

Trabajo Final de Carrera.



PROYECTO DE UN SISTEMA DE COMUNICACIONES VOIP IMPLEMENTADO CON RASPBERRY PI Y ASTERISK

Nombre del estudiante:

Oscar Castro Otero

E.T.T.

Consultor:

José López Vicario

"Ansiamos nuevas sensaciones pero enseguida nos volvemos indiferentes a ellas. Las maravillas del ayer son los sucesos corrientes de hoy".

Nikola Tesla

Descripción del proyecto.

El proyecto consiste en implantar un sistema de comunicaciones VoIP puro (sin conexiones RDSI ni analógicas) mediante la utilización de equipos de comunicaciones implementadas con Raspberry Pi y Asterisk en una empresa multinacional dedicada a los servicios con sedes en distintos países para que sustituya las tradicionales centralitas PBX propietarias por sistemas basados Asterisk.

También se realizará una comparación del coste entre los dos sistemas (sistemas tradicionales vs sistemas VoIP Asterisk) y una comparación de costes entre distintos proveedores de telefonía IP.

La empresa multinacional AQUADOM a la que se le desea implantar el sistema VoIP tiene 4 sedes principales en España (Barcelona, Madrid, Valladolid y Santiago de Compostela), una sede central en Santiago de Chile con oficinas en Valparaíso y Viña del Mar, otra sede central en Bogotá con oficinas en Medellín y Cali y por último otra sede en Oporto. En el proyecto damos por supuesto que la empresa ya tiene servicios de internet en cada una de las sedes.

Analizaremos cuáles son los problemas que se producen en las comunicaciones VoIP y las soluciones que se pueden utilizar para minimizar o eliminar estos problemas. Examinaremos nuestro caso y aportaremos las soluciones específicas para eliminar los problemas en nuestro sistema. Se realizará un análisis pormenorizado del ancho de banda para calcular cuánto ancho de banda necesitamos para las comunicaciones de nuestras sedes dependiendo del códec utilizado.

Al final del proyecto incluiremos un presupuesto de instalación de todo el sistema en la empresa, realizaremos un análisis de las velocidades y retardos que se requieren para la solución y se elaborará un plan de contingencias para el caso de que falle algún equipo.

Contenido

1.	Objetivos y planificación del proyecto	6
1.1.	Objetivos.	6
1.2.	Planificación del proyecto.	7
2.	VoIP vs Telefonía tradicional	10
2.1.	Ventajas de la Telefonía IP, ¿Porque utilizar VoIP?.....	10
2.2.	Estudio comparativo de diversos ITSP (proveedores de telecomunicaciones digitales basados en Voice over Internet Protocol (VoIP)).....	11
2.3.	Cálculo de ahorro telefonía IP vs Tradicional.....	12
3.	Diseño de la red de comunicaciones.....	15
3.1.	Diseño de la red entre sedes.....	15
3.2.	Análisis del hardware y del software que se utilizará en el desarrollo del proyecto. 17	
3.3.	Equipamiento para la solución.....	18
3.3.1.	Raspberry Pi.....	18
3.3.2.	PiTFT	19
3.3.3.	Alcatel Temporis IP600 IP	19
3.3.4.	Alcatel Temporis IP200 IP.....	20
4.	Garantizar un servicio óptimo.....	22
4.1.	QoS-Calidad de Servicio para VoIP	22
4.2.	Problemas que se producen en las comunicaciones VoIP y sus posibles soluciones . 22	
4.2.1.	Jitter.....	23
4.2.2.	Latencia	24
4.2.3.	Eco	25
4.2.4.	Pérdida de paquetes – packet loss.....	26
4.2.5.	Ancho de banda insuficiente.....	26
4.2.6.	Técnicas para lograr QoS.....	27
4.3.	Soluciones para evitar estos problemas en nuestro sistema.....	29
4.4.	Ancho de Banda.	30
3.	Presupuesto de instalación del sistema VoIP.....	33
4.	Velocidades y retardos que se requieren y plan de contingencias.....	36
4.1.	Velocidades y retardos.....	36
4.2.	Plan de contingencias.....	36
5.	Conclusiones.....	37
	Anexos.....	38
	Compra de los equipos.....	38
	Descarga del software, ensamblaje de los equipos e instalación del software.....	38

Instalación del sistema Raspbian OS	39
Configuración de Asterisk (Plan de marcación, teléfonos SIP y troncales IAX, troncales SIP y buzones de voz).....	46
Configuración de los terminales SIP (Alcatel Temporis IP) para conectarse a Asterisk.	56
Bibliografía.	60

Tabla de Figuras

Figura 1. Esquema de sedes	6
Figura 2. Comparativa de tarifas distintos proveedores IP	11
Figura 3. Tarifas SARENET para España	11
Figura 4. Tarifas SARENET para el resto de sedes	12
Figura 5. Factura tipo media de servicios de Movistar.	12
Figura 6. Resumen duración de llamadas de factura de Movistar	13
Figura 7. Tabla comparativa entre el coste Movistar y el coste con proveedor VOIP.	13
Figura 8. Ahorro total anual al utilizar proveedor VOIP	14
Figura 9. Esquema general del sistema. Sedes, direcciones IP y protocolos	16
Figura 10. Esquema tipo de red de una oficina	17
Figura 11. Raspberry Pi	18
Figura 12. PiTFT sobre una Raspberry Pi	19
Figura 13. Distribución del ancho de banda	24
Figura 14. Asignación de recursos según tipo de tráfico	27
Figura 15. Raspberry Pi funcionando con PiTFT ensamblada	39
Figura 16. Comparativa entre dos Raspberry Pi, una con PiTFT y otra sin ella	40
Figura 17. Menú de configuración de Raspbian	41
Figura 18. Página de configuración de teléfono Temporis IP200	58

1. Objetivos y planificación del proyecto

1.1. Objetivos.

Los objetivos del proyecto son ver cómo es posible el desarrollo de un sistema de comunicaciones VOIP para la empresa utilizando software libre cómo Asterisk integrado en dispositivos de bajo costo cómo son las Raspberry Pi dando un resultado óptimo si los juntamos con proveedores de servicios de telefonía IP, los cuales ofertan conexiones con casi cualquier lugar del Mundo a un precio muy competitivo. Veremos entonces si es viable implantar un sistema VoIP con Asterisk y si éste es rentable para la empresa a la que se lo queremos implementar.

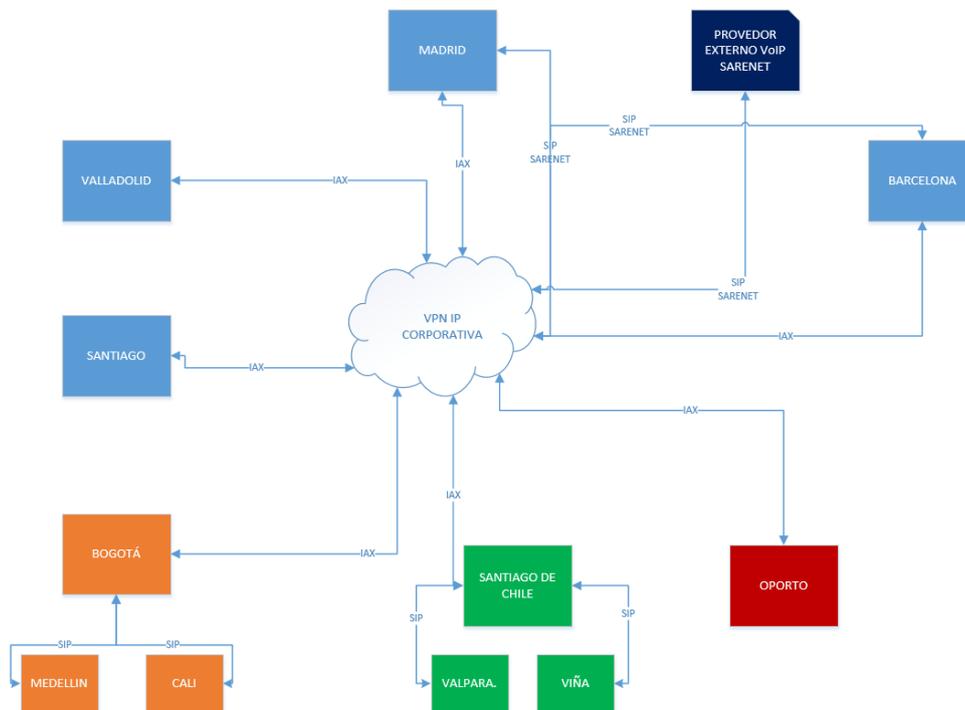


Figura 1. Esquema de sedes

El sistema se construirá mediante la instalación una Raspberry Pi con Asterisk en cada una de las sedes, menos en las oficinas de Cali y Medellín en Colombia y de Valparaíso y Viña del Mar en Chile las cuales conectarán directamente los teléfonos IP a las sedes de Bogotá y Santiago de Chile respectivamente. Las comunicaciones entre las sedes se realizarán mediante troncales IAX y las conexiones de los teléfonos IP se realizarán mediante el protocolo SIP. Se dispondrá de dos troncales SIP con el proveedor de servicios VoIP para llamadas externas, uno en Barcelona y el otro en Madrid. Se utilizan dos troncales SIP para evitar la interrupción del servicio si una de las dos sedes con trocales SIP cae.

En este proyecto optamos por la utilización de una Raspberry Pi con Asterisk en lugar de un sistema tradicional por las siguientes razones:

- Es un sistema libre. Podemos añadir las extensiones que nosotros creamos convenientes, añadir IVR sin necesidad de nuevas tarjetas y costosas licencias.
- No dependemos de los técnicos de los fabricantes Siemens, Alcatel, Avaya, etc...
- Es un sistema barato, ya que se pueden encontrar Raspberry Pi por menos de 50€.
- Tiene un tamaño pequeño que permite transportarlo de una forma más rápida y barata.
- Es muy fácil de clonar ya que sólo tenemos que clonar la tarjeta SD del Sistema.

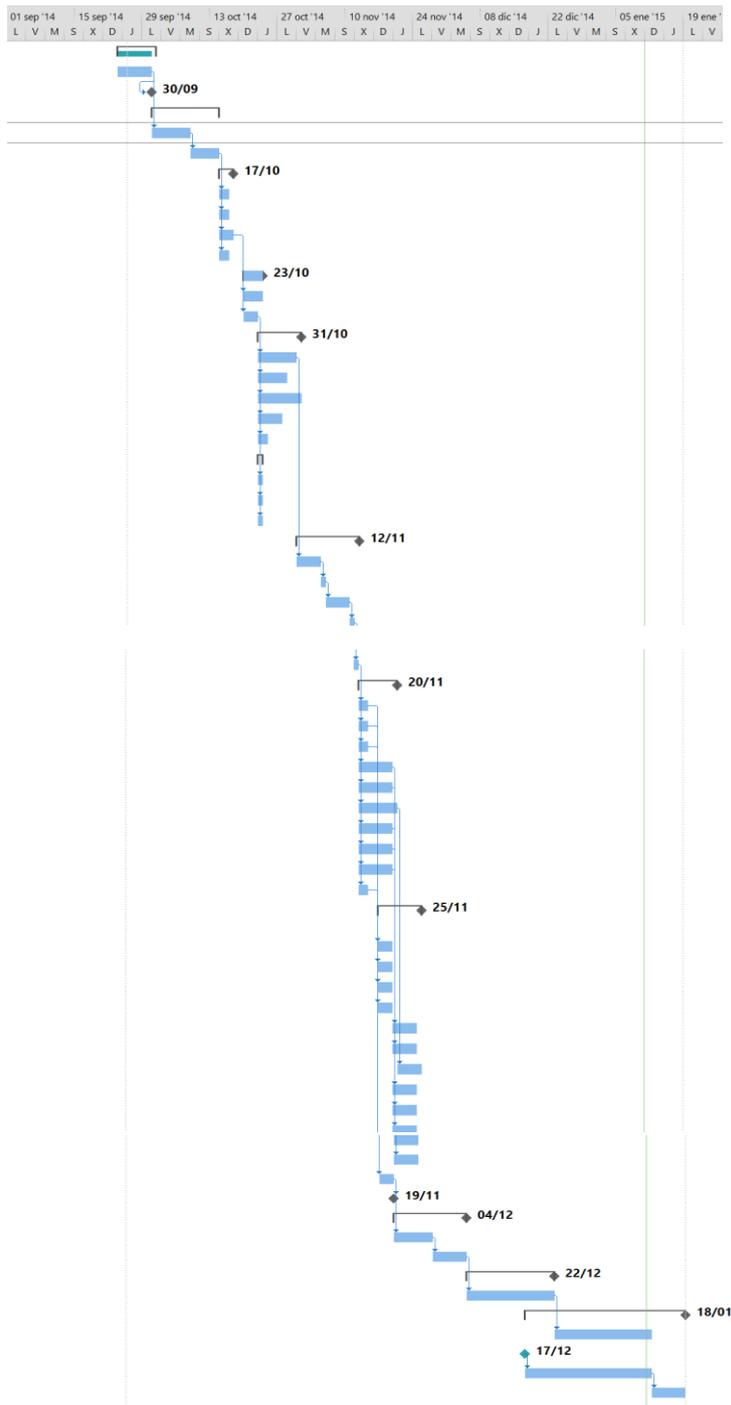
1.2. Planificación del proyecto.

Una vez establecidos los objetivos a alcanzar y las herramientas con que dispondremos para lograrlo, definimos las tareas específicas para la consecución de este trabajo:

Nombre de tarea	Duración	Comienzo	Fin
PLAN DE TRABAJO	6 días	mié 24/09/14	mié 01/10/14
Definición del plan de trabajo	5 días	mié 24/09/14	mar 30/09/14
Entrega del plan de trabajo PEC-1	0 días	mié 01/10/14	mié 01/10/14
ESTUDIO DE LAS NECESIDADES DE LA EMPRESA	10 días	mié 01/10/14	mar 14/10/14
Análisis de requerimientos	6 días	mié 01/10/14	mié 08/10/14
Búsqueda de solución óptima	4 días	jue 09/10/14	mar 14/10/14
ESTUDIO NECESIDADES DE EQUIPAMIENTO	3 días	mié 15/10/14	vie 17/10/14
Número de Raspberry Pi necesarias	2 días	mié 15/10/14	jue 16/10/14
Números de teléfonos IP necesarios	2 días	mié 15/10/14	jue 16/10/14
Proveedores de comunicaciones	3 días	mié 15/10/14	vie 17/10/14
Estudios de costes IP del sistema	2 días	mié 15/10/14	jue 16/10/14
ESTUDIO DE LA TIPOLOGIA DE RED ÓPTIMA	4 días	lun 20/10/14	jue 23/10/14
Definir sedes	4 días	lun 20/10/14	jue 23/10/14
Definir sedes troncales	3 días	lun 20/10/14	mié 22/10/14
APROVISAMIENTO DE EQUIPOS	7 días	jue 23/10/14	vie 31/10/14
Compra de Raspberry Pi	6 días	jue 23/10/14	jue 30/10/14
Compra de teléfonos IP	4 días	jue 23/10/14	mar 28/10/14
Compra de pantallas Adafruit	7 días	jue 23/10/14	vie 31/10/14
Compra de carcasas	3 días	jue 23/10/14	lun 27/10/14
Compra de alimentadores AC-USB	2 días	jue 23/10/14	vie 24/10/14
DESCARGA DE SOFTWARE	1 día	jue 23/10/14	jue 23/10/14
Descarga de Raspbian OS	1 día	jue 23/10/14	jue 23/10/14
Descarga de drivers	1 día	jue 23/10/14	jue 23/10/14
Descarga de Asterisk	1 día	jue 23/10/14	jue 23/10/14
INSTALACIÓN DE SOFTWARE EN RASPBERRY	9 días	vie 31/10/14	mié 12/11/14
Instalar Raspbian en Raspberry	3 días	vie 31/10/14	mar 04/11/14
Instalar drivers para la pantalla Adafruit	1 día	mié 05/11/14	mié 05/11/14
Instalar Asterisk	3 días	jue 06/11/14	lun 10/11/14
Configurar direcciones IP de la Raspberry	1 día	mar 11/11/14	mar 11/11/14
Configurar conexiones SSH para conexión remota	1 día	mié 12/11/14	mié 12/11/14
ENVIO DE EQUIPOS A SEDES	6 días	jue 13/11/14	jue 20/11/14
Envío de Raspberry a Madrid	2 días	jue 13/11/14	vie 14/11/14

Envío de Raspberry a Barcelona	2 días	jue 13/11/14	vie 14/11/14
Envío de Raspberry a Valladolid	2 días	jue 13/11/14	vie 14/11/14
Envío de Raspberry a Medellín	5 días	jue 13/11/14	mié 19/11/14
Envío de Raspberry a Bogotá	5 días	jue 13/11/14	mié 19/11/14
Envío de Raspberry a Cali	6 días	jue 13/11/14	jue 20/11/14
Envío de Raspberry a Santiago de Chile	5 días	jue 13/11/14	mié 19/11/14
Envío de Raspberry a Viña del Mar	5 días	jue 13/11/14	mié 19/11/14
Envío de Raspberry a Valparaíso	5 días	jue 13/11/14	mié 19/11/14
Envío de Raspberry a Oporto	2 días	jue 13/11/14	vie 14/11/14
CONFIGURACIÓN DE PLAN DE MARCACIÓN, TELÉFONOS Y TRONCALES	7 días	lun 17/11/14	mar 25/11/14
Configuración del plan de marcación de Madrid	3 días	lun 17/11/14	mié 19/11/14
Configuración del plan de marcación de Barcelona	3 días	lun 17/11/14	mié 19/11/14
Configuración del plan de marcación de Santiago	3 días	lun 17/11/14	mié 19/11/14
Configuración del plan de marcación de Valladolid	3 días	lun 17/11/14	mié 19/11/14
Configuración del plan de marcación de Medellín	3 días	jue 20/11/14	lun 24/11/14
Configuración del plan de marcación de Bogotá	3 días	jue 20/11/14	lun 24/11/14
Configuración del plan de marcación Cali	3 días	vie 21/11/14	mar 25/11/14
Configuración del plan de marcación de Santiago de Chile	3 días	jue 20/11/14	lun 24/11/14
Configuración del plan de marcación de Valparaíso	3 días	jue 20/11/14	lun 24/11/14
Configuración del plan de marcación de Viña del Mar	3 días	jue 20/11/14	lun 24/11/14
Configuración del plan de marcación de Oporto	3 días	lun 17/11/14	mié 19/11/14
Entrega PEC-2	0 días	mié 19/11/14	mié 19/11/14
PRUEBAS DE INTEGRACIÓN DE SISTEMAS	11 días	jue 20/11/14	jue 04/12/14
Probar la interconexión entre todas las sedes	6 días	jue 20/11/14	jue 27/11/14
Probar la interconexión con el ISPT	5 días	vie 28/11/14	jue 04/12/14
DOCUMENTACIÓN Y ENTREGA DEL SISTEMA	12 días	vie 05/12/14	lun 22/12/14
Documentar el proyecto	12 días	vie 05/12/14	lun 22/12/14
ELABORACIÓN MEMORIA Y PRESENTACIÓN	23 días	mié 17/12/14	dom 18/01/15
Puesta en limpio de la memoria	14 días	mar 23/12/14	dom 11/01/15
Entrega PEC-3	0 días	mié 17/12/14	mié 17/12/14
Entrega de la memoria del proyecto	18 días	mié 17/12/14	dom 11/01/15
Elaboración de la presentación del TFC	5 días	lun 12/01/15	dom 18/01/15

DIAGRAMA DE GANTT



2. VoIP vs Telefonía tradicional

Una vez que ya hemos definido lo que queremos hacer y los tiempos para llevarlo a cabo detallaremos a continuación las diferencias entre la telefonía tradicional y la telefonía VoIP y enumeraremos las ventajas de utilizar una solución VoIP para un entorno empresarial con sedes tanto nacionales como internacionales. Realizaremos una comparativa entre los distintos ITSP y analizaremos el ahorro de costes que se podría producir remplazando la solución de telefonía existente por esta otra solución VoIP.

2.1. Ventajas de la Telefonía IP, ¿Porque utilizar VoIP?

La primera ventaja y la más importante es el coste, una llamada mediante telefonía VoIP es en la mayoría de los casos mucho más barata que su equivalente en telefonía convencional.

Esto es básicamente debido a que se utiliza la misma red para la transmisión de datos y voz, la telefonía convencional tiene costos fijos que la telefonía IP no tiene, de ahí que ésta es más barata. Usualmente para una llamada entre dos teléfonos IP la llamada es gratuita, cuando se realiza una llamada de un teléfono IP a un teléfono convencional el coste corre a cargo del teléfono IP.

Existen otras ventajas más allá del coste para elegir a la telefonía IP:

1. Con VoIP uno puede realizar una llamada desde cualquier lado que exista conectividad a internet. Dado que los teléfonos IP transmiten su información a través de internet estos pueden ser administrados por su proveedor desde cualquier lugar donde exista una conexión. Esto es una ventaja para las personas que suelen viajar mucho, estas personas pueden llevar su teléfono consigo siempre teniendo acceso a su servicio de telefonía IP.
2. La mayoría de los proveedores de VOIP entregan características por las cuales las operadoras de telefonía convencional cobran tarifas aparte. Un servicio de VOIP incluye:
 - Identificación de llamadas.
 - Servicio de llamadas en espera
 - Servicio de transferencia de llamadas
 - Repetir llamada
 - Devolver llamada
 - Llamada de 3 líneas (three-way calling).
3. En base al servicio de identificación de llamadas existen también características avanzadas referentes a la manera en que las llamadas de un teléfono en particular son respondidas. Por ejemplo, con una misma llamada en Telefonía IP puedes:
 - Desviar la llamada a un teléfono particular.
 - Enviar la llamada directamente al correo de voz
 - Dar a la llamada una señal de ocupado.
 - Mostrar un mensaje de fuera de servicio

2.2. Estudio comparativo de diversos ITSP (proveedores de telecomunicaciones digitales basados en Voice over Internet Protocol (VoIP)).

En primer lugar realizamos una comparativa entre los diversos proveedores VoIP que podemos encontrar en el mercado para escoger el que mejores tarifas nos ofrece para nuestros dos troncales SIP para las llamadas externas.

Comparativa de precios de telefonía por internet

PROVEEDOR	ESPAÑA			POLONIA		USA	AUSTRALIA		FRANCIA		ALEMANIA		UK		PORTUGAL		MEXICO		RUSIA	
	Fijo	Estab. llamada en móvil	Móvil	Fijo	Móvil		Fijo	Móvil	Fijo	Móvil	Fijo	Móvil	Fijo	Móvil	Fijo	Móvil	Fijo	Móvil	Fijo	Móvil
SARENET	0,0100	0,0000	0,0200	0,0150	0,0150	0,0150	0,0180	0,0370	0,0150	0,0210	0,0150	0,0240	0,0150	0,0150	0,0170	0,0150	0,0290	0,0150	0,0150	0,1470
ALTECOM	0,0117	0,0000	0,0195	0,0186	0,0218	0,0114	0,0531	0,0545	0,0101	0,0351	0,0122	0,0341	0,1229	0,3128	0,0144	0,0227	0,0144	0,0401	0,0570	0,1935
4G PHONE	0,0146	0,0000	0,1881	0,0223	0,1704	0,0230	0,0146	0,1420	0,0146	0,1435	0,0146	0,2049	0,0146	0,1842	0,0146	0,2080	0,0821		0,0461	0,0415
ADIPTTEL	0,0175	0,0000	0,0604	0,0254	0,2363	0,4215	2,0603	0,0256	0,0399	0,0399	0,0470	0,0422	0,3121	0,0239	0,8537	0,0116	0,0617	0,2411	0,0317	0,0790
VOZELIA*	0,0150	0,1500	0,0320	0,0195	0,0217	0,0195	0,0520	0,0550	0,0195	0,0391	0,0195	0,0334	0,0195	0,0277	0,0195	0,0226	0,0195	0,0481	0,0195	0,1935
INGETEL	0,0180	0,0000	0,0800	0,0178	0,1311	0,0120	0,0243	0,1009	0,0092	0,0443	0,0114	0,1001	0,0092	0,0642	0,0182	0,0590	0,0426	0,0757	0,0517	0,0929
SKYPE*	0,0170	0,0000	0,0250	0,0170	0,0130	0,0170	0,0170	0,0430	0,0170	0,0250	0,0170	0,0450	0,0170	0,0250	0,0170	0,0250	0,0170	0,0220	0,0170	0,0350
ADAM TELEFONIA IP	0,0140	0,0000	0,0500	0,0184	0,3574	0,0086	0,1480	0,0672	0,0044	0,0368	0,0120	0,0454	0,0046	0,2805	0,0102	0,3060	0,0098	0,0498	0,0188	0,0542
VOZ TELECOM	0,0200	0,0000	0,0600	0,0500	0,3000	0,0400	0,0600	0,2600	0,0300	0,2300	0,0300	0,2300	0,0140	0,2300	0,0300	0,2300	0,1500	0,1800	0,1000	0,2600
TELSOME	0,0185	0,0500	0,0726	0,0200	0,0600	0,0900	0,0200	0,0600	0,0200	0,0600	0,0200	0,0600	0,0200	0,3060	0,0200	0,0600	0,1200	0,2840	0,1971	0,2199
VYKE	0,0120	0,0000	0,0410	0,0120	0,0240	0,0120	0,0120	0,0580	0,0120	0,0290	0,0120	0,0350	0,0120	0,0410	0,0120	0,0410	0,0350	0,0460	0,0240	0,1040

* Todas las llamadas realizadas con Skype están sujetas a un cargo por conexión de 0,049 € (sin IVA)
Tarifas en €/min IVA no incluido.

Figura 2. Comparativa de tarifas distintos proveedores IP.

Tenemos que encontrar el proveedor que mejor se adapte a nuestros intereses. Si comprobamos las tarifas vemos que el proveedor que tiene mejor tarifa para llamadas nacionales, que son las que mayormente se van realizar, es SARENET que tiene una tarifa a fijos de 0,01€/min (que es la que más vamos a utilizar) y a móviles de 0,02€/min.

Para España tenemos las siguientes tarifas:

precio min	precio establecimiento	Tipo
0.010€	0.000€	Spain / Fijo
0.015€	0.000€	Spain / Mobile
0.020€	0.000€	Spain / MOVILES
0.310€	0.510€	Spain / Premium
0.015€	0.000€	Spain Fixed / Madrid
0.360€	1.620€	Spain Information Services

Figura 3. Tarifas SARENET para España

Para destinos internacionales tenemos las siguientes tarifas.

precio min	precio establecimiento	Tipo
0.015€	0.000€	Portugal Azores & Madeira
0.015€	0.000€	Portugal Lisbon
0.017€	0.000€	Portugal Meo Mobile
0.017€	0.000€	Portugal Mobile Other
0.015€	0.000€	Portugal Olo
0.017€	0.000€	Portugal Optimus Mobile
0.015€	0.000€	Portugal Other
3.078€	0.000€	Portugal Special
0.017€	0.000€	Portugal Vodafone Mobile
0.023€	0.000€	Colombia Armenia
0.016€	0.000€	Colombia Baranquilla
0.023€	0.000€	Colombia Bogota
0.017€	0.000€	Colombia Bucaramanga
0.015€	0.000€	Colombia Cali
0.023€	0.000€	Colombia Cartagena
0.024€	0.000€	Colombia Comcel Mobile
0.108€	0.000€	Colombia Lex
0.015€	0.000€	Colombia Manizales
0.023€	0.000€	Colombia Medellin
0.077€	0.000€	Colombia Mobile
0.041€	0.000€	Colombia Movistar Mobile
0.024€	0.000€	Colombia Other
0.015€	0.000€	Colombia Pereira
0.069€	0.000€	Colombia Tigo Mobile
precio min	precio establecimiento	Tipo
0.377€	0.000€	Chile Easter Island
0.051€	0.000€	Chile Mobile
0.015€	0.000€	Chile Other
0.617€	0.000€	Chile Rural
0.015€	0.000€	Chile Santiago
0.754€	0.000€	Chile Special Services

Figura 4. Tarifas SARENET para el resto de sedes.

2.3. Cálculo de ahorro telefonía IP vs Tradicional

Una vez que hemos seleccionado el proveedor de VoIP para nuestro estudio podemos realizar el cálculo comparativo de cuanto nos cuesta una solución tradicional a través del proveedor de telefonía Movistar y cuanto nos cuesta una solución VoIP pura. Para ello se han estudiado las últimas facturas que se ha emitido por Movistar y se ha cogido un mes que represente el consumo medio mensual.

RESUMEN POR SERVICIO	
SERVICIOS	110,8295
CONSUMO	135,3950
Voz	110,8295
Línea (1 Ago. a 31 Ago.)	108,7992
Línea adicional acceso básico centralita RDSI: 39.2864	
Línea adicional acceso básico centralita RDSI: 72.5329	
Múltiple Numeración RDSI (1 Ago. a 31 Ago.)	2,0303
Voz	135,3950
Metropolitanas	79,8266
Provinciales	7,7941
Interprovinciales	42,9619
A Números 800/900	Gratis
A Servicios Avanzados (Novecientos)	4,7782
A S ² de Información y Emergencia	0,0342

Figura 5. Factura tipo media de servicios de Movistar.

Resumen de consumo (1 Jul. - 31 Jul.)

Voz				135,3950
Tipos de consumo	Llamadas	Duración	Importe	
Metropolitanas	439	13h 3m 12s	79,8266	
Provinciales	38	2h 5m 04s	7,7941	
Interprovinciales	86	6h 40m 46s	42,9619	
A Números 800/900	3	2m 24s	Gratis	
A Servicios Avanzados (Novcientos)	19	54m 05s	4,7782	
A S ² de Información y Emergencia	1	22s	0,0342	

Figura 6. Resumen duración de llamadas de factura de Movistar

Comparando la factura de Movistar con la que se correspondería si nuestro proveedor fuese de VoIP tenemos las siguientes tablas comparativas.

Tabla comparativa de gasto medio mensual antes de impuestos de una oficina de 18-20 usuarios (Movistar vs VoIP)

OFICINAS DE 18- 20 USUARIOS	PROVEEDOR TRADICIONAL					
	GASTO FIJO		CONSUMO		TOTAL	ANUAL
	RDSI	36,26 €	Metropolitanas	61,14 €		
RDSI	36,26 €	Provinciales	10,07 €			
Numeracion r	2,03 €	Interprovinciales	60,87 €			
		Moviles	0,98 €			
		Enlace móvil	57,38 €			
		Internacional	90,40 €			
	74,55 €		280,84 €	355,39 €	4.264,68 €	
PROVEEDOR VOIP						
	GASTO FIJO	CONSUMO		TOTAL	ANUAL	
	Numero	10,00 €				
		Metropolitanas	4,83 €			
		Provinciales	0,93 €			
		Interprovinciales	5,34 €			
		Moviles	20,14 €			
		Internacional	4,40 €			
	10,00 €		35,64 €	45,64 €	547,68 €	

Figura 7. Tabla comparativa entre el coste Movistar y el coste con proveedor VOIP.

En la tabla comparativa tenemos dos apartados:

- **Gasto fijo:**
Son los gastos fijos de comunicaciones, en el caso de las centrales tradicionales son los gastos de los dos RDSI que hay contratados. En el caso de la solución VoIP es el coste del número externo asociado a la centralita.
- **Consumo:**
Son lo que se ha consumido en llamadas tanto a fijos como a móvil. Los consumos se han obtenido de las facturas emitidas por Movistar en el caso del proveedor tradicional y se han estimado en el caso del proveedor VoIP al conocer el número de minutos que se ha llamado a móvil y fijo multiplicado por el valor en euros de la tarificación del proveedor VoIP (0,01€ a fijos y 0,02€ a móvil).

Ahorro de coste en centralitas de 18 – 20 usuarios (las cuatro centralitas de España) anual.

PROVEDOR TRADICIONAL		COSTE AÑO
4	4.264,68 €	17.058,72 €
PROVEDOR VOIP		COSTE AÑO
4	547,68 €	2.190,72 €
AHORRO ANUAL		14.868,00 €

Figura 8. Ahorro total anual al utilizar proveedor VOIP

Si comparamos el gasto entre las dos soluciones al año vemos que tenemos un ahorro de costes de **14.868€/año**.

3. Diseño de la red de comunicaciones.

Una vez que hemos definido que vamos a utilizar un sistema VoIP para implantar el sistema de comunicaciones y una vez que hemos definido el proveedor VoIP que vamos a utilizar para las llamadas externas a continuación en este punto detallaremos la topología de red. Definimos que soluciones hardware de red encontraremos en cada una de las sedes que tiene la compañía y las direcciones IP de cada uno de los equipos que nos servirá posteriormente para interconectar todos los Asterisk entre sí.

3.1. Diseño de la red entre sedes.

De las 11 sedes que cuenta la empresa tenemos que definir cuáles serán las troncales en España responsables de interconectar todos el sistema. Hemos definido dos troncales (Madrid y Barcelona) para que no tengamos problema en caso de que una de las centrales troncales falle, siempre tendremos la otra para que las comunicaciones se mantengan.

La infraestructura de la red de datos de los que dispone la empresa es la siguiente:

- Barcelona: VPNIP Metrolan (Fibra) 10/2 con IP 10.64.0.19
- Madrid: VPNIP Metrolan (Fibra) 10/2 con IP 10.64.1.19
- Valladolid: VPNIP ADSL iTelma Plus 10/1 con IP 10.64.2.19
- Santiago de Compostela: VPNIP Metrolan (Fibra) 10/2 con IP 10.64.3.19
- Oporto: VPNIP ADSL con IP 10.67.6.19
- Bogotá: VPNIP ADSL con IP 10.67.7.19
- Medellín: VPNIP ADSL con IP 10.67.8.19
- Cali: VPNIP ADSL con IP 10.67.9.19
- Santiago de Chile: VPNIP ADSL con IP 10.67.10.19
- Valparaíso: VPNIP ADSL con IP 10.67.11.19
- Viña del Mar: VPNIP ADSL con IP 10.67.12.19

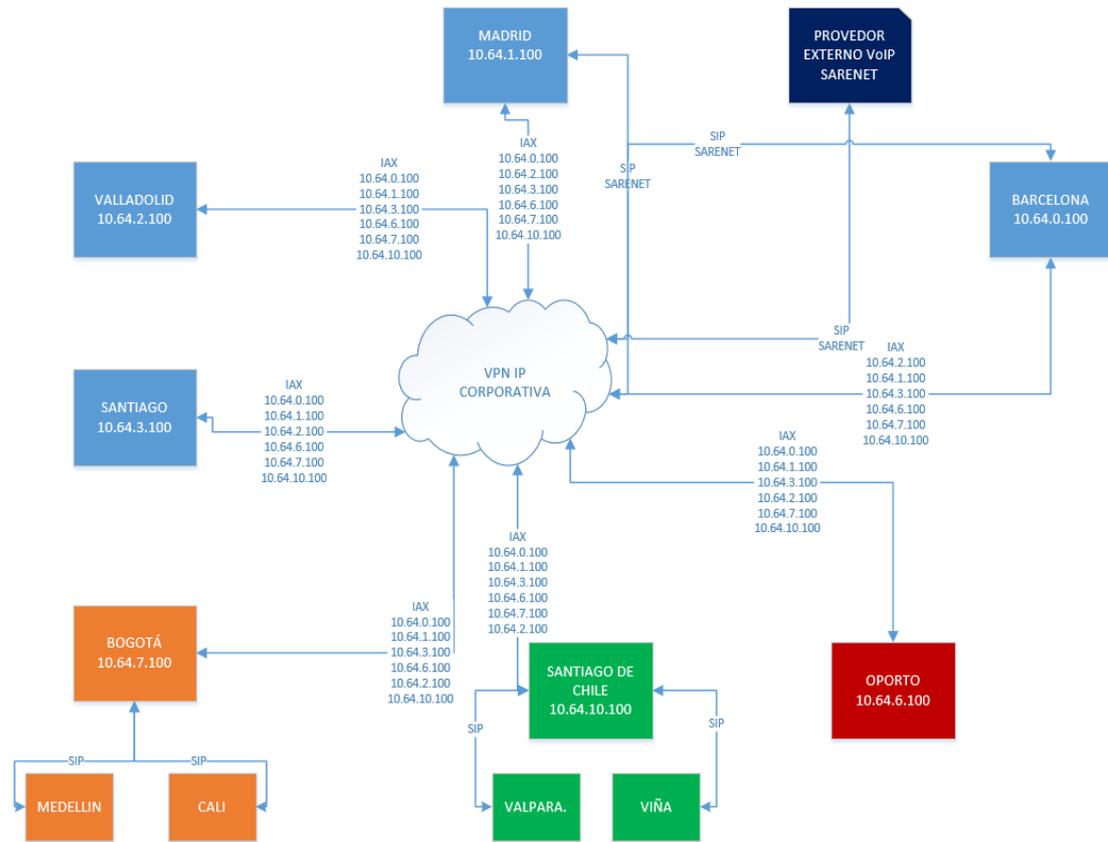


Figura 9. Esquema general del sistema. Sedes, direcciones IP y protocolos

Todos los equipos de red de los que disponemos son CISCO 1800 con el tráfico de red priorizada para comunicaciones IP a través de la VPN IP.

Tenemos pues dos troncales SIP en Madrid y Barcelona por ser las sedes que más usuarios tienen registrados y con las mejores comunicaciones. Ambas cuentan con conexiones por fibra Metrolan 10/2 de Movistar.

En cada oficina disponemos de un equipamiento de red compuesto por un Router, un Switch una Raspberry Pi y los teléfonos IP Temporis de Alcatel. En todas las oficinas menos en las secundarias de Colombia (Medellín y Cali) y de Chile (Valparaíso y Viña del Mar) que no necesitan Raspberry Pi ya que se registran contra las Raspberry de las sedes centrales de sus respectivos países.

El esquema general de las oficinas lo podemos observar en la Figura 10 que se muestra a continuación.

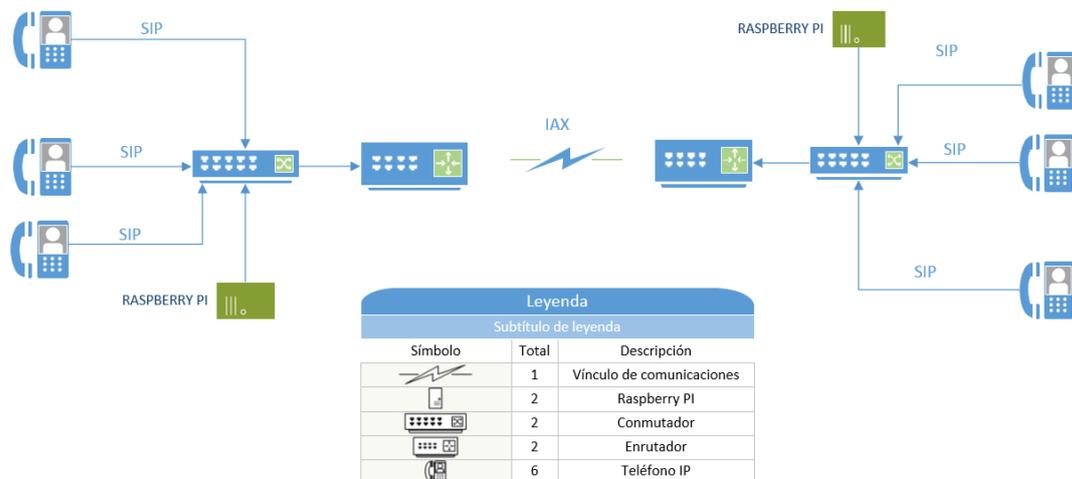


Figura 10. Esquema tipo de red de una oficina.

3.2. Análisis del hardware y del software que se utilizará en el desarrollo del proyecto.

En este punto ahora que ya tenemos la topología de red de comunicaciones que queremos implantar, la tecnología que vamos a utilizar y el proveedor VoIP que nos va dar salida de comunicaciones externa, a continuación detallamos todo el hardware que vamos a necesitar para implementar la solución en todas las sedes de la empresa.

Si recopilamos el número máximo de teléfonos que podríamos necesitar para cada una de las sedes de la red de comunicaciones obtenemos el siguiente listado:

- **Barcelona:** 2 Raspberry Pi + 20 terminales SIP.
- **Madrid:** 2 Raspberry Pi + 19 terminales SIP.
- **Valladolid:** 2 Raspberry Pi + 18 terminales SIP.
- **Santiago de Compostela:** 2 Raspberry Pi + 20 terminales SIP.
- **Oporto:** 2 Raspberry Pi + 7 terminales SIP.
- **Bogotá:** 2 Raspberry Pi + 5 terminales SIP.
- **Medellín:** 4 terminales SIP.
- **Cali:** 4 terminales SIP.
- **Santiago de Chile:** 2 Raspberry Pi + 6 terminales SIP.
- **Valparaíso:** 6 terminales SIP.
- **Viña del Mar:** 5 terminales SIP.

Para cada oficina necesitaremos una Raspberry Pi (excepto en las oficinas de Valparaíso, Viña del Mar, Cali y Medellín que conectan los teléfonos directamente a la central de cada país) para hacer las labores de centralita y de Gateway VoIP y otra clonada que sólo utilizaremos en caso

de que se estropee la original y tengamos que enviar otra de repuesto. Por lo tanto necesitaremos 16 Raspberry Pi.

El software que vamos a necesitar es por un lado la distribución de Linux para Raspberry Pi Raspbian OS y el software Asterisk. Se tendrán que instalar ambos en cada una de las Raspberry Pi.

3.3. Equipamiento para la solución.

A continuación vamos a describir cada uno de los componentes necesarios para implementar el proyecto de VoIP con Asterisk y Raspberry Pi.

3.3.1. Raspberry Pi.

Raspberry Pi es un ordenador en miniatura muy potente y ligero con procesador ARM, que se puede utilizar para muchas de las aplicaciones de un PC de sobremesa. Su gran funcionalidad y

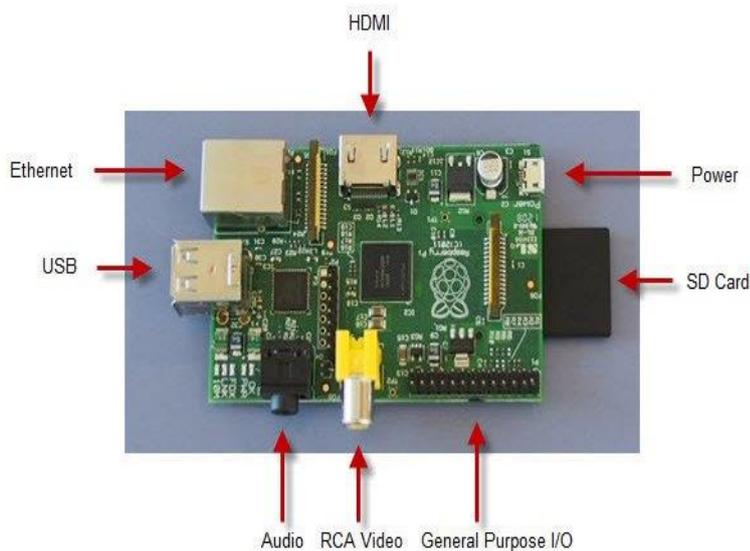


Figura 11. Raspberry Pi

amplia gama de accesorios permiten una variedad de usos casi infinita. Raspberry Pi es una placa computadora (SBC) de bajo costo desarrollada en Reino Unido por la Fundación Raspberry Pi, con el objetivo de estimular la enseñanza de ciencias de la computación en las escuelas.

El diseño de la Raspberry Pi incluye un System-on-a-chip Broadcom BCM2835, que contiene un procesador central (CPU) ARM1176JZF-S a 700 MHz (el firmware incluye unos modos Turbo para que el usuario pueda hacerle overclock de hasta 1 GHz

sin perder la garantía), un procesador gráfico (GPU) VideoCore IV, y 512 MB de memoria RAM aunque originalmente al ser lanzado eran 256 MB. El diseño no incluye un disco duro o una unidad de estado sólido, ya que usa una tarjeta SD para el almacenamiento permanente; tampoco incluye fuente de alimentación o carcasa.

- Agenda: 300 registros (capacidad de crear grupos y de mover a los contactos a una lista negra)
- Identificador de llamadas
- Multillamada (hasta 20)
- Indicador visual para las llamadas entrantes, llamadas perdidas, mensajes de voz, sin servicio...
- Gestión eficaz de las llamadas: llamada en espera, renvío de llamadas, transferencia de llamada ...
- Conferencia a 3
- Mute (silencio)
- Navegador XML
- 8 melodías
- Función "No molestar"
- Bloqueo del teclado
- Volumen ajustable (manos libres, teléfono, auricular y melodías)
- 8 idiomas disponibles
- Protocolos VoIP soportados: SIPv2, SIPv1
- Codecs de audio: G722.2, G711A, G711u, G726, G723.1, G729ab, VAD, CNG, AEC, PLC, AJB, AGC...
- Soporte ajustable con varias posiciones
- Montaje mural posible

3.3.4. Alcatel Temporis IP200 IP

El Alcatel Temporis IP200 es un terminal económico con muchas características (pantalla de 2 líneas, 2 cuentas SIP, conector para auriculares con tecla de acceso directo, auto alimentación PoE...).



Gran interoperabilidad

El Alcatel Temporis IP200 es un terminal con una gran interoperabilidad, que funciona fácilmente con la mayoría de los proveedores VoIP y con la mayor parte de centrales PBX IP.

Gestión eficaz de las llamadas

Silencio, espera, transferencia, desvío de llamadas, transferencia de llamada entrante sin descolgar, re-llamada, conferencia de tres (todos estos servicios de telefonía deben ser validados con su operador).

Características técnicas:

- Pantalla: 2 líneas de 15 caracteres + 1 línea de iconos
- 2 cuentas SIP
- Auto alimentación PoE (fuente de alimentación opcional)
- Manos libres full-dúplex
- 2 puertos Ethernet (10/100 Mbps - 1 LAN y 1 PC)

- Toma de auriculares RJ9
- Tecla directa casco
- 10 teclas programables
- Registro de llamadas: 100 entradas (efectuadas, recibidas, perdidas, reenviadas)
- Agenda: 300 registros (capacidad de crear grupos y de mover a los contactos a una lista negra)
- Identificador de llamadas
- Multillamada (hasta 20)
- Indicador visual para las llamadas entrantes, llamadas perdidas, mensajes de voz, sin servicio...
- Gestión eficaz de las llamadas: llamada en espera, envío de llamadas, transferencia de llamada ...
- Conferencia a 3
- Mute (silencio)
- 8 melodías
- Función "No molestar"
- Bloqueo del teclado
- Volumen ajustable (manos libres, teléfono, auricular y melodías)
- 8 idiomas disponibles
- Protocolos VoIP soportados: SIPv2, SIPv1
- Codecs de audio: G722.2, G711A, G711u, G726, G723.1, G729ab, VAD, CNG, AEC, PLC, AJB, AGC...
- Soporte ajustable con varias posiciones
- Montaje mural posible

4. Garantizar un servicio óptimo.

Una vez que ya hemos definido el tipo de solución que vamos a utilizar (VoIP), una vez que hemos definido la tipología de la red y las características del hardware que vamos a necesitar tenemos que a la hora de implantar un servicio telefónico en una empresa nos tendremos que asegurar que el servicio que vamos a implantar es el óptimo y cumple las expectativas que se esperan de él. Para garantizar este servicio óptimo tendremos que analizar los distintos problemas que pueden surgir y encontrar soluciones a estos problemas para que estos no ocurran o si ocurren afecten lo menos posible a las comunicaciones.

A continuación describiremos los que es QoS, los problemas que nos podemos encontrar a la hora de desplegar este sistema y las distintas opciones que tenemos de solucionarlos.

4.1. QoS-Calidad de Servicio para VoIP

QoS-Calidad de Servicio para VoIP QoS o Calidad de Servicio (Quality of Service, en inglés) son las tecnologías que garantizan la transmisión de cierta cantidad de información en un tiempo dado (throughput). Calidad de servicio es la capacidad de dar un buen servicio. Es especialmente importante para ciertas aplicaciones tales como la transmisión de vídeo o voz.

El auge de la telefonía IP es algo evidente y la principal razón es el reaprovechamiento de los recursos y la disminución en el coste de llamadas a través de Internet. Sin embargo, si de algo adolece todavía la VoIP es de la calidad de los sistemas telefónicos tradicionales. Los problemas de esta calidad son muchas veces inherentes a la utilización de la red (Internet y su velocidad y ancho de banda) y podrán irse solventando en el futuro. Mientras tanto, cuanto mejor conozcamos los problemas que se producen y sus posibles soluciones mayor calidad disfrutaremos.

4.2. Problemas que se producen en las comunicaciones VoIP y sus posibles soluciones

Detallaremos a continuación cuales son los distintos problemas que tendremos que tener en cuenta a la hora de implantar el sistema de comunicaciones. A final detallaremos que soluciones utilizaremos para minimizar estos problemas en nuestro sistema de comunicaciones.

4.2.1. Jitter

Causas:

El jitter es un efecto de las redes de datos no orientadas a conexión y basadas en conmutación de paquetes. Como la información se discretiza en paquetes cada uno de los paquetes puede seguir una ruta distinta para llegar al destino.

El jitter se define técnicamente como la variación en el tiempo en la llegada de los paquetes, causada por congestión de red, pérdida de sincronización o por las diferentes rutas seguidas por los paquetes para llegar al destino.

Las comunicaciones en tiempo real (como VoIP) son especialmente sensibles a este efecto. En general, es un problema frecuente en enlaces lentos o congestionados. Se espera que el aumento de mecanismos de QoS (calidad del servicio) como prioridad en las colas, reserva de ancho de banda o enlaces de mayor velocidad (100Mb Ethernet, E3/T3, SDH) puedan reducir los problemas del jitter en el futuro aunque seguirá siendo un problema por bastante tiempo.

Valores recomendados:

El jitter entre el punto inicial y final de la comunicación debiera ser inferior a 100 ms. Si el valor es menor a 100 ms el jitter puede ser compensado de manera apropiada. En caso contrario debiera ser minimizado.

Posibles soluciones:

La solución más ampliamente adoptada es la utilización del jitter buffer. El jitter buffer consiste básicamente en asignar una pequeña cola o almacén para ir recibiendo los paquetes y sirviéndolos con un pequeño retraso. Si alguno paquete no está en el buffer (se perdió o no ha llegado todavía) cuando sea necesario se descarta. Normalmente en los teléfonos IP (hardware y software) se pueden modificar los buffers. Un aumento del buffer implica menos pérdida de paquetes pero más retraso. Una disminución implica menos retardo pero más pérdida de paquetes.

4.2.2. Latencia

Causas:

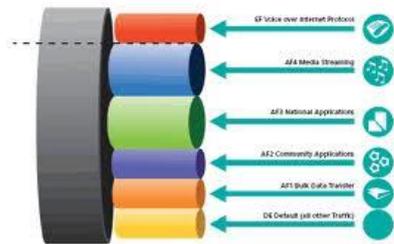


Figura 13. Distribución del ancho de banda

A la latencia también se la llama retardo. No es un problema específico de las redes no orientadas a conexión y por tanto de la VoIP. Es un problema general de las redes de telecomunicación. Por ejemplo, la latencia en los enlaces vía satélite es muy elevada por las distancias que debe recorrer la información. La latencia se define técnicamente en VoIP como el tiempo que tarda **un paquete en llegar desde la fuente al destino.**

Las comunicaciones en tiempo real (como VoIP) y full-dúplex son sensibles a este efecto. Es el problema de "pisarnos". Al igual que el jitter, es un problema frecuente en enlaces lentos o congestionados.

Es un problema general de las redes de telecomunicación. Por ejemplo, la latencia en los enlaces vía satélite es muy elevada por las distancias que debe recorrer la información.

La latencia se define técnicamente en VoIP como el tiempo que tarda un paquete en llegar desde la fuente al destino.

Las comunicaciones en tiempo real (como VoIP) y full-dúplex son sensibles a este efecto. Es el problema de "pisarnos". Al igual que el jitter, es un problema frecuente en enlaces lentos o congestionados.

Valores recomendados:

La latencia o retardo entre el punto inicial y final de la comunicación debiera ser inferior a 150 ms. El oído humano es capaz de detectar latencias de unos 250 ms, 200 ms en el caso de personas bastante sensibles. Si se supera ese umbral la comunicación se vuelve molesta.

Posibles soluciones:

No hay una solución que se pueda implementar de manera sencilla. Muchas veces depende de los equipos por los que pasan los paquetes, es decir, de la red misma. Se puede intentar reservar un ancho de banda de origen a destino o señalar los paquetes con valores de **ToS** para intentar que los equipos sepan que se trata de tráfico en tiempo real y lo traten con mayor prioridad pero actualmente no suelen ser medidas muy eficaces ya que no disponemos del control de la red.

Si el problema de la latencia está en nuestra propia red interna podemos aumentar el ancho de banda o velocidad del enlace o priorizar esos paquetes dentro de nuestra red.

ToS (Type of Service)

Tipo de servicio. Se suele corresponder con un campo de 8 bits de la cabecera de los datagramas IP que identifica la prioridad relativa de un paquete sobre otro. Los dispositivos de red usan este paquete para priorizar paquetes de forma adecuada y ponerles en las diferentes colas. (Bits 0-2: Precedente Bit 3: 0 = retardo normal, 1 = bajo retardo Bit 4: 0 = Throughput normal, 1 = Throughput Alto Bit 5: 0 = Fiabilidad Normal, 1 = Alta fiabilidad Bits 6-7: Reservados)

4.2.3. Eco

Causas:

El eco se produce por un fenómeno técnico que es la conversión de 2 a 4 hilos de los sistemas telefónicos o por un retorno de la señal que se escucha por los altavoces y se cuela de nuevo por el micrófono. El eco también se suele conocer como reverberación.

El eco se define como una reflexión retardada de la señal acústica original.

El eco es especialmente molesto cuanto mayor es el retardo y cuanto mayor es su intensidad con lo cual se convierte en un problema en VoIP puesto que los retardos suelen ser mayores que en la red de telefonía tradicional.

Valores recomendados:

El oído humano es capaz de detectar el eco cuando su retardo con la señal original es igual o superior a 10 ms. Pero otro factor importante es la intensidad del eco ya que normalmente la señal de vuelta tiene menor potencia que la original. Es tolerable que llegue a 65 ms y una atenuación de 25 a 30 dB.

Posibles soluciones:

En este caso hay dos posibles soluciones para evitar este efecto tan molesto.

- **Supresores de eco** - Consiste en evitar que la señal emitida sea devuelta convirtiendo por momentos la línea full-dúplex en una línea half-dúplex de tal manera que si se detecta comunicación en un sentido se impide la comunicación en sentido contrario. El tiempo de conmutación de los supresores de eco es muy pequeño. Impide una comunicación full-duplex plena.
- **Canceladores de eco** – Es el sistema por el cual el dispositivo emisor guarda la información que envía en memoria y es capaz de detectar en la señal de vuelta la misma información (tal vez atenuada y con ruido). El dispositivo filtra esa información y cancela esas componentes de la voz. Requiere mayor tiempo de procesamiento.

4.2.4. Pérdida de paquetes – packet loss

Causas:

Las comunicaciones en tiempo real están basadas en el protocolo UDP. Este protocolo no está orientado a conexión y si se produce una pérdida de paquetes no se renvían. Además la pérdida de paquetes también se produce por descartes de paquetes que no llegan a tiempo al receptor.

Sin embargo la voz es bastante predictiva y si se pierden paquetes aislados se puede recomponer la voz de una manera bastante óptima. El problema es mayor cuando se producen pérdidas de paquetes en ráfagas.

Valores recomendados:

La pérdida de paquetes máxima admitida para que no se degrade la comunicación deber ser inferior al 1%. Pero es bastante dependiente del códec que se utiliza. Cuanto mayor sea la compresión del códec más pernicioso es el efecto de la pérdida de paquetes. Una pérdida del 1% degrada más la comunicación si se usa el códec G.729 en vez del G.711.

Posibles soluciones:

Para evitar la pérdida de paquetes una técnica muy eficaz en redes con congestión o de baja velocidad es no transmitir los silencios. Gran parte de las conversaciones están llenas de momentos de silencio. Si solo transmitimos cuando haya información audible liberamos bastante los enlaces y evitamos fenómenos de congestión.

De todos modos este fenómeno puede estar también bastante relacionado con el jitter y el jitter buffer.

4.2.5. Ancho de banda insuficiente.

Causas:

El ancho de banda de las comunicaciones es limitado y suele estar compartido por numerosas aplicaciones (web, correo electrónico, tráfico FTP, descarga de archivos,...).

En conexiones a Internet el ancho de banda se define técnicamente como la cantidad de información o de datos que se puede enviar a través de una conexión de red en un período de tiempo dado. El ancho de banda se indica generalmente en bites por segundo (BPS), Kilobits por segundo (kbps), o Megabits por segundo (Mbps).

Si nuestras comunicaciones de VoIP comparten ancho de banda con otras aplicaciones puede que no tengamos suficiente capacidad para mantener correctamente una comunicación de Voz IP.

Valores recomendados:

El ancho de banda está fuertemente relacionado con el códec o codificación que estemos usando. Por ejemplo para una comunicación usando el códec G.711 codificamos la voz a 64 Kbps. Como tenemos que añadirle cabeceras para empaquetar los paquetes de voz podemos necesitar aproximadamente 80 Kbps de ancho de banda para una sola conversación (depende de los protocolos sobre los que encapsulemos)

Si utilizamos por ejemplo un códec como G.729 más comprimido y que codifica la voz a 8 Kbps necesitaremos, al añadirle las cabeceras unos 24 Kbps de ancho de banda para mantener una conversación.

Posibles soluciones:

Si tenemos problemas de ancho de banda podemos abordar el problema desde varios frentes:

- Aumentar el ancho de banda de las redes por las que circulen nuestras comunicaciones (normalmente pagando más)
- Reducir el consumo que hagan otras aplicaciones del ancho de banda (especialmente las descargas de archivos mediante redes de intercambio.
- Usar un códec con mayor compresión que usen menos ancho de banda. (Ej.: G729)

4.2.6. Técnicas para lograr QoS

Teniendo en cuenta que estamos hablando de la calidad de servicio en una interfaz y no a nivel de una red global, podemos decir lo que se intenta explicar en la siguiente figura:

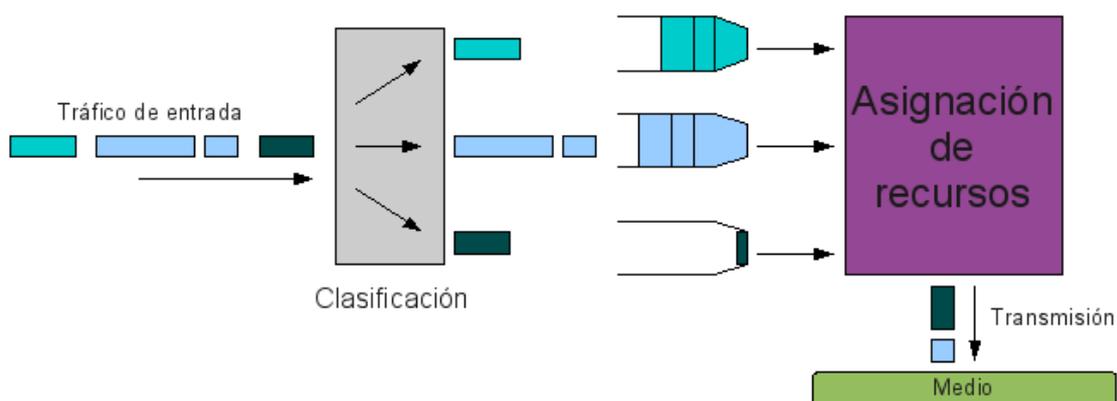


Figura 14. Asignación de recursos según tipo de tráfico

Este es un modelo que es bastante útil para comprender de qué formas podemos intentar garantizar la QoS y con cuáles conseguirlo. En esta figura están representadas las dos acciones fundamentales asociadas a garantizar la QoS:

- 1. Clasificación:** El tráfico que entra al equipo y que se ha de transmitir se tiene que clasificar. Pueden usarse muchos criterios de clasificación: Por equipo destino, por marcas en los paquetes, por aplicación... Es algo que siempre hay que hacer ya que si no el propio concepto de QoS no existe. Básicamente, la clasificación es buscar a qué parámetros de QoS negociados o contratados pertenece un paquete (o tráfico) en particular: Tráfico máximo en ráfaga, tráfico mínimo sostenido, latencia máxima, variación en la latencia...
- 2. Asignación de recursos:** Una vez que se tiene el tráfico clasificado, y por tanto se saben qué parámetros de QoS se deben cumplir, hay que asignar los recursos en la interfaz. Hay que permitir que los paquetes se transmitan al medio (el aire o un cable). La fase de clasificación es común a todos los tipos de interfaz que necesitan garantizar la QoS, pero la principal diferencia viene en la fase de asignación de recursos. Existen dos mecanismos que son lo bastante generales como para merecer que les demos un nombre “QoS a nivel 3 (L3QoS o IPQoS)” y “QoS a nivel 2 (L2QoS o MACQoS)”.

L3QoS: QoS a nivel IP

Las técnicas que se usan en este tipo de mecanismos de QoS son los típicos de los conformadores de tráfico o traffic shapers (TS). El TS clasifica el tráfico que entra en función de los criterios que se establezcan para cada una de los contratos de QoS. Es también conocida como QoS a nivel IP.

Una vez que el tráfico está clasificado, el TS asigna de una forma estadística los recursos de transmisión al medio. Por ejemplo si la cola de un servicio de baja latencia está muy llena, intentará vaciarla lo más rápido posible o por ejemplo si la cola de un servicio con tasa mínima garantizada tiene paquetes, intentará mantener en promedio a la salida esa tasa.

Estas técnicas de QoS a nivel 3, a veces llamados a nivel IP, son las clásicas basadas en colas de prioridades asociadas al DSCP o al TOS de las cabeceras IP, por ejemplo.

El problema que presentan las técnicas L3QoS es que no se conoce con exactitud la capacidad y la disponibilidad del medio sobre el que se transmiten. Imaginemos que tenemos un medio sin cables. El tráfico bruto puede depender del usuario al que se transmita, ya que podrían estar más lejos u obstruidos. Usar técnicas de L3QoS en estos casos, al desconocer la capacidad real por usuario destino, por ejemplo, lleva a una ineficiencia insalvable: “No se puede garantizar una QoS en términos absolutos, solo relativos”.

Esto quiere decir que si tenemos un servicio de 1Mbps y otro de 2Mbps, la única garantía que puede hacer un sistema de L3QoS es que el tráfico del primero va a ser la mitad que el del segundo, pero no puede garantizar cuál va a ser en realidad ese mínimo, ya que desconoce el estado y disponibilidad del medio.

Este problema aún se agrava mucho más en el caso en el que el medio está gestionado en contienda (WiFi, ethernet...) En estos casos el propio uso del medio es estadístico, ni siquiera el nivel 2 puede saber si podrá transmitir en un momento dado. Es más, en el caso de que la red

empiece a cursar mucho tráfico, es posible que un paquete jamás sea transmitido debido a las continuas colisiones.

L2QoS: QoS a nivel MAC

Cuando la asignación de recursos se hace a nivel 2, el sistema que va asignando los slots de transmisión conoce en todo momento tanto la disponibilidad del medio como la calidad o tráfico neto que es capaz de transmitir para cada usuario. Es también conocida como QoS a nivel MAC.

Esto hace posible implementar algoritmos que permitan garantizar de forma absoluta la asignación de tráfico.

4.3. Soluciones para evitar estos problemas en nuestro sistema.

Con el fin de evitar todos estos problemas que se han expuesto con anterioridad hemos creado una tabla con las diferentes soluciones que hemos aportado para solucionar cada uno de los tipos de problemas que nos podemos encontrar a la hora de implantar el sistema.

PROBLEMAS	SOLUCIÓN EN EL SISTEMA
Jitter	Utilizamos el jitter buffer en Asterisk: /etc/Asterisk/sip_general_custom.conf jbenable=yes jbimpl=adaptive
Latencia	Priorizar los paquetes de comunicaciones en los router CISCO que tenemos. Si calculamos la latencia los sitios más problemáticos son Chile y Colombia con latencias máximas de 160 y 180 respectivamente. Aunque son unos parámetros altos no impiden una buena recepción de la comunicación.
Eco	Cancelador de eco en Asterisk por software: Mediante el parámetro 'echocancel=yes'
Ancho de banda	Se utilizan redes con ancho de banda de subida superior a 2 Mbit cuando el ancho de banda necesario para 10 llamadas simultáneas son 112Kbps como se verá en el posteriormente.

4.4. Ancho de Banda.

El ancho de banda necesita un análisis individual ya que será el que nos indique que tipos de códec de comunicaciones podremos utilizar dependiendo del ancho de banda disponible.

Uno de los grandes problemas que se presentan cuando se instalan extensiones remotas a servidores Asterisk, en enlaces de datos de baja velocidad (menos de 1Mbps) es el limitado recurso de ancho de banda, problema que se agudiza aún más cuando estas extensiones se comunican por medio de Internet.



No es recomendable colocar extensiones en enlaces de datos de baja velocidad sin aplicar técnicas de calidad de servicio (QoS) ya que la calidad de la voz se verá afectada significativamente.

Un punto fundamental en el diseño de redes o rediseño de redes existentes para el transporte de voz sobre IP (VoIP), es el cálculo del ancho de banda necesario para la prestación adecuada del servicio.

El requerimiento del ancho de banda necesario en un enlace para el transporte de voz por paquetes sobre IP es el resultante de 2 factores:

1. El número de llamadas concurrentes.

Se suele denominar “llamadas concurrentes” a la estimación de la cantidad máxima de llamadas simultáneas que se podrán cursar sobre un enlace. Esta estimación debe considerar tanto la cantidad actual de llamadas telefónicas simultáneas entre diferentes puntos, como el posible margen de crecimiento y las políticas de la organización al respecto.

2. El requerimiento de ancho de banda para cursar cada conversación telefónica.

Cuando se implementa voz sobre IP se asume un conjunto de elecciones que impactan en ese requerimiento: CODEC, opciones de compresión, enlaces sobre los que se rutearán las llamadas, etc.

Para desarrollar el método utilizaremos un ejemplo que nos permita comprenderlo más fácilmente.

Calcular el tamaño de las tramas de voz.

Para esta tarea, podemos utilizar el siguiente procedimiento:

- Este parámetro es el resultado del CODEC utilizado, que da como resultado el tamaño de la porción de datos. A esto debe sumarse el tamaño de los encabezados de capa 4, capa 3 y capa 2.

Tamaño de trama = Payload + Enc. 4 + Enc. 3 + Enc. 2

Por ejemplo, las tramas obtenidas al utilizar G.729 tienen una longitud de 20 Bytes, a eso debemos sumarle los encabezados RTP, UDP e IP necesarios, que son 40B adicionales, y luego el encabezado de capa de enlace, que suponiendo que se trata de una trama PPP agrega 6B.

Tamaño de trama = 20B + 40B + 6B = 66B

- Dado el peso del encabezado en el tamaño de la trama a transmitir, en enlaces de bajo ancho de banda (menos de 768 Kbps) es conveniente aplicar compresión de los encabezados de capa 3 y capa 4, lo que se suele denominar compresión de RTP (cRTP). Esto reduce esos 40B iniciales a 2 o 4B. De este modo nuestro cálculo queda:

Tamaño de trama = 20B + 2B + 6B = 28B

- A fin de continuar el cálculo, es necesario convertir el tamaño expresado en Bytes a bits (1B = 8b):

28 Bytes x 8 bits/Byte = 224 bits/trama

Calcular el ancho de banda requerido por una llamada.

- Los CODECs actualmente utilizados para la digitalización de voz (G.711, G.728, G.729) generan 50 tramas por segundo.
- Para calcular el ancho de banda requerido para cada llamada debemos multiplicar el tamaño de cada trama por la cantidad de tramas que se envían por segundo:
 $BW/llamada = \text{tamaño de la trama} \times \text{tramas por segundo}$

$BW/llamada = 224 \text{ bits/trama} \times 50 \text{ tramas/seg.} = 11200 \text{ bps/llamada}$

Calcular el ancho de banda requerido en la implementación.

- Se debe considerar el número de llamadas concurrentes, y multiplicar el ancho de banda requerido para cada llamada por el número de llamadas concurrentes. $BW \text{ requerido} = BW/\text{llamada} \times \text{llamadas concurrentes}$. Para nuestro ejemplo vamos a suponer que se trata de cursar un máximo de 10 llamadas concurrentes generadas utilizando CODEC G.729 sobre un enlace PPP con cRTP.

$$BW \text{ requerido} = 11,200 \text{ Kbps} \times 10 = 112 \text{ Kbps}$$

Es preciso tener en cuenta que:

- El tamaño de la porción de datos depende del CODEC implementado.
- La porción correspondiente a los encabezados de capa 3 y 4 es de 40B si no se implementa compresión, y de 2 o 4B si se implementa cRTP.
- La porción correspondiente al encabezado de capa 2 está en función del enlace de que se trate. En este ejemplo he considerado un enlace PPP.
- Cuando se implementa voz sobre VPN IPSec, se debe considerar el overhead correspondiente a la implementación de IPSec que se esté aplicando.

3. Presupuesto de instalación del sistema VoIP.

En este punto ya tenemos definido el esquema de la red, la tecnología que vamos a utilizar y el hardware que queremos instalar y el método de garantizar un buen servicio. A continuación detallamos el presupuesto de instalación de todo el equipamiento necesario para la instalación del sistema de comunicaciones. Necesitaremos por un lado las Raspberry Pi y sus respectivas pantallas y fuentes de alimentación. En este caso necesitamos 2 Raspberry Pi para cada una de las sedes, una para instalar y otra clonada para quedar en retén para casos de emergencia.

Por otro lado tenemos los terminales telefónicos IP, tienen que soportar el protocolo SIP, que es el protocolo con el que vamos a conectar los teléfonos con el Asterisk. Necesitamos 7 teléfonos gama alta Temporis IP600 IP para cada una de las secretarías de cada una de las sedes de la empresa y 107 teléfonos gama media IP200 IP para cada uno de los usuarios de las oficinas de la empresa.

Alcatel Temporis IP200 IP

Precio	Unidades	Total
58,56 €	107	6.265,92



2x15 caracteres LCD + 1 Línea de iconos

2 cuentas SIP

Power over Ethernet

Manos libres Full duplex

Puerto RJ9 para auriculares y tecla para auriculares

Log de llamadas: 100 entradas (recibido, marcado, perdido, transferido)

Libreta de direcciones: local (300 entradas) con soporte a grupo, lista negra

Display con identificador de llamada (nombre & número)

Disponible en 8 idiomas

Alcatel Temporis IP600 IP

Precio:	Unidades	Total
80,96 €	7	566,72 €



LCD gráfico de 132x64 con luz de fondo y contraste ajustable

3 cuentas SIP

Power over Ethernet

Manos libre Full duplex

Puerto RJ9 para auriculares y tecla para auriculares

Log de llamadas: 100 entradas (recibido, marcado, perdido, transferido)

Libreta de direcciones: local (300 entradas) con soporte a grupo, lista negra y soporte LDAP y XML

Display con identificador de llamada (nombre & número)

Disponible en 11 idiomas

Alcatel Temporis Power Supply Unit 5V

Precio	Unidades	Total
9,90 €	114	1.128,60 €



5V Power supply unit for Alcatel Temporis series

Raspberry Pi + carcasa

Precio	Unidades	Total
39,95 €	14	559,30 €



ARM1176JZF-S (armv6k) a 700 MHz3

Broadcom VideoCore IV3

256 MiB

Tarjeta SD o SDHC

Linux ARM (Debian, Fedora, Arch Linux), RISC OS2

Cargador de red micro USB

Precio	Unidades	Total
--------	----------	-------

3,95 €

14

55,30 €



Cargador de red micro USB 800mAh.
 Entrada: 110 - 220V
 Salida: 5V

PiTFT - Assembled 320x240 2.8" TFT

Precio
42 €

Unidades
14

Total
588,00 €



Resolución 320x240
2,8" Resistiva

Diseño de la solución de comunicaciones

Total de horas de ingeniería para el diseño de la red de comunicaciones

1.500,00€

Mano de obra de instalación

Total horas de mano de obra de instalación de los equipos.

900,00€

Total presupuesto de la solución

11.583,64€

4. Velocidades y retardos que se requieren y plan de contingencias.

4.1. Velocidades y retardos

Tal como calculamos en capítulos anteriores necesitamos un ancho de banda para las comunicaciones superior a 112 Kbps que es lo necesario para poder dar salida a 10 llamadas concurrentes. También necesitamos un retardo inferior a 200ms para que no influya en la calidad de las llamadas.

En cuanto al ancho de banda tenemos 2Mbps de salida en cada una de la sedes con Macrolan de Barcelona, Santiago de Compostela y Madrid, 1 Mbps en la sede de Valladolid y cada una de las sedes internacionales (Oporto, Colombia y Chile). También recordamos que en cada uno de los router estamos priorizando el tráfico de VoIP.

Los retardos que encontramos entre las sedes nacionales si utilizamos el PING oscilan entre los 27 y 45 ms entre las sedes que se encuentran en España y los 141-156ms con Colombia y 170-190ms con Chile.

Con todo esto podemos decir que tenemos un ancho de banda suficiente para implantar la solución y unos retardos en las comunicaciones que no impiden una buena comunicación entre sedes.

4.2. Plan de contingencias.

En caso de que tengamos cualquier problema con los equipos siempre podremos utilizar la de repuesto que hemos clonado con la configuración y lista para funcionar. Sólo tendremos que conectarla a la red de datos como estaba conectada la anterior y ya tendríamos el sistema funcionando con normalidad.

Se dispondrá de dos troncales SIP con el proveedor de servicios VoIP para llamadas externas, uno en Barcelona y el otro en Madrid. Se utilizan dos troncales SIP para evitar la interrupción del servicio si una de las dos sedes con trocales SIP cae. Cuando alguna de las sedes que no son Madrid y Barcelona necesita realizar una llamada a números locales o a números internacionales sale por el troncal IAX2 que tiene asignado (Madrid o Barcelona), trata de realizar la comunicación y si no es capaz (porque está ocupado el canal o no disponible) marca el troncal IAX2 secundario para establecer comunicación con Madrid o Barcelona según sea el caso.

También hemos de salientar que dado el tamaño que la Raspberry PI tiene se puede enviar otra clonada para reemplazar la que se ha estropeado en 48h desde España a cualquiera de las sedes de la empresa.

5. Conclusiones.

En este proyecto hemos comprobado cómo podemos mediante los dispositivos Raspberry PI y el software Asterisk dar el mismo servicio de comunicaciones que las centrales tradicionales suministradas por empresas como Siemens, Alcatel, Avaya, Panasonic, etc...

Podemos mediante proveedores VoIP de comunicaciones reducir considerablemente los costes de comunicaciones, hemos calculado de alrededor de unos 15.000 euros al año, lo que implica que en un año podemos amortizar el gasto en el que hemos incurrido para comprar todo el equipamiento que asciende a 11.583,64 euros.

Por último y no menos importante, podemos modificar las capacidades de nuestra centralita VoIP a nuestro antojo. Podemos incluir un IVR sin coste, poner buzones de voz a todos los usuarios que queramos habilitar colas, enviar fax a correos electrónicos, etc...

Anexos.

Compra de los equipos.

En este paso tenemos lo que es la compra de los equipos Raspberry Pi, la compra de los terminales SIP, la compra de los visores TFT de Adafruit, compra de carcasas para los equipos y por último la compra de los alimentadores AC-USB. Existen muchos proveedores que suministran este tipo de equipamiento pero nosotros aportamos los siguientes enlaces en los que se pueden obtener los productos:

PiTFT Touchscreen

http://www.amazon.es/PiTFT-Mini-Kit-Touchscreen-Raspberry/dp/B00H9B1DTA/ref=sr_1_3?ie=UTF8&qid=1416406703&sr=8-3&keywords=raspberry+pi

Raspberry Pi

http://www.amazon.es/Raspberry-Pi-RBCA000-Model-1176JZF-S/dp/B008PT4GGC/ref=sr_1_6?ie=UTF8&qid=1416406703&sr=8-6&keywords=raspberry+pi

Alcatel temporis IP 200

<http://www.voipango.com/en/voip-telephones/alcatel/alcatel-temporis-ip200-ip-phone-without-psu.html>

Alcatel temporis IP 600

<http://www.voipango.com/en/voip-telephones/alcatel/alcatel-temporis-ip600-ip-phone-without-psu.html>

Power supply

<http://www.voipango.com/en/voip-telephones/alcatel/alcatel-temporis-power-supply-unit-5v.html>

Descarga del software, ensamblaje de los equipos e instalación del software.

En esta parte analizaremos como descargar el software necesario (Raspbian OS, Asterisk y drivers para la TFT de Adafruit).

Analizaremos como ensamblar los terminales con la pantalla y la instalación de la carcasa para cada equipo.

Por último indicaremos como instalar el software que hemos descargado previamente en la Raspberry Pi, dependencias necesarias para una correcta instalación de Asterisk sobre Raspbian OS y los primeros pasos de configuración del Sistema Operativo.

Lo primero que hay que hacer es descargar la imagen de la última versión Debian para Raspberry.

http://downloads.raspberrypi.org/raspbian_latest

Instalación del sistema Raspbian OS.

Una vez terminada la descarga, se descomprime. Tendremos un archivo img:

[2014-09-09-wheezy-raspbian.img](#)

Luego se descarga Win32 Disk Manager:

<http://downloads.sourceforge.net/project/win32diskimager/Archive/win32diskimager-v0.9-binary.zip>

Se descomprime y se guardan los archivos en una carpeta. Se pone una tarjeta de memoria en el lector del ordenador (en este caso 8GB clase 10) y se ejecuta el siguiente archivo:

[Win32DiskImager.exe](#)

Se selecciona la imagen descargada, la unidad donde se va a escribir y por último se presiona el botón Write. Empezará la escritura de la Raspbian en la tarjeta de memoria. En algunos casos, para que funcione, hay que cerrar todas las ventanas abiertas (navegador, explorador, etc.).



Figura 15. Raspberry Pi funcionando con PITFT ensamblada



Figura 15. Raspberry Pi en caja de metacrilato.

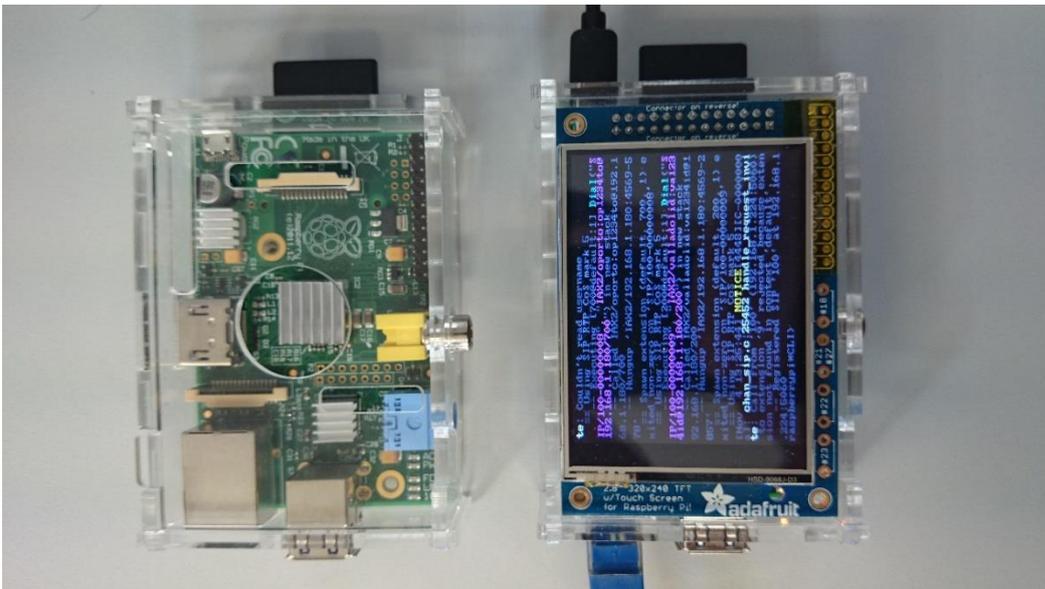


Figura 16. Comparativa entre dos Raspberry Pi, una con PiTFT y otra sin ella.

Una vez terminada la escritura se saca la tarjeta de memoria del computador y se inserta en la ranura del Raspberry pi. Se conecta la alimentación y se espera que el sistema arranque. En mi caso para conocer la IP asignada al equipo he consultado el servidor DHCP de mi router. Con la IP se accede vía SSH con los siguientes datos:

user: pi
password: Raspberry

Lo primero que hay que hacer es cambiar la contraseña del usuario pi y del usuario root:

passwd

*Cambiando la contraseña de pi.
 (actual) contraseña de UNIX:
 Introduzca la nueva contraseña de UNIX:
 Vuelva a escribir la nueva contraseña de UNIX:
 passwd: contraseña actualizada correctamente*

Luego la contraseña de root:

sudo passwd root

*Introduzca la nueva contraseña de UNIX:
 Vuelva a escribir la nueva contraseña de UNIX:
 passwd: contraseña actualizada correctamente*

Se ejecuta la utilidad de configuración de Raspbian:

raspi-config

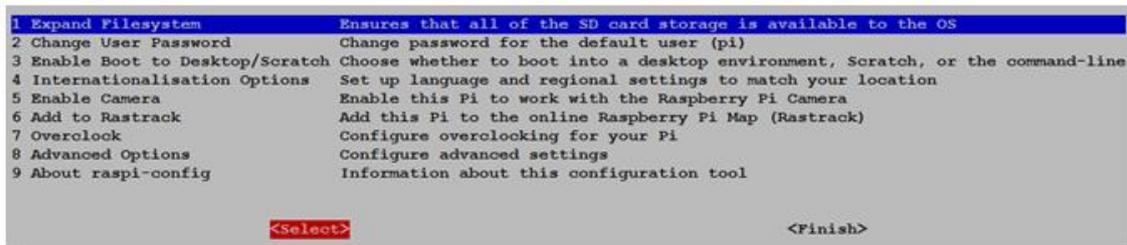


Figura 17. Menú de configuración de Raspbian

Se escoge la opción 8 → A6 para actualizar la utilidad. Luego la opción A2 para configurar el nombre de dominio.

Se escoge la opción 1 para asignar todo el espacio disponible en la tarjeta al sistema operativo.

Se escoge la opción 4 para la configuración del idioma, la zona horaria y el idioma del teclado

Una vez terminado se sale de la utilidad.

La primera operación es actualizar el sistema:

```
# apt-get update
```

```
# apt-get upgrade
```

La operación puede tardar un poco dependiendo de cuantos paquetes se descargarán. Se continúa configurando una IP estática. Se crea una copia del archivo original:

```
# cp /etc/network/interfaces /etc/network/interfaces.old
```

Se abre el archivo:

```
# nano /etc/network/interfaces
```

y se modifica para que quede:

```
# auto lo
```

```

# iface lo inet loopback
# iface eth0 inet dhcp
# The loopback interface
auto lo
iface lo inet loopback
auto eth0
iface eth0 inet static
#your static IP
address 10.64.3.100
# your gateway IP
gateway 10.64.3.19
netmask 255.255.255.0
#your network address "family"
network 10.64.0.0
broadcast 10.64.0.255

```

```

allow-hotplug wlan0
iface wlan0 inet manual
wpa-roam /etc/wpa_supplicant/wpa_supplicant.conf
iface default inet dhcp

```

En mi caso el router tiene la IP 10.64.3.19 y la IP estática que configuré es 10.64.3.100.

Luego se reinicia el servicio de red:

```
# /etc/init.d/networking restart
```

Ahora reiniciamos el sistema:

```
# reboot
```

Volvemos a entrar vía SSH con la nueva IP configurada.

Ahora instalamos una serie de paquetes y dependencias para la instalación de Asterisk:

```

#apt-get install libsqlite3-dev libncurses5-dev libxml2-dev libnewt-dev libssl-dev libiksemel-
dev libgnutls-dev libcurl3-dev libspandsp2 libspandsp-dev mysql-client mysql-server
libmysqld-dev unixodbc-dev libmyodbc unixodbc libportaudio-dev libical-dev libneon27-dev
portaudio19-dev libspeex-dev libvorbis-dev libsrtp-dev iptables-persistent libsox2 libsox-dev
sox sendmail

```

Descargamos la versión 11.2 certified de Asterisk:

```
# cd /usr/src
```

```

# wget http://downloads.asterisk.org/pub/telephony/certified-asterisk/certified-asterisk-
11.2-cert2.tar.gz

```

Descomprimos el archivo y entramos en la carpeta creada:

```
# tar -xf certified-asterisk-11.2-cert2.tar.gz
```

```
# cd certified-asterisk-11.2-cert2
```

Compilamos:

```
# ./configure
```

```
# make menuselect
```

Seleccionamos los módulos que queremos y salimos del menú de configuración:

```
# make
```

```
# make install
```

```
# make samples
```

Ya se puede iniciar Asterisk:

```
#!/etc/init.d/asterisk start
```

Ahora vamos a instalar la pantalla TFT de Adafruit.

Descargamos e script de instalación:

```
wget http://adafru.it/pitftsh
mv pitftsh pitft.sh
chmod +x pitft.sh
```

```
pi@raspberrypi:~$ wget http://adafru.it/pitftsh
--2014-08-25 20:49:32-- http://adafru.it/pitftsh
Resolving adafru.it (adafru.it)... 207.58.139.245
Connecting to adafru.it (adafru.it)|207.58.139.245|:80... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://gist.githubusercontent.com/ladyada/f86f777df68fe41181b0/raw/ [following]
--2014-08-25 20:49:32-- https://gist.githubusercontent.com/ladyada/f86f777df68fe41181b0/raw/
Resolving gist.githubusercontent.com (gist.githubusercontent.com)... 192.30.252.157
Connecting to gist.githubusercontent.com (gist.githubusercontent.com)|192.30.252.157|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/plain]
Saving to: `pitftsh'

  [ <=>          ] 13,001  --.-K/s  in 0.05s

2014-08-25 20:49:38 (260 KB/s) - `pitftsh' saved [13001]

pi@raspberrypi:~$ mv pitftsh pitft.sh
pi@raspberrypi:~$ chmod +x pitft.sh
pi@raspberrypi:~$
```

Lo ejecutamos

```
sudo ./pitft.sh -t 28r -r
```

```

Connecting to adafruit-download.s3.amazonaws.com (adafruit-download.s3.amazonaws.com) |17
6.32.102.57|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 184818 (180K) [application/octet-stream]
Saving to: `libraspberrypi-bin-adafruit.deb'

100%[=====>] 184,818      306K/s   in 0.6s

2014-08-25 20:52:12 (306 KB/s) - `libraspberrypi-bin-adafruit.deb' saved [184818/184818]

--2014-08-25 20:52:12--  http://adafruit-download.s3.amazonaws.com/libraspberrypi-dev-ad
afruit.deb
Resolving adafruit-download.s3.amazonaws.com (adafruit-download.s3.amazonaws.com) ... 176
.32.102.57
Connecting to adafruit-download.s3.amazonaws.com (adafruit-download.s3.amazonaws.com) |17
6.32.102.57|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 672150 (656K) [application/octet-stream]
Saving to: `libraspberrypi-dev-adafruit.deb'

100%[=====>] 672,150      684K/s   in 1.0s

2014-08-25 20:52:13 (684 KB/s) - `libraspberrypi-dev-adafruit.deb' saved [672150/672150]

--2014-08-25 20:52:13--  http://adafruit-download.s3.amazonaws.com/libraspberrypi-doc-ad
afruit.deb
Resolving adafruit-download.s3.amazonaws.com (adafruit-download.s3.amazonaws.com) ... 176
.32.102.57
Connecting to adafruit-download.s3.amazonaws.com (adafruit-download.s3.amazonaws.com) |17
6.32.102.57|:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 31480442 (30M) [application/octet-stream]
Saving to: `libraspberrypi-doc-adafruit.deb'

3% [>
] 1,033,977      674K/s

```

```

Ign http://raspberrypi.collabora.com wheezy/rpi Translation-en
Ign http://archive.raspberrypi.org wheezy/main Translation-en
Hit http://mirrordirector.raspbian.org wheezy/contrib armhf Packages
Get:10 http://mirrordirector.raspbian.org wheezy/non-free armhf Packages [49.3 kB]
Hit http://mirrordirector.raspbian.org wheezy/rpi armhf Packages
Ign http://mirrordirector.raspbian.org wheezy/contrib Translation-en_GB
Ign http://mirrordirector.raspbian.org wheezy/contrib Translation-en
Ign http://mirrordirector.raspbian.org wheezy/main Translation-en_GB
Ign http://mirrordirector.raspbian.org wheezy/main Translation-en
Ign http://mirrordirector.raspbian.org wheezy/non-free Translation-en_GB
Ign http://mirrordirector.raspbian.org wheezy/non-free Translation-en
Ign http://mirrordirector.raspbian.org wheezy/rpi Translation-en_GB
Ign http://mirrordirector.raspbian.org wheezy/rpi Translation-en
Fetched 6,992 kB in 48s (143 kB/s)
Reading package lists... Done
[PITFT] Installing Adafruit kernel...
total 54408
-rw-r--r-- 1 root root 444188 Aug 25 20:57 libraspberrypi0-adafruit.deb
-rw-r--r-- 1 root root 184818 Aug 25 20:57 libraspberrypi-bin-adafruit.deb
-rw-r--r-- 1 root root 672150 Aug 25 20:57 libraspberrypi-dev-adafruit.deb
-rw-r--r-- 1 root root 31480442 Aug 25 20:57 libraspberrypi-doc-adafruit.deb
-rw-r--r-- 1 root root 22918510 Aug 25 20:57 raspberrypi-bootloader-adafruit-20140724-1.
deb
Selecting previously unselected package libraspberrypi-bin-adafruit.
dpkg: considering removing libraspberrypi-bin in favour of libraspberrypi-bin-adafruit .
..
dpkg: yes, will remove libraspberrypi-bin in favour of libraspberrypi-bin-adafruit
(Reading database ... 69052 files and directories currently installed.)
Unpacking libraspberrypi-bin-adafruit (from ../libraspberrypi-bin-adafruit.deb) ...
Selecting previously unselected package libraspberrypi-dev-adafruit.
dpkg: considering removing libraspberrypi-dev in favour of libraspberrypi-dev-adafruit .
..
dpkg: yes, will remove libraspberrypi-dev in favour of libraspberrypi-dev-adafruit
Unpacking libraspberrypi-dev-adafruit (from ../libraspberrypi-dev-adafruit.deb) ...

```

```

ikernelhack'
Removing 'diversion of /boot/start_x.elf to /usr/share/rpikernelhack/start_x.elf by rpik
ernelhack'
Setting up libraspberrypi0-adafruit (1.20130902-1) ...
Setting up libraspberrypi-bin-adafruit (1.20130902-1) ...
Setting up libraspberrypi-dev-adafruit (1.20130902-1) ...
Setting up libraspberrypi-doc-adafruit (1.20130902-1) ...
[PITFT] Updating /etc/modules...
Adding spi-bcm2708
Adding fbtft_device
[PITFT] Updating /etc/modprobe.d/adafruit.conf...
[PITFT] Updating X11 default calibration...
[PITFT] Updating X11 setup tweaks...
Moving 99-fbturbo.conf to /home/pi
Adding 'export FRAMEBUFFER=/dev/fb1'
[PITFT] Updating TSLib default calibration...
[PITFT] Updating SysFS rules for Touchscreen...
[PITFT] Installing evtest tslib libts-bin...
Reading package lists... Done
Building dependency tree
Reading state information... Done
Note, selecting 'libts-0.0-0' instead of 'tslib'
libts-0.0-0 is already the newest version.
libts-0.0-0 set to manually installed.
The following NEW packages will be installed:
  evtest libts-bin
0 upgraded, 2 newly installed, 0 to remove and 48 not upgraded.
Need to get 55.0 kB of archives.
After this operation, 219 kB of additional disk space will be used.
Get:1 http://mirrordirector.raspbian.org/raspbian/ wheezy/main libts-bin armhf 1.0-11 [3
3.1 kB]
Get:2 http://mirrordirector.raspbian.org/raspbian/ wheezy/main evtest armhf 1:1.30-1 [21
.9 kB]
Fetched 55.0 kB in 5s (9,200 B/s)

```

Si queremos o no que aparezca la consola de texto.

```

3.1 kB]
Get:2 http://mirrordirector.raspbian.org/raspbian/ wheezy/main evtest armhf 1:1.30-1 [21
.9 kB]
Fetched 55.0 kB in 5s (9,200 B/s)
Selecting previously unselected package libts-bin.
(Reading database ... 68940 files and directories currently installed.)
Unpacking libts-bin (from ../libts-bin_1.0-11_armhf.deb) ...
Selecting previously unselected package evtest.
Unpacking evtest (from ../evtest_1:1.30-1_armhf.deb) ...
Processing triggers for man-db ...
Setting up libts-bin (1.0-11) ...
Setting up evtest (1:1.30-1) ...
Would you like the console to appear on the PiTFT display? [y/n] y

```

Si queremos que actúe como botón on/off

```

.9 kB]
Fetched 55.0 kB in 5s (9,200 B/s)
Selecting previously unselected package libts-bin.
(Reading database ... 68940 files and directories currently installed.)
Unpacking libts-bin (from ../libts-bin_1.0-11_armhf.deb) ...
Selecting previously unselected package evtest.
Unpacking evtest (from ../evtest_1:1.30-1_armhf.deb) ...
Processing triggers for man-db ...
Setting up libts-bin (1.0-11) ...
Setting up evtest (1:1.30-1) ...
Would you like the console to appear on the PiTFT display? [y/n] y
[PITFT] Updating console to PiTFT...
Would you like GPIO #23 to act as a on/off button? [y/n] n

```

Ejecutamos el comando **sudo reboot** para reiniciar y usar nuestra pantalla.


```

[incoming]
exten => s,1,Goto(default,300,1)                ;Número principal suena en
                                                ;la operadora

exten => t,1,Goto(default,300,1);
exten => i,1,Goto(default,300,1);

[from-sarnet]
exten => s,1,Answer
exten => s,n,Wait(1)
exten => s,n,Goto(IVR,s,1)

[local]
ignorepat => 0
exten => _09XXXXXXXX,1,Goto(trunkdial,${EXTEN},1)
exten => _08XXXXXXXX,1,Goto(trunkdial,${EXTEN},1)
exten => _06XXXXXXXX,1,Goto(trunkdial,${EXTEN},1)
exten => _07XXXXXXXX,1,Goto(trunkdial,${EXTEN},1)

include => default

[longdistance]
ignorepat => 0
exten => _00XXXXXXXXXXXX,1,Goto(trunkdial,${EXTEN},1)
include=>local

[trunkdial]
exten => _0.,1,Dial(${TRUNK}/${EXTEN:${TRUNKMSD}})
exten => _0.,2,Congestion(5)
exten => _0.,3,Hangup

[default]
include=>local
include=>incoming
exten => s,1,Goto(default,300,1)
exten => t,1,Goto(default,300,1)
exten => i,1,Goto(default,300,1)

;Secretaria extension 300
exten => 300,1,Macro(stdexten,${EXTEN},SIP/${EXTEN})

;Departamento de Sistemas
exten => _30[25],1,Macro(stdexten,${EXTEN},SIP/${EXTEN})

;Departamento financiero, extensiones 305-309
exten => _30[5-9],1,Macro(stdexten,${EXTEN},SIP/${EXTEN})

;Departamento Técnico, extensiones 310-313
exten => _31[0-3],1,Macro(stdexten,${EXTEN},SIP/${EXTEN})

;Departamento RRHH, extensiones 314-317
exten => _31[4-7],1,Macro(stdexten,${EXTEN},SIP/${EXTEN})

;Administracion, ext 320, no necesita buzón de voz
exten => 320,1,Macro(novm,SIP/${EXTEN})

```

TFC: COMUNICACIONES VOIP. RASPBERRY Pi + ASTERISK

;Para entrar en el menú del buzón de voz marcamos extensión 800

exten => 800,1,Answer

exten => 800,2,VoicemailMain

exten => _85X,1,Answer

exten => _85X,2,MeetMe(\${EXTEN})

exten => 888,1,Goto(dialext,s,1)

; Si llamamos al patrón de las extensiones de cada una de las otras sedes

; salimos por el troncal IAX2 definido para cada una de las sedes (usuario y contraseña)

;Oficina de Valladolid

exten => _2XX,1,Dial(IAX2/valladolid:va1234id@10.64.2.100/\${EXTEN})

;Oficina de Madrid

exten => _4XX,1,Dial(IAX2/madrid:ma1234id@10.64.1.100/\${EXTEN})

;Oficina de Santiago de Compostela

exten => _1XX,1,Dial(IAX2/santiago:sa1234go@10.64.3.100/\${EXTEN})

;Oficina de Bogotá

exten => _5[0-4],1,Dial(IAX2/colombia:co1234ia@10.64.7.100/\${EXTEN})

;Oficina de Santiago de Chile

exten => _5[5-9]X,1,Dial(IAX2/chile:ch1234le@10.64.10.100/\${EXTEN})

;Oficina de Oporto

exten => _7XX,1,Dial(IAX2/oporto:op1234to@10.64.6.100/\${EXTEN})

[dialext]

include => default

exten => s,1,Answer

exten => s,2,DigitTimeout(5)

exten => s,3,ResponseTimeout(20)

exten => s,4,Background(pleaseenterextension) ; "Por favor introduzca la
; extensión del usuario que quiere llamar."

exten => 9,1,Directory(default) ; pulse 9 para entrar en el directorio

exten => 9,2,Goto(dialext,9,1)

exten => 0,1,Goto(default,100,1) ; envía a la operadora como cortesía
; si se pulsa la tecla 0

exten => i,1,Playback(privacy-incorrect)

exten => i,2,Goto(dialext,s,1)

exten => t,1,Goto(dialext,i,1)

#include "IVR"

sip.conf

[general]

context=default

port=5060

bindaddr=0.0.0.0

disallow=all

allow=ulaw

[sarevoz]

type=peer

```
host=sarevoz.com
disallow=all
allow=g729
allow=alaw
defaultuser=24162
fromuser=24162
fromdomain=sarevoz.com
secret=iwonttellyou
directmedia=no
sendrpid=pai
```

```
[300]
type=friend
context=default
callerid=Marina Palacios<300>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=300
```

```
[301]
type=friend
context=default
callerid=Javier Garcia<301>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=301
```

```
[302]
type=friend
context=default
callerid=Elena Sanchez<302>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=302
```

```
[303]
type=friend
context=default
callerid=Juan Crespo<303>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=303
```

```
[304]
type=friend
context=default
callerid=Leonor Sanchez<304>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=304
```

```
[305]
type=friend
```

context=default
callerid= Saul Garcia<305>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=305

[306]
type=friend
context=default
callerid=John Daniels<306>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=306

[307]
type=friend
context=default
callerid=Oscar Pintos<307>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=307

[308]
type=friend
context=default
callerid=Sergio Vazquez<308>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=308

[309]
type=friend
context=default
callerid=Pedro Grande<309>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=309

[310]
type=friend
context=default
callerid=Mar Castro<310>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=310

[311]
type=friend
context=default
callerid=Elena Losada<311>

host=dynamic
secret=123456
dtmfmode=inband
mailbox=311

[312]
type=friend
context=default
callerid=Lucia Cabanelas<312>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=312

[313]
type=friend
context=default
callerid=Manuel Alvarez<313>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=313

[314]
type=friend
context=default
callerid=Cristina Lopez<314>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=314

[315]
type=friend
context=default
callerid=Hugo Macario<315>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=315

[316]
type=friend
context=default
callerid=Antonio Cerdan<316>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=316

[317]
type=friend
context=default
callerid=Orlando Docampo<317>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=317

```
[318]
type=friend
context=default
callerid=Felix Rodriguez<318>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=318
```

```
[319]
type=friend
context=default
callerid=Francisco Bretal<319>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=319
```

```
[320]
type=friend
context=default
callerid=Cristina Casas<320>
host=dynamic
secret=123456
dtmfmode=inband
mailbox=320
```

voicemail.conf

```
[general]
; Enviar archivos en las notificaciones de e-mail
attach=yes
; Usar el formato wav para los mensajes de voz
format=wav
; Limitar el tiempo máximo del mensaje de voz a 180 segundos
maxmessage=180
; Limitar el tiempo mínimo del mensaje a 3 segundos
minmessage=3
; Anunciar el número que llamó antes de repetir el mensaje
saycid=yes
; Limitar el número de intentos de registro a 3
maxlogins=3
; Define los contextos internos para especificar que vienen de una extensión interna
cidinternalcontexts=default,
```

```
[zonemessages]
madrid=Europe/Paris|'vm-received' Q 'digits/at' R
europa=Europe/Berlin|'vm-received' Q 'digits/at' kM
```

```
[default]
300 => 4321,Marina Palacios,mpalacios@aquadom.com,,delete=1
301 => 4321,Javier Garcia,jgarcia@aquadom.com,,delete=1
```

```
302 => 4321,Elena Sanchez,esanchez@aquadom.com,,delete=1
303 => 4321,Juan Crespo,jcrespo@aquadom.com,,delete=1
304 => 4321,Leonor Sanchez,lsanchez@aquadom.com,,delete=1
305 => 4321,Saul Garcia,sgarcia@aquadom.com,,delete=1
306 => 4321,John Daniels,jdaniels@aquadom.com,,delete=1
307 => 4321,Oscar Pintos,opintos@aquadom.com,,delete=1
308 => 4321,Sergio Vazquez,svazquez@aquadom.com,,delete=1
309 => 4321,Pedro Grande,pgrande@aquadom.com,,delete=1
310 => 4321,Mar Castro,mcastro@aquadom.com,,delete=1
311 => 4321,Elena Losada,elosada@aquadom.com,,delete=1
312 => 4321,Lucia Cabanelas,lcabanelas@aquadom.com,,delete=1
313 => 4321,Manuel Alvarez,malvarez@aquadom.com,,delete=1
314 => 4321,Cristina Lopez,clopez@aquadom.com,,delete=1
315 => 4321,Hugo Macario,hmacario@aquadom.com,,delete=1
316 => 4321,Antonio Cerdan,acerdan@aquadom.com,,delete=1
317 => 4321,Orlando Docampo,odocampo@aquadom.com,,delete=1
318 => 4321,Felix Rodrigue,frodriguez@aquadom.com,,delete=1
319 => 4321,Francisco Bretal,fbretal@aquadom.com,,delete=1
320 => 4321,Cristina Casas,ccasas@aquadom.com,,delete=1
```

iax.conf

```
[general]
port=4569
bandwidth=low
disallow=all
allow=gsm
jitterbuffer=yes
tos=lowdelay
```

```
[santiago]
type=peer
host=dynamic
trunk=yes
auth=md5,plaintext,rsa
secret=sa1234go
username=santiago
qualify=yes
context=default
```

```
[madrid]
type=peer
host=dynamic
trunk=yes
auth=md5,plaintext,rsa
secret=ma1234id
username=madrid
qualify=yes
context=default
```

```
[valladolid]
type=peer
host=dynamic
trunk=yes
```

```
auth=md5,plaintext,rsa
secret=va1234id
username=valladolid
qualify=yes
context=default
```

```
[barcelona]
type=friend
host=dynamic
trunk=yes
auth=md5,plaintext,rsa
secret=ba1234na
username=barcelona
qualify=yes
context=default
```

```
[colombia]
type=peer
host=dynamic
trunk=yes
auth=md5,plaintext,rsa
secret=co1234ia
username=colombia
qualify=yes
context=default
```

```
[chile]
type=peer
host=dynamic
trunk=yes
auth=md5,plaintext,rsa
secret=ch1234le
username=chile
qualify=yes
context=default
```

```
[oportor]
type=peer
host=dynamic
trunk=yes
auth=md5,plaintext,rsa
secret=op1234to
username=oportun
qualify=yes
context=default
```

IVR

```
[IVR]
exten => s,1,Wait(1) ;espera un segundo
exten => s,2,Set(CHANNEL(language)=es) ; pone como predefinidas las voces en español
exten => s,3,Set(TIMEOUT(digit)=7) ; 7 segundos es el tiempo que espera entre el primer dígito
; y los sucesivos
exten => s,4,Set(TIMEOUT(response)=10) ; 10 segundos es el tiempo que espera para que
; el llamante pulsa una tecla
```

```

exten => s,5,BackGround(custom/espeng) ;presenta el menu vocal y al mismo tiempo escucha si el llamante
; pulsa alguna tecla
exten => s,6,WaitExten() ; espera que el llamante presione alguna tecla
exten => 1,1,goto(IVR1,s,1) ; si presiona 1 va al contexto IVR1, extension s, prioridad 1
exten => 2,1,goto(IVR2,s,1) ; si presiona 2 va al contexto IVR2, extension s, prioridad 1
exten => i,1,Playback(invalid) ; si el numero digitado no es valido (ni 1 ni 2) comunica el error
exten => i,2,Playback(goodbye) ; se despide
exten => i,3,Hangup ; cuelga la llamada
exten => t,1,goto(IVR,s,2) ; si dentro de 10 segundo el llamante no presiona
; ninguna tecla vuelve a presentar el menu vocal
exten => h,1,Hangup ; si el llamante cuelga ejecuta la extension h

```

[IVR1]

```

exten => s,1,Set(TIMEOUT(digit)=7) ;
exten => s,2,Set(TIMEOUT(response)=10)
exten => s,3,Set(CHANNEL(language)=en) ; define como idioma predefinido el inglés y usas las
; voces en este idioma
exten => s,4,BackGround(custom/engmenu) ; presenta en menu en ingles
exten => s,5,WaitExten() ; Espera que el llamante pulse una tecla

exten => 1,1,Playback(pls-wait-connect-call) ; Si presiona 1 lo pone en comunicacion con
; la oficina de Santiago (extension 100)
exten => 1,2,Dial(IAX2/santiago:sa1234go@10.64.3.100/100)

exten => 2,1,Playback(pls-wait-connect-call) ; Si presiona 2 lo pone en comunicacion con
; la oficina de Valladolid (extension 200)
exten => 2,2,Dial(IAX2/valladolid:va1234id@10.64.2.100/200)

exten => 3,1,Playback(pls-wait-connect-call) ; Si presiona 3 lo pone en comunicacion con
; la oficina de Madrid (extension 400)
exten => 3,2,Dial(IAX2/madrid:ma1234id@10.64.1.100/400)

exten => 4,1,Playback(pls-wait-connect-call) ; Si presiona 4 lo pone en comunicacion con
; la oficina de Colombia (extension 500)
exten => 4,2,Dial(IAX2/colombia:co1234ia@10.64.7.100/500)

exten => 5,1,Playback(pls-wait-connect-call) ; Si presiona 5 lo pone en comunicacion con
; la oficina de Chile (extension 550)
exten => 5,2,Dial(IAX2/chile:ch1234le@10.64.10.100/550)

exten => 6,1,Playback(pls-wait-connect-call) ; Si presiona 6 lo pone en comunicacion con
; la oficina de Oporto (extension 700)
exten => 6,2,Dial(IAX2/op1234to@10.64.6.100/700)

exten => 7,1,Playback(pls-wait-connect-call) ; Si presiona 7 lo pone en comunicacion con
; la oficina de Barcelona (extension 300)
exten => 7,2,Dial(IAX2/ba1234na@10.64.10.100/300)

```

```

exten => i,1,Playback(invalid)
exten => i,2,Playback(goodbye)
exten => i,3,hangup
exten => t,1,goto(IVR1,s,1)
exten => h,1,Hangup

```

[IVR2]

```

exten => s,1,Set(TIMEOUT(digit)=7)
exten => s,2,Set(TIMEOUT(response)=10)

```

```

exten => s,3,Set(CHANNEL(language)=es) ; define como idioma predefinido el inglés y usas las voces en este idioma
exten => s,4,BackGround(custom/espmenu) ; presenta en menu en español
exten => s,5,WaitExten() ; espera a que el llamante pulse una tecla

exten => 1,1,Playback(pls-wait-connect-call) ; Si presiona 1 lo pone en comunicacion con
;la oficina de Santiago (extension 100)
exten => 1,2,Dial(IAX2/santiago:sa1234go@10.64.3.100/100)

exten => 2,1,Playback(pls-wait-connect-call) ; Si presiona 2 lo pone en comunicacion con
;la oficina de Valladolid (extension 200)
exten => 2,2,Dial(IAX2/valladolid:va1234id@10.64.2.100/200)

exten => 3,1,Playback(pls-wait-connect-call) ; Si presiona 3 lo pone en comunicacion con
;la oficina de Madrid (extension 400)
exten => 3,2,Dial(IAX2/madrid:ma1234id@10.64.1.100/400)

exten => 4,1,Playback(pls-wait-connect-call) ; Si presiona 4 lo pone en comunicacion con
;la oficina de Colombia (extension 500)
exten => 4,2,Dial(IAX2/colombia:co1234ia@10.64.7.100/500)

exten => 5,1,Playback(pls-wait-connect-call) ; Si presiona 5 lo pone en comunicacion con
;la oficina de Chile (extension 550)
exten => 5,2,Dial(IAX2/chile:ch1234le@10.64.10.100/550)

exten => 6,1,Playback(pls-wait-connect-call) ; Si presiona 6 lo pone en comunicacion con
;la oficina de Oporto (extension 700)
exten => 6,2,Dial(IAX2/oporto:op1234to@10.64.6.100/700)

exten => 7,1,Playback(pls-wait-connect-call) ; Si presiona 7 lo pone en comunicacion con
;la oficina de Barcelona (extension 300)
exten => 7,2,Dial(IAX2/barcelona:ba1234na@10.64.10.100/300)

exten => i,1,Playback(invalid)
exten => i,2,Playback(goodbye)
exten => i,3,hangup
exten => t,1,goto(IVR1,s,1)
exten => h,1,Hangup

```

Configuración de los terminales SIP (Alcatel Temporis IP) para conectarse a Asterisk.

La configuración básica se puede realizar través de la página Web con la que cuenta el terminal Temporis IP200 ya que este equipo tiene un servidor web incorporado

Tendremos que pulsar el botón OK en el teclado del teléfono para ingresar en la página de estado y averiguar la dirección IP del Temporis IP 200.

Posteriormente introducimos la dirección IP del teléfono (por ejemplo, <http://192.168.1.100>) en la barra de direcciones del navegador web.

El nombre de usuario y contraseña de Administrador son:

admin / admin

Primero tendremos que cambiar la dirección IP del teléfono para ponerlo en el mismo rango que el servidor Asterisk.

Para ello escogemos Network → Internet Port (WAN)

Static IP Address: Ponemos la dirección IP asignada a este terminal. (Eje. 10.64.3.50)

Una vez dentro tendremos que dirigirnos al apartado de Configuración de la Cuenta (account data). Encontraremos los siguientes parámetros que tendremos que configurar.

Campo	Descripción
Estado de registro	Muestra el estado de registro del teléfono (sólo lectura)
Cuenta activa	Se puede escoger entre on/off para habilitar o deshabilitar la cuenta
Etiqueta	El nombre que muestra en el visor LCD del dispositivo
Nombre a mostrar	El nombre que se le muestra al otro teléfono cuando se llama
Nombre de registro	Nombre de la cuenta para registro que se ha puesto en sip.conf
Nombre de usuario	Nombre de la cuenta de usuario que está en sip.conf
Contraseña	Contraseña que hemos puesto en el archivo sip.conf
Servidor SIP	Dirección del servidor Asterisk

ALCATEL
ALINKS

Account: Account 1

Basic >>

Register Status	Registered	
Account Active	<input checked="" type="radio"/> On <input type="radio"/> Off	
Label	Ext 511	?
Display Name	John	?
Register Name	511	?
User Name	511	?
Password	*****	?
SIP Server	10.24.20.91	Port: 5060 ?
Enable Outbound Proxy Server	Disabled	?
Outbound Proxy Server		Port: 5060 ?
Transport	UDP	?
Backup Outbound Proxy Server		Port: 5060 ?
NAT Traversal	Disabled	?
STUN Server		Port: 3478 ?
Voice Mail	*2	?
Proxy Require		?
Anonymous Call	Off	?
On Code		?
Off Code		?
Anonymous Call Rejection	Off	?
On Code		?
Off Code		?
Missed call tag	Enabled	?
Auto Answer	Disabled	?
Ring Type	common	?

Codecs >> ?

Advanced >>

Confirm

NOTE

Display Name
SIP service subscriber name used for Caller ID display.

Register Name
SIP service subscriber ID used for authentication.

User Name
User account provided by VoIP service provider or PBX admin.

NAT Traversal
Control STUN server settings.

Proxy Require
Relevant for Nortel server only. If you wish to login to a Nortel server, value should be: com.nortelnetworks.firewall

Codecs
Select codec types and priority.

Advanced
Administrator oriented parameters.

Figura 18. Página de configuración de teléfono Temporis IP200

Ejemplo para el terminal **100** de Santiago de Compostela:

Campo	Descripción
Estado de registro	Registrado
Cuenta activa	On
Etiqueta	Mar Vieites
Nombre a mostrar	Mar Vieites
Nombre de registro	100
Nombre de usuario	100
Contraseña	123456
Servidor SIP	10.64.3.100

Bibliografía.

DAVID GOMILLION, BARRIE DEMPSTE. "Building Telephony Systems with Asterisk". Packt Publishing Ltd. Birmingham. 2ª Edición. Reino Unido 2006. Pág 120 -137.

JIM VAN MEGGELEN, LEIF MADSEN & JARED SMITH. Asterisk™ The Future of Telephony. O`Reilly. Estados Unidos. 2007. Pág 69-144 y 185-206.

JESÚS GIRÓN ISAAC. "Manual de configuración de Asterisk con SIP Trunking de VozTelecom (OIGAA DIRECT)". VozTelecom. 08/12/2008.

ELASTIX. QoS-Calidad de Servicio para VoIP. Disponible en: <http://elastixtech.com/qos-calidad-de-servicio-para-voip/>

ADAFRUIT. PiTFT Easy Install. Fecha de revisión: 2014-12-14. Disponible en : <https://learn.adafruit.com/adafruit-2-8-pitft-capacitive-touch/easy-install>

VOZTOVOICE. Instalar Asterisk 11.2 certificado en Raspberry pi Model B con Raspbian. Fecha de revisión: 25/11/2013. Disponible en: <https://www.voztovoice.org/?q=node/655>