Plataforma

de telecomunicaciones

de VoIP basada en software libre

Profesor MSc. Ing. Carlos Alberto Rodríguez López, Ing. Yuniel González López, Ing. Andy Hernández González, Universidad Central "Marta Abreu" de Las Villas, y MSc. Ing. Pérsida Elizabeth Gorrín Leyva, Especialista Superior en Automática, CEDAI, Villa Clara

crodrigz@uclv.edu.cu, persida@cedaivc.co.cu, yuniel@uclv.edu.cu, andy@uclv.edu.cu

I Introducción

La transmisión de tráfico de voz sobre redes de paquetes ha experimentado grandes progresos, tanto por el desarrollo de estándares como por la aparición de productos basados en la tecnología IP. Son también notables los avances en tecnologías inalámbricas y comunicaciones móviles celulares. El escenario actual de las telecomunicaciones es muy variado en tecnologías y servicios. En estos tiempos es común oír hablar de convergencia en las telecomunicaciones. No es fácil dar una única definición de convergencia, evidentemente según se aplique a uno u otro contexto se puede destacar una gran variedad de **convergencias**. Un paso fundamental para la convergencia total de las telecomunicaciones lo constituye la convergencia Fijo-Móvil que plantea la integración de redes fijas y móviles. Se trata de un concepto difícil de definir, un tema con algunas incertidumbres que está recibiendo la atención de todos los factores.

En una plataforma convergente Fijo-Móvil, los usuarios pueden pasar de una red a otra manteniendo la conexión sin interrupciones. Para lograr dicha plataforma se utilizan las tecnologías existentes que continúan ampliándose, de forma que los nuevos aparatos sean compatibles con diversas tecnologías como 802.11x, SIP, CDMA, GSM, UMTS y WLAN. Asimismo se necesita de servidores de comunicaciones que aseguren la compatibilidad entre todos los elementos de la infraestructura. En una solución de telecomunicaciones para la convergencia Fijo-Móvil deben estar presentes los terminales móviles personales, la red de convergencia Fijo-Móvil y los servicios convergentes adaptados a los diferentes dispositivos.

El marco descrito constituye un reto para las universidades en cuanto a la formación de profesionales; es también una oportunidad en términos de investigación. Las universidades junto a los profesionales vinculados a empresas del sector de las telecomunicaciones tienen el deber de contribuir con el desarrollo de esta área, en sintonía con los avances que se producen a nivel internacional, sin perder de vista las particularidades de nuestro contexto nacional, ponderando la soberanía tecnológica.

Consciente de esta realidad, la Facultad de Ingeniería Eléctrica de la Universidad Central "Marta Abreu" de las Villas ha proyectado desarrollar una infraestructura de laboratorio que permita alcanzar niveles de docencia e investigación adecuados. La solución está basada en software libre de código abierto lo que reduce los costos y facilita las investigaciones científicas y el desarrollo de aplicaciones (Alfonso, 2006 y 2008; Rodríguez, 2006; Rodríguez 2007). La figura 1 representa el esquema final de la plataforma.

El proyecto de plataforma de laboratorio, que además puede funcionar como sistema en producción y brindar servicios de telecomunicaciones privadas, se ejecuta en dos etapas:

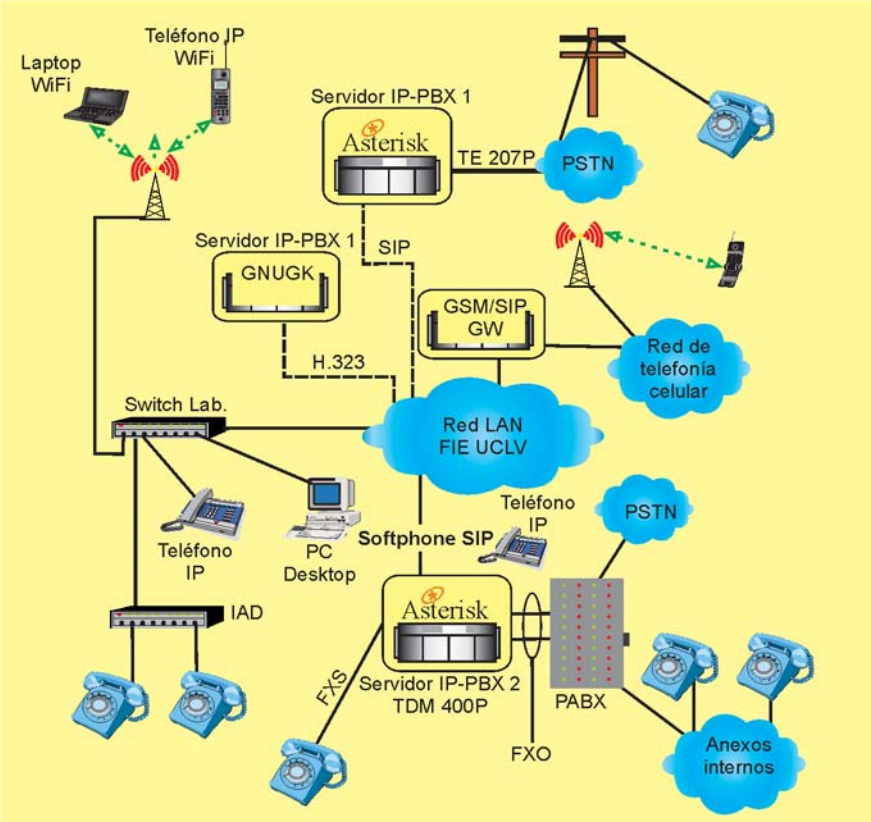

Figura 1 Plataforma de Telecomunicaciones Facultad de Ingeniería Eléctrica UCLV. (Fuente: elaboración propia).

la primera etapa permite la creación de una red SIP soportada en Asterisk —PBX de software libre— y otra H.323 controlada por GnuGK —*gatekeeper* de código abierto— ambas interconectadas entre sí y con el sistema telefónico. Estas redes están en funcionamiento y han constituido el soporte para trabajos docentes e investigativos en el área de VoIP. La segunda etapa pretende agregar tecnologías y servicios que permitan crear una red de pequeña escala donde se pueda representar el fenómeno de la convergencia en telecomunicaciones. Esta etapa incorpora tecnología inalámbrica WiFi y acceso a la red móvil.

2 Implementación de GnuGK para controlar la plataforma H.323

El *gatekeeper* seleccionado para implementar la plataforma H.323 es el GnuGk. Este proyecto comenzó en 1999 y fue lanzado bajo Licencia Pública General —*General Public License* (GPL)—. Es un software de código abierto que está basado en la pila de protocolos open H.323 y que posee las características regulares de un *gatekeeper* H.323 como son: la traducción de direcciones, control de admisión, autorización de llamada, control y gestión del ancho de banda, gestión de zona, entre otras. Igualmente presenta un amplio rango de métodos para la autenticación de usuarios. El GnuGk puede correr sobre ambientes Linux/UNIX, Windows y otros sistemas operativos, aunque algunas características no están disponibles aún para Windows. Es capaz de interactuar con otras herramientas que le permiten prestar varios servicios.

Por ejemplo, puede hacer un balance de la carga redistribuyéndola hacia *gatekeepers* alternativos, además puede implementar jerarquías de *gatekeepers* mediante la creación de *gatekeepers* padres e hijos e, incluso, alcanzar terminales que se encuentran detrás de Cajas de Traducción de Direcciones de Red —*Network Address Traslator* (NAT)— o simplemente cajas NAT (Stoeckigt 2004).

En el presente trabajo específicamente se ha hecho uso de su posibilidad de cambiar el modo del enrutado de la señalización y los canales de media, para mostrar los diferentes modelos de señalización. También se han configurado *gatekeepers* vecinos de forma tal que pueda apreciarse cómo se lleva a cabo la comunicación entre zonas mediante el intercambio de mensajes RAS de Petición de Localización —*Location ReQuest* (LRQ)— y Confirmación de Localización —*Location ConFirm* (LCF)—, aquí juega un papel fundamental la característica de la reescritura de números E.164. Esta situación combinada con los modos de enrutado de la señalización deriva en un conjunto de situaciones muy variadas, las cuales pueden contribuir de manera decisiva a la mejor comprensión del protocolo.

Se ha utilizado la posibilidad de monitorear y administrar remotamente el *gatekeeper* por diferentes vías: a través del puerto de estado y mediante la utilización de otras herramientas como el GnuGk Control Center. Se configuró un *gatekeeper* como padre y otro como hijo de manera que se pudo comprobar que las jerarquías funcionan perfectamente.

2.1 Diseño de la plataforma

El GnuGk es un *gatekeeper* implementado a base de software. Este puede correr sobre una PC de propósito general. Es importante señalar que no es muy exigente en cuanto a los requerimientos de hardware necesarios para su correcto funcionamiento,

aunque estos dependen de la versión que se vaya a utilizar. Fundamentalmente requiere que la PC donde va a correr tenga una tarjeta de red y, por supuesto, conexión a la red. En este trabajo se emplea la versión 2.2.3 del GnuGk.

2.2 Configuración del *gatekeeper*

El funcionamiento del *gatekeeper* está determinado por el fichero de configuración y por las opciones de línea de comando. Toda la configuración inicial se escribió en el fichero gatekeeper.ini. En este fichero se incluyen las secciones que definen la forma en que se realiza el monitoreo y la gestión remota y los modos de enrutamiento de la señalización.

2.3 Monitoreo y gestión remotos

Inicialmente es necesario explotar las posibilidades de monitoreo y gestión remotos mediante el uso de sesiones telnet o aplicaciones del hiperterminal, algo que la configuración por defecto no permite. A la vez es importante garantizar la seguridad del sistema, para que se pueda acceder al puerto de estado desde cualquier estación de trabajo; pero, estableciendo los mecanismos apropiados para asegurar que sólo el personal autorizado realice cambios en la configuración del *gatekeeper*. Todo lo anterior se configura en la sección GkStatus::Auth, como se muestra a continuación:

```
[GkStatus::Auth]
rule=explicit | password;
—otra combinación posible es
rule=explicit & password—
127.0.0.1=allow
a.b.c.d=allow
default=forbid
password=*****
gkadmin=username
```

Lo anterior expuesto significa que se podrá acceder al puerto de estado libremente desde las direcciones IP 127.0.0.1 —dirección IP de *loopback*— y a.b.c.d o desde cualquier IP mediante el uso del nombre de usuario —*username*— y la contraseña —*password*— correspondiente.

2.4 Definición del modo de enrutamiento para la señalización de la llamada

En el GnuGk los mensajes de señalización de llamada pueden ser manejados de dos maneras:

1. El primer método es **Señalización de Llamada en Modo Directo**, en el cual los mensajes de señalización de llamada son intercambiados directamente entre los *endpoints* o terminales (Figura 2).

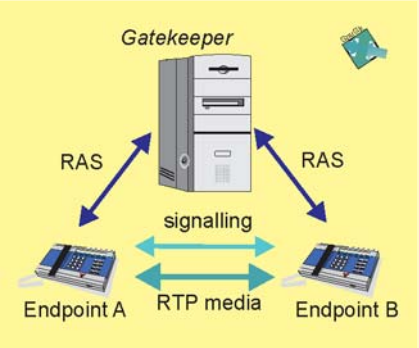

Figura 2 Modo de señalización directa. (Fuente: Willamowius 2007).

2. El segundo método es **Señalización de Llamada mediante *gatekeeper*** (Figura 3) en el que los mensajes de señalización de llamada son enrutados a través del mismo. En este caso, se puede seleccionar si se va a enrutar o no el canal de control H.245 —H.245 *control channel*— y los canales lógicos —*logical channels*— a través del *gatekeeper*, de lo cual, a su vez, se desprenden tres casos posibles (Willamowius 2007).

Caso I

El *gatekeeper* no enruta estos canales. El canal de control H.245 y los canales lógicos se establecen directamente entre los terminales.

Caso II

El canal de control H.245 se enruta a través del *gatekeeper*, mientras que los canales lógicos se establecen directamente entre los terminales.

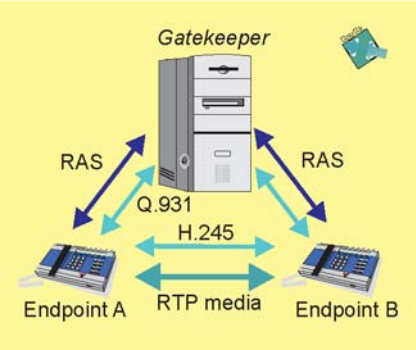

Figura 3 Enrutamiento del canal Q.931. (Fuente: Willamowius 2007).

Caso III

El *gatekeeper* enruta el canal de control H.245, así como también los canales lógicos, incluido el RTP/RTCP para audio.

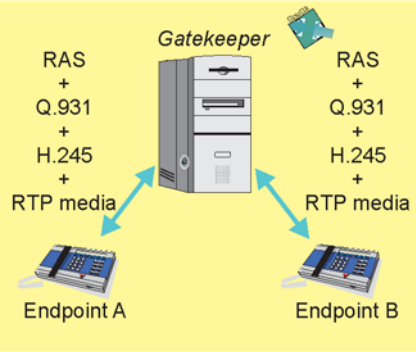

Figura 4 Modo proxy. (Fuente: Willamowius 2007).

Todos los casos anteriores se pueden configurar mediante las secciones RoutedMode y Proxy. A continuación se muestra un ejemplo en el que se ha configurado el *gatekeeper* de manera que sólo enrute la señalización Q.931 (Figura 4).

```
[RoutedMode]
GKRouted=1
H245Routed=0
[Proxy]
Enable=0
```

Los aspectos explicados previamente juegan un papel importante en la resolución y comprensión de las activida-

des docentes, puesto que permiten ilustrar los diferentes escenarios en que puede manejarse la señalización en los ambientes H.323.

2.5 Configuración de un *gatekeeper* vecino

La configuración de vecinos no forma parte de la configuración básica del *gatekeeper* sino que constituye un paso más avanzado dentro de la misma, con el objetivo de ilustrar la gestión de zonas y, además, la interoperabilidad entre ellas. Esto muestra algunas de las funcionalidades del *gatekeeper* en lo que a resolución de direcciones respecta, todo ello mediante el uso de mensajes RAS de petición de localización LRQ cuando el terminal llamado no pertenece a la misma zona del usuario que llama (Figura 5). La forma de implementar esta configuración es mediante las secciones RasSrv::Neighbors , Neighbor::gatekeepername. y RasSrv::RewriteE164.

```
[RasSrv::Neighbors]
Gatekeeper1=10.12.24.244;44
[Neighbor::Gatekeeper1]
GatekeeperIdentifier=Gatekeeper1
Host=10.12.24.244
```

SendPrefixes=44; —estos son los prefijos que el *gatekeeper* (gk) vecino puede recibir desde este gk. Aquellos que son aceptados se configuran en el gk vecino propiamente—.

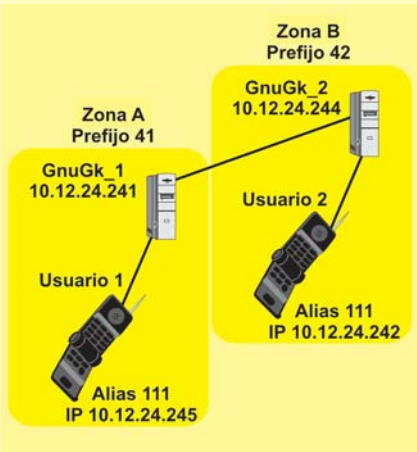

Figura 5 Configuración de *gatekeepers* como vecinos. (Fuente: elaboración propia).

AccepPrefixes=41; —estos son los prefijos que este gk aceptará provenientes de otros gk en la red—

```
[RasSrv::RewriteE164]
41...=...
```

2.6 Selección de Terminales

El usuario escogido para la implementación es SjPhone (Versión 1.40.258), que es un producto de SjLabs. Existen varias razones por las cuales se llegó a su elección; entre ellas pueden citarse, en primer lugar, el hecho de que es un software de libre distribución que se puede descargar de Internet; a parte de que es un usuario muy usado por la comunidad mundial y pueden encontrarse varios reportes satisfactorios de su utilización con el GnuGk.

SjPhone es un producto muy versátil pues puede emplearse tanto en ambientes H.323 o SIP utilizando un *gatekeeper* o un servidor Proxy SIP, respectivamente. También puede efectuar llamadas directas PC-PC empleando tanto H.323 como SIP y enrutar llamadas a través de una pasarela H.323.

Este usuario es capaz de manejar el canal de control de llamada H.245 de tres formas posibles conocidas como Fast Start, H.245 Tunneling y Early H.245, de manera que puede interactuar con las implementaciones de las versiones más antiguas y recientes del protocolo. Esto, en particular, es muy importante porque redunda en un menor tiempo de establecimiento de las llamadas con H.323, problema por el cual el protocolo fue muy criticado en sus inicios.

3 Asterisk como controlador de la plataforma SIP

La plataforma SIP soportada en Asterisk tiene más de tres años de explotación y ha sido objeto de discusión en otros trabajos. El servidor Asterisk es el elemento principal del proyecto y sus características de software libre de código abierto lo hacen especialmente apropiado desde el punto de vista económico y en cuanto a la flexibilidad para ajustarse a proyectos docente-investigativos.

La red SIP tiene como componentes principales:

- Asterisk —PBX de software libre—
- Teléfonos SIP —BUDGE TONE 100—
- Softphone —X LITE—
- Gateway VoIP —tarjeta wildcard TDM 400—

Las características generales de la implementación se muestran en la tabla 1.

Componente	Detalle
Nombre del producto	Asterisk@Home
Número de versión del software	1.3
Procesador	Pentium 4 CPU 1.80 GHz.
Sistema Operativo	Linux
Disco Duro	80 GB
RAM	128 MB
Soporte Telefónico	Digium Wildcard TDM 400 P
Teléfonos SIP	XLITE Soft Phone,Budgetone 100

Tabla 1 Características generales de la implementación. (Fuente: elaboración propia).

Como puede apreciarse en la figura 1, Asterisk está interconectada con la red telefónica privada, la interconexión con la pizarra LG GHX-16 se realiza con el empleo de puertos FXO en la tarjeta wildcard TDM 400, que se enlazan a extensiones de la pizarra. Asterisk aporta el servicio de recepcionista digital

para todo el sistema. Las llamadas de entrada se envían a Asterisk quien las enrutan hacia cualquiera de los destinos. A continuación se muestra un fragmento de la programación de dicho servicio:

```
[custom-op1]
exten => t,1,Flash
exten => t,2,Wait(1)
exten => t,3,SendDTMF(100)
exten => t,4,Wait(1)
exten => t,5,Hangup

[custom-op2]
exten => _1XX,1,Flash
exten => _1XX,2,Wait(1)
exten => _1XX,3,SendDTMF(${EXTEN})
exten => _1XX,4,Wait(1)
exten => _1XX,5,Hangup

[aa_1]
.
exten => s,7,Background(custom/aa_1)        ;
exten => t,1,Goto(custom-op1,t,1)       ;
exten => _107,1,Goto(custom-op1,t,1) ;
exten=>_10[012345689],1, Goto (custom-op2 ,BYEXTENSION, 1)     ;
exten=>_11[015],1,Goto(custom-op2, BYEXTENSION,1
```

3.1 Interconexión de las plataformas H.323 y SIP

Con el objetivo de ilustrar la interoperabilidad entre los protocolos H.323 y SIP, se interconectaron las plataformas correspondientes.

Luego de implementar la pasarela H.323 de Asterisk, al cual se denominó con el alias de "gw1", se pasa a configurar el *gatekeeper* para que pueda direccionar hacia él las llamadas que tienen como destino usuarios de la plataforma SIP.

Para lograr lo planteado se requiere asignar un plan de prefijos para hacer llamadas desde una plataforma hasta la otra.

Se realiza la configuración en la sección RasSrv::GWPrefixes, la que le indica al GnuGk cuáles de las llamadas deben ser enrutadas a la pasarela H.323 de Asterisk.

```
[RasSrv::GWPrefixes]
gw1=932
```

Esta entrada le dice al *gatekeeper* que enrute todas las llamadas con números E.164 que comiencen con el prefijo 932 a la pasarela de Asterisk o gw1. Finalmente el escenario queda como se muestra a continuación:

Figura 6 Interconexión de las plataformas H.323 y SIP. (Fuente: elaboración propia)

3.2 Papel del analizador de red en el trabajo

Un analizador de red o *sniffer* es una implementación de software y hardware que puede interceptar y visualizar el tráfico de datos digitales que está siendo cursado sobre una red o sobre una parte de esta. Como el flujo de datos viaja de un lado a otro sobre la red, el analizador puede capturar estos paquetes para decodificarlos y analizar su contenido de acuerdo con el RFC apropiado u otras especificaciones.

Se utilizan 2 analizadores —Agilent Advisor y Wireshark— como herramientas de trabajo fundamentales para el análisis de distintas situaciones que se presentan a lo largo del trabajo.

Entre las opciones que brindan estos analizadores se encuentra la de incluir varios filtros de captura. Algunos de los filtros predeterminados más importantes que pueden configurarse son los correspondientes a los protocolos TCP, IP, RAS, Q.931, RTP, RTCP, SIP, UDP, entre muchos otros.

Como ocurre casi siempre durante la implementación práctica de un sistema determinado, se cometen ciertos errores, los cuales muchas veces son difíciles de detectar si no se cuenta con una herramienta adecuada para ello. En el presente trabajo, el Agilent Advisor ha jugado un papel decisivo en este sentido. Por lo demás, ha servido para escudriñar el funcionamiento de las distintas implementaciones de los protocolos que la sustentan.

El Agilent Advisor es muy importante en la puesta a punto del sistema pues es capaz de ilustrar de forma más explícita los problemas que atenten contra el adecuado funcionamiento del mismo. También es de gran relevancia en la elaboración de actividades docentes y en trabajos de investigación. Algunos ejemplos de capturas realizadas con este software se aprecian en las figuras siguientes:

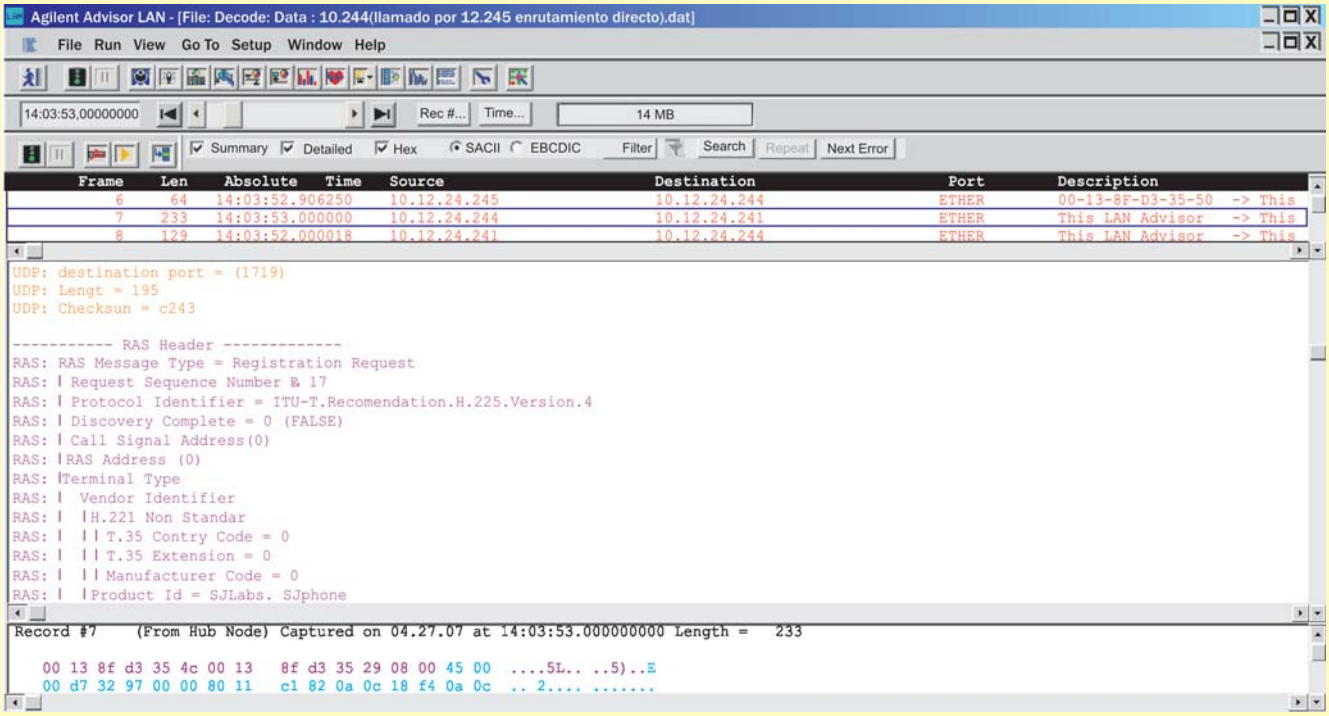

Figura 7 Capturas realizadas con el Agilent Advisor de un mensaje RAS. (Fuente: elaboración propia).

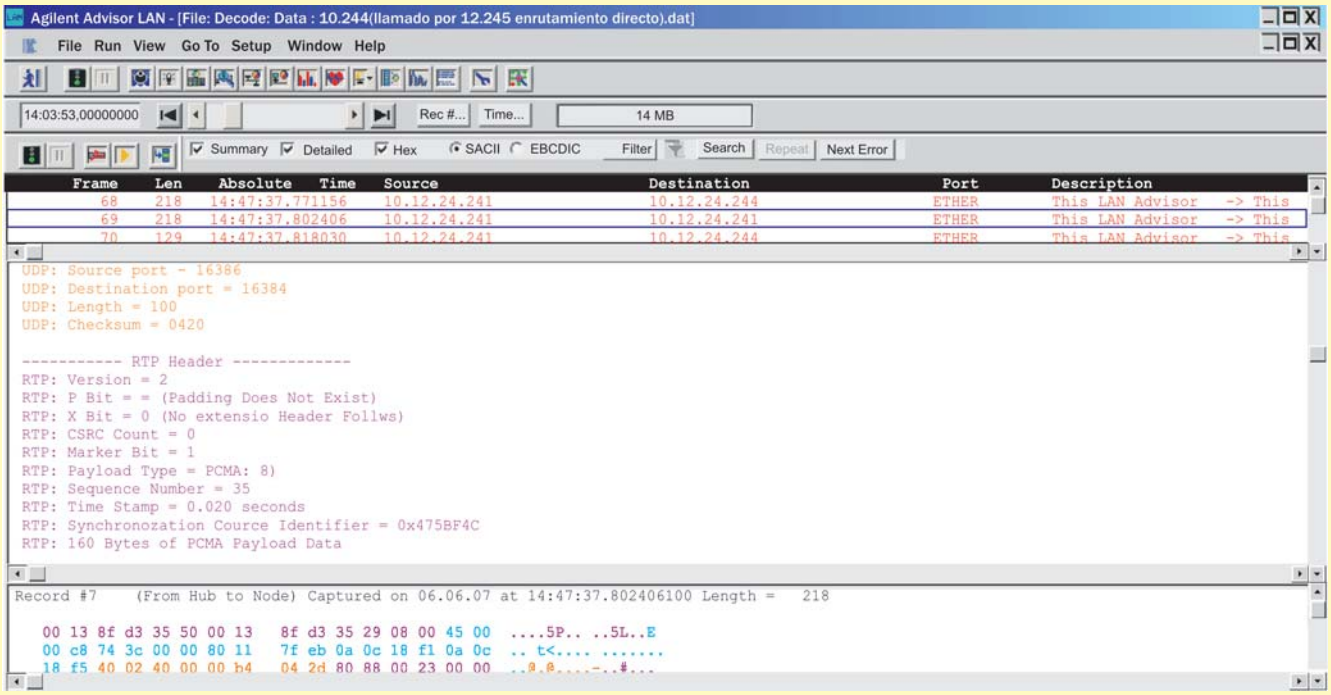

Figura 8 Capturas con Agilent Advisor de un paquete RTP. (Fuente: elaboración propia).

4 Conclusiones

Como parte de la ejecución del proyecto de telecomunicaciones que se desarrolla en la Facultad de Ingeniería Eléctrica de la Universidad Central "Marta Abreu" de Las Villas, se ha logrado poner a punto una plataforma para VoIP con fines docente-investigativos que ha probado, así, su robustez como sistema en producción.

La alternativa escogida se basa en software libre de código abierto lo que redunda en beneficios como costo, flexibilidad y soberanía tecnológica.

Tanto Asterisk como GnuGk han funcionado en correspondencia con las exigencias impuestas. El impacto en actividades de tipo docente-investigativas ha sido notable y se refleja en la calidad de la docencia y en la producción científica.

5 Referencias bibliográficas

Alfonso Reguera, V.; Álvarez Paliza, F.; Godoy Jr. W.; García Fernández, E. "On the Impact of Active Queue Management on VoIP Quality of Service". *Computer Communications*, vol. 31, no. 1 (2008): 73-87.

Alfonso Reguera V. "Evaluación de la calidad del servicio de VoIP en presencia de AQM". *Revista Ingeniería Electrónica. Automática y* Telecomunicaciones, vol *2-3, no. XXVII* (2006): 40-47.

Rec. UIT-T H.323 - Sistemas de Comunicación Multimedios Basadas en Paquetes, julio/2003. http://www.itu.int/rec/T-REC-H.323/en. (acceso abril 20, 2007).

Rodríguez, J. A. "Implementación de un sistema de facturación sobre la plataforma Asterisk PBX". Tesis de Pregrado, Universidad Central "Marta Abreu" de las Villas, Santa Clara, 2006.

Rodriguez López, C.; Alfonso Reguera V.; Gorrin Leyva P.; Montejo Sanchez S. "Evaluación del desempeño de una PBX soportada en software libre". Trabajo presentado en el Congreso y Feria Internacional Informática, La Habana, 2007.

Stoeckigt, K. "Building and Maintaining GnuGK". http://www.rzg.mpg.de/vc/docs/telozconf_kfs_092004.pdf. (acceso marzo 19, 2007).

Willamowius, J. "OpenH323 Gatekeeper - The GNU Gatekeeper". http://www.gnugk.org. (acceso abril 25, 2007).