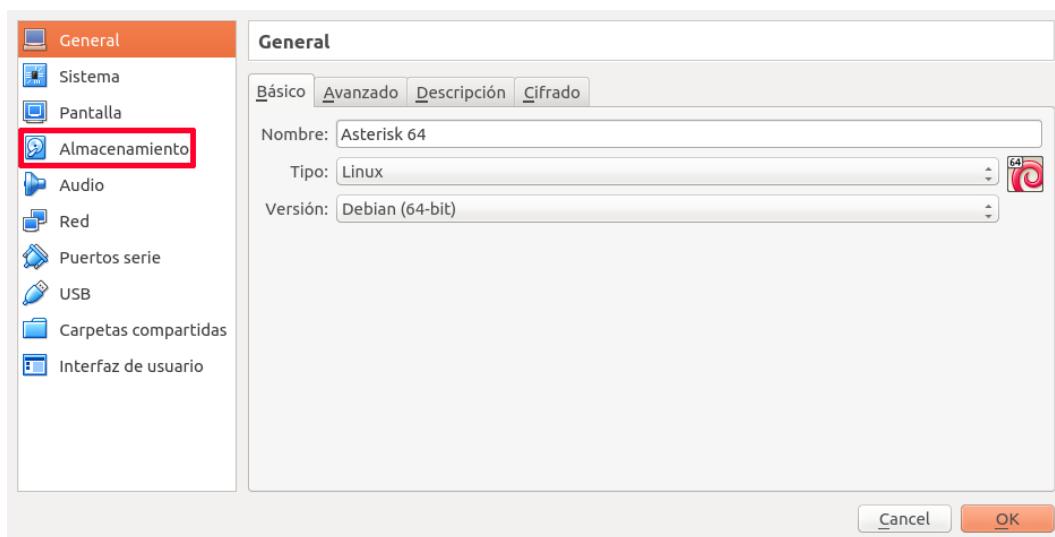
Nombre, memoria y HD



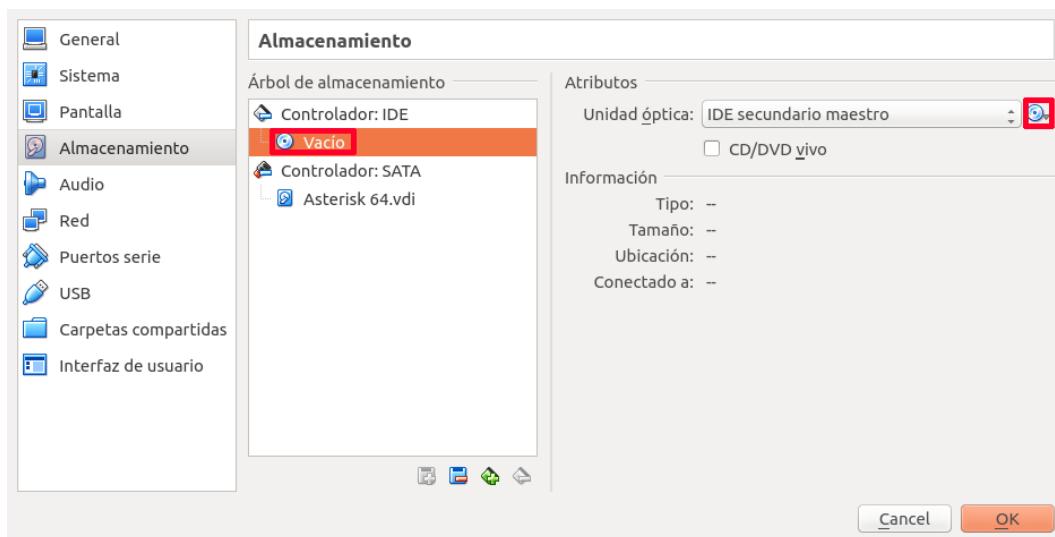
Ubicación, tamaño y tipo HD



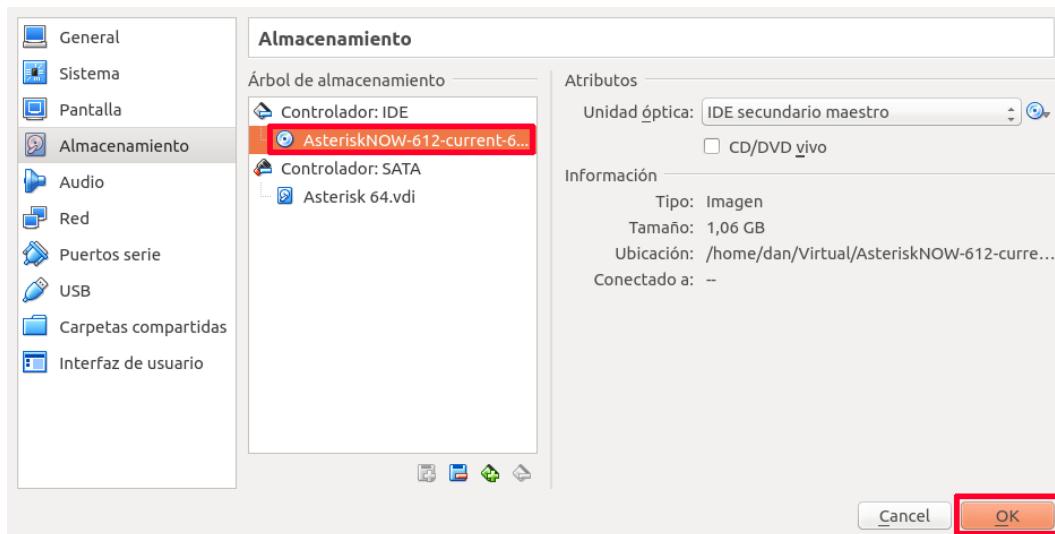
Inicio Asterisk 64bits



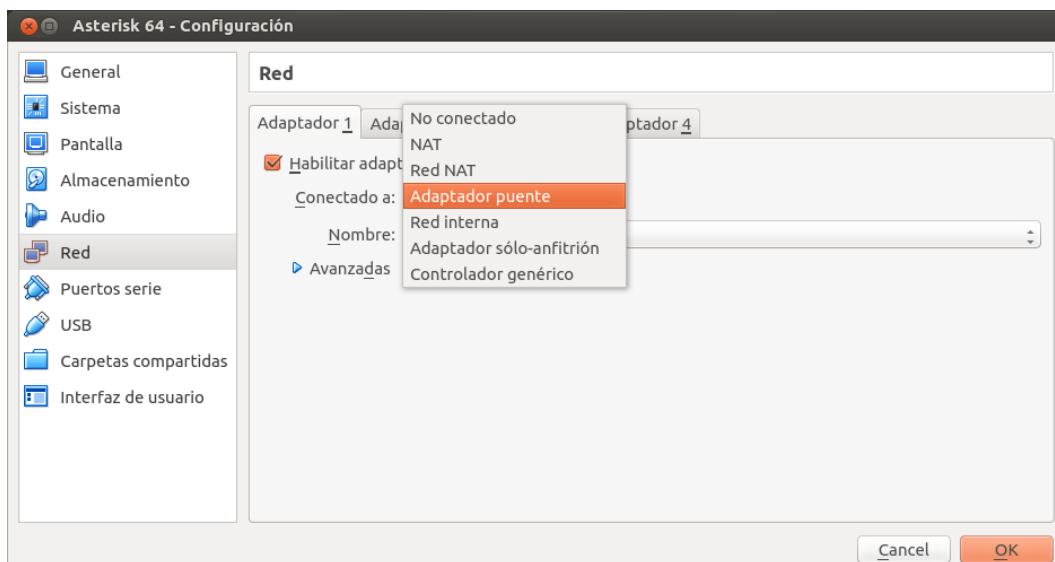
Instalación Asterisk 1/3



Instalación Asterisk 2/3



Instalación Asterisk 3/3



Configuración Asterisk

Configuración VirtualBox

1. Haremos la instalación en un entorno virtual. Usaremos para este cometido la aplicación **Oracle VM VirtualBox**⁴
Para hacer una nueva instalación, pulsaremos el botón **Nueva**. Para instalar un nuevo sistema operativo.
2. Aparecerá una nueva ventana, la cual tendrá en el encabezamiento **Crear máquina virtual**. Daremos un nombre (Asterisk 64), tipo de núcleo (linux), versión (Debian (64-bit)), tamaño de memoria (1536), Disco duro (Crear un disco duro virtual ahora)

⁴<https://www.virtualbox.org/>

Finalizaremos esta parte pulsando el botón **Crear**.

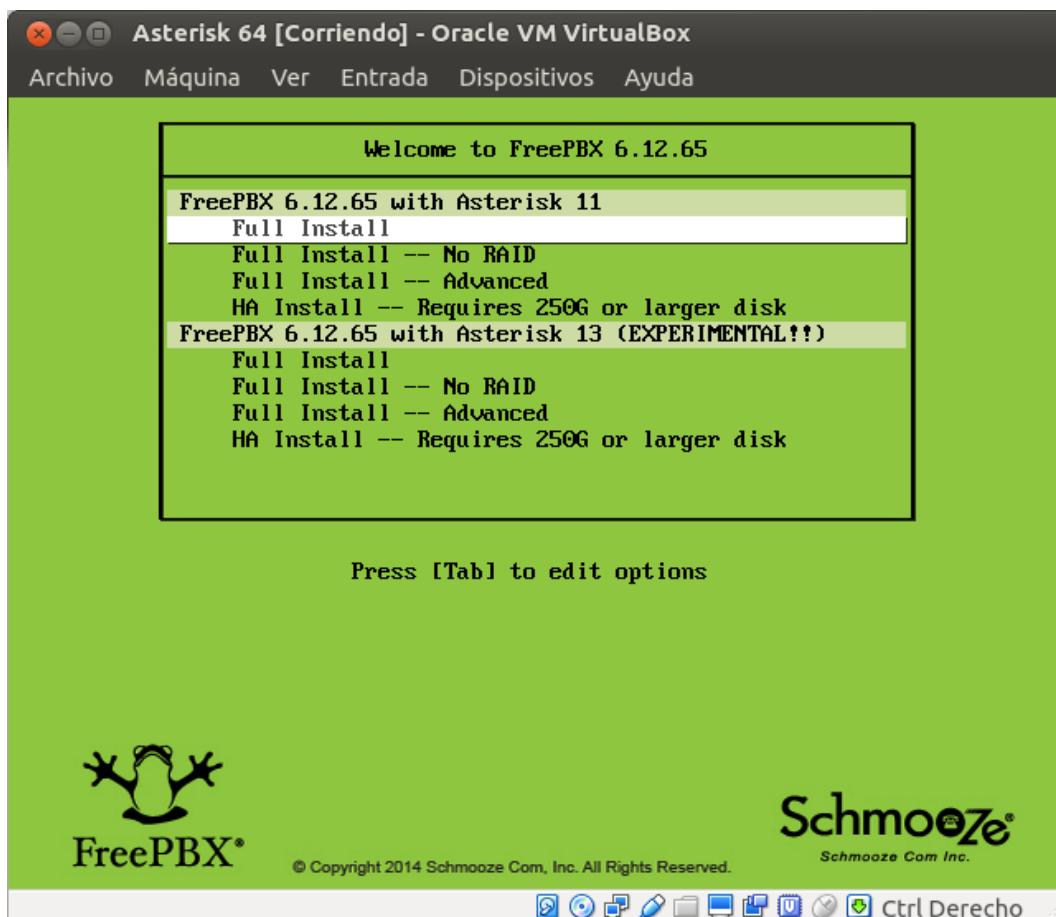
3. Seguidamente aparecerá la ventana Crear disco duro virtual, en el que se dispondrá la ubicación del sistema operativo. El archivo se ubicará de forma predeterminada (/home-/user/Virtualbox vms/) con una extensión vdi.
4. La configuración de esta parte será la siguiente; ubicación de archivo (Asterisk 64), tamaño de archivo (10.00 GB), tipo de archivo disco duro (VDI (VirtualBox Disk Image)), Almacenamiento en unidad de disco duro física (Reservado dinámicamente).

Para finalizar como en el punto anterior, pulsaremos el botón de **Crear**.

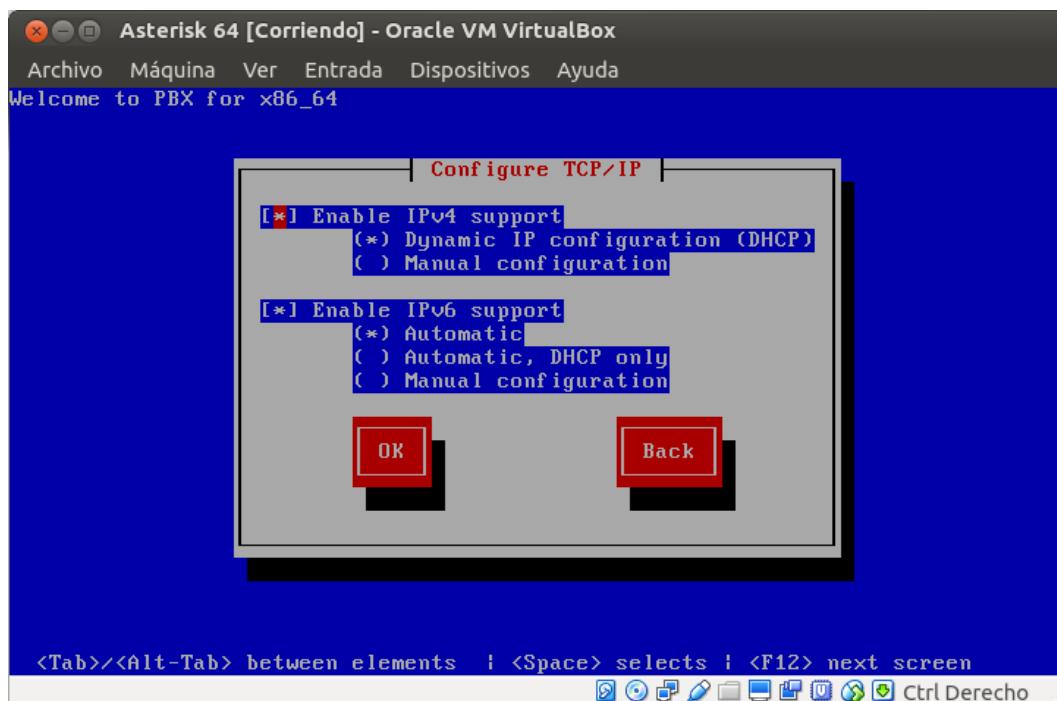
5. De esta manera, haremos la reserva de requisitos del sistema para la posterior instalación del sistema operativo (Asterisk 64).
6. Pulsaremos el botón **Configuración** y aparecerá la ventana configuración de la reserva de sistema y pulsaremos en sobre la opción **Almacenamiento**.
7. Una vez dentro de la pestaña de Almacenamiento, tendremos que cargar la ISO que nos descargamos anteriormente. Esto lo haremos de la siguiente manera.

Pulsaremos debajo del apartado Controlador IDE, en el que pondrá **Vacío**. Nos dirigiremos a la parte derecha y pulsamos el icono con forma de disco que está situado a la derecha de **unidad óptica**. Al pulsarlo, se abrirá otra pantalla, en la cual, pondremos la ruta del archivo ISO antes descargado.

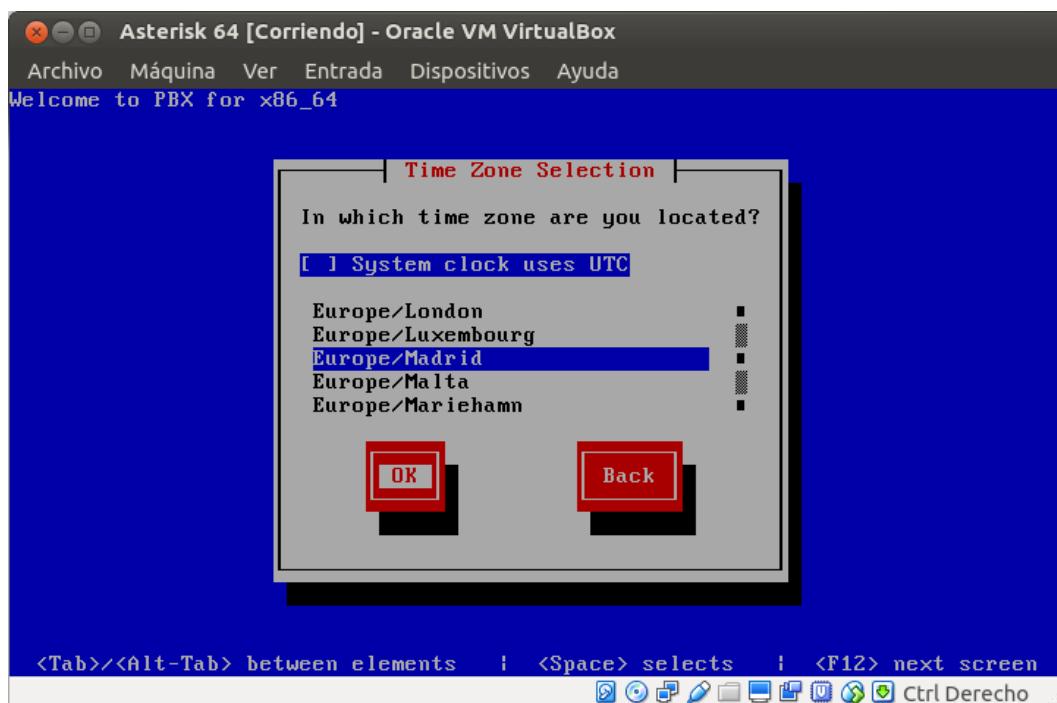
8. Para finalizar la configuración, pulsaremos la opción **Red**, veremos que tenemos activado el Adaptador 1 y en el apartado **Conectado a** elegiremos la opción **Adaptador puente**.



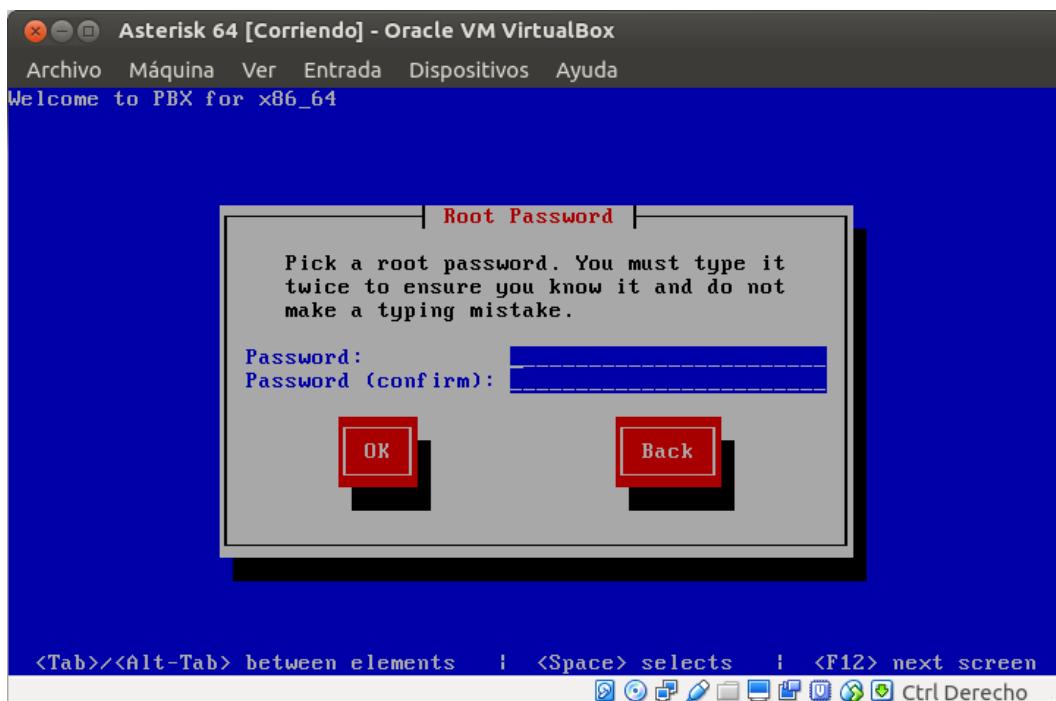
Instalación FreePBX



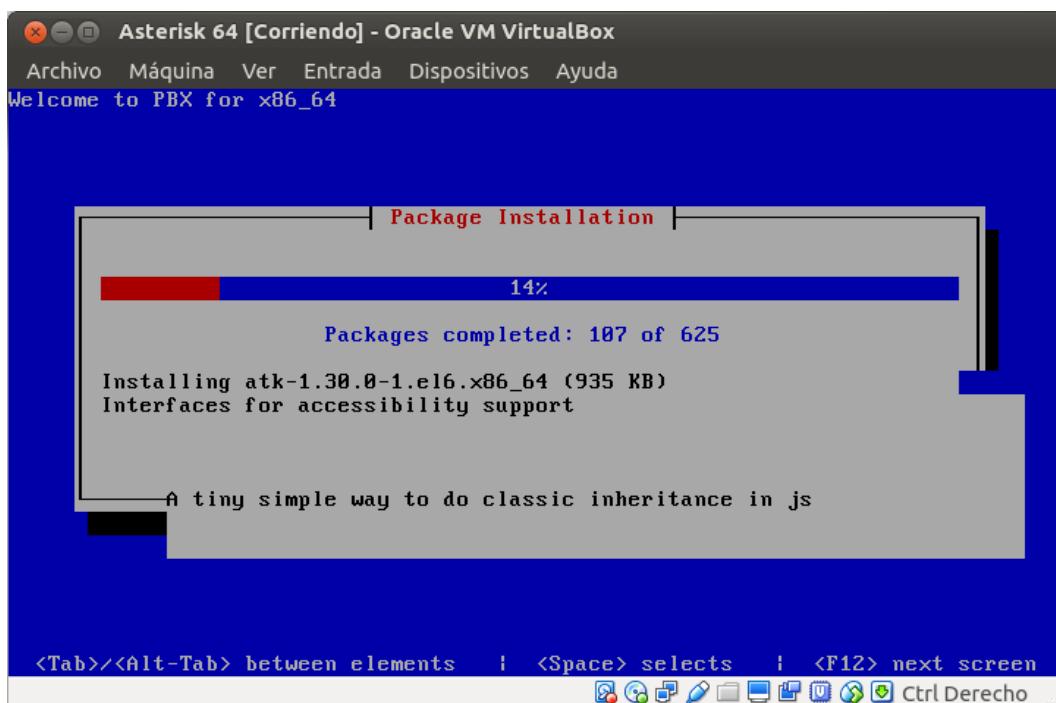
Configuración TCP/IP



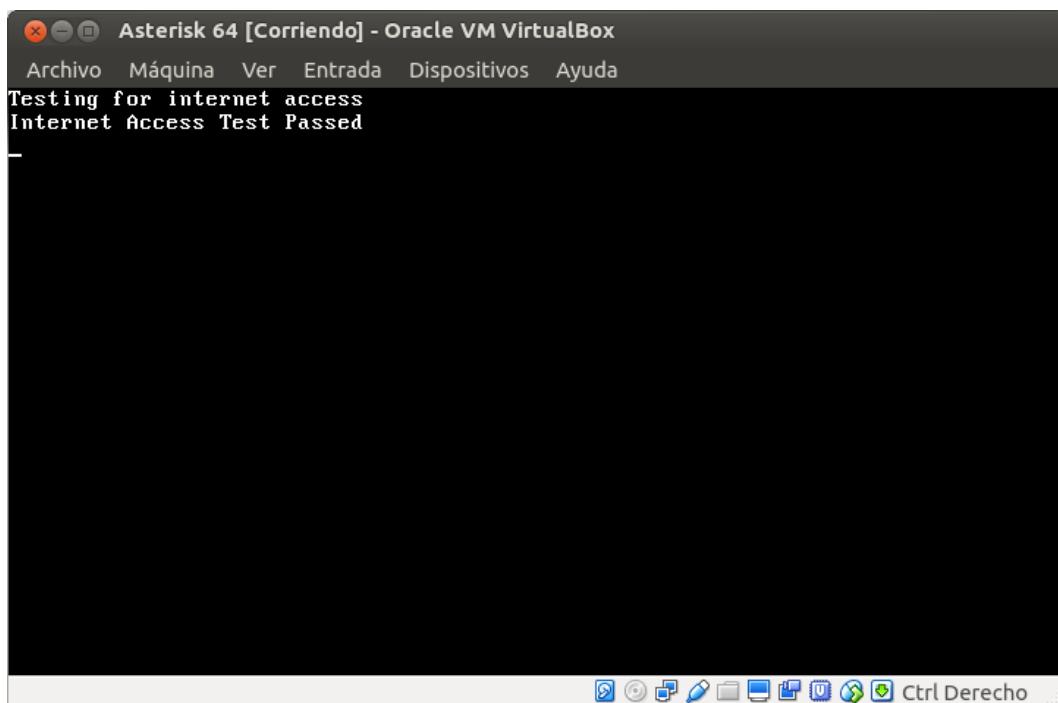
Configuración Time Zone



Configuración Root Password

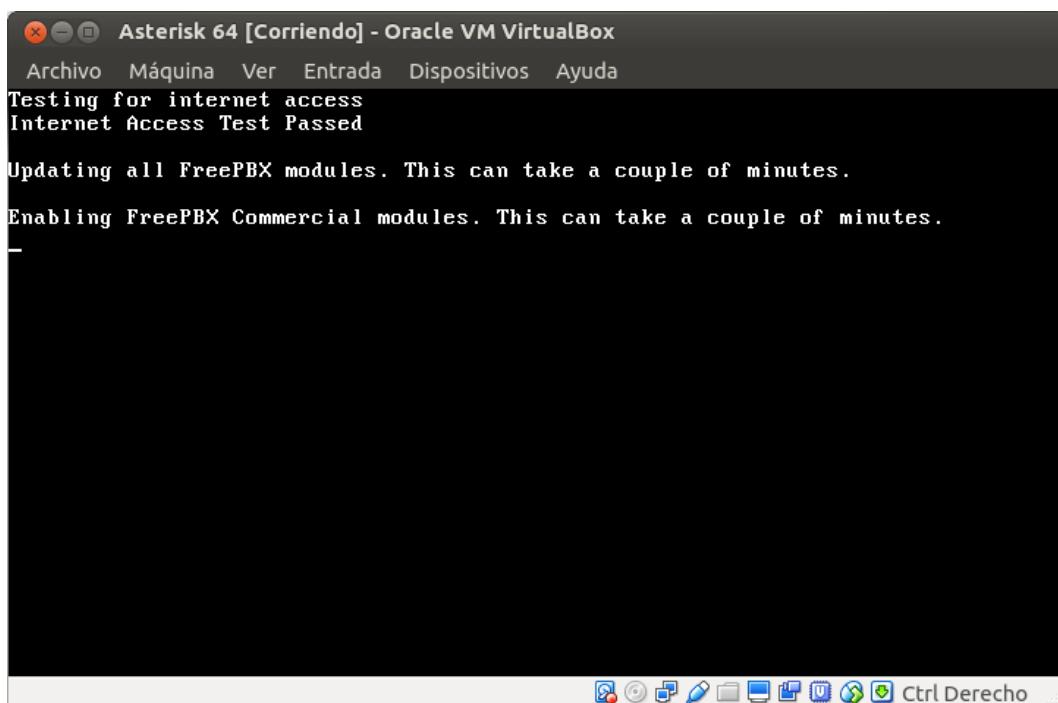


Instalación de paquetes



```
Asterisk 64 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Testing for internet access
Internet Access Test Passed
```

[Testeo internet](#)

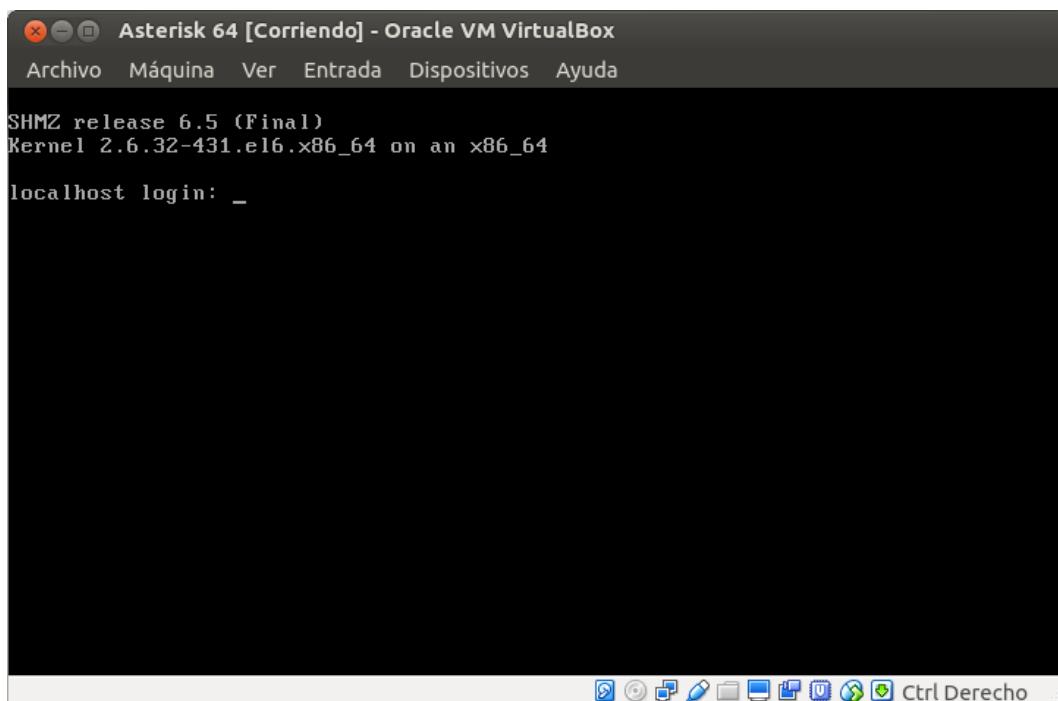


```
Asterisk 64 [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
Testing for internet access
Internet Access Test Passed

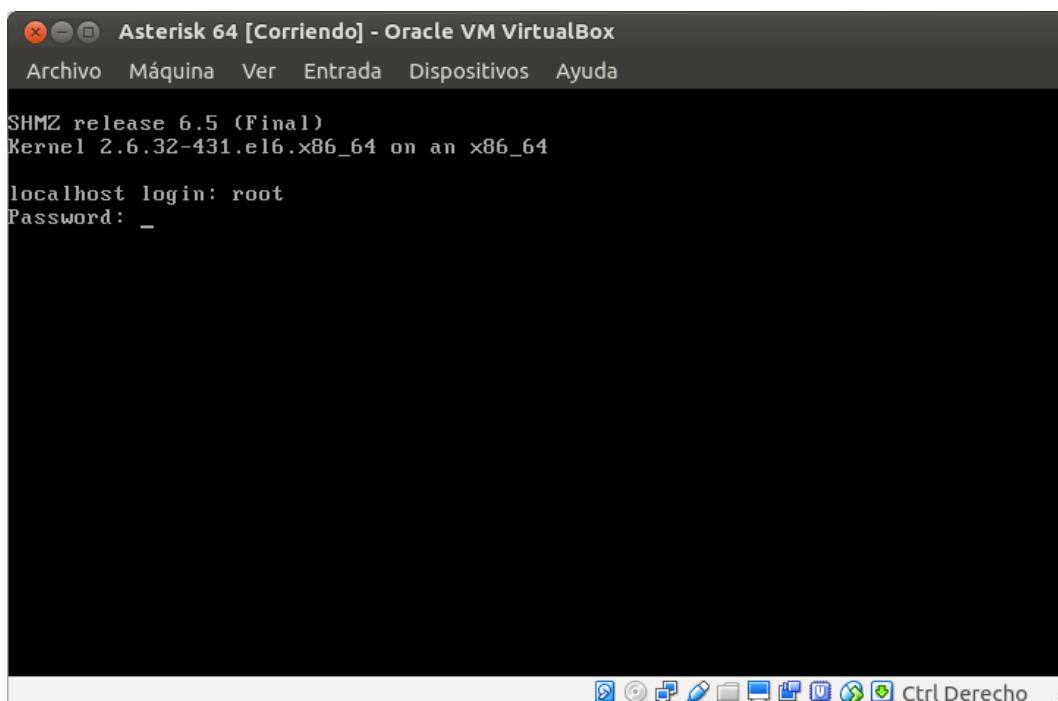
Updating all FreePBX modules. This can take a couple of minutes.

Enabling FreePBX Commercial modules. This can take a couple of minutes.
```

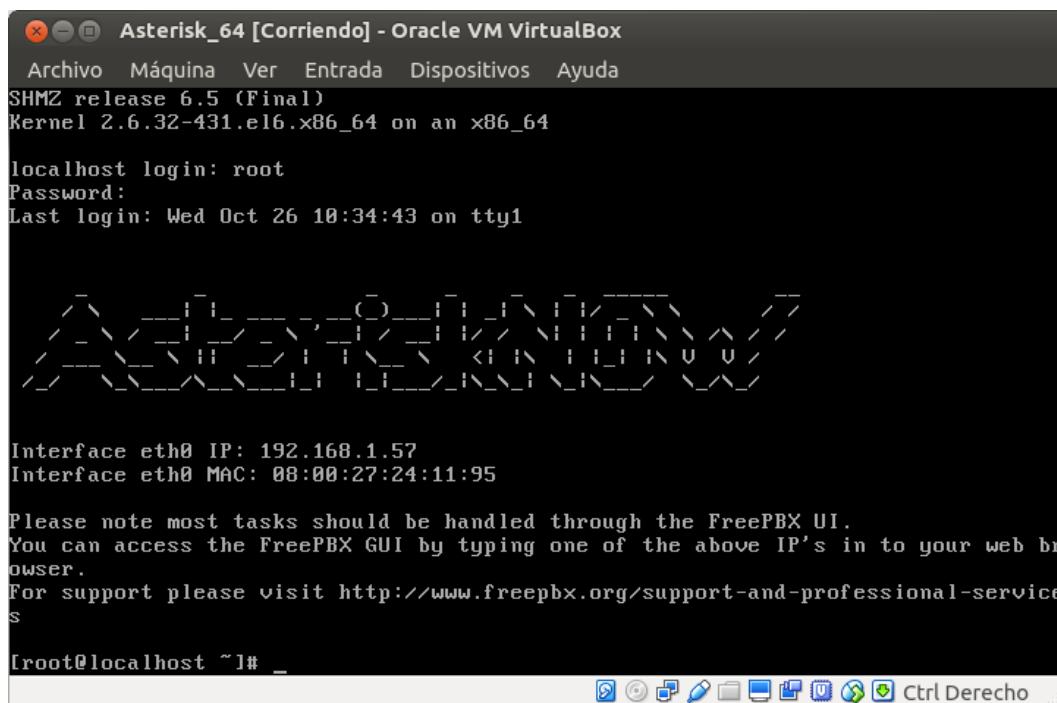
[Actualización y activación de módulos](#)



[Login](#)



[Login + password](#)



Acceso al programa y pantalla de inicio en terminal

Instalación Asterisk

1. Una vez cargado el archivo, procederá la instalación de AsteriskNOW. Elegiremos la opción primera **FreePBX 6.12.65 with Asterisk 11** que es actualmente la versión estable.
2. Durante la instalación irán apareciendo pantallas de configuración. La primera de ellas es la configuración TCP/IP. En este apartado, elegiremos la **opción dinámica del IPv4** y la **opción automática del IPv6**.
3. En este punto, se nos activa la pantalla de elección de huso horario. Buscaremos la **opción Europe/Madrid** y pulsaremos **OK**.
4. Aparecerá la pantalla de configuración de **password** para el usuario **root**.
5. A partir de este punto comenzará la instalación. La cual, durará alrededor de unos 20 minutos.
6. Cuando la barra de instalación llegue al final, desaparecerá la pantalla azul y aparecerá la pantalla en negro. En la que nos indicará, que se está haciendo un testeo al acceso a internet.
7. Al concluir el anterior testeo, el programa FreePBX se actualizará y cuando finalice, se configurarán los módulos comerciales. Esta configuración tomará cierto tiempo, y con este paso concluirá la instalación.
8. En este punto nos pedirá el usuario que se va a autenticar. Que en este caso será **root** y pulsaremos **intro** para confirmar.

9. Seguidamente nos pedirá el **password** que en uno de los pasos anteriores hemos introducido durante la instalación.
10. Habiendo solventado el paso anterior, nos aparecerá la siguiente pantalla, en la que observaremos un número determinado de información. Debajo de la palabra password, aparecerá la última conexión en fecha y hora. Más abajo, veremos un dato importante que usaremos para acceder a FreePBX gui. Accederemos de la siguiente manera. Pondremos en nuestro navegador la ip que se muestra en pantalla.

Welcome to FreePBX Administration!

Please provide the core credentials that will be used to administer your system

Initial setup

Username: admin user name
Password: Admin password
Confirm Password: Admin password
Admin Email address: Email Address

[Create Account](#)



Administrar usuario y contraseña en Asterisk.

FreePBX Administration

User Control Panel

Operator Panel

Get Support



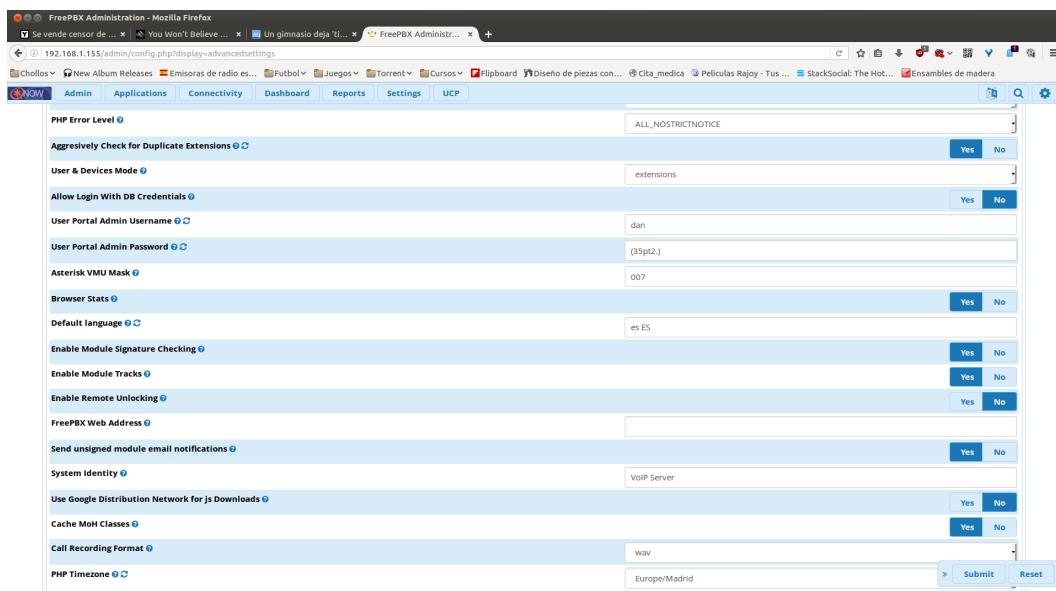
Configuración Asterisk

The screenshot shows the FreePBX Administration interface. On the left, there's a sidebar with a summary of system components (Asterisk, MySQL, Web Server, Fail2Ban, System Registration, Prosody (XMPP), XMPP Presence) and a list of critical errors found. The main dashboard shows IPBX Statistics, Uptime (26 minutes, 12 seconds ago), and Load Averages (0.04, 0.07, 0.08 for 1 Minute, 5 Minutes, and 15 Minutes respectively). There are also sections for Asterisk IAX Settings, Asterisk Logfile Settings, Asterisk Manager User, Asterisk SIP Settings, EndPoint Manager, Fax Configuration, High Availability, Music on Hold, PIN Sets, Route Congestion Messages, Text To Speech Engines, and Voicemail Admin.

Instalación Asterisk 3/3

This screenshot shows the SIP Channel Driver settings page in the FreePBX Administration interface. It includes fields for chan_sip, Use bad-number Context, Use Google DNS for ENUM, Waiting Period to Stop Asterisk (set to 120), Asterisk Dial Options (Ttr), Asterisk Outbound Trunk Dial Options, Attended Transfer Alert Info, Blind Transfer Alert Info, Country Indication Tones (Spain), Disallow transfer features for inbound callers, Display CallerID on Calling Phone, Display Dialed Number on Calling Phone, Display Presence State of Callee, Internal Alert Info (inherit), Ringtime Default (15), Speaking Clock Time Format (24 Hour Format), and a Follow Me Module section with a 'Create Follow Me at Extension Creation Time' checkbox. At the bottom right are 'Submit' and 'Reset' buttons.

Configuración Asterisk



Instalación Asterisk 3/3

Configuración Asterisk

1. Introduciremos un **usuario** y una **contraseña**, además de **confirmar la contraseña**, introduciémos un **email** de administración.
2. Nos aparecerá la pantalla de gestión de asterisk. Pulsaremos la opción **FreePBX Administration**. Nos pedirá el usuario y la contraseña que hemos introducido anteriormente, que nos dará acceso al entorno de gestión de FreePBX. Pulsaremos **Settings**, abriéndose un desplegable, en el cual elegiremos la primera opción **Advanced Settings**.
3. Se abrirá una nueva pantalla, en la cual se tendrá que modificar algunos puntos para adaptarlo a nuestras necesidades. Primeramente cambiaremos la opción **Country Indication Tones** en la que pondremos **Spain**. Seguidamente, cambiaremos la opción **Speaking Clock Time Format** que está justamente más abajo y la pondremos **24 hour format**
4. En la misma pantalla nos dirigiremos más hacia abajo seguiremos modificando algunas opciones. Modificaremos la opción **Default language**, en la que escribiremos **es ES**. También modificaremos la opción **PHP Timezone**, la situaremos en **Europe/Madrid**.

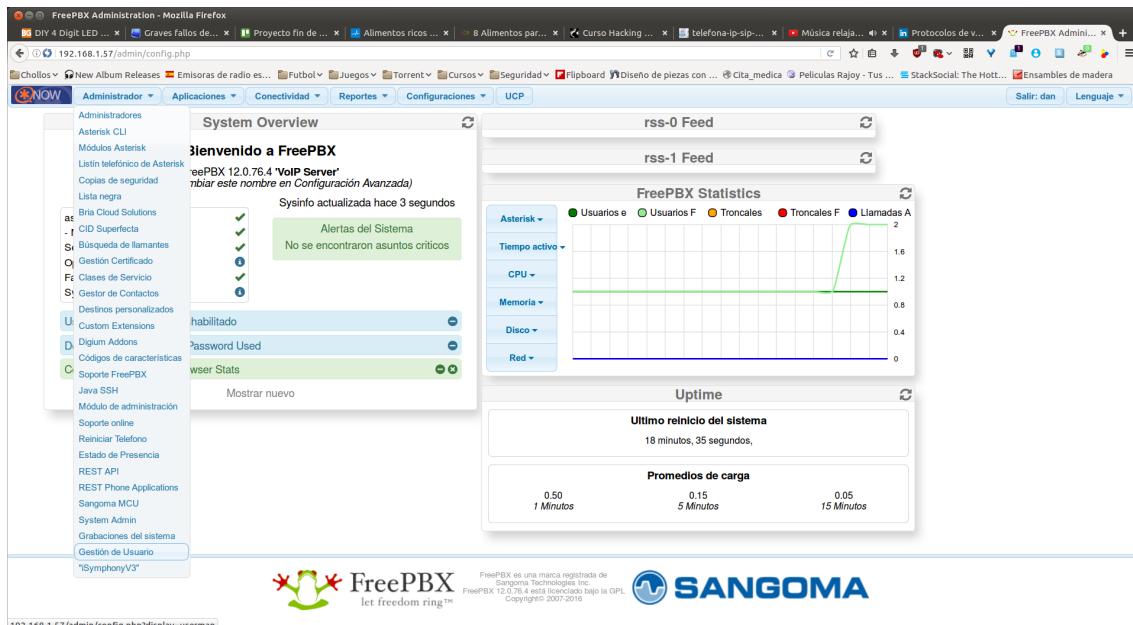
Creación de usuarios

Tenemos la oportunidad de crear usuarios de dos maneras.

- Desde la pantalla de gestión del FreePBX (**GUI**).
- Desde el **terminal**.

Estas dos maneras de configurar usuarios se explicarán a continuación:

Creación de usuarios desde GUI



Pantalla principal FreePBX

Gestor de Usuario FreePBX

Agregar Usuario FreePBX

1. En la pantalla principal, nos dirigiremos a la pestaña de "**Administrador**". Al desplegarse esta opción, buscaremos y pulsaremos la opción de "**Gestión de usuario**".
2. Nos abrirá el Gestor de Usuario. Nos dirigiremos a la derecha de la pantalla, en el que observaremos la opción "**Agregar nuevo usuario**", que se encuentra justamente debajo de "**Lista de Usuarios**"
3. Llegamos a la pantalla de "**Agregar Usuario**" e introduciremos una configuración simple.

- **Nombre de Inicio de Sesión:** 5201 <*Número de Usuario*>

- **Contraseña:** d5201s <contraseña>
- **Nombre a Mostrar:** Daniel Sanchez <Nombre a mostrar>

Para finalizar, nos dirigiremos al final, arrastrando el scroll hacia abajo y pulsaremos el botón de "**Guardar**".

Creación de usuarios desde Terminal

Crearemos 5 usuarios. Les daremos la siguiente numeración y contraseña a cada uno.

- Usuario: "5201" Contraseña "d5201s".
- Usuario: "5202" Contraseña "m5202r".
- Usuario: "5203" Contraseña "f5203p".
- Usuario: "5204" Contraseña "p5204v".
- Usuario: "5205" Contraseña "i5205r".

Se explicará los pasos a tomar para la creación de los usuarios y la configuración de las extensiones.

Habiendo introducido el localhost login; "*root*" y el Password; "*password introducido en la configuración inicial*". introduciremos los siguientes comando en el terminal:

1. "**service asterisk start**", que nos devolverá una mensaje "*Asterisk is already running*".
2. "**asterisk -r**", accederemos a la configuración por terminal.
3. "**sip show peers**", para visualizar los usuarios SIP que están registrados en el programa, que en nuestro caso no habrá ninguno.
4. "**core stop now**", detendremos la aplicación Asterisk.
5. Nos dirigiremos a la carpeta donde se configuran los usuarios en Asterisk. "**cd etc/asterisk/**". Pulsaremos **intro** y haremos que muestre todos los archivos con el comando "**ls**".
6. Accederemos al archivo "**sip.conf**", lo haremos introduciendo el comando "**nano sip.conf**".
7. Buscaremos la sección "**[general]**", donde se llevará a cabo la configuración de los usuarios.

Los apartados de esta sección se explicarán a continuación:

- **udpbindaddr:** Especifica la IP y el puerto por el cual se escucharán las peticiones de entrada. En este caso, será 0.0.0.0 (para que se escuchen todas las peticiones) y se usará el puerto por defecto 5060.
- **context.** Está ligado directamente al dialplan⁵ Se define por defecto para todas las conexiones SIP.
- **language:** Parámetro de lenguaje. Lo pondremos en español.
- **disallow:** Deniega una acción, IP o codec.
- **allow:** Permite una acción, IP o codec.
- **videosupport:** Parámetro de soporte de video.

⁵dialplan: Llamadas entrantes y llamadas salientes

- **maxcallbitrate**: Ancho de banda máximo para llamadas.
- **tos** (Type of service): Dispone de 6 bits en el protocolo IPV4 para indicar al router que paquetes tienen prioridad de paso. Forma parte de la capa 3 del modelo OSI.

Nombre DSCP	Prioridad IP
CS0	0 - Mejor esfuerzo
CS1, AF11 - 13	1 - Prioritario
CS2, AF21 - 23	2 - Inmediato
CS3, AF31 - 33	3 - Relámpago (Señal de voz)
CS4, AF41 - 43	4 - Relámpago sobrecargado (Comunicación de video RTP)
CS5, EF	5 - Crítico (Comunicación de voz RTP)
CS6	Internet
CS7	Red

Quality of service: Tos

- **Cos**: Dispone de 3 bits (PCP) en el protocolo VLAN 802.1Q para indicar que paquetes tienen prioridad de paso. Forma parte de la capa 2 del modelo OSI.

PCP	Prioridad	Tipos de tráfico
1	0 (Más bajo)	En segundo plano
0	1	Mejor esfuerzo
2	2	Excelente esfuerzo
3	3	Aplicaciones críticas
4	4	Video
5	5	Voz
6	6	Internet
7	7	Red

Quality of service: Cos

```
[general]
udpbindaddr=0.0.0.0:5060
context=default
language=es
disallow=all

; codec de audio (Llamadas)
allow=alaw
allow=ulaw
allow=gsm
allow=g722
allow=ilbc
allow=speex
allow=speex16
allow=speex32

; codec de video (Videollamadas)
allow=h264
allow=h263p
allow=h263
```

```

videosupport=yes
maxcallbitrate=512

;QOS sip, audio y video
tos_sip=cs3
tos_audio=ef
tos_video=af41
cos_sip=3
cos_audio=5
cos_video=4

```

8. Nos dirigiremos al final del archivo e introduciremos los usuarios que anteriormente hemos indicado.

Vamos a explicar primeramente que significa cada punto.

Englobaremos los patrones comunes, los usaremos en cada usuario. De esta manera ahorraremos tiempo a la hora de configuración.

- **type:** Se refiere al tipo de cliente. Puede ser de 3 tipos:
 - **peer.** Autentica llamadas y envía llamadas (solo salientes).
 - **user.** Autentica llamadas y recibe llamadas (solo entrantes).
 - **friend.** Autentica llamadas, envía y recibe llamadas (entrantes y salientes).
- **dtmfmode**⁶: Señalización (tonos de llamadas) de los sistemas de voz (rfc2833/info). Asignaremos "rfc2833".
- **host:** Indica la dirección IP o el nombre del host del cliente. Si es dinámico por DHCP, se pondrá *dynamic*.
- **context:** Dialplan que se asocia al usuario. En este caso se asociarán al apartado de "extensiones-internas".
- **canreinvite:** Este parámetro especifica si se fuerza que el streaming de audio pase por Asterisk. Asignándole "no" se obligará a que pase por Asterisk.
- **qualify:** Monitorización del estado de la extensión.
- **username:** Nombre del usuario.
- **secret:** Contraseña para autenticarse en el sistema.
- **callerid:** Nombre del usuario y su extensión.
- **mailbox:** Buzón de voz de la extensión.

```

[comunes] (!)
type=friend
dtmfmode=rfc2833
host=dynamic
context=extensiones-internas
camreinvite=no
qualify=yes

```

⁶DTMF: Multifrecuencia de doble tono

```
[5201] (comunes)

username=5201
secret=d5201s
callerid="Daniel Sanchez" <5201>
mailbox=5201@default
_____

[5202] (comunes)

username=5202
secret=m5202r
callerid="Maria Ramirez" <5202>
mailbox=5202@default
_____

[5203] (comunes)

username=5203
secret=f5203p
callerid="Fernando Perez" <5203>
mailbox=5203@default
_____

[5204] (comunes)

username=5204
secret=p5204v
callerid="Pablo Valdivia" <5204>
mailbox=5204@default
_____

[5205] (comunes)

username=5205
secret=i5205r
callerid="Irene Ramirez" <5205>
mailbox=5205@default
```

Cerramos el archivo pulsando "**control + x**", nos indicará que salvaremos el archivo pulsando "**y**", o por el contrario pulsando "**n**" destruiremos los cambios. Pulsaremos "**y**" y seguidamente le daremos a la tecla "**intro**" para indicar que vamos a sobrescribir el archivo.

9. Tecleamos en el terminal "**nano extensions.conf**" para configurar las extensiones.
10. Buscaremos el contexto "[default]", e introduciremos lo siguiente:

```
[default]

exten => _X,1,Hangup(21) "Recibe todos los context".
exten => s,1,Hangup(21) "Rechaza todo por seguridad".
```

Este contexto [default] se usará cuando el usuario no este definido en ningún otro contexto. Esta configuración se realiza por seguridad y colgará todas las llamadas bajo este contexto evitando que un usuario sin identificar pueda hacer llamadas.

También se creará una cabecera llamada "[extensiones-internas]", la que especificamos en el archivo sip.conf en los apartados de configuración de usuarios "context=extensiones-internas".

exten => patrón, índice, acción(parámetro)

- **exten.** Es una palabra reservada que especifica las líneas de dialplan.
- **patrón.** Todas las extensiones de 4 cifras que comiencen por 52 y le sigan 2 números que están comprendidas entre el 0 y el 9.
- **índice.** Orden secuencial de las acciones a tomar en la extensión.
- **acción(parámetros).** Acción que se debe ejecutar.
- **_52XX:** Extensión es de 4 dígitos que comiencen por 52.
- **1:** Numeración de la acción del dialplan.
- **Dial(SIP/EXTEN) :** Llamada a la extensión que se indique. *Dial()*, envío de la llamada. SIP, es el protocolo número de la extensión.
- **Hangup (16):** Normal call clearing. Cuelga la llamada si no se puede realizar la comunicación (no disponible, no descuelga, rechaza la llamada).

```
[extensiones-internas]

exten => _52XX,1,NoOp(Llamadas a terminal interno)
exten => _52XX,2,Dial(SIP/${EXTEN})
exten => _52XX,3,Hangup(16)
```

Al finalizar pulsaremos de nuevo "**control + x**" e "**intro**" para salvar los datos que se han modificado en el archivo.

11. Tecleamos en el terminal "**nano iax.conf**" para configurar la calidad de servicio en este archivo.

```
tos=ef
```

12. Escribiremos en el terminal "**service asterisk start**", pulsaremos "**intro**", y seguidamente, introduciremos el comando "**asterisk -r**". De esta manera, podremos ver que los datos introducidos de los clientes están en todos funcionales.
13. Actualizaremos los archivos "sip.conf" y "extensions.conf", a través de los comandos "**dialplan reload**", seguidamente pulsaremos "**intro**". Y para finalizar, introduciremos el comando "**sip reload**" y de nuevo pulsaremos "**intro**".

De esta manera, quedarán configurados los clientes.

Opciones de configuración dependiendo si es user, peer o friend

User	Peer	Explicación y opciones
context	context	Contexto relacionado con el dialplan (usuario o peer).
permit	permit	Permite una IP.
deny	deny	Deniega una IP.
secret	secret	Contraseña del registro de los terminales.
md5secret	md5secret	Encripta la contraseña en formato md5.
dtmfmode	dtmfmode	Modo de transmisión de tonos (RFC2833 o INFO).
canreinvite	canreinvite	Permitir o denegar el intercambio de mensajes RTP entre terminales.
callgroup	callgroup	Grupo de contactos.
pickupgroup	pickupgroup	Grupo de contactos para la opción pickup()
language	language	Señales para un país (indications.conf).
allow	allow	Habilita codec (audio, video y texto).
disallow	disallow	Deshabilita codec.
insecure	insecure	Debe autenticarse siempre.
callerid		Identificador del contacto.
accountcode		Vinculación con temas de facturación.
incominglimit		Límite de llamadas salientes simultáneas para un contacto.
restrictcid	mailbox	Ocultar el ID del llamante.
	username	Extensión para el contestador.
	fromdomain	Nombre del contacto del llamante.
	fromuser	Sitúa el campo From: en los mensajes SIP.
		Sitúa nombre del contacto en el From:.

User	Peer	Explicación y opciones
	host	dirección física u host del dispositivo remoto.
	port	Puerto UDP de Asterisk.
	qualify	Determina la situación del dispositivo para ser alcanzado.
	defaultip	IP por defecto del cliente host:.
	rtptimeout	Finaliza la llamada recibiendo un mensaje de timeout.
	rtpholdtimeout	Finaliza la llamada recibiendo un mensaje de timeout. Si no existe tráfico rtp (on hold).

Opciones dialplan

⁷<http://www.voipforo.com/asterisk/configuracion-sip-conf.php>

Anexo III: Router Neutro (firmware dd-wrt)

Firmware: DD-WRT v24-sp2 (03/25/13) std
Time: 00:01:20 up 1 min, load average: 0.22, 0.09, 0.03
WAN: Disabled

Setup		Wireless	Services	Security	Access Restrictions	NAT / QoS	Administration	Status
Basic Setup	DDNS	MAC Address Clone	Advanced Routing	Networking	EoIP Tunnel			
WAN Setup								
WAN Connection Type Connection Type: <input type="button" value="Disabled"/>				Automatic Configuration - DHCP: This setting is most commonly used by cable operators.				
Optional Settings Router Name: DD-WRT Hostname: <input type="text"/> Domain Name: <input type="text"/> MTU: <input type="button" value="Auto"/> <input type="button" value="1500"/> STP: <input type="radio"/> Enable <input checked="" type="radio"/> Disable				Hostname: Enter the hostname provided by your ISP.				
				Domain Name: Enter the domain name provided by your ISP.				
				Local IP Address: This is the LAN-side IP address of the router.				
				Subnet Mask: This is the subnet mask of the router.				
Network Setup								
Router IP Local IP Address: 192.168.10.62 Subnet Mask: 255.255.255.0 Gateway: 192.168.10.33 Local DNS: 0.0.0.0				DHCP Server: Allows the router to manage your IP addresses.				
				Start IP Address: The address you would like to start with.				
				Maximum DHCP Users: You may limit the number of addresses your router hands out. 0 means only predefined static leases will be handed out.				
				Maximum DHCP Users: You may limit the number of addresses your router hands out. 0 means only predefined static leases will be handed out.				
WAN Port								
Assign WAN Port to Switch : <input type="checkbox"/>				Time Settings: Choose the time you are in a summer time (DST) period. The router can use local time or UTC time.				
WAN Port								
Assign WAN Port to Switch : <input type="checkbox"/>								
Network Address Server Settings (DHCP)								
DHCP Type : <input type="button" value="DHCP Server"/>				Time Settings: Choose the time you are in a summer time (DST) period. The router can use local time or UTC time.				
DHCP Server : <input type="radio"/> Enable <input checked="" type="radio"/> Disable								
Start IP Address : 192.168.10.34								
Maximum DHCP Users : 20								
Client Lease Time : 1440 min								
Static DNS 1 : 0.0.0.0								
Static DNS 2 : 0.0.0.0								
Static DNS 3 : 0.0.0.0								
WINS : 105								
Use DNSMasq for DHCP : <input type="checkbox"/>								
Use DNSMasq for DNS : <input type="checkbox"/>								
DHCP-Authoritative : <input type="checkbox"/>								

Firmware: DD-WRT v24-sp2 (03/25/13) std
Time: 00:03:23 up 3 min, load average: 0.07, 0.07, 0.04
WAN IP: 0.0.0.0

Wireless Physical Interface ath0 [2.4 GHz]

Physical Interface ath0 - SSID [PFC-DSB] HWAddr [E8:94:F6:D4:45:45]

Wireless Mode: AP
Wireless Network Mode: Mixed
Channel Width: Full (20 MHz)
Wireless Channel: Auto
Wireless Network Name (SSID): PFC-DSB
Wireless SSID Broadcast: Enable Disable
Advanced Settings:

Wireless Network Mode:
If you wish to exclude Wireless-G clients, choose *B-Only* mode. If you would like to disable wireless access, choose *Disable*.
Note : when changing wireless mode, some advanced parameters are susceptible to be modified ("Afterburner", "Basic Rate" or "Frame Burst").

Sensitivity Range:
Adjusts the ACK timing. 0 disables ack timing completely for Broadcom firmwares. On Atheros based firmware, 0 enables auto ACK timing mode.

Configuración punto de acceso wireless

Firmware: DD-WRT v24-sp2 (03/25/13) std
Time: 00:03:52 up 4 min, load average: 0.04, 0.07, 0.04
WAN IP: 0.0.0.0

Wireless Security ath0

Physical Interface ath0 SSID [PFC-DSB] HWAddr [E8:94:F6:D4:45:45]

Security Mode: WPA2 Personal
WPA Algorithms: AES
WPA Shared Key: Alphanumeronic(35pl2.) Unmask
Key Renewal Interval (in seconds): 3600 (Default: 3600, Range: 1 - 99999)

Security Mode:
You may choose from Disable, WEP, WPA Personal, WPA Enterprise, or RADIUS. All devices on your network must use the same security mode. With N-Mode you must use WPA2/AES.

Configuración de seguridad wireless

Firmware: DD-WRT v24-sp2 (03/25/13) std
Time: 00:07:43 up 7 min, load average: 0.02, 0.06, 0.04
WAN IP: 0.0.0.0

Firewall **VPN Passthrough**

Security

Firewall Protection:
Enable or disable the SPI firewall.

SPI Firewall: Enable Disable

Configuración firewall wireless

Firmware: DD-WRT v24-sp2 (03/25/13) std
Time: 00:14:56 up 15 min, load average: 0.25, 0.11, 0.07
WAN IP: 0.0.0.0

NAT / QoS

Quality Of Service (QoS)

QoS Settings

Start QoS	<input checked="" type="radio"/> Enable <input type="radio"/> Disable
Port	LAN & WLAN
Packet Scheduler	HTB
Queueing Discipline	SFQ
Uplink (kbps)	0
Downlink (kbps)	0

TCP-Packet Priority

Prioritize small TCP-packets with the following flags:

<input checked="" type="checkbox"/> ACK	<input checked="" type="checkbox"/> SYN	<input checked="" type="checkbox"/> FIN	<input type="checkbox"/> RST
---	---	---	------------------------------

Services Priority

Delete	Service Name	Priority
<input type="checkbox"/>	sip	Premium
<input type="checkbox"/>	rtp	Premium

Add

Help more...

QoS Settings

Uplink:
Set this to 80%-95% (max) of your total upload limit.

Downlink:
Set this to 80%-100% (max) of your total download limit.

Services Priority:
You may control your data rate with respect to the application that is consuming bandwidth.

Netmask Priority:
You may specify priority for all traffic from a given IP address or IP range.

MAC Priority:
You may specify priority for all traffic from a device on your network by giving the device a device name, specifying priority and entering its MAC address.

Default Bandwidth Level
Enable Per User Default Limits:
Enable the default level per user or set the level for all users.

Configuración Quality of service wireless

Configuraremos el router de la siguiente manera:

1. Setup -> Basic Setup

- WAN Setup -> Connection type.
 - *Conection type* -> Disable
- WAN Setup -> Optional Setting.
 - *STP* -> Disable
- Network Setup -> Router IP (DHCP).
 - *Local IP Address* -> 192.168.10.62
 - *Subnet mask* -> 255.255.255.224
 - *Gateway* -> 192.168.10.33
- Network Setup -> Network Address Server Settings (DHCP).
 - *DHCP Server* -> Disable

2. Wireless -> Basic Setup

- Wireless Physical Interface ath0 (2.4 GHz).
 - *Wireless mode* -> Ap
 - *Wireless Network Mode* -> Mixed
 - *Channel Width* -> Full (20 MHz)
 - *Wireless Channel* -> Auto
 - *Wireless Network Name (SSID)* -> "NOMBRE DE LA RED"
 - *Wireless SSID Broadcast* -> Enable

3. Wireless -> Wireless Security

- Wireless Security ath0.
 - *Security mode* -> WPA2 Personal
 - *WPA Algorithm* -> AES
 - *WPA Shared Key* -> "PASSWORD"

4. Security -> Firewall

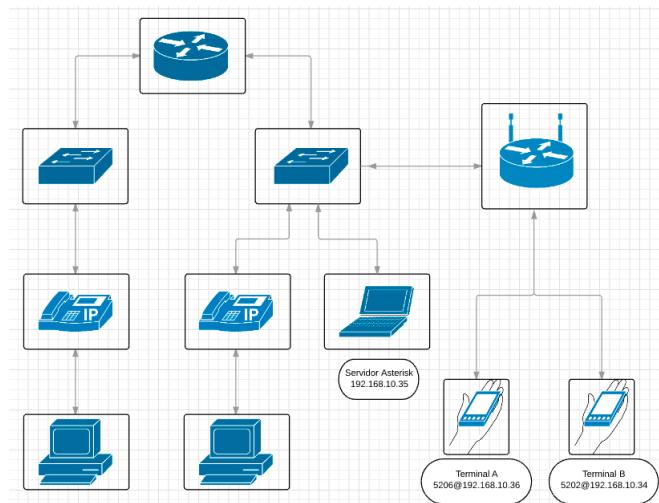
- Security -> Firewall Protection.
 - *SPI Firewall* -> Disable

5. NAT/QoS -> QoS

- Quality of Service (QoS) -> QoS Settings.
 - *Start QoS* -> Enable
 - *Port* -> LAN/WAN
- Quality of Service (QoS) -> TCP-Packet Priority.

- *Sip -> Premium*
- *rtp -> Premium*

Anexo IV: Monitorización protocolo SIP Wireshark



Red a monitorizar

Monitorización codec GSM

Observamos al monitorizar la realización de una llamada con el protocolo SIP + codec GSM que se activan diferentes eventos. Los explicaremos seguidamente:

18545 823.00382.. 192.168.10.36	192.168.10.35	SIP/SDP	858 Request: INVITE sip:5202@192.168.10.35
18546 823.00427.. 192.168.10.35	192.168.10.36	SIP	555 Status: 401 Unauthorized
18547 823.00428.. 192.168.10.35	192.168.10.36	SIP	555 Status: 401 Unauthorized
18548 823.10973.. 192.168.10.36	192.168.10.35	SIP	412 Request: ACK sip:5202@192.168.10.35
18549 823.11132.. 192.168.10.36	192.168.10.35	SIP/SDP	1025 Request: INVITE sip:5202@192.168.10.35
18550 823.11196.. 192.168.10.35	192.168.10.36	SIP	498 Status: 100 Trying
18551 823.11196.. 192.168.10.35	192.168.10.36	SIP	498 Status: 100 Trying
18552 823.11304.. 192.168.10.35	192.168.10.34	SIP/SDP	1483 Request: INVITE sip:5202@192.168.10.34:45910;transport=udp
18553 823.11304.. 192.168.10.35	192.168.10.34	SIP/SDP	1483 Request: INVITE sip:5202@192.168.10.34:45910;transport=udp
18554 823.33736.. 192.168.10.34	192.168.10.35	SIP	309 Status: 100 Trying

Establecimiento de la llamada

INVITE

```
▼ Session Initiation Protocol (INVITE)
  ▼ Request-Line: INVITE sip:5202@192.168.10.35 SIP/2.0
    Method: INVITE
  ▼ Request-URI: sip:5202@192.168.10.35
    Request-URI User Part: 5202
    Request-URI Host Part: 192.168.10.35
    [Resent Packet: False]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.f1W2EL0Ve;rport
      Transport: UDP
      Sent-by Address: 192.168.10.36
      Sent-by port: 54524
      Branch: z9hG4bK.f1W2EL0Ve
      RPort: rport
    ▼ From: <sip:5206@192.168.10.35>;tag=146-avb06
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: 146-avb06
    ▼ To: sip:5202@192.168.10.35
      ▼ SIP to address: sip:5202@192.168.10.35
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.35
    ▼ CSeq: 20 INVITE
      Sequence Number: 20
      Method: INVITE
    Call-ID: 1mWGQqm10
    Max-Forwards: 70
    Supported: replaces, outbound
    Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO, UPDATE
    Content-Type: application/sdp
    Content-Length: 230
  ▼ Contact: <sip:5206@192.168.10.36:54524;transport=udp>;+sip.instance=<urn:uuid:24db554b-01f9-4900-aa1b-5594643cd600>
    ▼ Contact URI: sip:5206@192.168.10.36:54524;transport=udp
      Contact URI User Part: 5206
      Contact URI Host Part: 192.168.10.36
      Contact URI Host Port: 54524
      Contact URI parameter: transport=udp
      Contact parameter: +sip.instance=<urn:uuid:24db554b-01f9-4900-aa1b-5594643cd600>\r\n
    User-Agent: LinphoneAndroid/3.2.4 (belle-sip/1.5.0)
  ▼ Message Body
    ▼ Session Description Protocol
      Session Description Protocol Version (v): 0
      ▼ Owner/Creator, Session Id (o): 5206 2621 3506 IN IP4 192.168.10.36
        Owner Username: 5206
        Session ID: 2621
        Session Version: 3506
        Owner Network Type: IN
        Owner Address Type: IP4
        Owner Address: 192.168.10.36
        Session Name (s): Talk
      ▼ Connection Information (c): IN IP4 192.168.10.36
        Connection Network Type: IN
        Connection Address Type: IP4
        Connection Address: 192.168.10.36
      ▼ Bandwidth Information (b): AS:500
        Bandwidth Modifier: AS [Application Specific (RTP session bandwidth)]
        Bandwidth Value: 500 kb/s
      ▼ Time Description, active time (t): 0 0
        Session Start Time: 0
        Session Stop Time: 0
      ▼ Session Attribute (a): rtcp-xr:rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-metrics
        Session Attribute Fieldname: rtcp-xr
        Session Attribute Value: rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-metrics
      ▼ Media Description, name and address (m): audio 7076 RTP/AVP 3 101
        Media Type: audio
        Media Port: 7076
        Media Protocol: RTP/AVP
        Media Format: GSM 06.10
        Media Format: DynamicRTP-Type-101
      ▼ Media Attribute (a): rtpmap:101 telephone-event/8000
        Media Attribute Fieldname: rtpmap
        Media Format: 101
        MIME Type: telephone-event
        Sample Rate: 8000
```

INVITE

El frame **18545** realiza la comunicación con el protocolo **IPv4**⁸. La petición la realiza el terminal A (**Source: 192.168.10.36**) y es dirigido al Servidor Asterisk (**Destination: 192.168.10.35**). Los paquetes tienen prioridad de envío (DSCP: AF31), no hay fragmentación del paquete. Este paquete se envía a partir del protocolo UDP. En la comunicación el terminal A utiliza el puerto 56526 y es recepcionado por el servidor en el puerto 5060 (puerto predefinido).

En el apartado **Session Initiation Protocol** cabe destacar los sub-apartados Request-Line, que especifica el método de comunicación del siguiente paquete es una solicitud de comunicación (INVITE), el tipo de comunicación (SIP 2.0), el contacto (5202) y la IP del Servidor Asterisk (192.168.10.35).

Referente al **Request URI**⁹ (Petición de identificación de usuario) que se realiza hacia la base de datos de contactos que está contenida en el servidor Asterisk. En este caso, se busca en la base de datos (Servidor Asterisk: 192.168.10.35) el usuario 5202.

En el apartado **Message Header**, se especifican varios puntos, de los cuales cabe destacar los siguientes apartados. Los cuales enumeraré y detallaré a continuación:

- **Via.** En el se detallan la comunicación a usar (SIP versión 2.0)el tipo de transporte (udp), la dirección IP (192.168.10.36) desde donde se envía el paquete, el puerto (54524) que se usa para el envío.
- **From.** Aparece el número del usuario (5206) (usuario que realiza la llamada), la dirección IP del servidor Asterisk (192.168.10.35).
- **To.** Aparece el número del usuario (5202) (usuario al que llaman), la dirección IP del servidor Asterisk (192.168.10.35).
- **CSeq.**
- **Contact.** Los datos más importantes a tener en cuenta en este apartado, son los datos del usuario que realiza la llamada. Vemos que está dividido en pequeñas partes:
 - Contact URI User Part: Número del contacto que realiza la llamada (5206).
 - Contact URI Host Part: IP del dispositivo que realiza la llamada (192.168.10.36).
 - Contact URI Host Port: Puerto por el cual se realiza la petición SIP (54524).
 - Contact URI parameter: Tipo de trasnporte utilizado (udp).
 - User-Agent: Tipo de programa que controla la comunicación (LinphoneAdroid/3.2.4).
- **Owner/Creator.**
 - Owner Username: Usuario que realiza la llamada (5206).
 - Session ID: Identificación de la sesión (2621).
 - Session Version: Versión de la sesión (3506)
 - Owner Address Type: Tipo de protocolo que se usa en la comunicación (IPv4)
 - Owner Address: Dirección IP que origina la comuncación (192.168.10.36).
- **Connection Information.**

⁸IPv4: Internet protocol Version 4

⁹URI: Uniform Resource Identifiers

- Connection Address Type: Tipo de protocolo que se usa en la comunicación (IPv4).
- Connection Address: Dirección IP desde donde se origina la comunicación (192.168.10.36).

- **Bandwidth Information.**

- Bandwidth Modifier: Adecuar el ancho de banda a las necesidades del protocolo a usar (RTP).
- Bandwidth Value: Ancho de banda (500 kb/s).

- **Media Description.**

- Media Type: Tipo de comunicación multimedia (audio).
- Media Port: Puerto usado para la comunicación multimedia (7076).
- Media Protocol: Protocolo usado para realizar la comunicación (RTP).
- Media Format: Codec usado para la comunicación multimedia (GSM).

- **Media Attribute.**

- MiME Type: Dispositivo que realiza la comunicación (telephone-event).
- Sample Rate: Ratio de codec que se usa en la comunicación (8000).

STATUS

```

▼ Session Initiation Protocol (401)
  ▼ Status-Line: SIP/2.0 401 Unauthorized
    Status-Code: 401
    [Resent Packet: False]
    [Request Frame: 18545]
    [Response Time (ms): 0]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.f1W2EL0Ve;received=192.168.10.36;rport=54524
      Transport: UDP
      Sent-by Address: 192.168.10.36
      Sent-by port: 54524
      Branch: z9hG4bK.f1W2EL0Ve
      Received: 192.168.10.36
      RPort: 54524
    ▼ From: <sip:5206@192.168.10.35>;tag=14G-avb06
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: 14G-avb06
    ▼ To: sip:5202@192.168.10.35;tag=a381a1df5
      ▼ SIP to address: sip:5202@192.168.10.35
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.35
        SIP to tag: a381a1df5
      Call-ID: 1mWGaQqm10
    ▼ CSeq: 20 INVITE
      Sequence Number: 20
      Method: INVITE
    Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
    Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
    Supported: replaces, timer
    ▼ WWW-Authenticate: Digest algorithm=MD5, realm="asterisk", nonce="782aac88"
      Authentication Scheme: Digest
      Algorithm: MD5, realm="asterisk"
      Nonce Value: "782aac88"
      Content-Length: 0
  ▼ Session Initiation Protocol (401)
    ▼ Status-Line: SIP/2.0 401 Unauthorized
      Status-Code: 401
      [Resent Packet: True]
      [Suspected resend of frame: 18546]
      [Request Frame: 18545]
      [Response Time (ms): 0]

```

STATUS

El frame número **18546**, es el correspondiente al **Status**, es el encargado del control del estado. El servidor Asterisk (192.168.10.35) realiza una petición hacia el terminal A (192.168.10.36). Se demuestra esta situación viendo que en el apartado Internet Protocol Version 4 el **Source** (Origen) -> 192.168.10.35 y **Destination** (Destino) -> 192.168.10.36.

En el apartado llamada User Datagram Protocol (UDP), existen varios puntos a tener en cuenta.

- **Source Port**: Puerto de origen que comunica este frame. Siendo el puerto predefinida en el Servidor Asterisk para la comunicación VoIP (5060).
- **Destination Port**: Puerto de destino en el cual, va a recibir este frame el Terminal A (54524).
- **Length**: Tamaño del frame (521).

En el apartado Session Initiation Protocol (SIP), destacamos varios puntos que se explicarán a continuación.

- **Status-code**: Este código se refiere a una comunicación no autorizada. Esto puede ser motivado a que está activada la opción de aceptar cualquier terminal que realice una comunicación (401).
- **Request Frame**: Es una respuesta al frame 18545, que en este caso es la respuesta al frame anterior en el que se realizaba el INVITE (18545).
- **Responde Time(ms)**: Tiempo de respuesta 0.

En la cabecera del mensaje (Message Header) como en el anterior frame, cabe destacar en el apartado **Via**:

el tipo de transporte (udp) los datos sobre la dirección IP del emisor del paquete (192.168.10.36), el puerto emisor (54524) y la dirección IP de la cual se ha recibido el paquete.

En el apartado **From** comentaré los siguientes subapartados:

- **SIP from address User Part**: Usuario que hace la petición de comunicación SIP (5206).
- **SIP from address Host Part**: Dirección IP del terminal al que se consulta la existencia del usuario en la base de datos del servidor Asterisk (192.168.10.35).

En el apartado **To** comentaré los siguientes subapartados:

- **SIP to address User Part**: Usuario que debe recibir la petición de comunicación SIP (5202).
- **SIP to address Host Part**: Dirección IP del terminal al que se consulta la existencia del usuario en la base de datos del servidor Asterisk (192.168.10.35).

El frame STATUS se manda de manera duplicada, esto es debido a si es necesario reenviar el mismo paquete. Este paquete tendrá todos los apartados iguales salvo en la Session Initiation Protocol (401), subapartado Status-Line. Se activara la opción Resent Packet (True) y la opción Suspected resend of frame (18546). Esto es debido a la posibilidad de perdida del paquete o retraso del mismo.

ACK sip:5202@192.168.10.35

```

└─> Session Initiation Protocol (ACK)
    └─> Request-Line: ACK sip:5202@192.168.10.35 SIP/2.0
        Method: ACK
    └─> Request-URI: sip:5202@192.168.10.35
        Request-URI User Part: 5202
        Request-URI Host Part: 192.168.10.35
    [Resent Packet: False]
    [Request Frame: 18545]
    [Response Time (ms): 105]
    └─> Message Header
        └─> Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.f1W2EL0Ve;rport
            Transport: UDP
            Sent-by Address: 192.168.10.36
            Sent-by port: 54524
            Branch: z9hG4bK.f1W2EL0Ve
            RPort: rport
            Call-ID: lmWGaQqm10
        └─> From: <sip:5206@192.168.10.35>;tag=14G-avb06
            └─> SIP from address: sip:5206@192.168.10.35
                SIP from address User Part: 5206
                SIP from address Host Part: 192.168.10.35
                SIP from tag: 14G-avb06
        └─> To: <sip:5202@192.168.10.35>;tag=as381a1df5
            └─> SIP to address: sip:5202@192.168.10.35
                SIP to address User Part: 5202
                SIP to address Host Part: 192.168.10.35
                SIP to tag: as381a1df5
        └─> Contact: <sip:5206@192.168.10.36:54524;transport=udp>;+sip.instance=<urn:uuid:24db554b-01f9-4900-aa1b-5594643cd600>
            └─> Contact URI: sip:5206@192.168.10.36:54524;transport=udp
                Contact URI User Part: 5206
                Contact URI Host Part: 192.168.10.36
                Contact URI Host Port: 54524
                Contact URI parameter: transport=udp
                Contact parameter: +sip.instance=<urn:uuid:24db554b-01f9-4900-aa1b-5594643cd600>\r\n
            Max-Forwards: 70
        └─> CSeq: 20 ACK
            Sequence Number: 20
            Method: ACK

```

ACK sip:5202@192.168.10.35

El frame **18548** es el acuse de recibo o confirmación (ACK) de la anterior petición (**STATUS 401**(autenticación para realizar la llamada)). Este frame lo envía el terminal A (192.168.10.36) como respuesta a la anterior petición de autenticación en el servidor Asterisk (192.168.10.35), se sigue usando IPv4 con un tamaño de cabecera de 20 bytes.

En el apartado **User Datagram Protocol**, el puerto emisor es 54524 (Terminal A) el puerto de destino es el 5060 (Servidor Asterisk).

En el apartado **Session Initiation Protocol**, destaca el subapartado llamado Method: que lleva el nombre de ACK para diferenciarlo de otro tipo de paquetes. En el Request-URI se especifica el usuario al que se llama (5202) y la dirección IP del Servidor Asterisk (192.168.10.35). En el subapartado Request Frame nos muestra el número del frame INVITE (18545).

En el apartado **Message Header**, en lo referente al subapartado nos indica el tipo de transporte (udp), la dirección IP emisora del paquete (192.168.10.36), el puerto por el cual se envían (54524). En el subapartado From, nos indica que la petición la realiza el usuario (5206) y la dirección IP del servidor Asterisk (192.168.10.35). En el subapartado To, nos indica el usuario al que se quiere llamar (5202) y la dirección IP del servidor Asterisk (192.168.10.35).

En el subapartado **Contact**, nos indican todos los datos del usuario que emite la llamada. Número de usuario (5206), dirección IP del terminal desde el cual se hace la llamada (192.168.10.36), puerto desde el cual se realiza la llamada (54524) y el tipo de transporte (udp).

INVITE sip:5202@192.168.10.35

El frame **18549** es idéntico que el primer frame (18545 - INVITE). Esta comunicación se realiza desde el terminal A (192.168.10.36) hacia el servidor Asterisk (192.168.10.35). Se usa el protocolo IPv4, el tamaño de la cabecera es de tamaño de unos 20 bytes. El paquete no está fragmentado. El puerto origen desde el que se realiza la comunicación es el 54524 y el puerto destino es el 5060.

En el apartado **Session Initiation Protocol**, existen varios subapartados de los cuales cabe destacar el Request-Line, en el que se especifican el tipo de frame (**INVITE**), el tipo de comunicación (sip), el número del contacto (5202) y la ip del servidor Asterisk (192.168.10.35) al cual se le pregunta si existe este usuario en la base de datos.

En el apartado **Message Header**, hay varios subapartados que a continuación se explicarán. En el subapartado Via muestra el protocolo que se va a utilizar (SIP), el modo de transporte (UDP), la dirección IP del terminal A (192.168.10.36) que en su caso es el emisor de este paquete, el puerto de emisión (54524).

En el apartado **From**, se observa el protocolo usado para la comunicación (SIP), el número del usuario que realiza la llamada (5206) y la dirección IP del servidor Asterisk (192.168.10.35).

En el apartado **To**, se observa el protocolo usado para la comunicación (SIP), el número del usuario al cual se va a llamar (5202) y la dirección IP del servidor Asterisk (192.168.10.35).

En el apartado **Contact**, se muestran todos los datos del terminal A. Protocolo a usar (SIP), número del usuario (5206), dirección IP del terminal (192.168.10.36), puerto desde el cual se comunica (54524) y el tipo de transporte que se usa para la comunicación (UDP). También tenemos que tener en cuenta el subapartado User-Agent, en el que observamos el nombre del programa que usa el terminal para realizar la llamada (LiphoneAndroid/3.2.4)

En el apartado **Message Body** que contiene el apartado Session Description Protocol (SDP), este último tiene varios subapartados que se explicarán a continuación.

En el subapartado Owner, se trata de los datos del que realiza la petición de llamada. Entre ellos vemos los siguientes; Número de usuario del propietario (5206), tipo de protocolo que usa (IPv4), dirección IP del usuario (192.168.10.36) y el nombre de sesión (Talk).

En el subapartado Connection Information, contiene los siguientes subapartados; Tipo de protocolo que usa en la comunicación (IPv4) y dirección IP del usuario (192.168.10.36).

En el subapartado Bandwidth Information, contiene los siguientes subapartados; adecuación del ancho de banda para poder reaizar las llamadas a partir del protocolo RTP, valor máximo del ancho de banda es de 500kb/s

En el subapartado de Media Description, cabe destacar el tipo de comunicación multimedia (audio), puerto con el que se realiza la comunicación multimedia (7076), protocolo multimedia a usar (RTP), codec a usar (GSM 06.10).

En el subapartado Media Attribute, cabe destacar MIME type (telephone-event) y Sample Rate (8000 HZ).

▼ Session Initiation Protocol (INVITE)

- ▼ Request-Line: INVITE sip:5202@192.168.10.35 SIP/2.0
 - Method: INVITE
- ▼ Request-URI: sip:5202@192.168.10.35
 - Request-URI User Part: 5202
 - Request-URI Host Part: 192.168.10.35
 - [Resent Packet: False]
- ▼ Message Header
 - ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.HizgKHyI2;rport
 - Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.HizgKHyI2;rport
 - Transport: UDP
 - Sent-by Address: 192.168.10.36
 - Sent-by port: 54524
 - Branch: z9hG4bK.HizgKHyI2
 - RPort: rport
 - ▼ From: <sip:5206@192.168.10.35>;tag=14G-avb06
 - SIP from address: sip:5206@192.168.10.35
 - SIP from address User Part: 5206
 - SIP from address Host Part: 192.168.10.35
 - SIP from tag: 14G-avb06
 - ▼ To: sip:5202@192.168.10.35
 - SIP to address: sip:5202@192.168.10.35
 - SIP to address User Part: 5202
 - SIP to address Host Part: 192.168.10.35
 - ▼ CSeq: 21 INVITE
 - Sequence Number: 21
 - Method: INVITE
 - Call-ID: 1mWGaQqm1o
 - Max-Forwards: 70
 - Supported: replaces, outbound
 - Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO, UPDATE
 - Content-Type: application/sdp
 - Content-Length: 230
- ▼ Contact: <sip:5206@192.168.10.36:54524;transport=udp>;+sip.instance=<urn:uuid:24db554b-01f9-4900-aa1b-5594643cd600>
 - Contact URI: sip:5206@192.168.10.36:54524;transport=udp
 - Contact URI User Part: 5206
 - Contact URI Host Part: 192.168.10.36
 - Contact URI Host Port: 54524
 - Contact URI parameter: transport=udp
 - Contact parameter: +sip.instance=<urn:uuid:24db554b-01f9-4900-aa1b-5594643cd600>\r\n
- User-Agent: LinphoneAndroid/3.2.4 (belle-sip/1.5.0)
- ▼ Authorization: Digest realm="asterisk", nonce="782aac88", algorithm=MD5, username="5206", uri="sip:5202@192.168.10.35", re
 - Authentication Scheme: Digest
 - Realm: "asterisk"
 - Nonce Value: "782aac88"
 - Algorithm: MD5, username="5206"
 - Authentication URI: "sip:5202@192.168.10.35"
 - Digest Authentication Response: "6b18b00f994dd76de724c16457864bab"
- ▼ Message Body
 - ▼ Session Description Protocol
 - Session Description Protocol Version (v): 0
 - ▼ Owner/Creator, Session Id (o): 5206 2621 3506 IN IP4 192.168.10.36
 - Owner Username: 5206
 - Session ID: 2621
 - Session Version: 3506
 - Owner Network Type: IN
 - Owner Address Type: IP4
 - Owner Address: 192.168.10.36
 - Session Name (s): Talk
 - ▼ Connection Information (c): IN IP4 192.168.10.36
 - Connection Network Type: IN
 - Connection Address Type: IP4
 - Connection Address: 192.168.10.36
 - ▼ Bandwidth Information (b): AS:500
 - Bandwidth Modifier: AS [Application Specific (RTP session bandwidth)]
 - Bandwidth Value: 500 kb/s
 - ▼ Time Description, active time (t): 0 0
 - Session Start Time: 0
 - Session Stop Time: 0
 - ▼ Session Attribute (a): rtcp-xr:rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-metrics
 - Session Attribute Fieldname: rtcp-xr
 - Session Attribute Value: rcvr-rtt=all:10000 stat-summary=loss,dup,jitt,TTL voip-metrics
 - ▼ Media Description, name and address (m): audio 7076 RTP/AVP 3 101
 - Media Type: audio
 - Media Port: 7076
 - Media Protocol: RTP/AVP
 - Media Format: GSM 06.10
 - Media Format: DynamicRTP-Type-101
 - ▼ Media Attribute (a): rtpmap:101 telephone-event/8000
 - Media Attribute Fieldname: rtpmap
 - Media Format: 101
 - MIME Type: telephone-event
 - Sample Rate: 8000

[INVITE sip:5202@192.168.10.35](#)

STATUS: 100 Trying

```

Session Initiation Protocol (100)
  ▼ Status-Line: SIP/2.0 100 Trying
    Status-Code: 100
    [Resent Packet: False]
    [Request Frame: 18549]
    [Response Time (ms): 0]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.HizgKHyI2;received=192.168.10.36;rport=54524
      Transport: UDP
      Sent-by Address: 192.168.10.36
      Sent-by port: 54524
      Branch: z9hG4bK.HizgKHyI2
      Received: 192.168.10.36
      RPort: 54524
    ▼ From: <sip:5206@192.168.10.35>;tag=14G-avb06
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: 14G-avb06
    ▼ To: sip:5202@192.168.10.35
      ▼ SIP to address: sip:5202@192.168.10.35
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.35
      Call-ID: lmWGaQqm10
    ▼ CSeq: 21 INVITE
      Sequence Number: 21
      Method: INVITE
    Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
    Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
    Supported: replaces, timer
    ▼ Contact: <sip:5202@192.168.10.35:5060>
      ▼ Contact URI: sip:5202@192.168.10.35:5060
        Contact URI User Part: 5202
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
    Content-Length: 0

```

STATUS: 100 Trying

Este frame (**18550**) lo realiza el servidor Asterisk (192.168.10.35) hacia el terminal A (192.168.10.36). El tipo de comunicación que se realiza en esta comunicación es el IPv4. El tamaño de la cabecera es de 20 bytes. La calidad del servicio se ve reflejado en el apartado Differentiated Services Field (CS3). El protocolo que se utiliza para la comunicación es UDP. Los puertos elegido para esta función es el puerto 5060 del servidor Asterisk y el puerto 54524 del terminal A.

En el apartado **Session Initiation Protocol** (SIP), vemos que el código del status (100) que hace referencia a la opción Trying y a su vez es la respuesta a la petición realizada en el frame 18549 (INVITE).

En el apartado **Message Header**, se encuentran diversos subapartados que a continuación explicaré. En el subapartado **Via**, se muestran los datos del terminal A. Tipo de transporte (UDP), dirección IP (192.168.10.36), puerto desde el cual se realiza la llamada (54524). En el subapartado **From**, se muestra el protocolo a usar (SIP), el número del usuario que realiza la llamada (5206) y la dirección IP del servidor Asterisk (192.168.10.35).

En el subapartado **To**, se muestra el protocolo a usar (SIP), el número del usuario al que se va a llamar (5202) y la dirección del servidor Asterisk (192.168.10.35).

En el subapartado **Contact**, se muestra el protocolo a usar (SIP), el número del usuario al que se va a llamar (5202), la dirección IP del servidor Asterisk (192.168.10.35) y el puerto que va a usar el servidor Asterisk para la comunicación SIP (5060).

INVITE sip:5202@192.168.10.34:45910; transport=udp

```

▼ Session Initiation Protocol (INVITE)
  ▼ Request-Line: INVITE sip:5202@192.168.10.34:45910;transport=udp SIP/2.0
    Method: INVITE
    ▶ Request-URI: sip:5202@192.168.10.34:45910;transport=udp
      [Resent Packet: False]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK18b15dad
      Transport: UDP
      Sent-by Address: 192.168.10.35
      Sent-by port: 5060
      Branch: z9hG4bK18b15dad
      Max-Forwards: 70
    ▼ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
      SIP Display info: "Alpha"
      ▶ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: as2a70f0bd
    ▼ To: <sip:5202@192.168.10.34:45910;transport=udp>
      ▶ SIP to address: sip:5202@192.168.10.34:45910;transport=udp
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.34
        SIP to address Host Port: 45910
        SIP To URI parameter: transport=udp
    ▼ Contact: <sip:5206@192.168.10.35:5060>
      ▶ Contact URI: sip:5206@192.168.10.35:5060
        Contact URI User Part: 5206
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
      Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▼ CSeq: 102 INVITE
      Sequence Number: 102
      Method: INVITE
      User-Agent: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
      Date: Wed, 01 Feb 2017 12:20:45 GMT
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
      Supported: replaces, timer
      Content-Type: application/sdp
      Content-Length: 619
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK18b15dad
      Transport: UDP
      Sent-by Address: 192.168.10.35
      Sent-by port: 5060
      Branch: z9hG4bK18b15dad
      Max-Forwards: 70
    ▼ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
      SIP Display info: "Alpha"
      ▶ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: as2a70f0bd
    ▼ To: <sip:5202@192.168.10.34:45910;transport=udp>
      ▶ SIP to address: sip:5202@192.168.10.34:45910;transport=udp
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.34
        SIP to address Host Port: 45910
        SIP To URI parameter: transport=udp
    ▼ Contact: <sip:5206@192.168.10.35:5060>
      ▶ Contact URI: sip:5206@192.168.10.35:5060
        Contact URI User Part: 5206
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
      Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▼ CSeq: 102 INVITE
      Sequence Number: 102
      Method: INVITE
      User-Agent: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
      Date: Wed, 01 Feb 2017 12:20:45 GMT
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
      Supported: replaces, timer
      Content-Type: application/sdp
      Content-Length: 819

```

INVITE sip:5202@192.168.10.34:45910; transport=udp

En el siguiente frame (**18552**), se realiza desde el servidor Asterisk un INVITE con la dirección IP 192.168.10.35, hacia el terminal B con la dirección IP (192.168.10.34). Observando los apartados Internet Protocol Version, observamos que estamos tratando con la versión IPv4. El puerto que usa el servidor Asterisk (192.168.10.35) para el envío de este frame es el 5060, y el puerto que usa el terminal B (192.168.10.34) para la recepción de este frame es el 45910.

En el apartado **Session Initiation Protocol**, observamos diferentes subapartados que se explicarán a continuación.

En el apartado **Request-line**, se muestran los datos del tipo de frame (INVITE), el tipo de protocolo a usar (SIP), el número del usuario (5202), la dirección IP (192.168.10.34) y tipo de protocolo que se usa para el transporte de paquetes (UDP).

En el apartado **Message Header**, nos encontramos diferentes subapartados que vamos a explicar a continuación.

En el subapartado **Vía**, se muestran el tipo de protocolo a usar (SIP), la dirección IP del servidor Asterisk (192.168.10.35) y el puerto a usar (5060).

En el subapartado **From**, se muestran el nombre del usuario (SIP Display info : "Alpha"), el tipo de protocolo (SIP), el número del usuario que realiza la llamada (5206) y la dirección IP del servidor Asterisk (192.168.10.35).

En el subapartado **To**, se muestran el protocolo que a usar (SIP), el número del ususario al que se le llama (5202), la dirección IP del terminal B (192.168.10.34), el puerto a usar para esta comunicación (45910) y el tipo de protocolo usado para transportar los paquetes (UDP).

En el subapartado **Contact**, se muestran los datos del usuario que realiza la llamada, el número del usuario (5206), la direccion IP del servidor Asterisk (192.168.10.35) y el puerto del servidor Asterisk (5060) desde el cual se realiza las llamadas SIP.

En el apartado **Session Description Protocol** (SDP), se observan diferentes subapartados que se explicarán a continuación.

En el subapartado **Owner/Creator**, se muestran los datos del servidor Asterisk. Entre los datos que se muestran encontramos el tipo de usuario que es propietario (root), la versión del protocolo IP (IPv4), la dirección IP del servidor Asterisk (192.168.10.35) y el nombre del programa que origina este frame (Asterisk PBX 11.10.0).

En el subapartado **Connection Information**, se muestra la versión del protocolo IP (IPv4), y la dirección IP del servidor Asterisk (192.168.10.35).

En el subapartado **Bandwidth Information**, se muestra el tipo de comunicación que se va a realizar (RTP) y el ancho de banda que se ha configurado como máximo (512 kb/s).

En el subapartado **Media Description**, se muestran todos los codecs que dispone activos el servidor Asterisk para poder hacer efectiva la comunicación SIP. Se observan el tipo de comunicación del que se trata (audio), el puerto que se usará para la comunicación (13040), el tipo de protocolo para la comunicación (RTP), y todos los codec que se encuentran activados en el servidor Asterisk (GSM, G.711 u, G.711-a, G.722, iLBC, Speex, Speex16, Speex32).

Hay que tener en cuenta 3 atributos multimedia.

- **telephone-event/8000.**: Para los eventos de telefonía usa un ratio de unos 8000 Hz.
- **fntp: 101 0 - 16.**: Son las señales codificadas de DTMF, su payload type es 101 (dynamic), utiliza desde 0 - 16 para los eventos de los tonos DTMF.
- **ptime: 20**. Es el tiempo de empaquetado.

En el subapartado **Media Description**, se muestran los codec de video (h.264, h.263, h.263p y h.261), el tipo de comunicación (video), el puerto a usar en caso de comunicación por video (14464), el protocolo a usar (RTP).

```
▼ Message Body
  ▼ Session Description Protocol
    Session Description Protocol Version (v): 0
    ▼ Owner/Creator, Session Id (o): root 136809880 136809880 IN IP4 192.168.10.35
      Owner Username: root
      Session ID: 136809880
      Session Version: 136809880
      Owner Network Type: IN
      Owner Address Type: IP4
      Owner Address: 192.168.10.35
      Session Name (s): Asterisk PBX 11.16.0
    ▼ Connection Information (c): IN IP4 192.168.10.35
      Connection Network Type: IN
      Connection Address Type: IP4
      Connection Address: 192.168.10.35
    ▼ Bandwidth Information (b): CT:512
      Bandwidth Modifier: CT [Conference Total(total bandwidth of all RTP sessions)]
      Bandwidth Value: 512 kb/s
    ▼ Time Description, active time (t): 0 0
      Session Start Time: 0
      Session Stop Time: 0
    ▼ Media Description, name and address (m): audio 13040 RTP/AVP 3 0 8 9 97 117 119 110 101
      Media Type: audio
      Media Port: 13040
      Media Protocol: RTP/AVP
      Media Format: GSM 06.10
      Media Format: ITU-T G.711 PCMU
      Media Format: ITU-T G.711 PCMA
      Media Format: ITU-T G.722
      Media Format: DynamicRTP-Type-97
      Media Format: DynamicRTP-Type-117
      Media Format: DynamicRTP-Type-119
      Media Format: DynamicRTP-Type-110
      Media Format: DynamicRTP-Type-101
    ▼ Media Attribute (a): rtpmap:3 GSM/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 3
      MIME Type: GSM
      Sample Rate: 8000
    ▼ Media Attribute (a): rtpmap:0 PCMU/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 0
      MIME Type: PCMU
      Sample Rate: 8000
    ▼ Media Attribute (a): rtpmap:8 PCMA/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 8
      MIME Type: PCMA
      Sample Rate: 8000
    ▼ Media Attribute (a): rtpmap:9 G722/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 9
      MIME Type: G722
      Sample Rate: 8000
    ▼ Media Attribute (a): rtpmap:97 ilBC/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 97
      MIME Type: ilBC
      Sample Rate: 8000
    ▼ Media Attribute (a): ftmp:97 mode=30
      Media Attribute Fieldname: ftmp
      Media Format: 97 [ilBC]
      Media format specific parameters: mode=30
    ▼ Media Attribute (a): rtpmap:117 speex/16000
      Media Attribute Fieldname: rtpmap
      Media Format: 117
      MIME Type: speex
      Sample Rate: 16000
    ▼ Media Attribute (a): rtpmap:119 speex/32000
      Media Attribute Fieldname: rtpmap
      Media Format: 119
      MIME Type: speex
      Sample Rate: 32000
    ▼ Media Attribute (a): rtpmap:110 speex/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 110
      MIME Type: speex
      Sample Rate: 8000
    ▼ Media Attribute (a): rtpmap:101 telephone-event/8000
      Media Attribute Fieldname: rtpmap
      Media Format: 101
      MIME Type: telephone-event
      Sample Rate: 8000
```

INVITE sip:5202@192.168.10.34:45910; transport=udp

```

▼ Media Attribute (a): fmtp:101 0-16
  Media Attribute Fieldname: fmtp
  Media Format: 101 [telephone-event]
  Media format specific parameters: 0-16
▼ Media Attribute (a): ptimetime:20
  Media Attribute Fieldname: ptimetime
  Media Attribute Value: 20
  Media Attribute (a): sendrecv
▼ Media Description, name and address (m): video 14464 RTP/AVP 99 104 98 34 31
  Media Type: video
  Media Port: 14464
  Media Protocol: RTP/AVP
  Media Format: DynamicRTP-Type-99
  Media Format: DynamicRTP-Type-104
  Media Format: DynamicRTP-Type-98
  Media Format: ITU-T H.263
  Media Format: ITU-T H.261
  Media Format: H264
  Sample Rate: 90000
▼ Media Attribute (a): rtpmap:99 H264/90000
  Media Attribute Fieldname: rtpmap
  Media Format: 99
  MIME Type: H264
  Sample Rate: 90000
▼ Media Attribute (a): fmtp:99 redundant-pic-cap=0;parameter-add=0;packetization-mode=0;level-asymmetry-allowed=0
  Media Attribute Fieldname: fmtp
  Media Format: 99 [H264]
  Media format specific parameters: redundant-pic-cap=0
  Media format specific parameters: parameter-add=0
  ▼ Media format specific parameters: packetization-mode=0
    [Packetization mode: Single NAL mode (0)]
  Media format specific parameters: level-asymmetry-allowed=0
▼ Media Attribute (a): rtpmap:104 MP4V-ES/90000
  Media Attribute Fieldname: rtpmap
  Media Format: 104
  MIME Type: MP4V-ES
  Sample Rate: 90000
▼ Media Attribute (a): rtpmap:98 H263-1998/90000
  Media Attribute Fieldname: rtpmap
  Media Format: 98
  MIME Type: H263-1998
  Sample Rate: 90000
▼ Media Attribute (a): fmtp:98 F=0;I=0;J=0;T=0;K=0;N=0;BPP=0;HRD=0
  Media Attribute Fieldname: fmtp
  Media Format: 98 [H263-1998]
  Media format specific parameters: F=0
  Media format specific parameters: I=0
  Media format specific parameters: J=0
  Media format specific parameters: T=0
  Media format specific parameters: K=0
  Media format specific parameters: N=0
  Media format specific parameters: BPP=0
  Media format specific parameters: HRD=0
▼ Media Attribute (a): rtpmap:34 H263/90000
  Media Attribute Fieldname: rtpmap
  Media Format: 34
  MIME Type: H263
  Sample Rate: 90000
▼ Media Attribute (a): fmtp:34 F=0;I=0;J=0;T=0;K=0;N=0;BPP=0;HRD=0
  Media Attribute Fieldname: fmtp
  Media Format: 34 [H263]
  Media format specific parameters: F=0
  Media format specific parameters: I=0
  Media format specific parameters: J=0
  Media format specific parameters: T=0
  Media format specific parameters: K=0
  Media format specific parameters: N=0
  Media format specific parameters: BPP=0
  Media format specific parameters: HRD=0
▼ Media Attribute (a): rtpmap:31 H261/90000
  Media Attribute Fieldname: rtpmap
  Media Format: 31
  MIME Type: H261
  Sample Rate: 90000
  Media Attribute (a): sendrecv

```

INVITE sip:5202@192.168.10.34:45910; transport=udp

STATUS: 100 Trying

```

▼ Session Initiation Protocol (100)
  ▼ Status-Line: SIP/2.0 100 Trying
    Status-Code: 100
    [Resent Packet: False]
    [Request Frame: 18552]
    [Response Time (ms): 224]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK18b15dad
      Transport: UDP
      Sent-by Address: 192.168.10.35
      Sent-by port: 5060
      Branch: z9hG4bK18b15dad
    ▼ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
      SIP Display info: "Alpha"
    ▼ SIP from address: sip:5206@192.168.10.35
      SIP from address User Part: 5206
      SIP from address Host Part: 192.168.10.35
      SIP from tag: as2a70f0bd
    ▼ To: <sip:5202@192.168.10.34:45910;transport=udp>
      ▼ SIP to address: sip:5202@192.168.10.34:45910;transport=udp
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.34
        SIP to address Host Port: 45910
        SIP To URI parameter: transport=udp
      Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▼ CSeq: 102 INVITE
      Sequence Number: 102
      Method: INVITE

```

STATUS: 100 Trying

El envío de este frame (**18554**), lo realiza el terminal B (192.168.10.34) hacia el servidor Asterisk (192.168.10.35), el puerto que se usa para transmitir este frame es el 45910 del terminal B y es recepcionado por el servidor Asterisk en el puerto 5060.

Este frame es la **respuesta al INVITE** que se realiza en el frame 18552, esto viene reflejado en Request Frame: 18552 (Session Initiation Protocol/Status-Line).

En el apartado **Message Header**, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (servidor Asterisk (192.168.10.35)) del frame 18552 (INVITE).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que envía el frame (192.168.10.35).
 - Puerto desde el que se realiza la comunicación (5060).
- Datos desde donde se envía la petición (From).
 - Nombre que se va a visualizar (Alpha).
 - Protocolo que se usará para la comunicación (SIP).
 - Número del usuario (5206).
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del usuario del terminal B (192.168.10.34).
 - Puerto que se usa el terminal B para la comunicación (45910).

Ringing

```
18556 823.86067... 192.168.10.34      192.168.10.35      SIP      406 Status: 180 Ringing |
18557 823.86119... 192.168.10.35      192.168.10.36      SIP      514 Status: 180 Ringing |
18558 823.86119... 192.168.10.35      192.168.10.36      SIP      514 Status: 180 Ringing |
```

Llamando (Wireshark)

Estos frames son los encargados de generar los impulsos de llamada (Ring). Los cuales se explicarán a continuación.

RINGING (Terminal B (192.168.10.34) -> Servidor Asterisk (192.168.10.35))

```
▼ Session Initiation Protocol (180)
  ▼ Status-Line: SIP/2.0 180 Ringing
    Status-Code: 180
    [Resent Packet: False]
    [Request Frame: 18552]
    [Response Time (ms): 748]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK18b15dad
      Transport: UDP
      Sent-by Address: 192.168.10.35
      Sent-by port: 5060
      Branch: z9hG4bK18b15dad
    ▼ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
      SIP Display info: "Alpha"
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: as2a70f0bd
    ▼ To: <sip:5202@192.168.10.34:45910;transport=udp>;tag=fEShCgR
      ▼ SIP to address: sip:5202@192.168.10.34:45910;transport=udp
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.34
        SIP to address Host Port: 45910
        SIP To URI parameter: transport=udp
        SIP to tag: fEShCgR
      Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▼ CSeq: 102 INVITE
      Sequence Number: 102
      Method: INVITE
      User-Agent: LinphoneAndroid/3.2.1 (belle-sip/1.5.0)
      Supported: replaces, outbound
```

RINGING (Terminal B (192.168.10.34) -> Servidor Asterisk (192.168.10.35))

En el frame **18556** el terminal B (192.168.10.34) realiza la petición (Ringing) hacia el servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, posee un tamaño de cabecera de 20 bytes. este paquete tiene prioridad, ya que tiene activado el QOS (DSCP: AF31 (Flash o Relámpago)). No hay fragmentación del paquete. El puerto que usa el terminal B para enviar este frame es el 45910 y el puerto que usa el servidor Asterisk para recibir este frame es el 5060.

En el apartado **Session Initiation Protocol** (100), observamos el tipo de protocolo a usar (SIP), siendo este un frame de respuesta al frame 18552 (INVITE).

En el apartado **Message Header**, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (servidor Asterisk (192.168.10.35)) del frame 18552 (INVITE).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que envía el frame (192.168.10.35).
 - Puerto desde el que se realiza la comunicación (5060).
- Datos desde donde se envía la petición (From).

- Nombre que se va a visualizar (Alpha).
- Protocolo que se usará para la comunicación (SIP).
- Número del usuario (5206).
- Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del usuario del terminal B (192.168.10.34).
 - Puerto que se usa el terminal B para la comunicación (45910).
- Programa que se usa para poder realizar esta comunicación (LinphoneAndroid/3.2.1)

RINGING (Servidor Asterisk (192.168.10.35) -> Terminal A (192.168.10.36))

```

▼ Session Initiation Protocol (180)
  ▼ Status-Line: SIP/2.0 180 Ringing
    Status-Code: 180
    [Resent Packet: False]
    [Request Frame: 18549]
    [Response Time (ms): 750]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.HizgKHyI2;received=192.168.10.36;rport=54524
      Transport: UDP
      Sent-by Address: 192.168.10.36
      Sent-by port: 54524
      Branch: z9hG4bK.HizgKHyI2
      Received: 192.168.10.36
      RPort: 54524
      From: <sip:5206@192.168.10.35>;tag=14G-avb06
        ▼ SIP from address: sip:5206@192.168.10.35
          SIP from address User Part: 5206
          SIP from address Host Part: 192.168.10.35
          SIP from tag: 14G-avb06
      ▼ To: sip:5202@192.168.10.35;tag=as7d4da5df
        ▶ SIP to address: sip:5202@192.168.10.35
        SIP to tag: as7d4da5df
        Call-ID: ImWGaQqm10
    ▼ CSeq: 21 INVITE
      Sequence Number: 21
      Method: INVITE
    Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
    Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
    Supported: replaces, timer
    ▼ Contact: <sip:5202@192.168.10.35:5060>
      ▼ Contact URI: sip:5202@192.168.10.35:5060
        Contact URI User Part: 5202
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
    Content-Length: 0
  
```

RINGING (Servidor Asterisk (192.168.10.35) -> Terminal A (192.168.10.36))

En el frame **18557** el servidor Asterisk (192.168.10.35) realiza una petición (Ringing) hacia el terminal A (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte (UDP), posee un tamaño de cabecera de 20 bytes. Este paquete posee prioridad ya que tiene activado el QOS (DSCP: CS3 (Flash o Relámpago)). El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal A para recibir este frame es el 54524.

En el apartado **Session Initiation Protocol** (100), observamos el tipo de protocolo a usar (SIP), siendo este un frame de respuesta al frame 18552 (INVITE).

En el apartado Message Header, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (servidor Asterisk (192.168.10.35)) del frame 18552 (INVITE).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que envía el frame (192.168.10.35).
 - Puerto desde el que se realiza la comunicación (5060).
- Datos desde donde se envía la petición (From).
 - Nombre que se va a visualizar (Alpha).
 - Protocolo que se usará para la comunicación (SIP).
 - Número del usuario (5206).
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del usuario del terminal B (192.168.10.34).
 - Puerto que se usa el terminal B para la comunicación (45910).
- Programa que se usa para poder realizar esta comunicación (LinphoneAndroid/3.2.1)

RINGING (Servidor Asterisk (192.168.10.35) -> Terminal A (192.168.10.36)) Duplicado

```

▼ Session Initiation Protocol (180)
  ▼ Status-Line: SIP/2.0 180 Ringing
    Status-Code: 180
    [Resent Packet: False]
    [Request Frame: 18549]
    [Response Time (ms): 750]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.HizgKHyI2;received=192.168.10.36;rport=54524
      Transport: UDP
      Sent-by Address: 192.168.10.36
      Sent-by port: 54524
      Branch: z9hG4bK.HizgKHyI2
      Received: 192.168.10.36
      RPort: 54524
    ▼ From: <sip:5206@192.168.10.35>;tag=14G-avb06
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: 14G-avb06
    ▼ To: sip:5202@192.168.10.35;tag=as7d4da5df
      ▶ SIP to address: sip:5202@192.168.10.35
      SIP to tag: as7d4da5df
      Call-ID: lmWGaQqm10
    ▼ CSeq: 21 INVITE
      Sequence Number: 21
      Method: INVITE
      Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
      Supported: replaces, timer
    ▼ Contact: <sip:5202@192.168.10.35:5060>
      ▼ Contact URI: sip:5202@192.168.10.35:5060
        Contact URI User Part: 5202
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
      Content-Length: 0

```

RINGING (Servidor Asterisk (192.168.10.35) -> Terminal A (192.168.10.36))

En el frame **18558** el servidor Asterisk (192.168.10.35) realiza una petición (Ringing) hacia el terminal A (192.168.10.36). Se usa como protocolo de internet IPv4, y el protocolo de transporte

es UDP, posee un tamaño de cabecera de 20 bytes. Este paquete posee prioridad ya que tiene activado el QOS (DSCP: CS3 (Flash o Relámpago)). El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal A para recibir este frame es el 54524.

En el apartado **Session Initiation Protocol** (100), observamos el tipo de protocolo a usar (SIP), siendo este un frame de respuesta al frame 18549 (INVITE).

En el apartado **Message Header**, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (Terminal A(192.168.10.36)) del frame 18549 (INVITE).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que envía el frame (192.168.10.36).
 - Puerto desde el que se realiza la comunicación (54524).
- Datos desde donde se envía la petición (From).
 - Protocolo que se usará para la comunicación (SIP).
 - Número del usuario (5206).
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Programa que se usa para poder realizar esta comunicación (LinphoneAndroid/3.2.1)

Establecimiento de la llamada

18577 829.45940... 192.168.10.34	192.168.10.35	SIP/SDP	1046 Status: 200 Ok
18579 829.46006... 192.168.10.35	192.168.10.34	SIP	480 Request: ACK sip:5202@192.168.10.34:45910;transport=udp
18580 829.46029... 192.168.10.35	192.168.10.34	SIP/SDP	480 Request: ACK sip:5202@192.168.10.34:45910;transport=udp
18581 829.46029... 192.168.10.35	192.168.10.36	SIP/SDP	779 Status: 200 OK
18582 829.46571... 192.168.10.34	192.168.10.35	STUN	779 Status: 200 OK
18583 829.46592... 192.168.10.35	192.168.10.34	STUN	62 Binding Request
18584 829.46593... 192.168.10.35	192.168.10.34	STUN	74 Binding Success Response MAPPED-ADDRESS: 192.168.10.34:7076
18585 829.46593... 192.168.10.34	192.168.10.35	RTCP	62 7077 → 13041 Len=20
18586 829.46599... 192.168.10.35	192.168.10.34	CLASSI...	74 Message: Binding Request
18587 829.46600... 192.168.10.35	192.168.10.34	CLASSI...	74 Message: Binding Request
18588 829.46605... 192.168.10.35	192.168.10.34	RTCP	74 13041 → 7077 Len=32
18589 829.46605... 192.168.10.35	192.168.10.34	RTCP	74 13041 → 7077 Len=32
18590 829.46609... 192.168.10.35	192.168.10.34	RTCP	74 13041 → 7077 Len=32
18591 829.46609... 192.168.10.35	192.168.10.34	RTCP	74 13041 → 7077 Len=32
18592 829.49037... 192.168.10.36	192.168.10.35	SIP	518 Request: ACK sip:5202@192.168.10.35:5060
18593 829.50988... 192.168.10.36	192.168.10.35	STUN	62 Binding Request
18594 829.50988... 192.168.10.36	192.168.10.35	RTCP	62 7077 → 12913 Len=20
18595 829.51000... 192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18596 829.51000... 192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18597 829.51004... 192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18598 829.51004... 192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18599 829.51008... 192.168.10.35	192.168.10.36	STUN	74 Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076
18600 829.51008... 192.168.10.35	192.168.10.36	STUN	74 Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076
18601 829.51011... 192.168.10.35	192.168.10.36	CLASSI...	74 Message: Binding Request
18602 829.51011... 192.168.10.35	192.168.10.36	CLASSI...	74 Message: Binding Request
18603 829.54312... 192.168.10.36	192.168.10.35	STUN	62 Binding Request

Estableciendo la llamada (Wireshark)

STATUS: 200 OK

En el frame **18577** el terminal B (192.168.10.34) realiza una petición (Status) hacia el servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. Este paquete posee prioridad ya que tiene activado el QOS (DSCP: AF31 (Flash o Relámpago)). El puerto que usa el terminal B para enviar este frame es el 45910 y el puerto que usa el servidor Asterisk para recibir este frame es el 5060. Este frame es una respuesta al frame 18552 (InVITE).

En el apartado **Message Header**, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (Terminal B(192.168.10.34)) del frame 18577 (STATUS-OK).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que se envía el frame (192.168.10.34).
 - Puerto desde el que se realiza la comunicación (45910).
- Datos del receptor (Servidor Asterisk (192.168.10.35))del frame 18577.
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que recibe el frame (192.168.10.35).
 - Puerto desde el que se realiza la comunicación (5060).
- Datos desde donde se envía la petición (From).
 - Nombre que se visualizará (Alpha).
 - Protocolo que se usará para la comunicación (SIP).
 - Número del usuario (5206).

```

▼ Session Initiation Protocol (200)
  ▼ Status-Line: SIP/2.0 200 Ok
    Status-Code: 200
    [Resent Packet: False]
    [Request Frame: 18552]
    [Response Time (ms): 6346]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK18b15dad
      Transport: UDP
      Sent-by Address: 192.168.10.35
      Sent-by port: 5060
      Branch: z9hG4bK18b15dad
    ▼ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
      SIP Display info: "Alpha"
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: as2a70f0bd
    ▼ To: <sip:5202@192.168.10.34:45910;transport=udp>;tag=fEShCgR
      ▼ SIP to address: sip:5202@192.168.10.34:45910;transport=udp
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.34
        SIP to address Host Port: 45910
        SIP To URI parameter: transport=udp
        SIP to tag: fEShCgR
      Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▼ CSeq: 102 INVITE
      Sequence Number: 102
      Method: INVITE
      User-Agent: LinphoneAndroid/3.2.1 (belle-sip/1.5.0)
      Supported: replaces, outbound
      Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, NOTIFY, MESSAGE, SUBSCRIBE, INFO, UPDATE
    ▼ Contact: <sip:5202@192.168.10.34:45910;transport=udp>;+sip.instance=<urn:uuid:5c413854-c489-41d8-8c9c-b498e44e2811>
      ▼ Contact URI: sip:5202@192.168.10.34:45910;transport=udp
        Contact URI User Part: 5202
        Contact URI Host Part: 192.168.10.34
        Contact URI Host Port: 45910
        Contact URI parameter: transport=udp
        Contact parameter: +sip.instance=<urn:uuid:5c413854-c489-41d8-8c9c-b498e44e2811>\r\n
      Content-Type: application/sdp
      Content-Length: 383
    ▼ Message Body
      ▼ Session Description Protocol
        Session Description Protocol Version (v): 0
      ▼ Owner/Creator, Session Id (o): 5202 1036 392 IN IP4 192.168.10.34
        Owner Username: 5202
        Session ID: 1036
        Session Version: 392
        Owner Network Type: IN
        Owner Address Type: IP4
        Owner Address: 192.168.10.34
      Session Name (s): Talk
      ▼ Connection Information (c): IN IP4 192.168.10.34
        Connection Network Type: IN
        Connection Address Type: IP4
        Connection Address: 192.168.10.34
      ▼ Bandwidth Information (b): AS:500
        Bandwidth Modifier: AS [Application Specific (RTP session bandwidth)]
        Bandwidth Value: 500 kb/s
      ▼ Time Description, active time (t): 0 0
        Session Start Time: 0
        Session Stop Time: 0
      ▼ Media Description, name and address (m): audio 7076 RTP/AVP 3 8 9 97 117 119 110 101
        Media Type: audio
        Media Port: 7076
        Media Protocol: RTP/AVP
        Media Format: GSM 06.10
        Media Format: ITU-T G.711 PCMA
        Media Format: ITU-T G.722
        Media Format: DynamicRTP-Type-97
        Media Format: DynamicRTP-Type-117
        Media Format: DynamicRTP-Type-119
        Media Format: DynamicRTP-Type-110
        Media Format: DynamicRTP-Type-101
      ▼ Media Attribute (a): rtpmap:97 ilBC/8000
        Media Attribute Fieldname: rtpmap
        Media Format: 97
        MTMF Type: ilBC

```

STATUS: 200 OK 2/1

```

▼ Media Attribute (a): rtpmap:97 ilBC/8000
  Media Attribute Fieldname: rtpmap
  Media Format: 97
  MIME Type: ilBC
  Sample Rate: 8000
▼ Media Attribute (a): fmtp:97 mode=30
  Media Attribute Fieldname: fmtp
  Media Format: 97 [ilBC]
  Media format specific parameters: mode=30
▼ Media Attribute (a): rtpmap:117 speex/16000
  Media Attribute Fieldname: rtpmap
  Media Format: 117
  MIME Type: speex
  Sample Rate: 16000
▼ Media Attribute (a): fmtp:117 vbr=on
  Media Attribute Fieldname: fmtp
  Media Format: 117 [speex]
  Media format specific parameters: vbr=on
▼ Media Attribute (a): rtpmap:119 speex/32000
  Media Attribute Fieldname: rtpmap
  Media Format: 119
  MIME Type: speex
  Sample Rate: 32000
▼ Media Attribute (a): fmtp:119 vbr=on
  Media Attribute Fieldname: fmtp
  Media Format: 119 [speex]
  Media format specific parameters: vbr=on
▼ Media Attribute (a): rtpmap:110 speex/8000
  Media Attribute Fieldname: rtpmap
  Media Format: 110
  MIME Type: speex
  Sample Rate: 8000
▼ Media Attribute (a): fmtp:110 vbr=on
  Media Attribute Fieldname: fmtp
  Media Format: 110 [speex]
  Media format specific parameters: vbr=on
  Media Format: 117
  MIME Type: speex
  Sample Rate: 16000
▼ Media Attribute (a): fmtp:117 vbr=on
  Media Attribute Fieldname: fmtp
  Media Format: 117 [speex]
  Media format specific parameters: vbr=on
▼ Media Attribute (a): rtpmap:119 speex/32000
  Media Attribute Fieldname: rtpmap
  Media Format: 119
  MIME Type: speex
  Sample Rate: 32000
▼ Media Attribute (a): fmtp:119 vbr=on
  Media Attribute Fieldname: fmtp
  Media Format: 119 [speex]
  Media format specific parameters: vbr=on
▼ Media Attribute (a): rtpmap:110 speex/8000
  Media Attribute Fieldname: rtpmap
  Media Format: 110
  MIME Type: speex
  Sample Rate: 8000
▼ Media Attribute (a): fmtp:110 vbr=on
  Media Attribute Fieldname: fmtp
  Media Format: 110 [speex]
  Media format specific parameters: vbr=on
▼ Media Attribute (a): rtpmap:101 telephone-event/8000
  Media Attribute Fieldname: rtpmap
  Media Format: 101
  MIME Type: telephone-event
  Sample Rate: 8000
▼ Media Description, name and address (m): video 0 RTP/AVP 0
  Media Type: video
  Media Port: 0
  Media Protocol: RTP/AVP
  Media Format: ITU-T G.711 PCMU
Media Attribute (a): inactive

```

STATUS: 200 OK 2/2

- Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del Terminal B (192.168.10.34).
 - Puerto que recibirá 45910.

En el apartado **Message Body**, se observan los diferentes datos sobre la comunicación del frame.

- Datos del creador (Owner/Creator).
 - Número del creador de este frame (5202).
 - Tipo de protocolo para realizar la comunicación (IPv4).
 - Dirección IP del que se envía el frame (192.168.10.34).
- Ancho de banda a usar (Bandwidth information).
 - Adaptación del ancho de banda según el protocolo (RTP).
 - Ancho de banda máximo (500kb/s).
- Información sobre los codecs multimedia a usar (Media Description).
 - Tipo de comunicación multimedia (audio).
 - Puerto usado para la comunicación (7076).
 - Protocolo usado para la comunicación (RTP).
 - Descripción de codecs disponibles a usar por el terminal B.
 - * GSM 06.10.
 - * G.711 PCMA.
 - * G.722.
 - * iLBC.
 - * speex 8000 hz.
 - * speex 16000 hz.
 - * speex 32000 hz.
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del Terminal B (192.168.10.34).
 - Puerto que recibirá 45910.

Request: ACK sip:5202@192.168.10.34:45910;transport=udp

```

▼ Session Initiation Protocol (ACK)
  ▼ Request-Line: ACK sip:5202@192.168.10.34:45910;transport=udp SIP/2.0
    Method: ACK
    ▶ Request-URI: sip:5202@192.168.10.34:45910;transport=udp
      [Resent Packet: False]
      [Request Frame: 18552]
      [Response Time (ms): 6347]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK5eaf0d6f
      Transport: UDP
      Sent-by Address: 192.168.10.35
      Sent-by port: 5060
      Branch: z9hG4bK5eaf0d6f
      Max-Forwards: 70
    ▼ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
      SIP Display info: "Alpha"
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: as2a70f0bd
    ▼ To: <sip:5202@192.168.10.34:45910;transport=udp>;tag=fEShCgR
      ▼ SIP to address: sip:5202@192.168.10.34:45910;transport=udp
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.34
        SIP to address Host Port: 45910
        SIP To URI parameter: transport=udp
        SIP to tag: fEShCgR
    ▼ Contact: <sip:5206@192.168.10.35:5060>
      ▼ Contact URI: sip:5206@192.168.10.35:5060
        Contact URI User Part: 5206
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
      Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▼ CSeq: 102 ACK
      Sequence Number: 102
      Method: ACK
    User-Agent: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
    Content-Length: 0
  
```

Request: ACK sip:5202@192.168.10.34:45910;transport=udp

En el frame **18578** el servidor Asterisk (192.168.10.35) realiza una petición (Request: ACK) hacia el terminal B (192.168.10.34). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. Este paquete posee prioridad ya que tiene activado el QOS (DSCP: CS3 (Flash o Relámpago)). El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal B para recibir este frame es el 45910.

En el apartado **Session Initiation Protocol** (ACK), siendo este un frame de respuesta al frame 18552 (INVITE).

- Tipo de protocolo a usar para la comunicación (SIP).
- Protocolo de transporte a usar (UDP).
- Dirección IP del dispositivo que envía el frame (192.168.10.34).
- Puerto a usar en la comunicación (45910).
- Número del terminal B (5202).

En el apartado **Message Header**, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (Servidor Asterisk (192.168.10.35)) del frame 18578 (ACK).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que envía el frame (192.168.10.35).
 - Puerto desde el que se realiza la comunicación (5060).

- Datos desde donde se envía la petición (From).
 - Nombre que se visualizará (Alpha).
 - Protocolo que se usará para la comunicación (SIP).
 - Número del usuario (5206).
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del terminal B (192.168.10.34).
 - Puerto a usar (45910).

Request: ACK sip:5202@192.168.10.34:45910;transport=udp Duplicado

```
▼ Session Initiation Protocol (ACK)
  ▼ Request-Line: ACK sip:5202@192.168.10.34:45910;transport=udp SIP/2.0
    Method: ACK
    ▶ Request-URI: sip:5202@192.168.10.34:45910;transport=udp
    [Resent Packet: False]
    [Request Frame: 18578]
    [Response Time (ms): 6347]
```

Request: ACK sip:5202@192.168.10.34:45910;transport=udp Duplicado

Este frame es idéntico al anterior con una única salvedad. Este frame es en respuesta al frame 18578 (frame anterior).

Status: 200 OK

En el frame **18580** el servidor Asterisk (192.168.10.35) realiza una petición (Status: 200 OK) hacia el terminal A (192.168.10.36). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. Este paquete posee prioridad ya que tiene activado el QOS (DSCP: CS3 (Flash o Relámpago)). El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal A para recibir este frame es el 54524.

En el apartado **Session Initiation Protocol** (Status - 200 OK), se observan diferentes datos sobre la comunicación del frame.

- Protocolo de comunicación a usar (SIP).
- Status - Code (200).
- Respuesta al frame 18549 (INVITE).

En el apartado Message Header, se observan los diferentes datos sobre la comunicación del frame.

- Datos del emisor (Terminal A(192.168.10.36)) del frame 18580 (STATUS - 200 OK).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que se envía el frame (192.168.10.36).

```

▼ Session Initiation Protocol (200)
  ▼ Status-Line: SIP/2.0 200 OK
    Status-Code: 200
    [Resent Packet: False]
    [Request Frame: 18549]
    [Response Time (ms): 6348]
  ▼ Message Header
    ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.HizgKHyI2;received=192.168.10.36;rport=54524
      Transport: UDP
      Sent-by Address: 192.168.10.36
      Sent-by port: 54524
      Branch: z9hG4bK.HizgKHyI2
      Received: 192.168.10.36
      RPort: 54524
    ▼ From: <sip:5206@192.168.10.35>;tag=14G-avb06
      ▼ SIP from address: sip:5206@192.168.10.35
        SIP from address User Part: 5206
        SIP from address Host Part: 192.168.10.35
        SIP from tag: 14G-avb06
    ▼ To: sip:5202@192.168.10.35;tag=as7d4da5df
      ▼ SIP to address: sip:5202@192.168.10.35
        SIP to address User Part: 5202
        SIP to address Host Part: 192.168.10.35
        SIP to tag: as7d4da5df
      Call-ID: lmWGaQqm10
    ▼ CSeq: 21 INVITE
      Sequence Number: 21
      Method: INVITE
    Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
    Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
    Supported: replaces, timer
    ▼ Contact: <sip:5202@192.168.10.35:5060>
      ▼ Contact URI: sip:5202@192.168.10.35:5060
        Contact URI User Part: 5202
        Contact URI Host Part: 192.168.10.35
        Contact URI Host Port: 5060
      Content-Type: application/sdp
      Content-Length: 237
  ▼ Message Body
    ▼ Session Description Protocol
      Session Description Protocol Version (v): 0
      ▼ Owner/Creator, Session Id (o): root 2002464001 2002464001 IN IP4 192.168.10.35
        Owner Username: root
        Session ID: 2002464001
        Session Version: 2002464001
        Owner Network Type: IN
        Owner Address Type: IP4
        Owner Address: 192.168.10.35
      Session Name (s): Asterisk PBX 11.16.0
      ▼ Connection Information (c): IN IP4 192.168.10.35
        Connection Network Type: IN
        Connection Address Type: IP4
        Connection Address: 192.168.10.35
      ▼ Time Description, active time (t): 0 0
        Session Start Time: 0
        Session Stop Time: 0
      ▼ Media Description, name and address (m): audio 12912 RTP/AVP 3 101
        Media Type: audio
        Media Port: 12912
        Media Protocol: RTP/AVP
        Media Format: GSM 06.10
        Media Format: DynamicRTP-Type-101
      ▼ Media Attribute (a): rtpmap:3 GSM/8000
        Media Attribute Fieldname: rtpmap
        Media Format: 3
        MIME Type: GSM
        Sample Rate: 8000
      ▼ Media Attribute (a): rtpmap:101 telephone-event/8000
        Media Attribute Fieldname: rtpmap
        Media Format: 101
        MIME Type: telephone-event
        Sample Rate: 8000
      ▼ Media Attribute (a): fmtp:101 0-16
        Media Attribute Fieldname: fmtp
        Media Format: 101 [telephone-event]
        Media format specific parameters: 0-16
      ▼ Media Attribute (a): ptimetime:20
        Media Attribute Fieldname: ptimetime
        Media Attribute Value: 20
      Media Attribute (a): sendrecv

```

Status: 200 OK

- Puerto desde el que se realiza la comunicación (54524).
- Datos del receptor (Terminal A (192.168.10.36)) del frame 18580.
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Dirección IP del que recibe el frame (192.168.10.36).
 - Puerto desde el que se realiza la comunicación (54524).
- Datos desde donde se envía la petición (From).
 - Protocolo que se usará para la comunicación (SIP).
 - Número del usuario (5206).
 - Dirección IP del servidor Asterisk (192.168.10.35).
- Datos del que recibe la petición (To).
 - Tipo de protocolo para el transporte de paquetes (UDP).
 - Número del usuario (5202).
 - Dirección IP del Servidor Asterisk (192.168.10.35).
 - Puerto que recibirá 5060.

En el apartado Message Body, se observan los diferentes datos sobre la comunicación del frame.

- Datos del creador (Owner/Creator).
 - Nombre del dispositivo (Asterisk PBX 11.16.0 (root))
 - Tipo de protocolo para realizar la comunicación (IPv4).
 - Dirección IP del que se envía el frame (192.168.10.35).
- Información sobre los codecs multimedia a usar (Media Description).
 - Tipo de comunicación multimedia (audio).
 - Puerto usado para la comunicación (12912).
 - Protocolo usado para la comunicación (RTP).
 - Codec a usar en la comunicación (GSM 06.10).
 - Dirección IP del servidor Asterisk (192.168.10.35).

STATUS: 200 OK Duplicado

```
▼ Session Initiation Protocol (200)
  ▼ Status-Line: SIP/2.0 200 OK
    Status-Code: 200
    [Resent Packet: True]
    [Suspected resend of frame: 18580]
    [Request Frame: 18549]
    [Response Time (ms): 6348]
```

STATUS: 200 OK Duplicado

Este frame es idéntico al anterior (STATUS - 200 OK), con la salvedad de que se trata de un reenvío del frame 18580 en respuesta a la petición del frame 18549 (INVITE).

STUN: Binding Request

```
▼ Session Traversal Utilities for NAT
  ▼ Message Type: 0x0001 (Binding Request)
    .... .0 ...0 .... = Message Class: 0x0000 Request (0)
    ..00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0 ..... .... .... = Message Method Assignment: IETF Review (0x0000)
  Message Length: 0
  Message Cookie: 2112a442
  Message Transaction ID: 40064a0970ed2302a4ed19f8
```

[STUN: Binding Request](#)

En el frame **18582** el Terminal B (192.168.10.34) realiza una petición (STUN¹⁰ Binding Request) hacia el Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal B para enviar este frame es el 7076 y el puerto que usa el servidor Asterisk para recibir este frame es el 13040. En el apartado Session Transversal Utilities for NAT (STUN), se activa el mensaje de Binding Request (petición de unión).

¹⁰STUN: Session Transversal Utilities for NAT

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.34:7076

```

▼ Session Traversal Utilities for NAT
  ▼ Message Type: 0x0101 (Binding Success Response)
    .... .1 ...0 .... = Message Class: 0x0010 Success Response (2)
    ..00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 12
    Message Cookie: 2112a442
    Message Transaction ID: 40064a0970ed2302a4ed19f8
  ▼ Attributes
    ▼ MAPPED-ADDRESS: 192.168.10.34:7076
      ▼ Attribute Type: MAPPED-ADDRESS (0x0001)
        0... .... .... .... = Attribute Type Comprehension: Required (0x0000)
        .0... .... .... .... = Attribute Type Assignment: IETF Review (0x0000)
        Attribute Length: 8
        Reserved: 00
        Protocol Family: IPv4 (0x01)
        Port: 7076
        IP: 192.168.10.34
  
```

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.34:7076

En el frame **18583** el Servidor Asterisk (192.168.10.35) responde al Terminal B (192.168.10.34) realiza una petición (STUN¹¹ Binding Success) hacia el. Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 13040 y el puerto que usa el terminal B para recibir este frame es el 7076. En el apartado Session Transversal Utilities for NAT (STUN), se activa el mensaje de Binding Success Response (Respuesta de unión exitosa) y el mapeo de la dirección ip y puerto del terminal B (192.168.10.34: 7076).

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.34:7076 Duplicado

```

▼ Session Traversal Utilities for NAT
  [Duplicated original message in: 18583]
  ▼ Message Type: 0x0101 (Binding Success Response)
    .... .1 ...0 .... = Message Class: 0x0010 Success Response (2)
    ..00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 12
    Message Cookie: 2112a442
    Message Transaction ID: 40064a0970ed2302a4ed19f8
  ▼ Attributes
    ▼ MAPPED-ADDRESS: 192.168.10.34:7076
      ▼ Attribute Type: MAPPED-ADDRESS (0x0001)
        0... .... .... .... = Attribute Type Comprehension: Required (0x0000)
        .0... .... .... .... = Attribute Type Assignment: IETF Review (0x0000)
        Attribute Length: 8
        Reserved: 00
        Protocol Family: IPv4 (0x01)
        Port: 7076
        IP: 192.168.10.34
  
```

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.34:7076 Duplicado

Frame (18583) Duplicado.

¹¹STUN: Session Transversal Utilities for NAT

RTCP (Terminal B (192.168.10.34:7077) - Servidor Asterisk (192.168.10.35:13041))

```
▼ User Datagram Protocol, Src Port: 7077 (7077), Dst Port: 13041 (13041)
  Source Port: 7077
  Destination Port: 13041
  Length: 28
  ▼ Checksum: 0x794f [validation disabled]
    [Good Checksum: False]
    [Bad Checksum: False]
    [Stream index: 1350]
  [RTCP frame length check: OK - 0 bytes]
```

[RTCP \(Terminal B \(192.168.10.34:7077\) - Servidor Asterisk \(192.168.10.35:13041\)\)](#)

En el frame **18585** el Terminal B (192.168.10.34) realiza una petición **RTCP**¹² hacia el Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal B para enviar este frame es el 7077 y el puerto que usa el servidor Asterisk para recibir este frame es el 13041.

CLASSIC STUN: Binding Request

```
▼ Simple Traversal of UDP Through NAT
  Message Type: Binding Request (0x0001)
  Message Length: 0x000c
  Message Transaction ID: 998d7564a6769c1facd08a19fe6c581f
  ▼ Attributes
    ▼ Attribute: USERNAME
      Attribute Type: USERNAME (0x0006)
      Attribute Length: 8
      Value: e0b10320c17f0000
```

[CLASSIC STUN: Binding Request](#)

En el frame **18586** el Servidor Asterisk (192.168.10.35) responde al Terminal B (192.168.10.34) realiza una petición (**CLASSIC STUN Binding Request**). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 13040 y el puerto que usa el terminal B para recibir este frame es el 7076. En el apartado Session Transversal Utilities for NAT (STUN), se activa el mensaje de Binding Request (Petición de unión).

CLASSIC STUN: Binding Request Duplicado

```
▼ Simple Traversal of UDP Through NAT
  Message Type: Binding Request (0x0001)
  Message Length: 0x000c
  Message Transaction ID: 998d7564a6769c1facd08a19fe6c581f
  ▼ Attributes
    ▼ Attribute: USERNAME
      Attribute Type: USERNAME (0x0006)
      Attribute Length: 8
      Value: e0b10320c17f0000
```

[CLASSIC STUN: Binding Request Duplicado](#)

En el frame **18587** el Servidor Asterisk (192.168.10.35) responde al Terminal B (192.168.10.34) realiza una petición (**CLASSIC STUN Binding Request**). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 13040 y el puerto que usa el terminal B para recibir este frame es el 7076. En el apartado Session Transversal Utilities for NAT (STUN), se activa el mensaje de Binding Request (Petición de unión).

¹²RTCP: Real-time Transport Control Protocol

RTCP (Servidor Asterisk (192.168.10.35:13041) - Terminal B (192.168.10.34:7077))

```
▼ User Datagram Protocol, Src Port: 13041 (13041), Dst Port: 7077 (7077)
  Source Port: 13041
  Destination Port: 7077
  Length: 40
  ▼ Checksum: 0x0000 (none)
    [Good Checksum: False]
    [Bad Checksum: False]
    [Stream index: 1428]
  [RTCP frame length check: OK - 0 bytes]
```

RTCP (Servidor Asterisk (192.168.10.35:13041) - Terminal B (192.168.10.34:7077))

En el frame **18588** el servidor Asterisk (192.168.10.35) realiza una petición **RTCP**¹³ hacia el Terminal B (192.168.10.34). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 13041 y el puerto que usa el Terminal B para recibir este frame es el 7077.

Request: ACK sip:5202@192.168.10.35:5060

```
▼ Session Initiation Protocol (ACK)
  ▼ Request-Line: ACK sip:5202@192.168.10.35:5060 SIP/2.0
    Method: ACK
    ▼ Request-URI: sip:5202@192.168.10.35:5060
      Request-URI User Part: 5202
      Request-URI Host Part: 192.168.10.35
      Request-URI Host Port: 5060
      [Resent Packet: False]
      [Request Frame: 18549]
      [Response Time (ms): 6379]
    ▼ Message Header
      ▼ Via: SIP/2.0/UDP 192.168.10.36:54524;rport;branch=z9hG4bK.UAXwXLdgQ
        Transport: UDP
        Sent-by Address: 192.168.10.36
        Sent-by port: 54524
        RPort: rport
        Branch: z9hG4bK.UAXwXLdgQ
      ▼ From: <sip:5206@192.168.10.35>;tag=14G-avb06
        ▼ SIP from address: sip:5206@192.168.10.35
          SIP from address User Part: 5206
          SIP from address Host Part: 192.168.10.35
          SIP from tag: 14G-avb06
      ▼ To: <sip:5202@192.168.10.35>;tag=as7d4da5df
        ▼ SIP to address: sip:5202@192.168.10.35
          SIP to address User Part: 5202
          SIP to address Host Part: 192.168.10.35
          SIP to tag: as7d4da5df
      ▼ CSeq: 21 ACK
        Sequence Number: 21
        Method: ACK
        Call-ID: lmWGaQqm10
        Max-Forwards: 70
      ▼ Authorization: Digest realm="asterisk", nonce="782aac88", algorithm=MD5, username="5206", uri="sip:5202@192.168.10.35", res...
        Authentication Scheme: Digest
        Realm: "asterisk"
        Nonce Value: "782aac88"
        Algorithm: MD5, username="5206"
        Authentication URI: "sip:5202@192.168.10.35"
        Digest Authentication Response: "6b18b00f994dd76de724c16457864bab"
        User-Agent: LinphoneAndroid/3.2.4 (belle-sip/1.5.0)
```

Request: ACK sip:5202@192.168.10.35:5060

En el frame **18592** el Terminal A (192.168.10.36) realiza una petición **ACK** hacia el servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el Terminal A para enviar este frame es el 54524 y el puerto que usa el servidor Asterisk para recibir este frame es el 5060.

En el apartado **Session Initiation Protocol**, existen varios subapartados de los cuales cabe destacar el Request-Line, en el que se especifican el tipo de frame (INVITE), el tipo de comuni-

¹³RTCP: Real-time Transport Control Protocol

cación (sip), el número del contacto (5202) y la ip del servidor Asterisk (192.168.10.35) al cual se le pregunta si existe este usuario en la base de datos.

En el apartado **Message Header**, hay varios subapartados que a continuación se explicarán. En el subapartado **Via** muestra el protocolo que se va a utilizar (SIP), el modo de transporte (UDP), la dirección IP del terminal A (192.168.10.36) que en su caso es el emisor de este paquete, el puerto de emisión (54524).

En el apartado **From**, se observa el protocolo usado para la comunicación (SIP), el número del usuario que realiza la llamada (5206) y la dirección IP del servidor Asterisk (192.168.10.35).

En el apartado **To**, se observa el protocolo usado para la comunicación (SIP), el número del usuario al cual se va a llamar (5202) y la dirección IP del servidor Asterisk (192.168.10.35).

En el apartado **Contact**, se muestran todos los datos del terminal A. Protocolo a usar (SIP), número del usuario (5206), dirección IP del terminal (192.168.10.36), puerto desde el cual se comunica (54524) y el tipo de transporte que se usa para la comunicación (UDP). También tenemos que tener en cuenta el subapartado **User-Agent**, en el que observamos el nombre del programa que usa el terminal para realizar la llamada (LimphoneAndroid/3.2.4)

STUN: Binding Request

```
▼ Session Traversal Utilities for NAT
  ▼ Message Type: 0x0001 (Binding Request)
    .... .0 ...0 .... = Message Class: 0x0000 Request (0)
    ..00 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 0
    Message Cookie: 2112a442
    Message Transaction ID: 654685b323afc706f57f19b1
```

STUN: Binding Request

En el frame **18593** el Terminal A (192.168.10.36) realiza una petición (**STUN**¹⁴ Binding Request) hacia el Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal A para enviar este frame es el 7076 y el puerto que usa el servidor Asterisk para recibir este frame es el 12912. En el apartado Session Transversal Utilities for NAT (STUN), se activa el mensaje de **Binding Request** (petición de unión).

RTCP (Terminal A (192.168.10.36:7077) - Servidor Asterisk (192.168.10.35:12913))

```
▼ User Datagram Protocol, Src Port: 7077 (7077), Dst Port: 12913 (12913)
  Source Port: 7077
  Destination Port: 12913
  Length: 28
  ▼ Checksum: 0x71d1 [validation disabled]
    [Good Checksum: False]
    [Bad Checksum: False]
    [Stream index: 1356]
    [RTCP frame length check: OK - 0 bytes]
```

RTCP (Terminal A (192.168.10.36:7077) - Servidor Asterisk (192.168.10.35:12913))

En el frame **18594** el Terminal A (192.168.10.36) realiza una petición **RTCP** hacia el Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte

¹⁴STUN: Session Transversal Utilities for NAT

es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el Terminal A para enviar este frame es el 7077 y el puerto que usa el servidor Asterisk para recibir este frame es el 12913.

RTCP (Servidor Asterisk (192.168.10.35:12913) - Terminal A (192.168.10.36:7077))

```
▼ User Datagram Protocol, Src Port: 12913 (12913), Dst Port: 7077 (7077)
  Source Port: 12913
  Destination Port: 7077
  Length: 40
  ▼ Checksum: 0x0000 (none)
    [Good Checksum: False]
    [Bad Checksum: False]
    [Stream index: 1430]
  [RTCP frame length check: OK - 0 bytes]
```

[RTCP \(Servidor Asterisk \(192.168.10.35:12913\) - Terminal A \(192.168.10.36:7077\)\)](#)

En el frame **18595** el servidor Asterisk (192.168.10.35) realiza una petición **RTCP** hacia el Terminal A (192.168.10.36) en respuesta del frame anterior. Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12913 y el puerto que usa el Terminal A para recibir este frame es el 7077.

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076

```
▼ Session Traversal Utilities for NAT
  ▼ Message Type: 0x0101 (Binding Success Response)
    ....1...0.... = Message Class: 0x0010 Success Response (2)
    ..00 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 12
    Message Cookie: 2112a442
    Message Transaction ID: 654685b323afc706f57f19b1
  ▼ Attributes
    ▼ MAPPED-ADDRESS: 192.168.10.36:7076
      ▼ Attribute Type: MAPPED-ADDRESS (0x0001)
        0... .... .... = Attribute Type Comprehension: Required (0x0000)
        .0. .... .... = Attribute Type Assignment: IETF Review (0x0000)
      Attribute Length: 8
      Reserved: 00
      Protocol Family: IPv4 (0x01)
      Port: 7076
      IP: 192.168.10.36
```

[STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076](#)

En el frame **18599** el Servidor Asterisk (192.168.10.35) responde al Terminal A (192.168.10.36) realiza una petición (**STUN Binding Response**) hacia el. Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12913 y el puerto que usa el Terminal A para recibir este frame es el 7076.

En el apartado **Session Transversal Utilities for NAT**(STUN), se activa el mensaje de Binding Success Response (Respuesta de unión exitosa) y el mapeo de la dirección ip y puerto del Terminal A (192.168.10.36: 7076).

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076 Duplicado

```
▼ Session Traversal Utilities for NAT
[Duplicated original message in: 18599]
  ▼ Message Type: 0x0101 (Binding Success Response)
    .... .1 ...0 .... = Message Class: 0x0010 Success Response (2)
    ..00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 12
    Message Cookie: 2112a442
    Message Transaction ID: 654685b323afc706f57f19b1
  ▼ Attributes
    ▼ MAPPED-ADDRESS: 192.168.10.36:7076
      ▼ Attribute Type: MAPPED-ADDRESS (0x0001)
        0... .... .... .... = Attribute Type Comprehension: Required (0x0000)
        .0.. .... .... .... = Attribute Type Assignment: IETF Review (0x0000)
        Attribute Length: 8
        Reserved: 00
        Protocol Family: IPv4 (0x01)
        Port: 7076
        IP: 192.168.10.36
```

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076 Duplicado

Frame **18600** es duplicado al frame 18599.

Comunicación entre los interlocutores RTP

18601 829.51011...	192.168.10.35	192.168.10.36	CLASSI...	74 Message: Binding Request
18602 829.51011...	192.168.10.35	192.168.10.35	CLASSI...	74 Message: Binding Request
18603 829.54312...	192.168.10.36	192.168.10.35	STUN	62 Binding Request
18604 829.54313...	192.168.10.36	192.168.10.35	RTCP	62 7077 → 12913 Len=20
18605 829.54322...	192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18606 829.54322...	192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18607 829.54325...	192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18608 829.54325...	192.168.10.35	192.168.10.36	RTCP	74 12913 → 7077 Len=32
18609 829.54328...	192.168.10.35	192.168.10.36	STUN	74 Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076
18610 829.54328...	192.168.10.35	192.168.10.36	STUN	74 Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076
18611 829.54333...	192.168.10.35	192.168.10.36	CLASSI...	74 Message: Binding Request
18612 829.54333...	192.168.10.35	192.168.10.36	CLASSI...	74 Message: Binding Request
18613 829.78346...	192.168.10.36	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x6AE792E1, Seq=0, Time=54400
18614 829.78369...	192.168.10.35	192.168.10.34	RTP	214 PT=ITU-T G.711 PCMA, SSRC=0x26D4D261, Seq=62781, Time=54400, Mark
18615 829.78369...	192.168.10.35	192.168.10.34	RTP	214 PT=ITU-T G.711 PCMA, SSRC=0x26D4D261, Seq=62781, Time=54400, Mark
18616 829.80510...	192.168.10.36	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x6AE792E1, Seq=1, Time=54560
18617 829.80522...	192.168.10.35	192.168.10.34	RTP	214 PT=ITU-T G.711 PCMA, SSRC=0x26D4D261, Seq=62782, Time=54560
18618 829.80522...	192.168.10.35	192.168.10.34	RTP	214 PT=ITU-T G.711 PCMA, SSRC=0x26D4D261, Seq=62782, Time=54560
18619 829.82385...	192.168.10.36	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x6AE792E1, Seq=2, Time=54720
18620 829.82403...	192.168.10.35	192.168.10.34	RTP	214 PT=ITU-T G.711 PCMA, SSRC=0x26D4D261, Seq=62783, Time=54720
18621 829.82403...	192.168.10.35	192.168.10.34	RTP	214 PT=ITU-T G.711 PCMA, SSRC=0x26D4D261, Seq=62783, Time=54720

Comunicación en la llamada

CLASSIC STUN: Binding Request

```
▼ Simple Traversal of UDP Through NAT
  Message Type: Binding Request (0x0001)
  Message Length: 0x000c
  Message Transaction ID: 421c0b6ebc0b19396c33fc6abfb01813
▼ Attributes
  ▼ Attribute: USERNAME
    Attribute Type: USERNAME (0x0006)
    Attribute Length: 8
    Value: 70960120c17f0000
```

CLASSIC STUN (Binding Request)

En el frame **18601** el Servidor Asterisk (192.168.10.35) responde al Terminal A (192.168.10.36) realiza una petición (**CLASSIC STUN Binding Request**). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12912 y el puerto que usa el terminal A para recibir este frame es el 7076.

En el apartado **Session Transversal Utilities for NAT** (STUN), se activa el mensaje de Binding Request (Petición de unión).

CLASSIC STUN: Binding Request Duplicado

```
▼ Simple Traversal of UDP Through NAT
  Message Type: Binding Request (0x0001)
  Message Length: 0x000c
  Message Transaction ID: 421c0b6ebc0b19396c33fc6abfb01813
▼ Attributes
  ▼ Attribute: USERNAME
    Attribute Type: USERNAME (0x0006)
    Attribute Length: 8
    Value: 70960120c17f0000
```

CLASSIC STUN: Binding Request

En el frame **18602** el Servidor Asterisk (192.168.10.35) responde al Terminal A (192.168.10.36) realiza una petición (**CLASSIC STUN Binding Request**). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12912 y el puerto que usa el terminal A para recibir este frame es el 7076.

En el apartado **Session Transversal Utilities for NAT** (STUN), se activa el mensaje de Binding Request (Petición de unión).

STUN: Binding Request

```
▼ Session Traversal Utilities for NAT
  ▶ Message Type: 0x0001 (Binding Request)
    Message Length: 0
    Message Cookie: 2112a442
    Message Transaction ID: a3e27c2d6ed5b424e6f684b1
```

[STUN \(Binding Request\)](#)

En el frame **18603** el Terminal A (192.168.10.36) realiza una petición (**STUN Binding Request**) hacia el Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal A para enviar este frame es el 7076 y el puerto que usa el servidor Asterisk para recibir este frame es el 12912.

En el apartado **Session Transversal Utilities for NAT** (STUN), se activa el mensaje de Binding Request (petición de unión).

RTCP (Terminal B (192.168.10.3:7077) - Servidor Asterisk (192.168.10.35:12913))

```
▼ User Datagram Protocol, Src Port: 7077 (7077), Dst Port: 12913 (12913)
  Source Port: 7077
  Destination Port: 12913
  Length: 28
  ▶ Checksum: 0x2860 [validation disabled]
    [Stream index: 1356]
  [RTCP frame length check: OK - 0 bytes]
```

[RTCP \(Terminal B \(192.168.10.3:7077\) - Servidor Asterisk \(192.168.10.35:12913\)\)](#)

En el frame **18604** el Terminal A (192.168.10.36) realiza una petición **RTCP** hacia el Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal A para enviar este frame es el 7077 y el puerto que usa el servidor Asterisk para recibir este frame es el 12913.

RTCP (Servidor Asterisk (192.168.10.35:12913) - Terminal B (192.168.10.36:7077))

```
▼ User Datagram Protocol, Src Port: 12913 (12913), Dst Port: 7077 (7077)
  Source Port: 12913
  Destination Port: 7077
  Length: 40
  ▶ Checksum: 0x0000 (none)
    [Stream index: 1430]
  [RTCP frame length check: OK - 0 bytes]
```

[RTCP \(Servidor Asterisk \(192.168.10.35:12913\) - Terminal B \(192.168.10.36:7077\)\)](#)

En el frame **18605** el Servidor Asterisk (192.168.10.35) realiza una petición **RTCP** hacia el Terminal A (192.168.10.36). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12913 y el puerto que usa el servidor Asterisk para recibir este frame es el 7077.

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076

```
▼ Session Traversal Utilities for NAT
  ▼ Message Type: 0x0101 (Binding Success Response)
    .... .1 ...0 .... = Message Class: 0x0010 Success Response (2)
    ..00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 12
    Message Cookie: 2112a442
    Message Transaction ID: a3e27c2d6ed5b424e6f684b1
  ▼ Attributes
    ▼ MAPPED-ADDRESS: 192.168.10.36:7076
      ▶ Attribute Type: MAPPED-ADDRESS (0x0001)
      Attribute Length: 8
      Reserved: 00
      Protocol Family: IPv4 (0x01)
      Port: 7076
      IP: 192.168.10.36
```

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076

En el frame **18609** el Servidor Asterisk (192.168.10.35) realiza una petición (**STUN Binding Request**) hacia Terminal A (192.168.10.36). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk es el 12912.

En el apartado **Session Transversal Utilities for NAT** (STUN), se activa el mensaje de Binding Success Response (Respuesta éxito de unión). El mapeo de la

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076 Duplicado

```
▼ Session Traversal Utilities for NAT
  [Duplicated original message in: 18609]
  ▼ Message Type: 0x0101 (Binding Success Response)
    .... .1 ...0 .... = Message Class: 0x0010 Success Response (2)
    ..00 000. 000. 0001 = Message Method: 0x0001 Binding (0x001)
    ..0. .... .... .... = Message Method Assignment: IETF Review (0x0000)
    Message Length: 12
    Message Cookie: 2112a442
    Message Transaction ID: a3e27c2d6ed5b424e6f684b1
  ▼ Attributes
    ▼ MAPPED-ADDRESS: 192.168.10.36:7076
      ▶ Attribute Type: MAPPED-ADDRESS (0x0001)
      Attribute Length: 8
      Reserved: 00
      Protocol Family: IPv4 (0x01)
      Port: 7076
      IP: 192.168.10.36
```

STUN: Binding Success Response MAPPED-ADDRESS: 192.168.10.36:7076 Duplicado

Frame duplicado del frame original (18609).

CLASSIC STUN: Binding Request

```
▼ Simple Traversal of UDP Through NAT
  Message Type: Binding Request (0x0001)
  Message Length: 0x000c
  Message Transaction ID: c9aa1f01c1aeb205c0c5b95e105df556
  ▼ Attributes
    ▼ Attribute: USERNAME
      Attribute Type: USERNAME (0x0006)
      Attribute Length: 8
      Value: 70960120c17f0000
```

CLASSIC STUN: Binding Request

En el frame **18611** el Servidor Asterisk (192.168.10.35) responde al Terminal A (192.168.10.36) realiza una petición (**CLASSIC STUN Binding Request**). Se usa como protocolo de internet IPv4, y el protocolo de transporte es UDP, posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12912 y el puerto que usa el terminal A para recibir este frame es el 7076. En el apartado **Session Transversal Utilities for NAT** (STUN), se activa el mensaje de Binding Request (Petición de unión).

CLASSIC STUN: Binding Request Duplicado

```

▼ Simple Traversal of UDP Through NAT
  Message Type: Binding Request (0x0001)
  Message Length: 0x000c
  Message Transaction ID: c9aa1f01c1aeb205c0c5b95e105df556
  ▼ Attributes
    ▼ Attribute: USERNAME
      Attribute Type: USERNAME (0x0006)
      Attribute Length: 8
      Value: 70960120c17f0000
  
```

CLASSIC STUN: Binding Request Duplicado

En el frame **18612** el Servidor Asterisk (192.168.10.35) responde al Terminal A (192.168.10.36) realiza una petición (**CLASSIC STUN Binding Request**). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 12912 y el puerto que usa el terminal A para recibir este frame es el 7076. En el apartado **Session Transversal Utilities for NAT (STUN)**, se activa el mensaje de Binding Request (Petición de unión).

RTP (Terminal A (192.168.10.36) - Servidor Asterisk (192.168.10.35))

```

▼ Real-Time Transport Protocol
  ▼ [Stream setup by SDP (frame 18581)]
    [Setup frame: 18581]
    [Setup Method: SDP]
    10.. .... = Version: RFC 1889 Version (2)
    ..0. .... = Padding: False
    ..0 .. .... = Extension: False
    .... 0000 = Contributing source identifiers count: 0
    0... .... = Marker: False
    Payload type: GSM 06.10 (3)
    Sequence number: 0
    [Extended sequence number: 65536]
    Timestamp: 54400
    Synchronization Source identifier: 0x6ae792e1 (1793561313)
    Payload: d820a2e15a50004924924924500049249245000492492...
  
```

RTP (Terminal A (192.168.10.36) - Servidor Asterisk (192.168.10.35))

En el frame **18613** el Terminal A (192.168.10.36) responde al Servidor Asterisk (192.168.10.35) realiza una petición **RTP**. Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal A para enviar este frame es el 7076 y el puerto que usa el servidor Asterisk para recibir este frame es el 12912.

En el apartado **Real-Time Transport Protocol**, se observa que se trata de una respuesta de la petición de streaming en el frame 18581.

RTP (Servidor Asterisk (192.168.10.35) - Terminal B (192.168.10.34))

```
▼ Real-Time Transport Protocol
  ▼ [Stream setup by SDP (frame 18552)]
    [Setup Frame: 18552]
    [Setup Method: SDP]
    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)
    Sequence number: 62781
    [Extended sequence number: 62781]
    Timestamp: 54400
    Synchronization Source identifier: 0x26d4d261 (651481697)
    Payload: d5d5d5d5d5d5d4d5d5d4d4d5d4d4d4d4d4d4d4d4d4...
```

RTP (Servidor Asterisk (192.168.10.35) - Terminal B (192.168.10.34))

En el frame **18614** el Servidor Asterisk (192.168.10.35) responde al Terminal B (192.168.10.34) realiza una petición **RTP**. Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 13040 y el puerto que usa el terminal B para recibir este frame es el 7076.

En el apartado **Real-Time Transport Protocol**, se observa que se trata de una respuesta de la petición de streaming en el frame 18552.

RTP (Servidor Asterisk (192.168.10.35) - Terminal B (192.168.10.34))

```
▼ Real-Time Transport Protocol
  ▼ [Stream setup by SDP (frame 18552)]
    [Setup Frame: 18552]
    [Setup Method: SDP]
    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)
    Sequence number: 62781
    [Extended sequence number: 62781]
    Timestamp: 54400
    Synchronization Source identifier: 0x26d4d261 (651481697)
    Payload: d5d5d5d5d5d5d4d5d5d4d4d5d4d4d4d4d4d4d4d4...
```

RTP (Servidor Asterisk (192.168.10.35) - Terminal B (192.168.10.34))

En el frame **18614** el Servidor Asterisk (192.168.10.35) responde al Terminal B (192.168.10.34) realiza una petición **RTP**. Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 13040 y el puerto que usa el terminal B para recibir este frame es el 7076.

En el apartado **Real-Time Transport Protocol**, se observa que se trata de una respuesta de la petición de streaming en el frame 18552.

Finalización de la llamada

21701 839.58981...	192.168.10.34	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x914716C7, Seq=490, Time=82160
21702 839.59020...	192.168.10.35	192.168.10.36	RTP	87 PT=GSM 06.10, SSRC=0x45308019, Seq=985, Time=82160
21703 839.59020...	192.168.10.35	192.168.10.36	RTP	87 PT=GSM 06.10, SSRC=0x45308019, Seq=985, Time=82160
21704 839.59897...	192.168.10.36	192.168.10.35	SIP	523 Request: BYE sip:5202@192.168.10.35:5060
21705 839.59953...	192.168.10.35	192.168.10.36	SIP	468 Status: 200 OK
21706 839.59953...	192.168.10.35	192.168.10.36	SIP	468 Status: 200 OK
21707 839.60500...	192.168.10.35	192.168.10.34	SIP	513 Request: BYE sip:5202@192.168.10.34:45910;transport=udp
21708 839.60501...	192.168.10.35	192.168.10.34	SIP	513 Request: BYE sip:5202@192.168.10.34:45910;transport=udp
21709 839.60973...	192.168.10.34	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x914716C7, Seq=491, Time=82320
21710 839.63064...	192.168.10.34	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x914716C7, Seq=492, Time=82480
21711 839.65062...	192.168.10.34	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x914716C7, Seq=493, Time=82640
21712 839.67051...	192.168.10.34	192.168.10.35	RTP	87 PT=GSM 06.10, SSRC=0x914716C7, Seq=494, Time=82800
21713 839.68725...	192.168.10.34	192.168.10.35	SIP	398 Status: 200 Ok

Finalización de la llamada

Request: BYE sip:5202@192.168.10.35:5060

```
▼ Session Initiation Protocol (BYE)
► Request-Line: BYE sip:5202@192.168.10.35:5060 SIP/2.0
▼ Message Header
► Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.Hgf208omJ;rport
► From: <sip:5202@192.168.10.35>;tag=14G-avb06
► To: <sip:5202@192.168.10.35>;tag=as7d4da5df
► CSeq: 22 BYE
  Call-ID: 1mWGaqQm10
  Max-Forwards: 70
  User-Agent: LinphoneAndroid/3.2.4 (belle-sip/1.5.0)
► Authorization: Digest realm="asterisk", nonce="782aac88", algorithm=MD5, username="5206", uri="sip:5202@192.168.10.35:5060", response="d0b90aa31d5b353ce46a2d..."
```

Request: BYE sip:5202@192.168.10.35:5060

En el frame **21704** el Terminal A (192.168.10.36) manda una petición "**BYE**" para terminar la llamada al Servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal A para enviar este frame es el 54524 y el puerto que usa el servidor Asterisk para recibir este frame es el 5060.

En el apartado **Session Initiation Protocol (BYE)**, se observa que se trata de una petición de finalización de la llamada por parte del terminal A. El servidor Asterisk recibe esta petición por el puerto predefinido 5060.

STATUS: 200 OK

```
▼ Session Initiation Protocol (200)
► Status-Line: SIP/2.0 200 OK
▼ Message Header
► Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.Hgf208omJ;received=192.168.10.36;rport=54524
► From: <sip:5206@192.168.10.35>;tag=14G-avb06
► To: <sip:5202@192.168.10.35>;tag=as7d4da5df
► Call-ID: 1mWGaqQm10
► CSeq: 22 BYE
  Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
  Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
  Supported: replaces, timer
  Content-Length: 0
```

STATUS: 200 OK

En el frame **21705** el servidor Asterisk (192.168.10.35) envía una respuesta (**STATUS: 200 OK**) al anterior frame (Request: BYE sip:5202@192.168.10.35:5060) al Terminal A (192.168.10.36). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal A para recibir este frame es el 54524.

En el apartado **Session Initiation Protocol (200)**, se observa que se trata de la respuesta de la petición de finalización de la llamada por parte del terminal A.

STATUS: 200 OK Duplicado

```
▼ Session Initiation Protocol (200)
  ▶ Status-Line: SIP/2.0 200 OK
  ▼ Message Header
    ▶ Via: SIP/2.0/UDP 192.168.10.36:54524;branch=z9hG4bK.Hgf208omJ;received=192.168.10.36;rport=54524
    ▶ From: <sip:5202@192.168.10.35>;tag=14c-avb06
    ▶ To: <sip:5202@192.168.10.35>;tag=as7d4da5df
    ▶ Call-ID: 1mWgAqqm10
    ▶ CSeq: 22 BYE
    ▶ Server: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
    ▶ Allow: INVITE, ACK, CANCEL, OPTIONS, BYE, REFER, SUBSCRIBE, NOTIFY, INFO, PUBLISH, MESSAGE
    ▶ Supported: replaces, timer
    ▶ Content-Length: 0
```

STATUS: 200 OK Duplicado

En el frame **21706** el servidor Asterisk (192.168.10.35) envía una respuesta (**STATUS: 200 OK**) al anterior frame (Request: BYE sip:5202@192.168.10.35:5060) al Terminal A (192.168.10.36). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal A para recibir este frame es el 54524.

En el apartado **Session Initiation Protocol (200)**, se observa que se trata de la respuesta de la petición de finalización de la llamada por parte del terminal A.

Request: BYE sip:5202@192.168.10.34:45910

```
▼ Session Initiation Protocol (BYE)
  ▶ Request-Line: BYE sip:5202@192.168.10.34:45910;transport=udp SIP/2.0
  ▼ Message Header
    ▶ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK6caa86e5
    ▶ Max-Forwards: 70
    ▶ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
    ▶ To: <sip:5202@192.168.10.34:45910;transport=udp>;tag=fEShCgR
    ▶ Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▶ CSeq: 103 BYE
    ▶ User-Agent: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
  ▶ X-Asterisk-HangupCause: Normal Clearing
  ▶ [Expert Info (Note/Undecoded): Unrecognised SIP header (x-asterisk-hangupcause)]
  ▶ X-Asterisk-HangupCauseCode: 16
  ▶ Content-Length: 0
```

Request: BYE sip:5202@192.168.10.34:45910

En el frame **21707** el servidor Asterisk (192.168.10.35) envía una petición "**BYE**" para terminar la llamada al Terminal B (192.168.10.34). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal B para recibir este frame es el 45910.

En el apartado **Session Initiation Protocol (BYE)**, se observa que se trata de una petición de finalización de la llamada por parte del servidor Asterisk. El terminal B recibe esta petición por el puerto 45910.

Request: BYE sip:5202@192.168.10.34:45910 Duplicado

```
▼ Session Initiation Protocol (BYE)
  ▶ Request-Line: BYE sip:5202@192.168.10.34:45910;transport=udp SIP/2.0
  ▼ Message Header
    ▶ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK6caa86e5
    ▶ Max-Forwards: 70
    ▶ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f0bd
    ▶ To: <sip:5202@192.168.10.34:45910;transport=udp>;tag=fEShCgR
    ▶ Call-ID: 0097d6bc241055d122ff9ddc4225b5f2@192.168.10.35:5060
    ▶ CSeq: 103 BYE
    ▶ User-Agent: FPBX-AsteriskNOW-12.0.76.4(11.16.0)
  ▶ X-Asterisk-HangupCause: Normal Clearing
  ▶ X-Asterisk-HangupCauseCode: 16
  ▶ Content-Length: 0
```

Request: BYE sip:5202@192.168.10.34:45910 Duplicado

En el frame **21708** el servidor Asterisk (192.168.10.35) envía una petición "**BYE**" para terminar la llamada al Terminal B (192.168.10.34). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el servidor Asterisk para enviar este frame es el 5060 y el puerto que usa el terminal B para recibir este frame es el 45910.

En el apartado **Session Initiation Protocol** (BYE), se observa que se trata de una petición de finalización de la llamada por parte del servidor Asterisk. El terminal B recibe esta petición por el puerto 45910.

STATUS: 200 OK

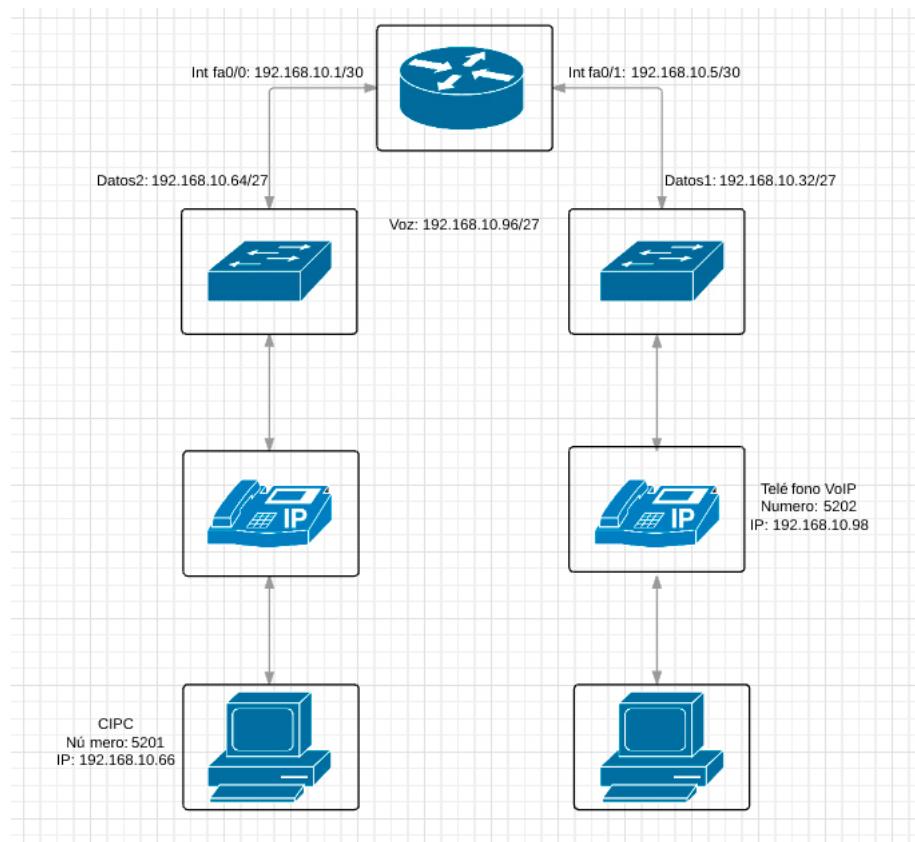
```
▼ Session Initiation Protocol (200)
  ▷ Status-Line: SIP/2.0 200 Ok
  ▷ Message-Header
    ▷ Via: SIP/2.0/UDP 192.168.10.35:5060;branch=z9hG4bK6caa86e5
    ▷ From: "Alpha" <sip:5206@192.168.10.35>;tag=as2a70f6bd
    ▷ To: <sip:5202@192.168.10.34:45910;transport=udp>;tag=fEShCgR
    ▷ Call-ID: 0097d6bc241055d122ff9dc4225b5f2@192.168.10.35:5060
    ▷ CSeq: 103 BYE
    ▷ User-Agent: LinphoneAndroid/3.2.1 (belle-sip/1.5.0)
    ▷ Supported: replaces, outbound
```

STATUS: 200 OK

En el frame **21713** el terminal B (192.168.10.34) envía una respuesta (**STATUS: 200 OK**) al anterior frame (Request: BYE sip:5202@192.168.10.35:5060) al servidor Asterisk (192.168.10.35). Se usa como protocolo de internet IPv4 y el protocolo de transporte es UDP. Posee un tamaño de cabecera de 20 bytes. El puerto que usa el terminal B para enviar este frame es el 45910 y el puerto que usa el servidor Asterisk para recibir este frame es el 5060.

En el apartado **Session Initiation Protocol** (200), se observa que se trata de la respuesta de la petición de finalización de la llamada por parte del terminal B.

Anexo V: Monitorización protocolo SCCP Wireshark



Red a monitorizar

19 14.376466	192.168.10.66	192.168.10.65	SKINNY	78 SoftKeyEvent
20 14.382035	192.168.10.65	192.168.10.66	SKINNY	90 CallState
21 14.384261	192.168.10.65	192.168.10.66	SKINNY	74 ClearPromptStatus
22 14.384318	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=25 Ack=57 Win=63572 Len=0
23 14.386887	192.168.10.65	192.168.10.66	SKINNY	82 SelectSoftKeys
24 14.388883	192.168.10.65	192.168.10.66	SKINNY	78 SetLamp
25 14.388929	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=25 Ack=109 Win=63520 Len=0
26 14.390882	192.168.10.65	192.168.10.66	SKINNY	70 SetSpeakerMode
27 14.402632	192.168.10.65	192.168.10.66	SKINNY	82 DisplayPromptStatusV2
28 14.402682	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=25 Ack=153 Win=63476 Len=0
29 14.404969	192.168.10.65	192.168.10.66	SKINNY	82 StartTone
30 14.588256	192.168.10.66	192.168.10.65	SKINNY	74 MediaPathEvent
31 14.789425	192.168.10.65	192.168.10.66	TCP	60 2000 → 49218 [ACK] Seq=181 Ack=45 Win=3796 Len=0
32 14.789496	192.168.10.66	192.168.10.65	SKINNY	102 KeypadButton KeypadButton
33 14.794625	192.168.10.65	192.168.10.66	SKINNY	78 StopTone
34 14.868466	192.168.10.66	192.168.10.65	SKINNY	78 KeypadButton
35 15.069517	192.168.10.65	192.168.10.66	TCP	60 2000 → 49218 [ACK] Seq=205 Ack=117 Win=3724 Len=0
36 15.069583	192.168.10.66	192.168.10.65	SKINNY	78 KeypadButton
37 15.073995	192.168.10.65	192.168.10.66	SKINNY	98 DialedNumber
38 15.075817	192.168.10.65	192.168.10.66	SKINNY	90 CallState
39 15.075856	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=141 Ack=285 Win=63344 Len=0
40 15.078903	192.168.10.65	192.168.10.66	SKINNY	90 CallState
41 15.080208	192.168.10.65	192.168.10.66	SKINNY	82 DisplayPromptStatusV2
42 15.080243	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=141 Ack=349 Win=63280 Len=0
43 15.082428	192.168.10.65	192.168.10.66	SKINNY	82 SelectSoftKeys
44 15.084492	192.168.10.65	192.168.10.66	SKINNY	78 SetLamp
45 15.084544	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=141 Ack=401 Win=63228 Len=0
46 15.086781	192.168.10.65	192.168.10.66	SKINNY	178 CallInfoV2
47 15.102665	192.168.10.65	192.168.10.66	SKINNY	78 SetLamp
48 15.102723	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=141 Ack=549 Win=63080 Len=0
49 15.124129	192.168.10.65	192.168.10.66	SKINNY	82 StartTone
50 15.178755	192.168.10.65	192.168.10.66	SKINNY	178 CallInfoV2
51 15.178831	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=141 Ack=701 Win=62928 Len=0

SCCP flow

Los frames que se intercambian más comunes en una comunicación SCCP, son los siguientes:

		Características	
Skinny Client Control Protocol	Data length: 16 Header version: Basic (0x00000000) Message ID: SoftKeyEvent (38) softKeyEvent: NewCall (0x00000002) lineInstance: 0 callReference: 0	19 - SKINNY: SoftKeyEvent (NewCall) -> Evento (Nueva llamada). Se va a realizar una nueva llamada.	
Skinny Client Control Protocol	Data length: 28 Header version: Basic (0x00000000) Message ID: CallState (273) callState: OffHook (0x00000001) lineInstance: 1 callReference: 32 privacy: None (0x00000000) precedence	20 - SKINNY: CallState (OffHook) -> El CallManager envía un frame de estado de la llamada (OffHook (Descuelgue de auricular)). El usuario ha descolgado el auricular del teléfono.	
Skinny Client Control Protocol	Data length: 12 Header version: Basic (0x00000000) Message ID: ClearPromptStatus (275) lineInstance: 1 callReference: 32	21 - SKINNY: ClearPromptStatus -> El Callmanager envía un frame para borrar el estado de ambos dispositivos (teléfono IP/aplicación voip).	

Características	
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: SelectSoftKeys (272) lineInstance: 1 callReference: 32 softKeySetIndex: Off Hook (0x00000004) ▼ validKeyMask 1..... = SoftKey1: Yes 1..... = SoftKey2: Yes 1..... = SoftKey3: Yes 1..... = SoftKey4: Yes 1..... = SoftKey5: Yes 1..... = SoftKey6: Yes 1..... = SoftKey7: Yes 1..... = SoftKey8: Yes 1..... = SoftKey9: Yes 1..... = SoftKey10: Yes 1..... = SoftKey11: Yes 1..... = SoftKey12: Yes 1..... = SoftKey13: Yes 1..... = SoftKey14: Yes 1..... = SoftKey15: Yes 1..... = SoftKey16: Yes </pre>	23 - SKINNY: SelectSoftKeys (OffHook) -> El CallManager envía un frame que activa las opciones disponibles (OffHook (Descuelgue de auricular)) en los dispositivos (teléfono IP/aplicación VoIP). .
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: SetLamp (134) stimulus: Line (0x00000009) stimulusInstance: 1 lampMode: On (0x00000002) </pre>	24 - SKINNY: SetLamp -> El CallManager envía un frame que activa la luz (roja) que lleva el auricular en su parte trasera.
<pre> Skinny Client Control Protocol Data length: 8 Header version: Basic (0x00000000) Message ID: SetSpeakerMode (136) speakerMode: On (0x00000001) </pre>	26 - SKINNY: SetSpeakerMode (On) -> El CallManager envía un frame para activar el auricular del teléfono.
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: DisplayPromptStatusV2 (325) timeOutValue: 0 lineInstance: 1 callReference: 32 promptStatus: \357\277\275 => "Enter Number" </pre>	27 - SKINNY: DisplayPromptStatusV2 -> El CallManager envía un frame para actualizar el estado del dispositivo (teléfono IP/aplicación VoIP). Visualiza por la pantalla del teléfono el mensaje; "Enter Number" ("Introducir número").
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: StartTone (130) tone: InsideDialTone (0x00000021) tone_output_direction: User (0x00000000) lineInstance: 1 callReference: 32 </pre>	29 - SKINNY: StartTone (InsideDialTone) -> El CallManager envía un frame para activar el tono continuo de dial e iniciar el marcado de números.
<pre> Skinny Client Control Protocol Data length: 12 Header version: Basic (0x00000000) Message ID: MediaPathEvent (73) mediaPathID: Handset (0x00000002) mediaPathEvent: Off (0x00000002) </pre>	30 - SKINNY: MediaPathEvent (Handset (Off)) -> El CallManager envía un frame desactivando el auricular.

	Características
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: KeypadButton (3) kpButton: Five (0x00000005) lineInstance: 1 callReference: 32 </pre>	32 - SKINNY: KeypadButton -> Se dispone a pulsar números en el teclado para realizar la llamada. En este caso se está pulsando los números 5 y seguidamente se pulsa el número 2.
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: KeypadButton (3) kpButton: Two (0x00000002) lineInstance: 1 callReference: 32 </pre>	
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: StopTone (131) lineInstance: 1 callReference: 32 </pre>	33 - SKINNY: StopTone -> El CallManager envía un frame para parar el tono continuo cuando se haya pulsado el primer número.
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: KeypadButton (3) kpButton: Zero (0x00000000) lineInstance: 1 callReference: 32 </pre>	34 - SKINNY: KeypadButton -> Se continua pulsando números para poder realizar la llamada. En este caso es 0.
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: KeypadButton (3) kpButton: Two (0x00000002) lineInstance: 1 callReference: 32 </pre>	36 - SKINNY: KeypadButton -> Se continua pulsando números para poder realizar la llamada. En este caso es 2.
<pre> Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: DialedNumber (285) dialedNumber: 5202 lineInstance: 1 callReference: 32 </pre>	37 - SKINNY: DialedNumber -> El CallManager envía un frame para conectar los dos números e iniciar la llamada. Se dispone a conectarse con el número (5202) al que se está llamando.
<pre> Skinny Client Control Protocol Data length: 28 Header version: Basic (0x00000000) Message ID: CallState (273) callState: Proceed (0x0000000c) lineInstance: 1 callReference: 32 privacy: None (0x00000000) ▼ precedence precedenceLevel: 4 precedenceDomain: 0 </pre>	38 - SKINNY: CallState (Proceed) -> El CallManager envía un frame actualizando el estado de los dispositivos (teléfonos IP/aplicación VoIP). Cambia el estado de la llamada a Proceed (procediendo a llamar).
<pre> Skinny Client Control Protocol Data length: 28 Header version: Basic (0x00000000) Message ID: CallState (273) callState: RingOut (0x00000003) lineInstance: 1 callReference: 32 privacy: None (0x00000000) ▼ precedence precedenceLevel: 4 precedenceDomain: 0 </pre>	40 - SKINNY: CallState (RingOut) -> El CallManager envía un frame actualizando el estado de los dispositivos (teléfonos IP/aplicación VoIP). Cambia el estado de la llamada a RingOut (Llamando).

Características	
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: DisplayPromptStatusV2 (325) timeOutValue: 0 lineInstance: 1 callReference: 32 promptStatus: \357\277\275\026 => "Ring Out" </pre>	41 - SKINNY: DisplayPromptStatusV2 (RingOut) -> El CallManager envía un frame para actualizar el estado de los dispositivos (teléfono IP/aplicación VoIP), al estado de RingOut (Llamando). Visualizando por pantalla del dispositivo (teléfono IP/aplicación VoIP) el texto "RingOut" (Llamando).
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: SelectSoftkeys (272) lineInstance: 1 callReference: 32 softKeySetIndex: Ring Out (0x00000008) ▼ validKeyMask 1 = SoftKey1: Yes 1 = SoftKey2: Yes 1 = SoftKey3: Yes 1 = SoftKey4: Yes 1 = SoftKey5: Yes 1 = SoftKey6: Yes 1 = SoftKey7: Yes 1 = SoftKey8: Yes 1 = SoftKey9: Yes 1 = SoftKey10: Yes 1 = SoftKey11: Yes 1 = SoftKey12: Yes 1 = SoftKey13: Yes 1 = SoftKey14: Yes 1 = SoftKey15: Yes 1 = SoftKey16: Yes </pre>	43 - SKINNY: SelectSoftKeys (RingOut) -> El CallMANager envía un frame que activa las opciones disponibles en el teléfono IP/aplicación voip. Se activa la opción RingOut (Llamando).
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: SetLamp (134) stimulus: Line (0x00000009) stimulusInstance: 1 lampMode: On (0x00000002) </pre>	44 - SKINNY: SetLamp (ON) -> El CallManager envía un frame manteniendo la alerta de luz (roja) de llamada activa y fija.
<pre> Skinny Client Control Protocol Data length: 116 Header version: Basic (0x00000000) Message ID: CallInfoV2 (330) lineInstance: 1 callReference: 32 callType: OutboundCall (0x00000002) originatingCallingReason: 0 lastRedirectionReason: 0 callInstance: 1 callSecurityStatus: NotAuthenticated (0x00000001) ▼ partyPIRestrictionBits 0 = CallingPartyName: No 0 = CallingPartyNumber: No 0 = CalledPartyName: No 0 = CalledPartyNumber: No 0 = CalledParty: No 0 = OriginalCalledPartyName: No 0 = OriginalCalledPartyNumber: No 0 = LastRedirectingPartyName: No 0 = LastRedirectingPartyNumber: No 0 = LastRedirectPartyName: No 0 = LastRedirectPartyNumber: No 0 = LastRedirectParty: No 0 = BitsReserved: No callingParty: 5201 alternateCallingParty: 5202 calledParty: 5202 lastRedirectingParty: 5201 cpnvoiceMailbox: 5202 cdpnvoiceMailbox: 5202 originalCdpnVoiceMailbox: 5202 lastRedirectingVoiceMailbox: Ordenador callingPartyName: Telefono_cisco calledPartyName: Telefono_cisco </pre>	46 - SKINNY: CallInfoV2 (OutBound-Call) -> El CallManager envía un frame con información relevante sobre la llamada que se va a realizar. Número 5201 (Ordenador) realizando la comunicación con el número 5202 (Teléfono_cisco).
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: SetLamp (134) stimulus: Line (0x00000009) stimulusInstance: 1 lampMode: On (0x00000002) </pre>	47 - SKINNY: SetLamp (ON) -> El CallManager envía un frame manteniendo la alerta de luz (roja) de llamada activa y fija.
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: StartTone (130) tone: AlertingTone (0x00000024) tone_output_direction: User (0x00000000) lineInstance: 1 callReference: 32 </pre>	49 - SKINNY: StartTone (Alerting-Tone) -> El CallManager envía un frame para activar el tono de llamada a los dispositivos (teléfono IP/aplicación voip). Dando tono en el número 5202 (Teléfono_cisco).

Características
<pre> Skinny Client Control Protocol Data length: 116 Header version: Basic (0x00000000) Message type: CallInfoV2 (330) lineInstance: 1 callReference: 32 callType: OutboundCall (0x00000002) originalCdpnRedirectReason: 0 lastRedirectingReason: 0 callInstance: callSecurityStatus: NotAuthenticated (0x00000001) ▼ partyPIRRestrictionBits 0..... = CallingPartyName: No 0..... = CallingPartyNumber: No 0..... = CalledPartyName: No 0..... = CalledPartyNumber: No 0..... = CalledParty: No 0..... = OriginalCalledPartyName: No 0..... = OriginalCalledPartyNumber: No 0..... = OriginalCalledParty: No 0..... = LastRedirectPartyName: No 0..... = LastRedirectPartyNumber: No 0..... = LastRedirectParty: No 0..... = BitsReserved: No callingParty: 5201 AlternateCallingParty: 5202 calledParty: 5202 lastRedirectingParty: 5201 cdpnVoiceMailbox: 5202 cdpnVoiceMailbox: 5202 originalCdpnVoiceMailbox: 5202 lastRedirectingVoiceMailbox: Ordenador callingPartyName: Telefono cisco calledPartyName: Telefono_cisco </pre>

62 21.178375	192.168.10.65	192.168.10.66	SKINNY	70 ClearPriNotify
63 21.180391	192.168.10.65	192.168.10.66	SKINNY	70 ClearPriNotify
64 21.180458	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=141 Ack=733 Win=62896 Len=0
65 21.182634	192.168.10.65	192.168.10.66	SKINNY	66 ClearNotify
66 21.184655	192.168.10.65	192.168.10.66	SKINNY	98 DisplayPromptStatusV2
67 21.184722	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=141 Ack=789 Win=62840 Len=0
68 21.657533	192.168.10.65	192.168.10.66	SKINNY	90 CallState
69 21.659424	192.168.10.65	192.168.10.66	SKINNY	82 DisplayPromptStatusV2
70 21.659478	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=141 Ack=853 Win=64240 Len=0
71 21.661775	192.168.10.65	192.168.10.66	SKINNY	82 SelectSoftKeys
72 21.664099	192.168.10.65	192.168.10.66	SKINNY	78 SetLamp
73 21.664174	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=141 Ack=905 Win=64188 Len=0
74 21.666368	192.168.10.65	192.168.10.66	SKINNY	78 StopTone
75 21.694532	192.168.10.65	192.168.10.66	SKINNY	98 ConnectionStatisticsReq
76 21.694598	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=141 Ack=973 Win=64120 Len=0
77 21.696804	192.168.10.65	192.168.10.66	SKINNY	98 ConnectionStatisticsReq
78 21.698729	192.168.10.65	192.168.10.66	SKINNY	174 OpenReceiveChannel
79 21.698780	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=141 Ack=1137 Win=63956 Len=0
80 21.701162	192.168.10.65	192.168.10.66	SKINNY	78 SetLamp
81 21.703828	192.168.10.66	192.168.10.65	SKINNY	86 OpenReceiveChannelAck
82 21.709310	192.168.10.65	192.168.10.66	SKINNY	178 CallInfoV2
83 21.794085	192.168.10.65	192.168.10.66	SKINNY	78 StopTone
84 21.794159	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=173 Ack=1309 Win=63784 Len=0
85 21.796484	192.168.10.65	192.168.10.66	SKINNY	182 StartMediaTransmission
86 21.796507	192.168.10.66	192.168.10.65	SKINNY	66 KeepAlive
87 21.799919	192.168.10.65	192.168.10.66	SKINNY	66 KeepAliveAck
88 21.799961	192.168.10.66	192.168.10.65	TCP	54 49218 - 2000 [ACK] Seq=185 Ack=1449 Win=63644 Len=0
89 21.839643	192.168.10.66	192.168.10.65	SKINNY	90 StartMediaTransmissionAck

SCCP flow

	Características
Skinny Client Control Protocol Data length: 8 Header version: Basic (0x00000000) Message ID: ClearPriNotify (289) priority: 4	62 - SKINNY: ClearPriNotify -> El CallManager envía un frame para borrar todas las notificaciones en los dispositivos (teléfonos IP/aplicación VoIP).
Skinny Client Control Protocol Data length: 8 Header version: Basic (0x00000000) Message ID: ClearPriNotify (289) priority: 5	63 - SKINNY: ClearPriNotify -> El CallManager envía un frame para borrar todas las notificaciones en los dispositivos (teléfonos IP/aplicación VoIP).
Skinny Client Control Protocol Data length: 4 Header version: Basic (0x00000000) Message ID: ClearNotify (277)	65 - SKINNY: ClearNotify -> El CallManager envía un frame para borrar todas las notificaciones.
Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: DisplayPromptStatusV2 (325) timeOutValue: 0 lineInstance: 0 callReference: 0 promptStatus: Cisco Unified CME	66 - SKINNY: DisplayPromptStatusV2 (PromptStatus) -> El CallManager envía un frame para actualizar el estado de los dispositivos (teléfono IP/acplicación VoIP). Visualizando por pantalla del dispositivo (teléfono IP/aplicación VoIP) el texto "Cisco Unified CME".

	Características
<pre> Skinny Client Control Protocol Data length: 28 Header version: Basic (0x00000000) Message ID: CallState (273) callState: Connected (0x00000005) lineInstance: 1 callReference: 32 privacy: None (0x00000000) ▶ precedence </pre>	68 - SKINNY: CallState (Connected) -> El CallManager envía un frame actualizando el estado de los dispositivos (teléfonos IP/aplicación VoIP). Cambia el estado de los dispositivos a Connected (Conectados).
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: DisplayPromptStatusV2 (325) timeOutValue: 0 lineInstance: 1 callReference: 32 promptStatus: \357\277\275\030 => "Connected" </pre>	69 - SKINNY: DisplayPromptStatusV2 (Connected) -> El CallManager envía un frame para actualizar el estado de los dispositivos (teléfono IP/aplicación VoIP). Se visualiza en la pantalla el mensaje Connected (Conectado).
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: SelectSoftKeys (272) lineInstance: 1 callReference: 32 softKeySetIndex: Connected (0x00000001) ▼ validKeyMask1 = SoftKey1: Yes1.. = SoftKey2: Yes1... = SoftKey3: Yes1... = SoftKey4: Yes0... = SoftKey5: Yes0... = SoftKey6: No1... = SoftKey7: No1... = SoftKey8: Yes1... = SoftKey9: Yes1... = SoftKey10: Yes1... = SoftKey11: Yes1... = SoftKey12: Yes1... = SoftKey13: Yes1... = SoftKey14: Yes1... = SoftKey15: Yes1... = SoftKey16: Yes </pre>	71 - SKINNY: SelectSoftKeys (Connected) -> El CallManager envía un frame que activa las opciones disponibles en los dispositivos (teléfono IP/aplicación voip). Se activa la opción Conected (Conectados).
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: SetLamp (134) stimulus: Line (0x00000009) stimulusInstance: 1 lampMode: On (0x00000002) </pre>	72 - SKINNY: SetLamp (ON) -> El CallManager envía un frame manteniendo la alerta de luz (roja) de llamada activa y fija.
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: StopTone (131) lineInstance: 1 callReference: 32 </pre>	74 - SKINNY: StopTone -> El CallManager envía un frame indicando que debe pararse el tono de llamada entre los números 5201 (Ordenador) y 5202 (Teléfono_cisco), señalando que están conectados.
<pre> Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: ConnectionStatisticsReq (263) directoryNum: 5201 callReference: 32 Stats Processing Mode: clearStats (0x00000000) </pre>	75 - SKINNY: ConnectionStaticsReq (clearStats) -> El CallManager envía un frame para resetear las estadísticas de la llamada a cero en el número 5201 (Ordenador).
<pre> Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: ConnectionStatisticsReq (263) directoryNum: 5201 callReference: 32 Stats Processing Mode: clearStats (0x00000000) </pre>	77 - SKINNY: ConnectionStaticsReq (clearStats) -> El CallManager envía un frame para resetear las estadísticas de la llamada a cero en el número 5201 (Ordenador).

Características	
<pre> Skinny Client Control Protocol Data length: 112 Header version: Basic (0x00000000) Message ID: OpenReceiveChannel (261) conferenceID: 2 passThruPartyID: 0 millisecondsPacketsize: 20 compressionType: Media_Payload_G711Ulaw64k (0x00000004) ▼ qualifierIn ecValue: Media_EchoCancellation_On (0x00000001) g723bitRate: Media_G723BRate_6_3 (0x00000002) callReference: 32 ▼ mRxMediaEncryptionKeyInfo algorithmID: NO_ENCRYPTION (0x00000000) keylen: 0 saltlen: 0 key [ref: keylen = 0, max:16] salt [ref: saltlen = 0, max:16] isMKIPresent: 0 keyDerivationRate: 0 streamPassThroughID: 0 Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: SetLamp (134) stimulus: Line (0x00000009) stimulusInstance: 1 lampMode: On (0x00000002) Skinny Client Control Protocol Data length: 24 Header version: Basic (0x00000000) Message ID: OpenReceiveChannelAck (34) openReceiveChannelStatus: Ok (0x00000000) ipAddr IPv4 Address: 192.168.10.66 portNumber: 24606 passThruPartyID: 0 callReference: 32 Skinny Client Control Protocol Data length: 116 Header version: Basic (0x00000000) Message ID: CallInfoV2 (330) lineInstance: 1 callReference: 32 callingParty: OutboundCall (0x00000002) originalCallingDirectReason: 0 lastRedirectingReason: 0 callInstance: 1 callSecurityStatus: NotAuthenticated (0x00000001) ▼ partyPIRestrictionBits = CallingPartyName: No = CallingPartyNumber: No = CalledPartyName: No = CalledPartyNumber: No = CalleeParty: No = OriginalCalledPartyName: No = OriginalCalledPartyNumber: No = LastRedirectPartyName: No = LastRedirectPartyNumber: No = LastRedirectParty: No = BitsReserved: No callingParty: 5201 alternativeLineIdentity: 5202 calledParty: 5202 lastRedirectingParty: 5201 cgpnVoiceMailbox: 5202 cdpnVoiceMailbox: 5202 originalCdpnVoiceMailbox: Ordenador lastRedirectingVoiceMailbox: Ordenador callingPartyName: Telefono_cisco calledPartyName: Telefono_cisco Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: StopTone (131) lineInstance: 1 callReference: 32 </pre>	<p>78 - SKINNY: OpenReceiveChannel -></p> <ul style="list-style-type: none"> • Tipo de compresión Media_Payload_G711Ulaw64k. • En escucha para cancelación de la llamada -> ON. • Velocidad de comunicación Media_G723BRate_6_3. • Sin encriptación. <p>80 - SKINNY: SetLamp (ON) -> El CallManager envía un frame manteniendo la alerta de luz (roja) de llamada activa y fija.</p> <p>81 - SKINNY: OpenReceiveChannelAck -> La aplicación VoIP envía un frame al CallManager indicando cual es su IP (192.168.10.66) y el puerto que va a usar para establecer la comunicación es el 22606.</p> <p>82 - SKINNY: CallInfoV2 (OutBoundCall) -> El CallManager envía un frame actualizando la información y especificando la existencia de una llamada (5201 (Ordenador) - 5202 (Teléfono_cisco)) en curso.</p> <p>83 - SKINNY: StopTone -> El CallManager envía un frame a ambos dispositivos (teléfono IP/aplicación voip) de parada de tono. Los números 5201 (Ordenador) y 5202 (Teléfono_cisco) están conectados.</p>

Características	
<pre> Skinny Client Control Protocol Data length: 120 Header version: Basic (0x00000000) Message ID: StartMediaTransmission (138) conferenceID: 2 passThruPartyID: 0 remoteIpAddr IPv4 Address: 192.168.10.98 remotePortNumber: 25714 millisecondPacketSize: 20 compressionType: Media_Payload_G711Ulaw64k (0x00000004) ▼ qualifierOut precedenceValue: 184 ssValue: Media_SilenceSuppression_Off (0x00000000) maxFramesPerPacket: 0 padding: 1 g723bitRate: Media_G723BRate_6_3 (0x00000002) callReference: 32 ▼ mTxMediaEncryptionKeyInfo algorithmID: NO_ENCRYPTION (0x00000000) keylen: 0 saltlen: 0 key [ref: keylen = 0, max:16] salt [ref: saltlen = 0, max:16] isMKIPresent: 0 keyDerivationRate: 0 streamPassThroughID: 0 associatedStreamID: 0 RFC2833PayloadType: 0 dtmfType: 0 mixingMode: 2 </pre> <pre> Skinny Client Control Protocol Data length: 28 Header version: Basic (0x00000000) Message ID: StartMediaTransmissionAck (340) conferenceID: 2 passThruPartyID: 0 callReference: 32 transmitIpAddr IPv4 Address: 192.168.10.66 transmitPort: 0 startMediaTransmissionStatus: Ok (0x00000000) </pre>	<p>85 - SKINNY: StartMediaTransmission -> El CallManager envía un frame a ambos dispositivos (teléfono IP/aplicación voip) para comenzar a compartir archivos de audio. IPv4 Address: 192.168.10.98, remotePortNumber 24714, compressionType: Media_Payload_G711Ulaw64k.</p> <ul style="list-style-type: none"> • IPv4 Address del destino de la llamada: 192.168.10.98. • Puerto: 25714. • Velocidad de comunicación Media_G723BRate_6_3. • Sin encriptación. <p>89 - SKINNY: StartMediaTransmission-Ack (startMediaTransmissionStatus) -> Los dispositivos envían un frame de vuelta al CallManager avisando de la recepción del frame anterior. IPv4 Address del emisor de la llamada: 192.168.10.66.</p>

963 30.226117	192.168.10.65	192.168.10.66	SKINNY	74 ClearPromptStatus
964 30.228135	192.168.10.65	192.168.10.66	SKINNY	86 DisplayPromptStatusV2
965 30.228158	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=317 Ack=1545 Win=63548 Len=0
966 30.238483	192.168.10.65	192.168.10.66	SKINNY	98 ConnectionStatisticsReq
967 30.238917	192.168.10.66	192.168.10.98	UDP	214 24606 → 25714 Len=172 [UDP CHECKSUM INCORRECT]
968 30.238958	192.168.10.66	192.168.10.65	SKINNY	126 ConnectionStatisticsRes
969 30.241482	192.168.10.65	192.168.10.66	SKINNY	82 CloseReceiveChannel
970 30.242964	192.168.10.98	192.168.10.66	UDP	214 25714 → 24606 Len=172
971 30.243482	192.168.10.65	192.168.10.66	SKINNY	82 StopMediaTransmission
972 30.243503	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=389 Ack=1645 Win=63448 Len=0
973 30.246482	192.168.10.65	192.168.10.66	SKINNY	98 ConnectionStatisticsReq
974 30.248483	192.168.10.65	192.168.10.66	SKINNY	90 CallState
975 30.248503	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=389 Ack=1725 Win=63368 Len=0
976 30.250484	192.168.10.65	192.168.10.66	SKINNY	82 SelectSoftKeys
977 30.252490	192.168.10.65	192.168.10.66	SKINNY	82 SelectSoftKeys
978 30.252512	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=389 Ack=1781 Win=63312 Len=0
979 30.254492	192.168.10.65	192.168.10.66	SKINNY	70 SetSpeakerMode
980 30.256554	192.168.10.65	192.168.10.66	SKINNY	78 StopTone
981 30.256573	192.168.10.66	192.168.10.65	SKINNY	126 ConnectionStatisticsRes
982 30.258779	192.168.10.65	192.168.10.66	SKINNY	78 SetLamp
983 30.260700	192.168.10.65	192.168.10.66	SKINNY	70 ClearPriNotify
984 30.260707	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=461 Ack=1861 Win=63232 Len=0
985 30.262290	192.168.10.98	192.168.10.66	UDP	214 25714 → 24606 Len=172
986 30.263375	192.168.10.65	192.168.10.66	SKINNY	70 ClearPriNotify
987 30.265389	192.168.10.65	192.168.10.66	SKINNY	66 ClearNotify
988 30.265404	192.168.10.66	192.168.10.65	TCP	54 49218 → 2000 [ACK] Seq=461 Ack=1889 Win=63204 Len=0
989 30.267618	192.168.10.65	192.168.10.66	SKINNY	98 DisplayPromptStatusV2

SCCP flow

Para concluir la llamada que está en curso, acontecen los siguientes frames:

		Características
<pre> Skinny Client Control Protocol Data length: 12 Header version: Basic (0x00000000) Message ID: ClearPromptStatus (275) lineInstance: 1 callReference: 32 </pre>		963 - SKINNY: ClearPromptStatus -> El CallManager envía un frame para borrar el estado de ambos dispositivos (teléfono IP/aplicación voip).
<pre> Skinny Client Control Protocol Data length: 24 Header version: Basic (0x00000000) Message ID: DisplayPromptStatusV2 (325) timeOutValue: 0 lineInstance: 1 callReference: 32 promptStatus: </pre>		964 - SKINNY: DisplayPromptStatusV2 -> El CallManager envía un frame para actualizar el estado de ambos dispositivos (teléfono IP/aplicación VoIP), quedando a la espera.
<pre> Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: ConnectionStatisticsReq (263) directoryNum: 5201 callReference: 32 Stats Processing Mode: doNotClearStats (0x00000001) </pre>		966 - SKINNY: ConnectionStaticsReq (doNotClearStats) -> El CallManager envía un frame de recopilación de información estadística de la llamada en curso entre los dispositivos (teléfono IP/aplicación voip).
<pre> Skinny Client Control Protocol Data length: 64 Header version: Basic (0x00000000) Message ID: ConnectionStatisticsRes (35) directoryNum: 5201 callReference: 32 Stats Processing Mode: doNotClearStats (0x00000001) numberPacketsSent: 420 numberOctetsSent: 67200 numberPacketsReceived: 420 numberOctetsReceived: 67200 numberPacketsLost: 0 jitter: 0 latency: 0 </pre>		968 - SKINNY: ConnectionStaticsRes (doNotClearStats) -> El CallManager envía un frame mostrando información recopilada en el anterior frame.

Características	
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: CloseReceiveChannel (262) conferenceID: 2 passThruPartyID: 0 callReference: 32 portHandlingFlag: CLOSE_PORT (0x00000000) Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: StopMediaTransmission (139) conferenceID: 2 passThruPartyID: 0 callReference: 32 portHandlingFlag: CLOSE_PORT (0x00000000) Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: ConnectionStatisticsReq (263) directoryNum: 5201 callReference: 32 Stats Processing Mode: clearStats (0x00000000) </pre>	969 - SKINNY: CloseReceiveChannel (CLOSE_PORT) -> Cierra el puerto de comunicaciones usado en la llamada (22606).
	971 - SKINNY: StopMediaTransmission (CLOSE_PORT) -> El CallMAnager envía un frame de parada a ambos dispositivos (teléfono IP/aplicación voip) de la transmisión de datos de audio.
	973 - SKINNY: ConnectionStaticsReq (clearStats) -> El CallManager envía un frame para borrar la información estadística de la llamada entre los dispositivos (teléfono IP/aplicación voip).
<pre> Skinny Client Control Protocol Data length: 28 Header version: Basic (0x00000000) Message ID: CallState (273) callState: OnHook (0x00000002) lineInstance: 1 callReference: 32 privacy: None (0x00000000) ► precedence </pre>	974 - SKINNY: CallState (On Hook) -> El CallManager envía un frame actualizando el estado de la llamada de los dispositivos (teléfono IP/aplicación voip). El estado actual es On Hook (Colgado).
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: SelectSoftkeys (272) lineInstance: 0 callReference: 32 softKeySetIndex: On Hook (0x00000000) ▼ validKeyMask 1 = SoftKey1: Yes 1 = SoftKey2: Yes 1 = SoftKey3: Yes 1 = SoftKey4: Yes 1 = SoftKey5: Yes 1 = SoftKey6: Yes 0 = SoftKey7: No 1 = SoftKey8: Yes 1 = SoftKey9: Yes 1 = SoftKey10: Yes 1 = SoftKey11: Yes 1 = SoftKey12: Yes 1 = SoftKey13: Yes 1 = SoftKey14: Yes 1 = SoftKey15: Yes 1 = SoftKey16: Yes </pre>	976 - SKINNY: SelectSoftKeys (On Hook) -> El CallManager envía un frame actualizando el estado de ambos dispositivos (teléfono IP/aplicación voip) posicionandolos en el estado de On Hook (Colgado).
<pre> Skinny Client Control Protocol Data length: 20 Header version: Basic (0x00000000) Message ID: SelectSoftkeys (272) lineInstance: 0 callReference: 0 softKeySetIndex: On Hook (0x00000000) ▼ validKeyMask 1 = SoftKey1: Yes 1 = SoftKey2: Yes 1 = SoftKey3: Yes 1 = SoftKey4: Yes 1 = SoftKey5: Yes 1 = SoftKey6: Yes 0 = SoftKey7: No 1 = SoftKey8: Yes 1 = SoftKey9: Yes 1 = SoftKey10: Yes 1 = SoftKey11: Yes 1 = SoftKey12: Yes 1 = SoftKey13: Yes 1 = SoftKey14: Yes 1 = SoftKey15: Yes 1 = SoftKey16: Yes </pre>	977 - SKINNY: SelectSoftKeys (On Hook) -> El CallManager envía un frame actualizando el estado de ambos dispositivos (teléfono IP/aplicación voip) posicionandolos en el estado de On Hook (Colgado).

Características	
<pre> Skinny Client Control Protocol Data length: 8 Header version: Basic (0x00000000) Message ID: SetSpeakerMode (136) speakerMode: Off (0x00000002) </pre>	979 - SKINNY: SetSpeakerMode (Off) -> El CallManager envía un frame que desactiva el auricular del teléfono IP.
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: StopTone (131) lineInstance: 1 callReference: 32 </pre>	980 - SKINNY: StopTone -> El CallManager envía un frame que mantiene desactivado el tono tras haber colgado el auricular del teléfono.
<pre> Skinny Client Control Protocol Data length: 64 Header version: Basic (0x00000000) Message ID: ConnectionStatisticsRes (35) directoryNum: 5201 callReference: 32 Stats Processing Mode: clearStats (0x00000000) numberPacketsSent: 421 numberOctetsSent: 67360 numberPacketsReceived: 421 numberOctetsReceived: 67360 numberPacketsLost: 0 jitter: 0 latency: 0 </pre>	981 - SKINNY: ConnectionStatisticsRes -> El CallManager envía un frame con el resultado estadístico de la llamada.
<pre> Skinny Client Control Protocol Data length: 16 Header version: Basic (0x00000000) Message ID: SetLamp (134) stimulus: Line (0x00000009) stimulusInstance: 1 lampMode: Off (0x00000001) </pre>	982 - SKINNY: SetLamp (Off) -> El CallManager envía un frame que desactiva luz (roja) del teléfono.
<pre> Skinny Client Control Protocol Data length: 8 Header version: Basic (0x00000000) Message ID: ClearPriNotify (289) priority: 4 </pre>	983 - SKINNY: ClearPriNotify () -> El CallManager envía un frame para borrar todas las notificaciones en los dispositivos (teléfonos IP/aplicación VoIP).
<pre> Skinny Client Control Protocol Data length: 8 Header version: Basic (0x00000000) Message ID: ClearPriNotify (289) priority: 5 </pre>	986 - SKINNY: ClearPriNotify () -> El CallManager envía un frame para borrar todas las notificaciones en los dispositivos (teléfonos IP/aplicación VoIP).
<pre> Skinny Client Control Protocol Data length: 4 Header version: Basic (0x00000000) Message ID: ClearNotify (277) </pre>	987 - SKINNY: ClearNotify () -> El CallManager envía un frame para borrar todas las notificaciones.
<pre> Skinny Client Control Protocol Data length: 36 Header version: Basic (0x00000000) Message ID: DisplayPromptStatusV2 (325) timeOutValue: 0 lineInstance: 0 callReference: 0 promptStatus: Cisco Unified CME </pre>	989 - SKINNY: DisplayPromptStatusV2 (Cisco Unified CME) -> El CallManager envía un frame para actualizar el estado de los dispositivos (teléfono IP/aplicación VoIP). Visualizando por pantalla del dispositivo (teléfono IP/aplicación VoIP) el texto "Cisco Unified CME".

Anexo VI: Conclusión

Diferencias generales

SIP	SCCP
<ul style="list-style-type: none">• Es open source (Código abierto). Posibilidad de descarga e instalación sin permiso y sin licencias establecidas por el fabricante.• Enfocado a la instalación y configuración de la PBX.• Muy flexible a la hora de configurar.• Muy buen soporte y documentación desde la comunidad (en la red).• Registra en el servidor Usuario y password, al realizarse la autenticación del registro del cliente.• Los frames generado contienen mucha información adicional.• Usa RTP (UDP) para el envío y recepción de paquetes.• Muy buen soporte, gran cantidad de documentación en la red.• Certificaciones oficiales.• Algo de complejo de implementar.• Muy barato.• Complejo de mantener.	<ul style="list-style-type: none">• Cisco crea software privativo. Cisco Unified Communication Manager (CUCM) y Cisco Call Manager Express (CME).• Enfocado a la infraestructura de red (Vlan, trunking, frame tagging, DHCP, TFTP, ...) más la instalación y configuración de la telefonía IP (CME y CUCM).• Muy rígido a la hora de configurar.• Muy buen soporte y su documentación es de Cisco.• Registra las MACs de los dispositivos.• Los frames generados son simples y ligeros.• Usa UDP para el envío y recepción de paquetes.• Muy buen soporte y su documentación es de Cisco.• Certificaciones oficiales renovables cada cierto tiempo.• Fácil de implementar.• Muy caro.• Fácil mantenimiento.

SIP vs SCCP

Tiempos de respuesta

SIP	SCCP
• Estableciendo la llamada = 0.41799549 seg	• Estableciendo la llamada = 1.463633 seg
• Finalizando la llamada = 0.077101084 seg	• Finalizando la llamada = 0.03926 seg

Tiempos de respuesta (SIP vs SCCP)

Tamaño de los frames

SIP Estableciendo la llamada
SIP/SDP -> INVITE: 858 bytes (6864 bits)
SIP -> Status: 555 bytes (4440 bits)
SIP -> Status: 555 bytes (4440 bits)
SIP -> Request Ack: 412 bytes (3296 bits)
SIP/SDP -> Request INVITE: 1825 bytes (8200 bits)
SIP -> Status 100 Trying: 498 bytes 3984 (bits)
SIP -> Status 100 Trying: 498 bytes 3984 (bits)
SIP/SDP -> Request INVITE: 1483 bytes (11864 bits)
SIP/SDP -> Request INVITE: 1483 bytes (11864 bits)
SIP -> Status 100 Trying: 309 bytes 2472 (bits)
SIP -> Status 100 Ringing: 406 bytes 3248 (bits)
SIP -> Status 100 Ringing: 514 bytes 4112 (bits)
SIP -> Status 100 Ringing: 514 bytes 4112 (bits)
SIP/SDP -> Status 200 OK: 1046 bytes (8368 bits)
SIP -> Request Ack: 480 bytes (3840 bits)
SIP -> Request Ack: 480 bytes (3840 bits)
SIP/SDP -> Status 200 OK: 779 bytes (6232 bits)
SIP/SDP -> Status 200 OK: 779 bytes (6232 bits)
STUN -> Binding Request: 62 bytes (496 bits)
STUN -> Binding Success: 74 bytes (592 bits)
STUN -> Binding Success: 74 bytes (592 bits)
RTCP: 62 bytes (496 bits)
CLASSIC-STUN -> Message Binding Request: 74 bytes (592 bits)
CLASSIC-STUN -> Message Binding Request: 74 bytes (592 bits)
RTCP: 62 bytes (496 bits)
RTCP: 62 bytes (496 bits)
RTCP: 62 bytes (496 bits)
RTCP: 62 bytes (496 bits)
SIP -> Request Ack: 518 bytes (4144 bits)
STUN -> Binding Request: 62 bytes (496 bits)
RTCP: 62 bytes (496 bits)
RTCP: 74 bytes (592 bits)

SIP Estableciendo la llamada
STUN -> Binding Success: 74 bytes (592 bits)
STUN -> Binding Success: 74 bytes (592 bits)
CLASSIC-STUN -> Message Binding Request: 74 bytes (592 bits)
CLASSIC-STUN -> Message Binding Request: 74 bytes (592 bits)
STUN -> Binding Request: 62 bytes (496 bits)
RTCP: 62 bytes (496 bits)
RTCP: 74 bytes (592 bits)
STUN -> Binding Success: 74 bytes (592 bits)
STUN -> Binding Success: 74 bytes (592 bits)
CLASSIC-STUN -> Message Binding Request: 74 bytes (592 bits)
CLASSIC-STUN -> Message Binding Request: 74 bytes (592 bits)
Total = 15944 bytes (127552 bits) = 0.12164306640625 megabits

Tamaño de los frames (SIP Estableciendo la llamada)

SIP Comunicación.
RTP: 87 bytes (696 bits)

SIP Comunicación

SIP Finalizando la llamada.
SIP -> Request BYE: 523 bytes (4384 bits)
SIP -> Status 200 OK: 468 bytes (3744 bits)
SIP -> Status 200 OK: 468 bytes (3744 bits)
SIP -> Request BYE: 513 bytes (4104 bits)
SIP -> Request BYE: 513 bytes (4104 bits)
Total = 2485 bytes (20080 bits) = 0.0191497802734375 megabits

Tamaño de los frames (SIP Finalizando la llamada)

SCCP Estableciendo la llamada.
SKinny -> SoftkeyEvent: 78 bytes (624 bits)
SKinny -> CallState: 90 bytes (720 bits)
SKinny -> ClearPromptStatus: 74 bytes (592 bits)
SKinny -> SelectSoftKeys: 82 bytes (656 bits)
SKinny -> SetLamp: 78 bytes (624 bits)
SKinny -> SetSpeakerMode: 78 bytes (624 bits)
SKinny -> DisplayPromptStatusV2: 82 bytes (656 bits)
SKinny -> StartTone: 82 bytes (656 bits)
SKinny -> MediaPathEvent: 74 bytes (592 bits)
SKinny -> KeypadButton: 102 bytes (816 bits)
SKinny -> StopTone: 78 bytes (624 bits)

SCCP Estableciendo la llamada.

SKinny -> KeypadButton: 78 bytes (624 bits)
SKinny -> KeypadButton: 78 bytes (624 bits)
SKinny -> DialedNumber: 98 bytes (784 bits)
SKinny -> CallState: 98 bytes (784 bits)
SKinny -> CallState: 98 bytes (784 bits)
SKinny -> DisplayPromptStatusV2: 82 bytes (656 bits)
SKinny -> SelectSoftKeys: 82 bytes (656 bits)
SKinny -> SetLamp: 78 bytes (624 bits)
SKinny -> CallInfo: 178 bytes (1424 bits)
SKinny -> SetLamp: 78 bytes (624 bits)
SKinny -> StartTone: 82 bytes (656 bits)
SKinny -> CallInfo: 178 bytes (1424 bits)
SKinny -> ClearPriNotify: 78 bytes (624 bits)
SKinny -> ClearPriNotify: 78 bytes (624 bits)
SKinny -> ClearNotify: 68 bytes (520 bits)
SKinny -> DisplayPromptStatusV2: 98 bytes (784 bits)
SKinny -> CallState: 98 bytes (784 bits)
SKinny -> DisplayPromptStatusV2: 82 bytes (656 bits)
SKinny -> SelectSoftKeys: 82 bytes (656 bits)
SKinny -> SetLamp: 78 bytes (624 bits)
SKinny -> StopTone: 78 bytes (624 bits)
SKinny -> ConnectionStaticsReq: 98 bytes (784 bits)
SKinny -> ConnectionStaticsReq: 98 bytes (784 bits)
SKinny -> OpenRecieveChannel: 174 bytes (1392 bits)
SKinny -> SetLamp: 78 bytes (624 bits)
SKinny -> OpenRecieveChannelAck: 86 bytes (688 bits)
SKinny -> OpenRecieveChannel: 174 bytes (1392 bits)
SKinny -> CallInfo: 178 bytes (1424 bits)
SKinny -> StopTone: 78 bytes (624 bits)
SKinny -> StartMediaTransmission: 182 bytes (1456 bits)
SKinny -> KeepAlive: 66 bytes (528 bits)
SKinny -> KeepAliveAck: 66 bytes (528 bits)
SKinny -> StartMediaTransmissionAck: 90 bytes (720 bits)

Total = 3634 bytes (29072 bits) = 0.0277252197265625 megabits

SCCP Estableciendo la llamada

Tamaño de los frames (SCCP Comunicación)

UDP: 214 bytes (1712 bits)

SCCP Comunicación

SCCP Finalizando la llamada.

SKinny -> ClearPromptStatus: 74 bytes (582 bits)
 SKinny -> DisplayPromptStatusV2: 86 bytes (688 bits)
 SKinny -> ConnectionStaticsReq: 98 bytes (784 bits)
 SKinny -> ConnectionStaticsRes: 126 bytes (1088 bits)
 SKinny -> CloseRecieveChannel: 82 bytes (656 bits)
 SKinny -> ConnectionStaticsReq: 98 bytes (784 bits)
 SKinny -> StopMediaTransmissionAck: 82 bytes (656 bits)
 SKinny -> ConnectionStaticsReq: 98 bytes (784 bits)
 SKinny -> CallSate: 98 bytes (784 bits)
 SKinny -> SelectSoftKeys: 82 bytes (656 bits)
 SKinny -> SelectSoftKeys: 82 bytes (656 bits)
 SKinny -> SetSpeakerMode: 78 bytes (624 bits)
 SKinny -> StopTone: 78 bytes (624 bits)
 SKinny -> ConnectionStaticsRes: 126 bytes (1088 bits)
 SKinny -> SetLamp: 78 bytes (624 bits)
 SKinny -> ClearPriNotify: 78 bytes (624 bits)
 SKinny -> ClearPriNotify: 78 bytes (624 bits)
 SKinny -> ClearNotify: 66 bytes (528 bits)
 SKinny -> DisplayPromptStatusV2: 98 bytes (784 bits)
 SKinny -> SelectSoftKeys: 82 bytes (656 bits)
 SKinny -> ClearPriNotify: 78 bytes (624 bits)
 SKinny -> ClearPriNotify: 78 bytes (624 bits)
 SKinny -> ClearNotify: 66 bytes (528 bits)
 SKinny -> DisplayPromptStatusV2: 98 bytes (784 bits)
 SKinny -> ClearPriNotify: 78 bytes (624 bits)
 SKinny -> ClearPriNotify: 78 bytes (624 bits)
 SKinny -> ClearNotify: 66 bytes (528 bits)
 SKinny -> DisplayPromptStatusV2: 98 bytes (784 bits)

Total = 2408 bytes (19264 bits) = 0.01837158203125 megabits

[Tamaño de los frames \(SCCP Finalizando la llamada\)](#)

Bibliografía

Libros

- [1] Barrie Dempster, David Gomillion, David Merel. *Asterisk 1.6*. Pack Publishing Ltd. Inglaterra. September 2009.
- [2] Nir Simionovich. *AsteriskNOW*. Inglaterra. Pack Publishing Ltd. Marzo 2008.
- [3] Cisco Systems, Inc. *Cisco IP Telephony Network Design Guide - Cisco CallManager Release 3.0(5)*. Cisco Systems, Inc. Estados Unidos. 2000 - 2001.
- [4] Jeremy Cioara, Mike Valentine. *CCNA Voice 640-461 - Official Cert Guide*. Pearson Education, Inc. Estados Unidos. 2012.
- [5] Russell Bryant, Leif Madsen and Jim Van Megelen. *Asterisk: The Definitive Guide, Fourth Edition*. O'Reilly. Estados Unidos. Mayo 2013.
- [6] Kevin Wallace. *Implementing Cisco Unified Communications Voice over IP and QoS (CVOICE) Foundation Learning Guide*. Cisco Press. Estados Unidos. Jan 2011.
- [7] Bruce Hartpence. *Packet Guide to Voice over IP*. O'Reilly. Estados Unidos. 2013.

Web

- [8] Techabulary
<https://www.techabulary.com/v/voip/>
Accedido 10 de Enero
- [9] Tecnología VoIP: Historia, evolución y Apps.
<http://www.jeronimoperez.com/blog/marketing-online/tecnologia-voip-historia-evolucion-y-apps/>
Accedido 11 de Enero

- [10] Antecedentes Históricos de telefonía IP.
<http://www.servervoip.com/blog/antecedentes-historicos-de-telefonia-ip/>
Accedido 11 de Enero
- [11] Estudio de H.323 y SIP Abel Sáez Incertis.
[http://www.grc.upv.es/docencia/tdm/trabajos2007/Abel_H.323%20vs%20SIP%20\(1\).pdf](http://www.grc.upv.es/docencia/tdm/trabajos2007/Abel_H.323%20vs%20SIP%20(1).pdf)
Accedido 14 de Enero
- [12] Techabulary
<https://www.techabulary.com/h/h323/>
Accedido 15 de Enero
- [13] Redes y servicios de telecomunicaciones
<http://blogtelecomunicaciones.ramonmillan.com/2008/06/principales-diferencias-entre-h323-y.html>
Accedido 15 de Enero
- [14] VoipForo - H.323
<http://www.voipforo.com/H323/H323componentes.php>
Accedido 15 de Enero
- [15] Techabulary
<https://www.techabulary.com/s/sip/>
Accedido 20 de Enero
- [16] VoIP: Funcionamiento básico del protocolo SIP
<https://www.securityartwork.es/2008/03/03/voip-protocolo-sip/>
Accedido 22 de Enero
- [17] SIP - Session Initiation Protocol
<http://www.quarea.com/es/sip-session-initiation-protocol>
Accedido 15 de Febrero
- [18] Session Initiation Protocol
https://es.wikipedia.org/wiki/Session_Initiation_Protocol
Accedido 5 de Febrero
- [19] Wikipedia - SIP-Status-Codes
<https://de.wikipedia.org/wiki/SIP-Status-Codes>
Accedido 6 de Febrero
- [20] NetVoIP
<https://netvoip.wordpress.com/2015/04/29/qos-sobre-nuestras-comunicaciones-voip-sip-y-rtp/>
Accedido 8 de Febrero
- [21] Informacion y Preguntas frecuentes acerca de SIP
<https://www.3cx.es/voip-sip/sip-faq/>
Accedido 12 de Febrero
- [22] Wikipedia - RTP audio video profile
https://en.wikipedia.org/wiki/RTP_audio_video_profile
Accedido 16 de Febrero
- [23] Wikipedia - IAX2
<https://es.wikipedia.org/wiki/IAX2>
Accedido 20 de Febrero

- [24] IAX - Inter-Asterisk eXchange protocol
<http://www.voipforo.com/IAX/IAX-arquitectura.php>
Accedido 21 de Febrero
- [25] Protocolo IAX
<http://elastixtech.com/protocolo-iax/>
Accedido 21 de Febrero
- [26] SCCP Call Signaling
<http://flylib.com/books/en/2.3.1.89/1/>
Accedido 9 de Mayo
- [27] Wikipedia - Skinny Call Control Protocol
https://en.wikipedia.org/wiki/Skinny_Call_Control_Protocol
Accedido 21 de Marzo
- [28] CCNP Self-Study: Understanding and Implementing Quality of Service in Cisco Multilayer Switched Networks
<http://www.ciscopress.com/articles/article.asp?p=170743&seqNum=7>
Accedido 11 de Mayo
- [29] Catalyst 3750 Switch Software Configuration Guide, 12.2(35)SE - Chapter: Configuring QoS
https://www.cisco.com/c/en/us/td/docs/switches/lan/catalyst3750/software/release/12-2_35_se/configuration/guide/scg/swqos.html#wp1032145
Accedido 12 de Mayo
- [30] Cisco Paso a Paso
<https://ronaldreales.wordpress.com/tag/encapsulation-dot1q/>
Accedido 13 de Mayo
- [31] Cisco Support Communioty - Unified Communications Call flow in an Enterprise Network
<https://supportforums.cisco.com/document/138351/unified-communications-call-flow-enterprise-network>
Accedido 13 de Mayo
- [32] Ejemplo de configuración de Cisco CallManager Express/Cisco Unity Express
https://www.cisco.com/c/es_mx/support/docs/voice-unified-communications/unity-express/62609-tdcmecu.html
Accedido 14 de Mayo
- [33] DSCP & TOS
<https://www.tucny.com/Home/dscp-tos>
Accedido 16 de Mayo
- [34] Tipos de Servicio con DSCP
<https://ccie-en-espanol.blogspot.com.es/2009/06/tipos-de-servicio-con-dscp.html>
Accedido 16 de Mayo
- [35] Los mejores trucos de Asterisk
<https://es.slideshare.net/david.motta/los-mejores-trucos-de-asterisk>
Accedido 20 de Mayo
- [36] Asterisk config sip.conf
<https://www.voip-info.org/wiki/view/Asterisk+config+sip.conf>

Accedido 21 de Mayo

[37] IP Quality of Service

<https://wiki.asterisk.org/wiki/display/AST/IP+Quality+of+Service>

Accedido 22 de Mayo

[38] Understanding QoS and How to Improve the Audio of your VoIP Calls

<http://blogs.digium.com/2015/12/23/understanding-qos-improve-audio-voip-calls/>

Accedido 25 de Mayo

[39] Asterisk sip tos

<https://www.voip-info.org/wiki/view/Asterisk+sip+tos>

Accedido 20 de Febrero

[40] Audio-Codecs zur Sprachdigitalisierung

<https://www.elektronik-kompendium.de/sites/net/0905121.htm> Accedido 20 de Mayo

[41] Codec Summary table

<http://www.en.voipforo.com/codec/codecs.php> Accedido 22 de Mayo

[42] Payload Type Definitions

<http://www.freesoft.org/CIE/RFC/1890/29.htm> Accedido 26 de Mayo

Normas y referencias

Advanced Encryption Standard, *AES*, (2001)

<https://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

Adaptative Multi-Rate o Compresión multi-tasa adaptativa, *AMR*, (1999)

<http://www.3gpp.org/DynaReport/26-series.htm>

Estándar para la codificación de audio usado en la telefonía, *G.711*, (1988)

<https://www.itu.int/rec/T-REC-G.711/es>

Estándar para la codificación de audio usado en la telefonía (codificación de audio de 7khz dentro de 64 kbit/s), *G.722*, (2012)

<https://www.itu.int/rec/T-REC-G.722/es>

Estándar para la codificación de audio usado en la telefonía (Codec de voz de doble velocidad para la trasnmisión en comunicaciones multimedia a 5,3 y 6,3 kbit/s), *G.723.1*, (2006)

<https://www.itu.int/rec/T-REC-G.723.1/es>

Modulación por impulso codificado diferencial adaptativa (MICDA), *G.726*, (1990)

<https://www.itu.int/rec/T-REC-G.726/es>

Codificación de señales vocales a 16 kbit/s utilizando predicción lineal con excitación por código de bajo retardo, *G.728*, (2012)

<https://www.itu.int/rec/T-REC-G.728/es>

Codificación de la voz a 8kbit/s mediante predicción lineal con excitación por código algebraico de estructura conjugada, *G.729*, (2012)

<https://www.itu.int/rec/T-REC-G.729/es>

Sistema global para las comunicaciones móviles o Global System for Mobile communications, *GSM*, (1992)

http://www.etsi.org/deliver/etsi_gts/06/0610/03.02.00_60/gsmts_0610sv030200p.pdf

Protocolos de señalización de llamada y paquetización de trenes de medios para sistemas de comunicación multimedia por paquetes, *H.225.0*, (2009)

<https://www.itu.int/rec/T-REC-H.225.0/es>

Seguridad y encriptado para terminales multimedias de la serie H (basados en las Recomendaciones UIT-T H.323 y H.245), *H.235*, (1998)

<https://www.itu.int/rec/T-REC-H.235/es>

Gestión de funciones y canales de medios adicionales para terminales de la serie H.300, *H.239*, (2014)

<http://www.itu.int/ITU-T/recommendations/rec.aspx?rec=12235&lang=es>

Protocolo de control para comunicación multimedia, *H.245*, (2011)

<https://www.itu.int/rec/T-REC-H.245/es>

Estándar para la codificación de video a 64kbit/s, *H.261*, (1993)

<https://www.itu.int/rec/T-REC-H.261-199303-I/en>

Protocolo funcional genérico para el soporte de servicios suplementarios en los sistemas UIT-T H.323, *H.450.1*, (2011)

<https://www.itu.int/rec/T-REC-H.450.1/es>

Negociación de protocolos de seguridad para proteger los mensajes de señalización de llamada UIT-T H.225.0, *H.460.22*, (2015)

<https://www.itu.int/rec/T-REC-H.460.22/es>

Interactive Connectivity Establishment, *ICE*, (2010)

<https://tools.ietf.org/html/rfc5245>

Internet Low Bitrate Codec, *iLBC*, (2004)

<http://ilbcfreeware.org/>

<https://www.ietf.org/rfc/rfc3951.txt>

Estándar de compresión digital audio y video, *MPG-4*, (2002)

<http://mpeg.chiariglione.org/standards/mpeg-4/mpeg-4.htm>

RTP Control Protocol, *RTCP*, (2003)

<https://tools.ietf.org/html/rfc3605>

Real-Time Transport Protocol, *RTP*, (2003)

<https://tools.ietf.org/search/rfc3550>

Session Description Protocol, *SDP*, (2006)

<https://tools.ietf.org/html/rfc4566>

Compresión de audio, *Speex*, (2007)

<https://www.speex.org/>

Procedimientos para la comunicación facsimil en tiempo real entre terminales facsimil del grupo 3 por redes con protocolo internet, *T.38*, (2015)

<https://www.itu.int/rec/T-REC-T.38/es>

Protocolo que da soporte en tiempo real y comunicación multipunto, *T.120*, (2007)
<https://www.itu.int/rec/T-REC-T.120/es>

Módem sobre redes de protocolo Internet, *V.150*, (2003)
<http://www.itu.int/itu-t/recommendations/rec.aspx?rec=6223&lang=es>

