PAQUETIZACION DE VOZ Y VIDEO SOBRE REDES IP

Dr. Ing. José Joskowicz
josej@fing.edu.uy
Instituto de Ingeniería Eléctrica, Facultad de Ingeniería
Universidad de la República
Montevideo, URUGUAY

Febrero 2017
Versión 2

Tabla de Contenido

1 Introducción

Las redes de voz [1] y las redes de datos [2] presentan tecnologías muy disímiles. Por un lado, la transmisión de voz, con una historia de más de 130 años, se basa en el establecimiento de vínculos permanentes entre dos puntos, diseñados para transmitir un tipo de señal específico: la voz humana, típica señal analógica, de ancho de banda acotado, que debe llegar a destino "inmediatamente" y ser lo más inteligible posible. Por otro lado, la transmisión de datos, con una historia relativamente reciente, se basa en la transmisión de información digital, utilizando técnicas de conmutación de paquetes, donde las pérdidas y los retardos no producen generalmente consecuencias importantes.

La integración de estas dos tecnologías no parece algo sencilla. Sin embargo, el tráfico multimedia sobre redes de paquetes ha ido en constante crecimiento en los últimos años. El tráfico de voz a nivel mundial ha tenido un incremento constante, aunque se nota un estancamiento en las tecnologías digitales históricas (TDM), tal como se puede ver en la siguiente gráfica [3]:

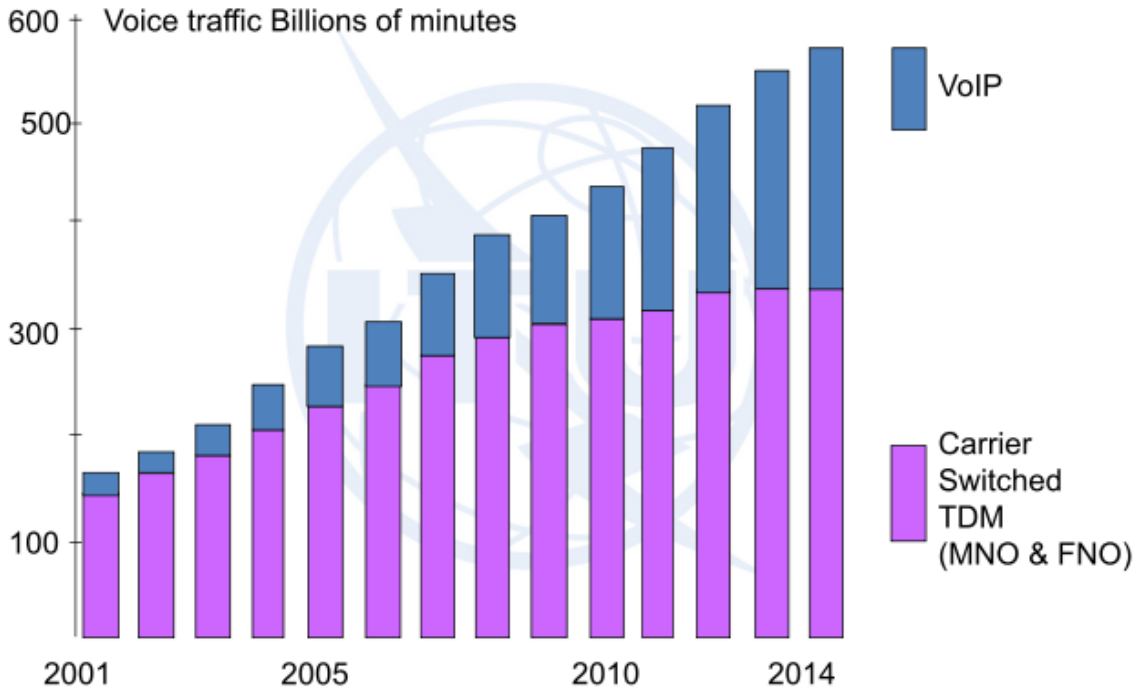

Figura 1.1

En estas notas se analizan los problemas tecnológicos que aparecen al cursar tráfico de voz y video sobre redes de datos, y se presentan los protocolos más utilizados de señalización de video-telefonía sobre IP.

2 Voz sobre redes de datos

2.1 Codificación de voz

La voz es codificada digitalmente para su transmisión. Los dispositivos de codificación y decodificación se denominan CoDec (Codificadores / Decodificadores). Los procesos de codificación y decodificación, así como los estándares más utilizadas, pueden ser consultados en [4]

Los codecs pueden ser clasificados según diferentes características, entre las que se encuentran su tasa de bits (bit rates), la calidad resultante del audio codificado, su complejidad, el tipo de tecnología utilizada y el retardo que introducen, entre otros. Originalmente, los primeros codecs fueron diseñados para reproducir la voz en la banda de mayor energía, entre 300 Hz a 3.4 kHz. Actualmente este tipo de codecs son caracterizados como de "banda angosta" (narrowband). En contraste, los codecs que reproducen señales entre 50 Hz y 7 kHz se han llamado de "banda ancha" (wideband). Más recientemente, ITU-T ha estandarizado codecs llamados de banda superancha (superwideband), para el rango de 50 Hz a 14 kHz y de banda completa (fullband), para el rango de 50 Hz a 20 kHz.

A modo de referencia, la Tabla 2.1 resume los codecs más conocidos.

Codecs de banda angosta (narrowband):

Codec	Nombre	Bit rate (kb/s)	Retardo (ms)	Comentarios
G.711	PCM: Pulse Code Modulation	64, 56	0.125	Codec "base", utiliza dos posibles leyes de compresión: μ-law y A-law [5]
G.723.1	Hybrid MPC-MLQ and ACELP	6.3, 5.3	37.5	Desarrollado originalmente para video conferencias en la PSTN, es actualmente utilizado en sistemas de VoIP [6]
G.728	LD-CELP: Low-Delay code excited linear prediction	40, 16, 12.8, 9.6	1.25	Creado para aplicaciones DCME (Digital Circuit Multiplex Encoding) [7]
G.729	CS-ACELP: Conjugate Structure Algebraic Codebook Excited Linear Prediction	11.8, 8, 6.4	15	Ampliamente utilizado en aplicaciones de VoIP, a 8 kb/s [8]
AMR	Adaptive Multi Rate	12..2 a 4.75	20	Utilizado en redes celulares GSM [9]

Codecs de banda ancha (wideband):

Codec	Nombre	Bit rate (kb/s)	Retardo (ms)	Comentarios
G.722	Sub-band ADPCM	48,56,64	3	Inicialmente diseñado para audio y videconferencias, actualmente utilizado para servicios de telefonía de banda ancha en VoIP [10]
G.722.1	Transform Coder	24,32	40	Usado en audio y videoconferencias [11]
G.722.2	AMR-WB	6.6, 8.85, 12.65, 14.25, 15.85, 18.25, 19.85, 23.05, 23.85	25.9375	Estandar en común con 3GPP (3GPP TS 26.171). Los bit rates más altos tienen gran inmunidad a los ruidos de fondo en ambientes adversos (por ejemplo celulares) [12]
G.711.1	Wideband G.711	64, 80, 96	11.875	Amplía el ancho de banda del codec G.711, optimizando su uso para VoIP [13]
G.729.1	Wideband G.729	8 a 32 kb/s	<49 ms	Amplía el ancho de banda del codec G.729, y es “compatible hacia atrás” con este codec. Optimizado su uso para VoIP con audio de alta calidad [14]
RtAudio	Real Time Audio	8.8, 18	40	Codec propietario de Microsoft, utilizado en aplicaciones de comunicaciones unificadas (OCS) [15]

Codecs de banda super ancha (superwideband):

Codec	Nombre	Bit rate (kb/s)	Retardo (ms)	Comentarios
SILK	SILK	8 a 24	25	Utilizado por Skype [16]

Codecs de banda completa (fullband):

Codec	Nombre	Bit rate (kb/s)	Retardo (ms)	Comentarios
G.719	Low-complexity, full-band	32 a 128	40	Es el primer codec “fullband” estandarizado por ITU [17]

Tabla 2.1

2.2 Transmisión de voz sobre redes de datos

Para poder transmitir las muestras codificadas de voz sobre redes de datos, es necesario armar “paquetes”. Un canal de voz consiste en un flujo de bits, dependientes del codec utilizado. Si, por ejemplo, la voz está codificada con el

codec G.711 en ley A, un canal de voz consiste en un flujo de 64 kb/s. (una "muestra" de voz, codificada con 8 bits, cada 125 µs). Para enviar este flujo sobre una red de datos, es necesario armar "paquetes". Si bien se podría formar un paquete con cada muestra de voz, esto generaría un sobrecarga ("overhead") demasiado importante (recordar que cada paquete requiere de cabezales). Por otro lado, si se espera a "juntar" demasiadas muestras de voz, para formar un paquete con mínima sobrecarga porcentual, se pueden introducir retardos no aceptables. Un paquete IP puede tener hasta 1500 bytes de información. Si con muestras del codec G.711 se quisiera completar los 1500 bytes del paquete IP, se introduciría un retardo de 125µs x 1500 = 187,5 ms. Esta demora no es aceptable en aplicaciones de conversaciones de voz.

Por esta razón, se toman generalmente "ventanas" de 10 a 30 ms (ver Figura 2.1). Las muestras codificadas de voz de cada una de estas ventanas se "juntan" y con ellas se arman paquetes. El tamaño de estas ventanas es configurable para algunos algoritmos de codificación, y está estandarizado para otros.

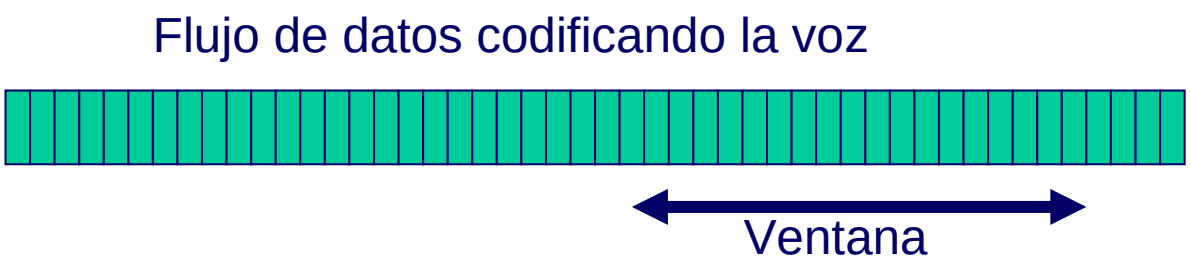

Figura 2.1

2.3 RTP – Real-Time Transport Protocol

El protocolo RTP, basado originalmente en el RFC 1889 y luego reemplazado por el RFC 3550 [18], establece los principios de un protocolo de transporte sobre redes que no garantizan calidad de servicio para datos "de tiempo real", como por ejemplo voz y video.

El protocolo establece la manera de generar paquetes que incluyen, además de los propios datos de "tiempo real" a transmitir, números de secuencia, marcas de tiempo, y monitoreo de entrega. Las aplicaciones típicamente utilizan RTP sobre protocolos de red "no confiables", como UDP. Los "bytes" obtenidos de cada conjunto de muestras de voz o video son encapsulados en paquetes RTP, y cada paquete RTP es a su vez encapsulado en segmentos UDP (Ver Figura 2.2).

RTP soporta transferencia de datos a destinos múltiples, usando facilidades de "multicast", si esto es provisto por la red.

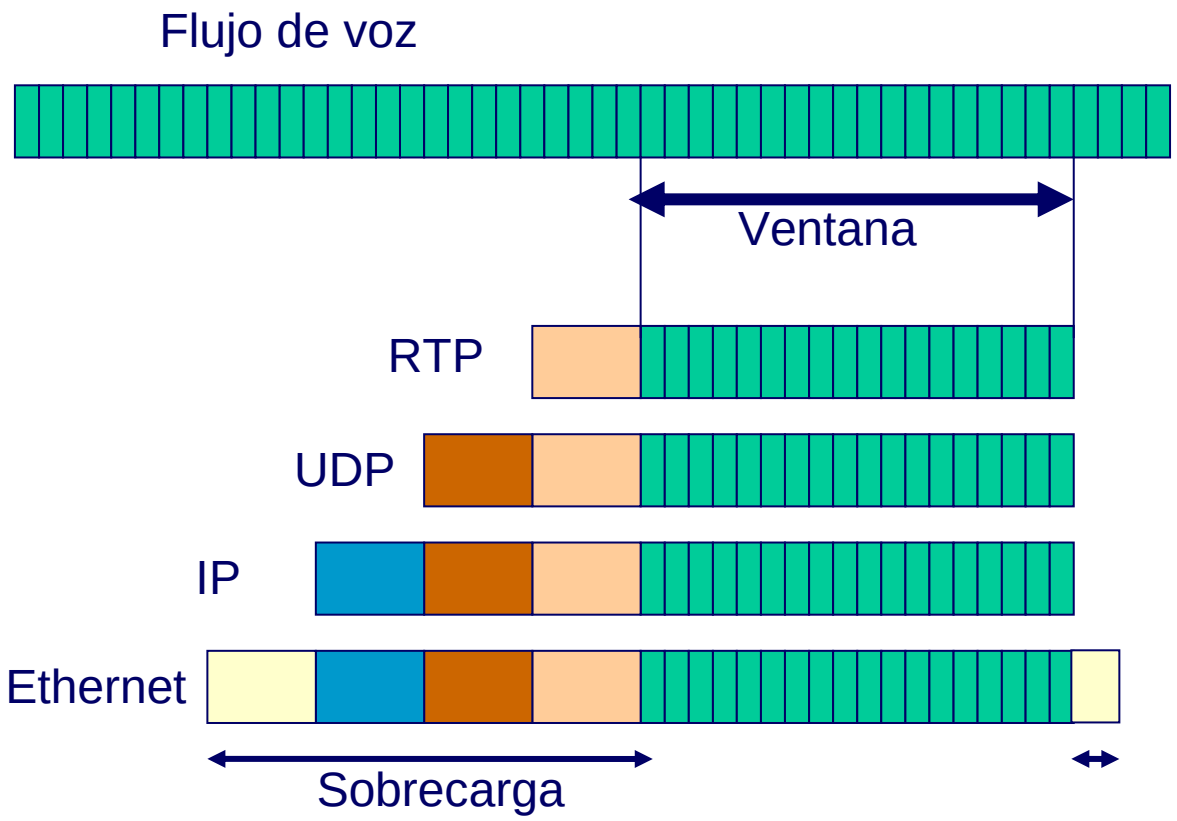

Figura 2.2

Cada paquete RTP consiste en un cabezal y los datos de voz. El cabezal contiene números de secuencia, marcas de tiempo, y monitoreo de entrega. El formato de éste cabezal es el mostrado en la Figura 2.3

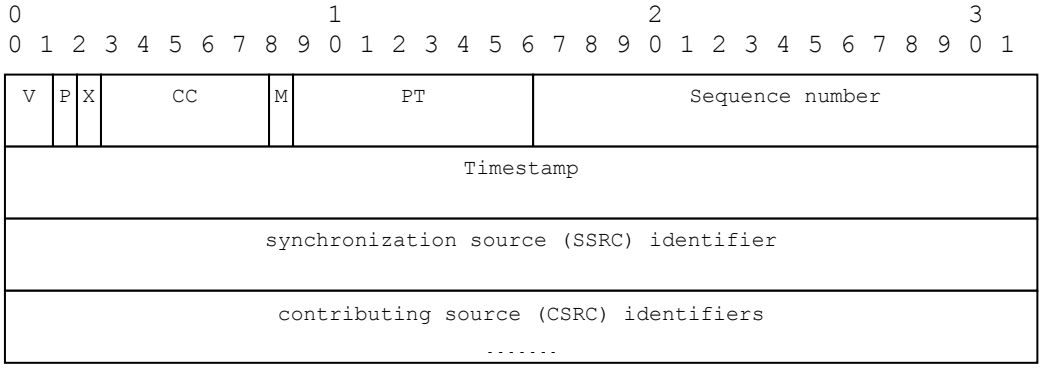

Figura 2.3

Los campos más relevantes son:

2.3.1 Versión (V)
La versión actual del protocolo es la 2.

2.3.2 CSRC count (CC - Contributing Sources Count)
El campo indica la cantidad de "fuentes" que contribuyen al audio incluido en el paquete. Pueden ser desde 0 a 15 (ver 2.3.7)

2.3.3 Tipo de información (PT - Payload Type)
El campo "payload" identifica el tipo de información que viaja en el paquete. Es un campo de 7 bits, lo que permite diferenciar hasta 128 tipos de información. Este campo indica el tipo de codificación de audio o video, o el contenido de información "especial". Los valores de este campo se definen en el RFC 3551 [19]. Algunos valores de ejemplo se muestran en la Tabla 2.2.

Payload Type	Formato	Medio	Clock Rate
0	PCM mu-law	Audio	8 kHz
3	GSM	Audio	8 kHz
4	G.723	Audio	8 kHz
8	PCM A-law	Audio	8 kHz
9	G.722	Audio	8 kHz
13	Confort Noise	Audio	
14	MPEG Audio	Audio	90 kHz
15	G.728	Audio	8 kHz
18	G.729	Audio	8 kHz
26	Motion JPEG	Video	90 kHz
31	H.261	Video	90 kHz
32	MPEG-1 o 2 Elementary Stream	Video	90 kHz
33	MPEG-1 o 2 Transport Stream	Video	90 kHz
34	H.263	Video	90 kHz
96 – 127	Dinámico		

Tabla 2.2

El valor 13 "Confort Noise" inidica que el paquete se trata de "ruido de confort", utilizado junto con técnicas de detección de actividad de voz (VAD, Voice Activity Detection). Los valores 96 a 127 son dinámicos, su utilización puede depender de las aplicaciones. Por ejemplo, una utilización de los valores dinámicos es para codificar los dígitos DTMF, de acuerdo al RFC 2833 [20].

2.3.4 Número de secuencia (Sequence Number)
El campo correspondiente al número de secuencia es de 16 bits. Con cada paquete enviado, el emisor incrementa en uno el número de secuencia. Esto permite al receptor detectar paquetes perdidos, o fuera de orden.

2.3.5 Marca de tiempo (Time Stamp)
Este campo es de 32 bits. Indica el momento al que corresponde la primera muestra de la ventana de información que viaja en el paquete. Este campo es utilizado por el receptor, para reproducir las muestras con la misma cadencia con

las que fueron obtenidas. Es a su vez útil para medir el “jitter”. En audio, el campo “Time Stamp” se mide en unidades de 125 μs (o sea, en unidades de muestreo). Si por ejemplo un paquete de 160 bytes de audio en Ley A contiene el campo TimeStamp con el valor 1, el siguiente paquete contendrá el campo TimeStamp en 160.

2.3.6 Identificador del origen (SSRC - Synchronization Source Identifier)

El campo correspondiente al SSRC es de 32 bits. Típicamente cada flujo en una sesión RTP tiene un identificador diferente. El origen establece este número, asegurando que no se repita.

2.3.7 Identificador del tributario (CSRC - Contributing Sources Identifier)

Pueden existir de 0 hasta 15 campos CSRC, de acuerdo al valor de CC. Esta lista identifica a cada uno de los interlocutores cuando el audio que se envía es producido en un mezclador o “mixer” (por ejemplo, cuando se envía el audio de varios participantes de una conferencia)

La Figura 2.4 muestra un ejemplo de un paquete RTP que contiene audio. En este caso, el audio fue codificado en G711 Ley μ, y existe una única fuente que contribuye al audio (CSRC = 0).

La Figura 2.5 muestra un ejemplo de un paquete RTP que codifica el dígito DTMF “2” según el RFC 2833.

La Figura 2.6 muestra un ejemplo de un paquete RTP que codifica la señal de “Confort Noise”.

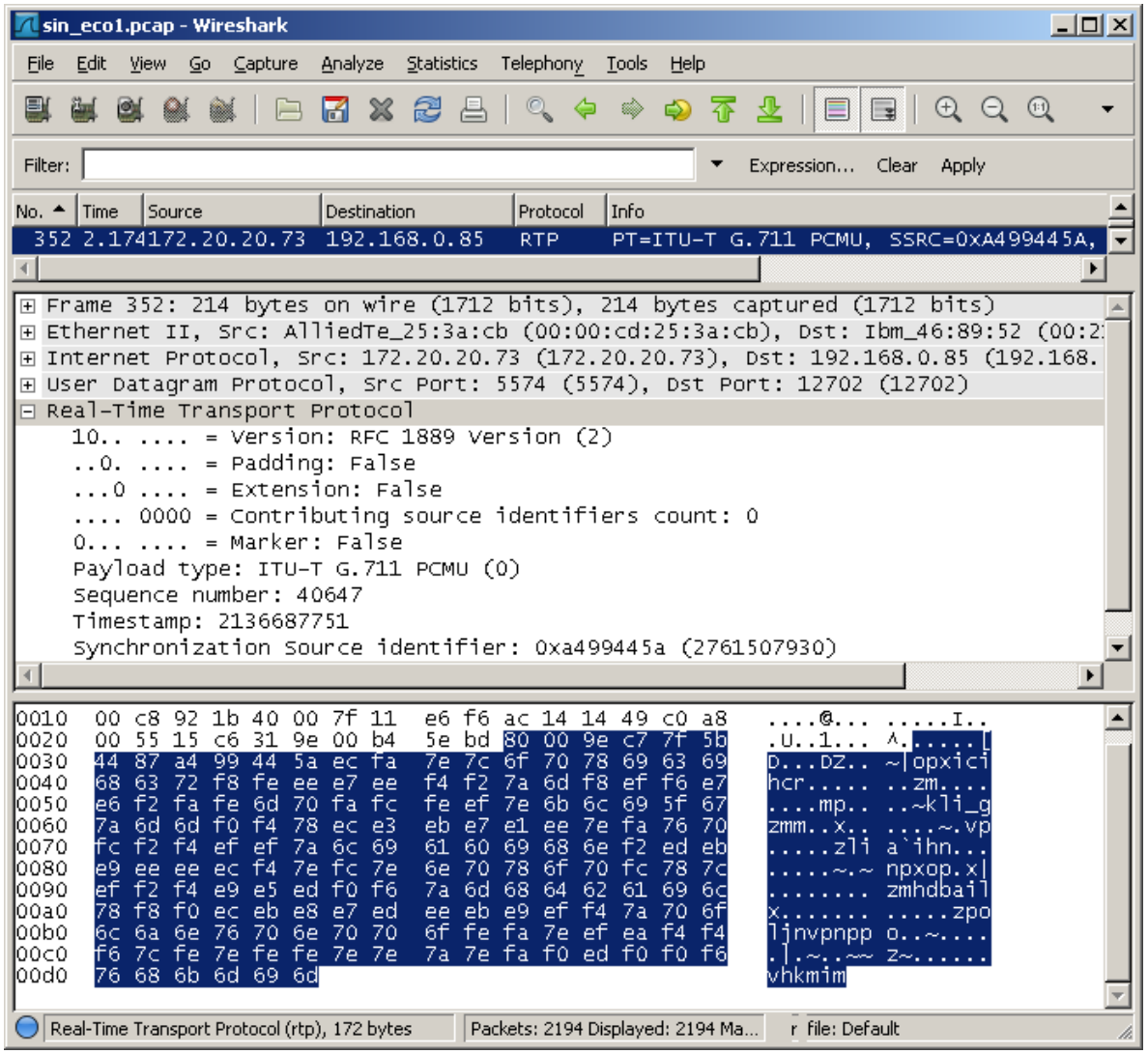
sin_eco1.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter: Expression... Clear Apply
No. Time Source Destination Protocol Info
352 2.174172.20.20.73 192.168.0.85 RTP PT=ITU-T G.711 PCMU, SSRC=0xA499445A,
Frame 352: 214 bytes on wire (1712 bits), 214 bytes captured (1712 bits)
Ethernet II, Src: AlliedTe_25:3a:cb (00:00:cd:25:3a:cb), Dst: Ibm_46:89:52 (00:2
Internet Protocol, Src: 172.20.20.73 (172.20.20.73), Dst: 192.168.0.85 (192.168.
User Datagram Protocol, Src Port: 5574 (5574), Dst Port: 12702 (12702)
Real-Time Transport Protocol
10.. .... = Version: RFC 1889 Version (2)
..0. .... = Padding: False
...0 .... = Extension: False
.... 0000 = Contributing source identifiers count: 0
0... .... = Marker: False
Payload type: ITU-T G.711 PCMU (0)
Sequence number: 40647
Timestamp: 2136687751
Synchronization Source identifier: 0xa499445a (2761507930)
0010 00 c8 92 1b 40 00 7f 11 e6 f6 ac 14 14 49 c0 a8 ....@... .....I..
0020 00 55 15 c6 31 9e 00 b4 5e bd 80 00 9e c7 7f 5b .U..1... ^.....[
0030 44 87 a4 99 44 5a ec fa 7e 7c 6f 70 78 69 63 69 D...DZ.. ~|opxici
0040 68 63 72 f8 fe ee e7 ee f4 f2 7a 6d f8 ef f6 e7 hcr..... ..zm....
0050 e6 f2 fa fe 6d 70 fa fc fe ef 7e 6b 6c 69 5f 67 ....mp.. ..~kli_g
0060 7a 6d 6d f0 f4 78 ec e3 eb e7 e1 ee 7e fa 76 70 zmm..x.. ....~.vp
0070 fc f2 f4 ef ef 7a 6c 69 61 60 69 68 6e f2 ed eb .....zli a`ihn...
0080 e9 ee ee ec f4 7e fc 7e 6e 70 78 6f 70 fc 78 7c .....~.~ npxop.x|
0090 ef f2 f4 e9 e5 ed f0 f6 7a 6d 68 64 62 61 69 6c .... zmhdbail
00a0 78 f8 f0 ec eb e8 e7 ed ee eb e9 ef f4 7a 70 6f x..... .....zpo
00b0 6c 6a 6e 76 70 6e 70 70 6f fe fa 7e ef ea f4 f4 ljnvpnpp o..~....
00c0 f6 7c fe 7e fe fe 7e 7e 7a 7e fa f0 ed f0 f0 f6 .|.~..~~ z~......
00d0 76 68 6b 6d 69 6d vhkmim
Real-Time Transport Protocol (rtp), 172 bytes
Packets: 2194 Displayed: 2194 Ma...
Profile: Default

Figura 2.4

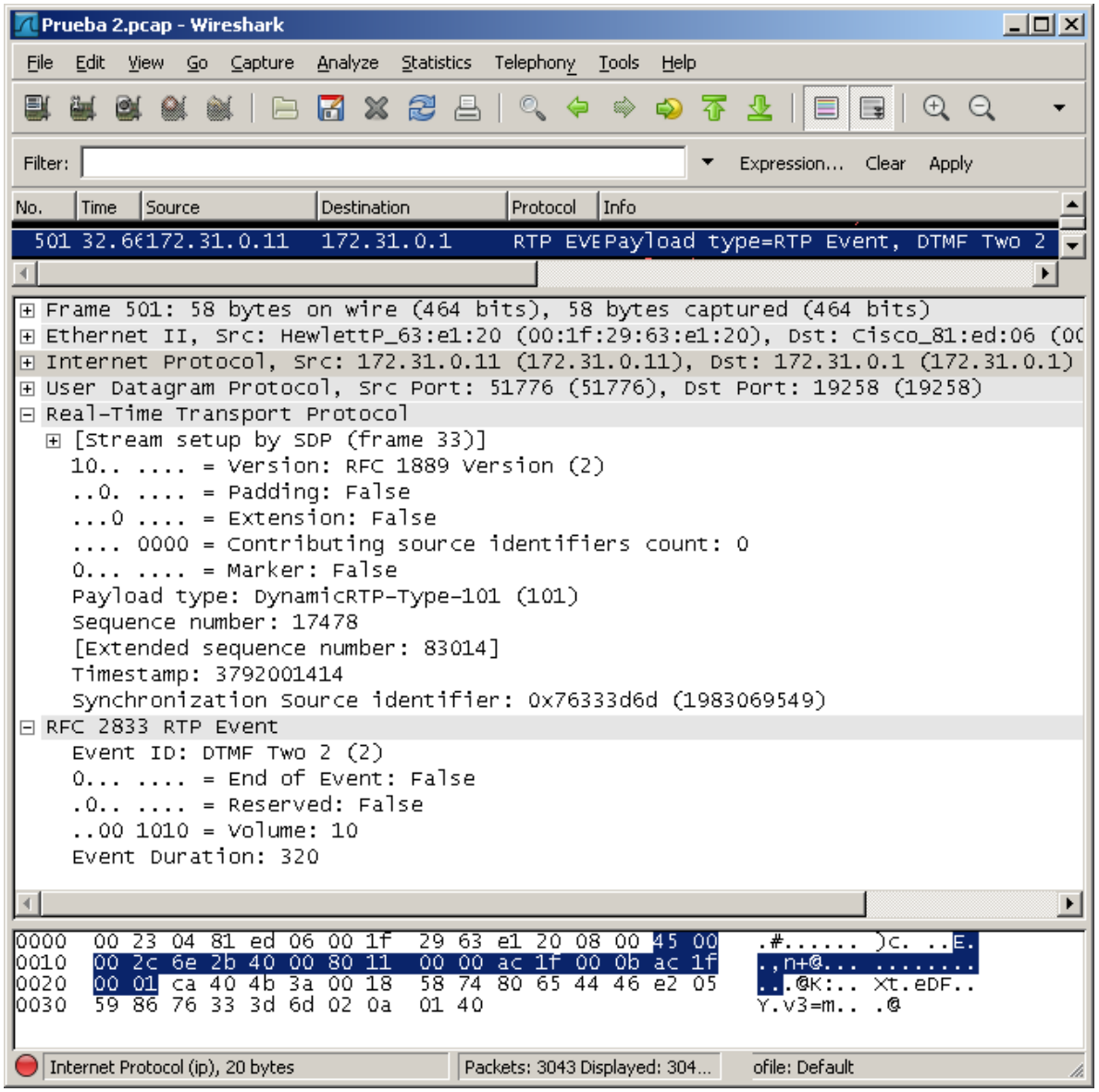
Prueba 2.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter: Expression... Clear Apply
No. Time Source Destination Protocol Info
501 32.6€172.31.0.11 172.31.0.1 RTP EVEPayload type=RTP Event, DTMF Two 2
Frame 501: 58 bytes on wire (464 bits), 58 bytes captured (464 bits)
Ethernet II, Src: HewlettP_63:e1:20 (00:1f:29:63:e1:20), Dst: Cisco_81:ed:06 (0C
Internet Protocol, Src: 172.31.0.11 (172.31.0.11), Dst: 172.31.0.1 (172.31.0.1)
User Datagram Protocol, Src Port: 51776 (51776), Dst Port: 19258 (19258)
Real-Time Transport Protocol
[Stream setup by SDP (frame 33)]
10.. .... = Version: RFC 1889 Version (2)
..0. .... = Padding: False
...0 .... = Extension: False
.... 0000 = Contributing source identifiers count: 0
0... .... = Marker: False
Payload type: DynamicRTP-Type-101 (101)
Sequence number: 17478
[Extended sequence number: 83014]
Timestamp: 3792001414
Synchronization Source identifier: 0x76333d6d (1983069549)
RFC 2833 RTP Event
Event ID: DTMF Two 2 (2)
0... .... = End of Event: False
.0.. .... = Reserved: False
..00 1010 = Volume: 10
Event Duration: 320
0000 00 23 04 81 ed 06 00 1f 29 63 e1 20 08 00 45 00 .#.....)c. ..E.
0010 00 2c 6e 2b 40 00 80 11 00 00 ac 1f 00 0b ac 1f .,n+@... ....
0020 00 01 ca 40 4b 3a 00 18 58 74 80 65 44 46 e2 05 ...@K:.. Xt.eDF..
0030 59 86 76 33 3d 6d 02 0a 01 40 Y.v3=m.. .@
Internet Protocol (ip), 20 bytes
Packets: 3043 Displayed: 304...
ofile: Default

Figura 2.5

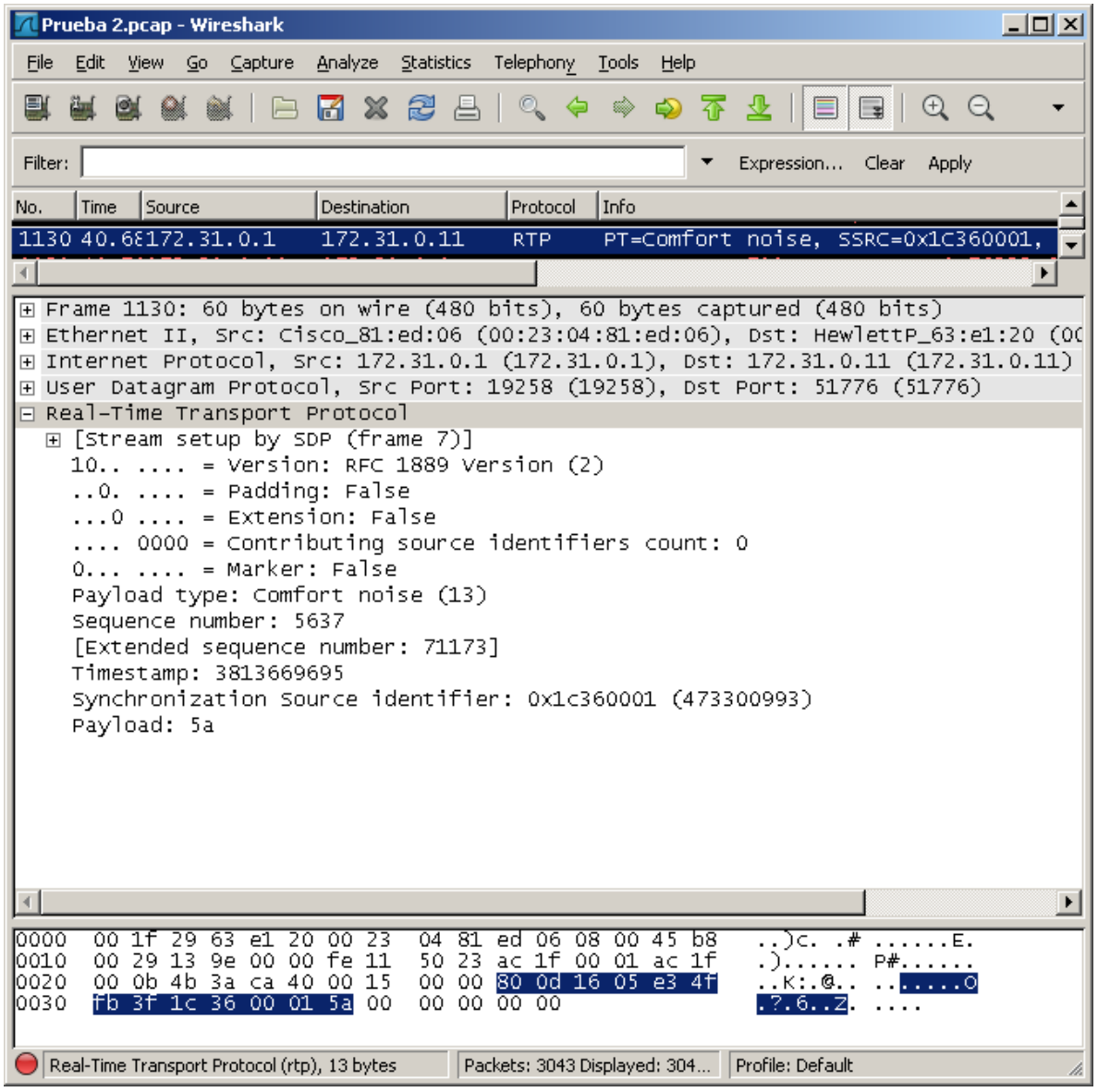

Figura 2.6

Una comunicación RTP entre dos extremos (por ejemplo, dos teléfonos IP) requiere del establecimiento de una “sesión”. Esta sesión se establece mediante mecanismos de señalización (por ejemplo, señalización SIP, que no se describe dentro de este capítulo). Como parte de la señalización se intercambia información de las capacidades de cada dispositivo (por ejemplo, si se soporta o no video, la lista de codecs de audio soportados, etc.), y las direcciones de red en las que cada dispositivo espera recibir el flujo de medio correspondiente, mediante el protocolo RTP. Esto se ejemplifica en la siguiente figura.

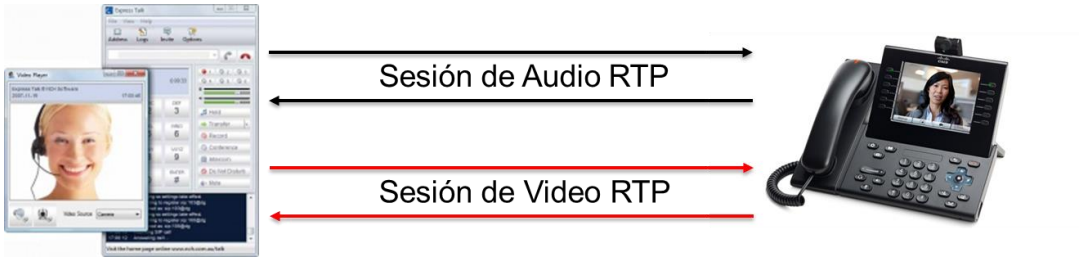

Figura 2.7

Cada sesión estable por lo tanto dos caminos independiente para el flujo de los medios, uno de ida y otro de vuelta, tal como se muestra en la siguiente figura. Allí se puede ver como la "aplicación" (por ejemplo, una aplicación de "softphone" corriendo en una computadora) utiliza un enconder y un decoder, los que se comunican con sus pares (decoder y encoder respectivamente) del extremo distante, a través de flujos RTP/UDP/IP. Como se mencionó, cada extremo debe conocer la dirección IP y el puerto UDP de su contraparte.

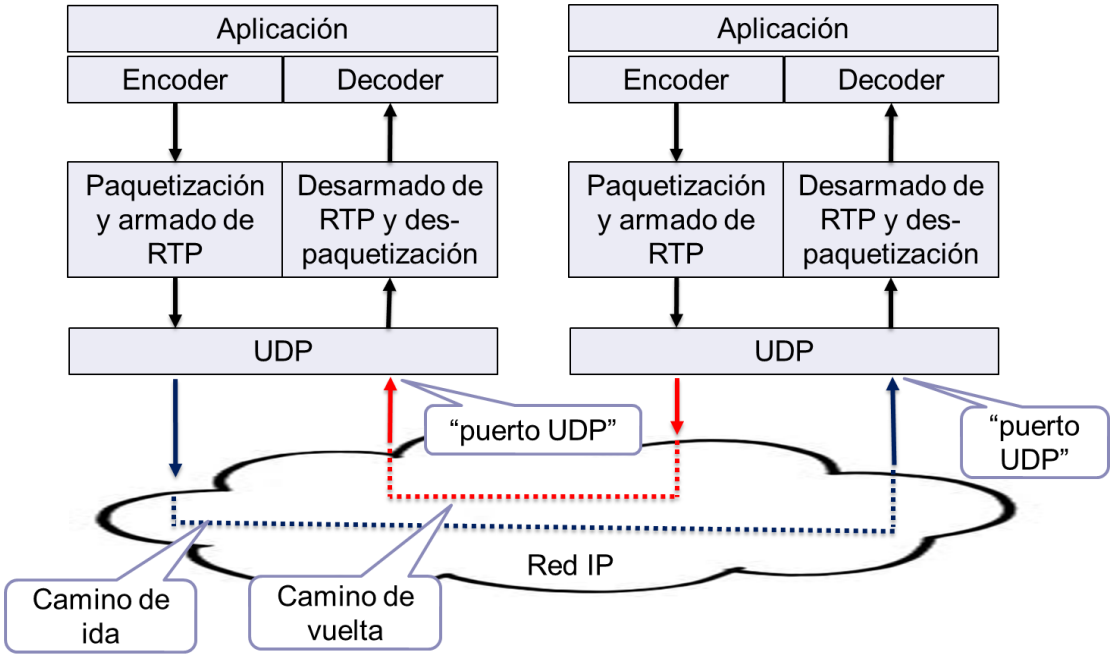

Figura 2.8

2.4 RTCP – RTP Control Protocol

El RFC 3550 establece, además del protocolo RTP, un protocolo de control, RTCP, encargado de enviar periódicamente paquetes de control entre los participantes de una sesión. El protocolo RTCP tiene las siguientes funciones principales:

- Proveer realimentación acerca de la calidad de los datos distribuidos (por ejemplo, de la calidad percibida de VoIP). Esta realimentación permite adaptar dinámicamente la codificación, o tomar acciones tendientes a solucionar problemas cuando se detecta degradación en la calidad de la comunicación
- Transporte del CNAME (Canonical Name) de cada originador. Este identificador permite asociar varios flujos RTP con el mismo origen (por ejemplo, flujos de audio y video provenientes del mismo emisor)
- Adaptar dinámicamente la frecuencia de envío de paquetes de control RTCP de acuerdo al número de participantes en la sesión. Dado que los paquetes se deben intercambiar “todos contra todos”, es posible saber cuántos participantes hay, y de esta manera calcular la frecuencia de envíos de esto paquetes.

Los paquetes RTCP pueden ser de los siguientes tipos:

- SR (Sender Report): Envía estadísticas de los participantes “origen” (sender)
- RR (Receiver Report): Envía estadísticas de los participantes “destino” (receivers)
- SDES (Source Description): Envía ítems de descripción del origen
- BYE: Indica el fin de la participación en el intercambio de mensajes RTCP
- APP: Funciones específicas para las aplicaciones participantes

En la Figura 2.9 se muestra un ejemplo de un paquete RTCP. En este ejemplo se envían en el mismo paquete un SR (Sender Report) y un SDES (Source Description).

Los flujos de RTP y RTCP recorren caminos independientes entre los extremos, como se muestra en la Figura 2.10.

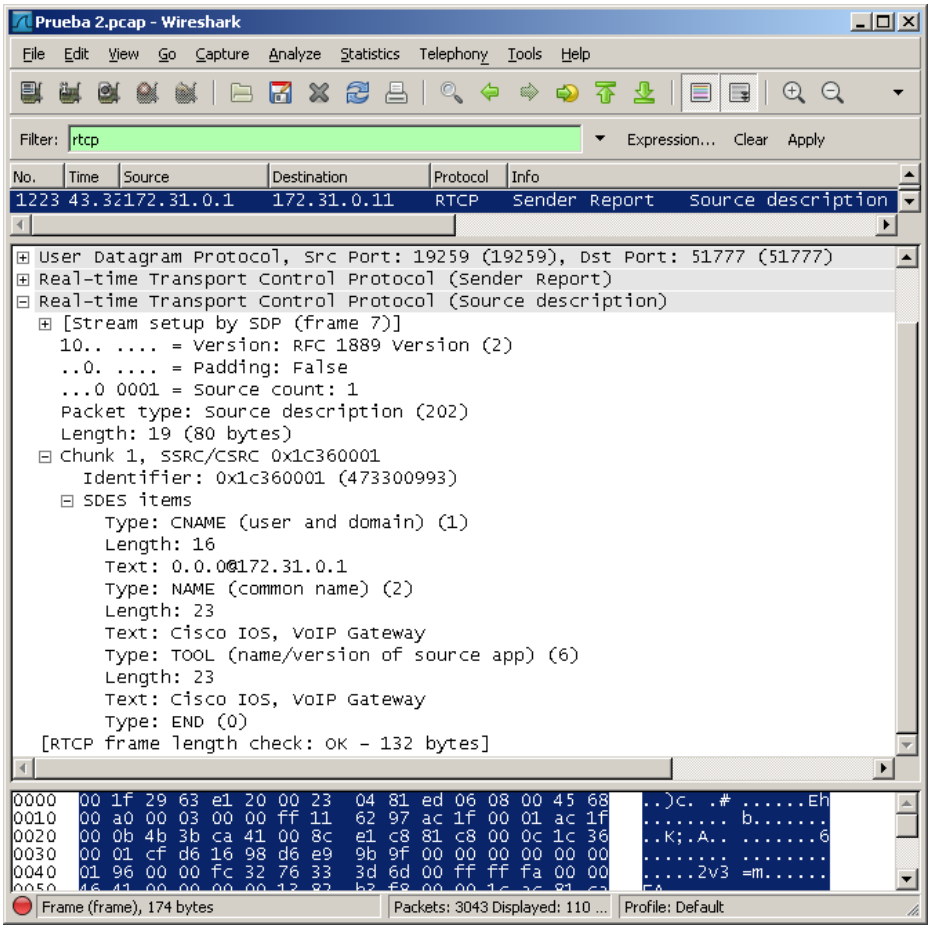

Figura 2.9

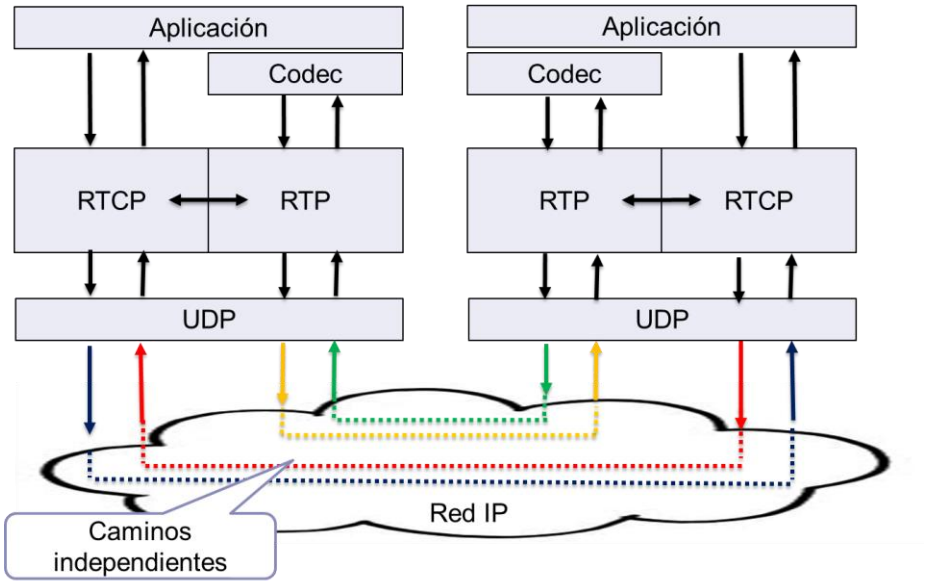

Figura 2.10

2.5 Jitter Buffer

Las redes de paquetes pueden presentar demoras, desde que un paquete es emitido, hasta que el mismo paquete es recibido. Estas demoras se pueden deber a diversos factores, incluyendo tiempos de transmisión y encolamiento en routers o diversos dispositivos de red. En la mayoría de las aplicaciones típicas de redes de datos, como correo electrónico o navegación en páginas web, estas (relativamente pequeñas) demoras no son perceptibles. Sin embargo, en aplicaciones multimedia, aún demoras del orden de pocas decenas de milisegundos pueden ser percibidas (como se describe en el módulo de “Calidad de voz y video”). La siguiente figura ejemplifica el proceso de generación de paquetes de voz, los que atraviesan una red de datos con 100 milisegundos de demora. Se genera un paquete codificando el audio de los 20 milisegundos previos, y se transmite. El tamaño de este paquete es muy pequeño, y también lo es el tiempo ε que tarda en ser generado y serializado. Luego, el paquete demora 100 milisegundos en transitar la red, a su destino. En este escenario, el audio decodificado tiene 120 + 2 ε milisegundos de retardo respecto del audio original.

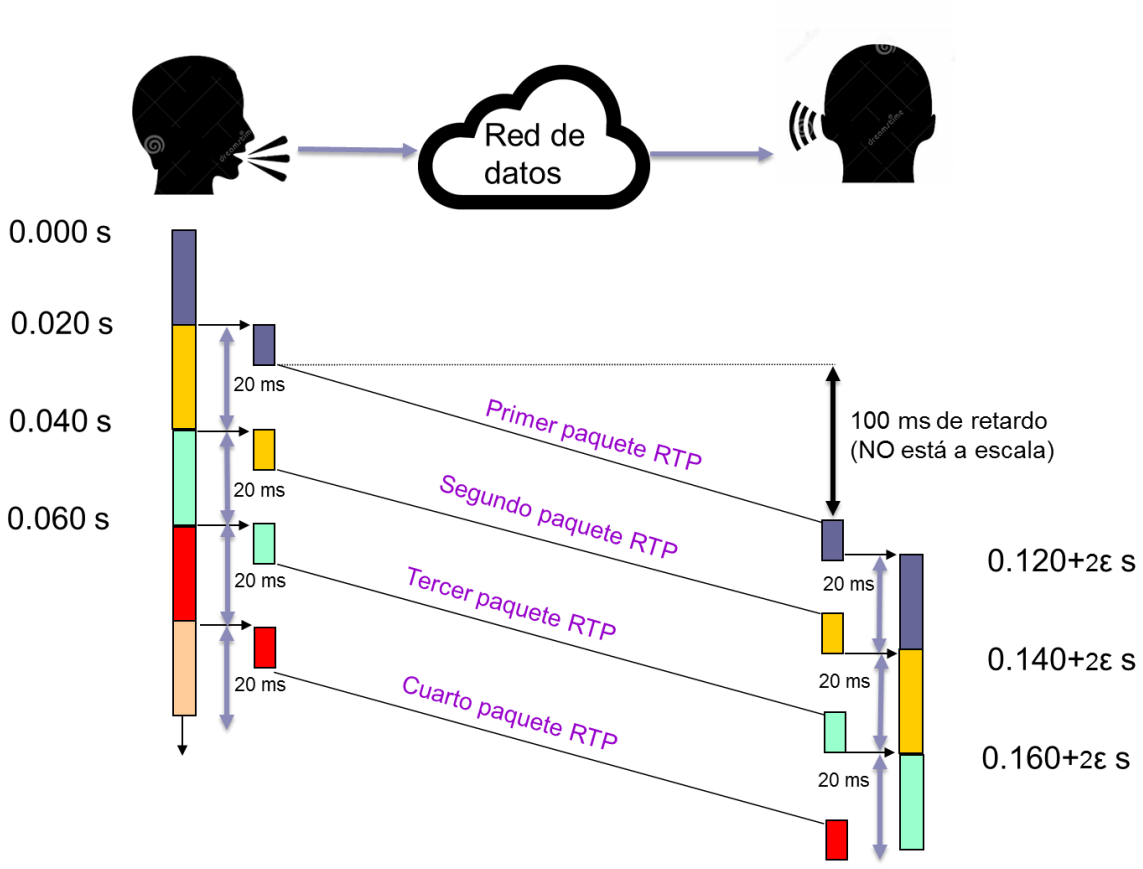

Figura 2.11

Las redes de datos pueden generar demoras diferentes para cada paquete. Esto puede ser debido a los procesos de encolamiento que se produce en algunos dispositivos de red, donde diferentes paquetes deben esperar diferente tiempo hasta ser procesados. Esta situación se representa en la siguiente figura. La variación de las demoras se conoce como “jitter”, y puede medirse según técnicas definidas en el RFC 3550.

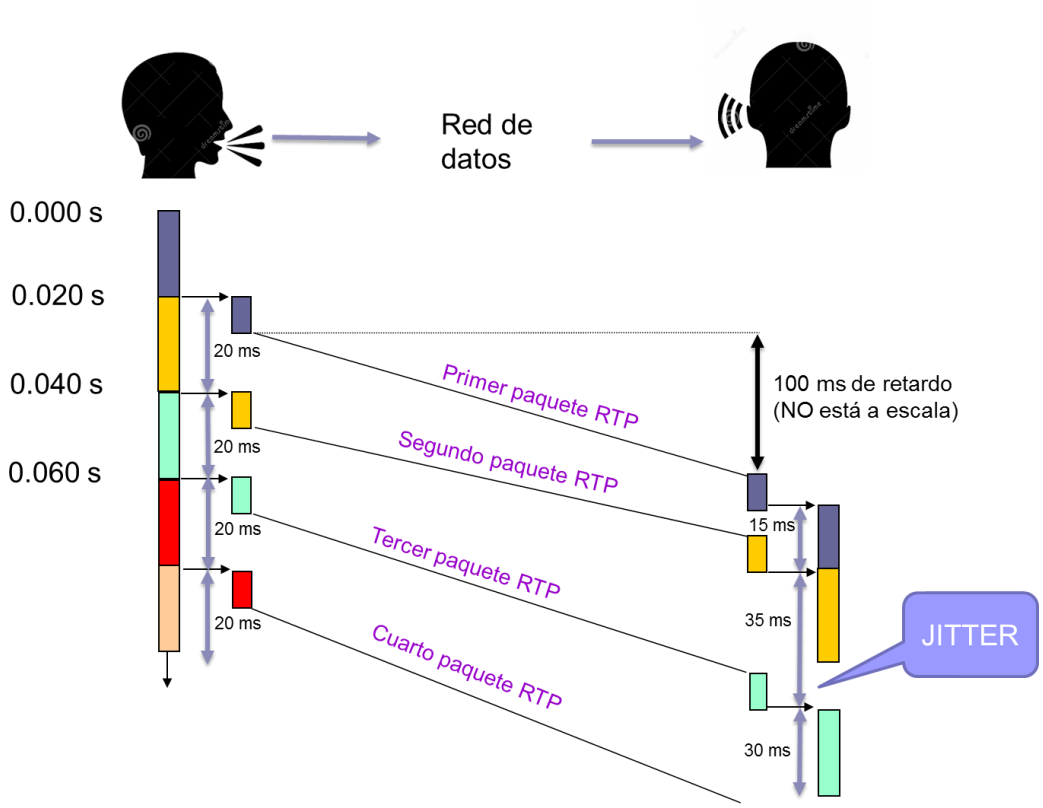

Figura 2.12

La existencia de “jitter” es perjudicial para las aplicaciones de tiempo real, ya que los paquetes recibidos deben ser decodificados a intervalos constantes, idénticos a los utilizados al momento de su codificación. Para realizarlo, el receptor requiere implementar un “buffer” (conocido como “jitter buffer”), para almacenar temporalmente los paquetes recibidos y decodificarlos apropiadamente, tal como se muestra en la siguiente figura. Este buffer resuelve efectivamente el problema de las demoras variables, pero a costa de agregar un retardo adicional en el sistema completo de transmisión. En aplicaciones de Voz sobre IP los jitter buffer típicamente pueden almacenar de 20 a 60 milisegundos de voz, aunque estos valores pueden variar y dependen de cada implementación.

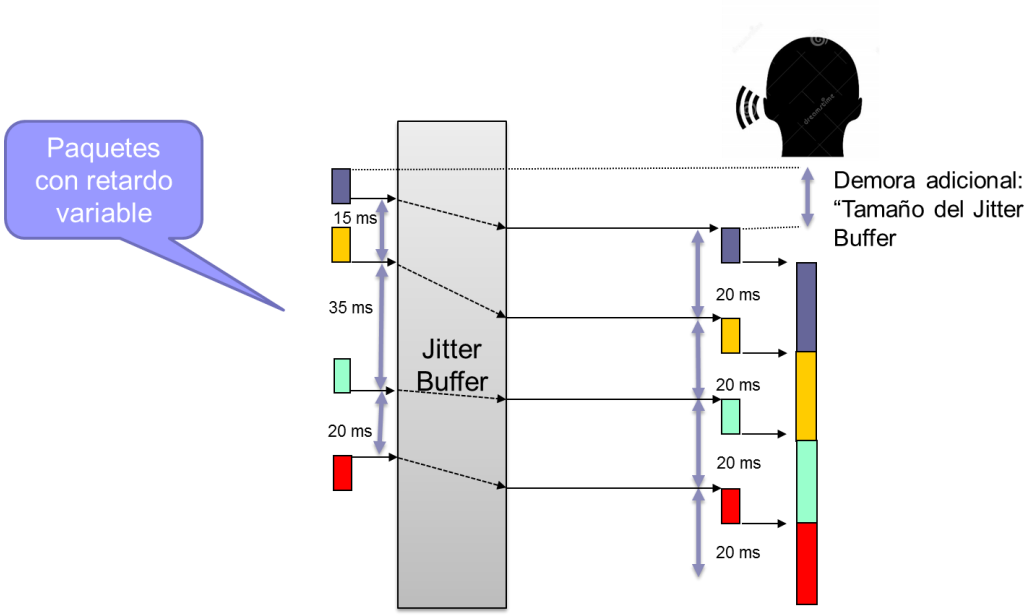

Figura 2.13

2.6 Ancho de banda en IP para voz

Dado que para el envío de voz sobre redes de datos es necesario armar "paquetes", el ancho de banda requerido dependerá de la "sobrecarga" ("overhead") que generen estos paquetes.
Como se ha visto, para el envío de voz sobre redes de paquetes se utiliza el estándar RTP. Éste protocolo a su vez se monta sobre UDP, el que a su vez se monta sobre IP, el que, en la LAN, viaja sobre Ethernet.
Esta suma de protocolos hace que el ancho de banda requerido para el tráfico de voz sobre Ethernet sea bastante mayor al ancho de banda del audio. A continuación se presenta un ejemplo, para el codec G.711.

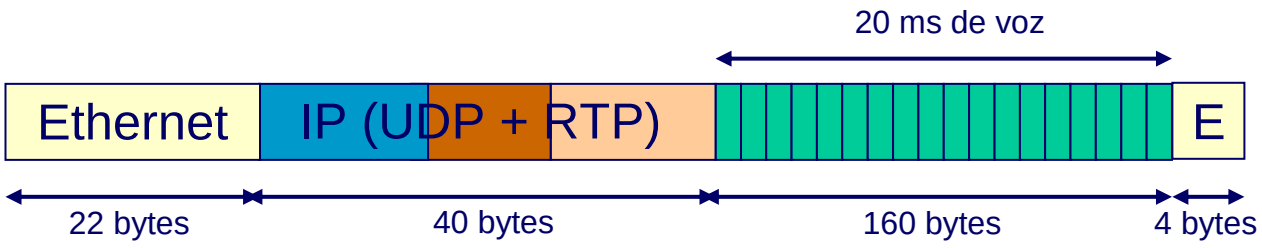

Figura 2.14

Para una ventana de 20 ms, y con codificación de audio G.711 Ley A, se obtienen 160 bytes de voz por trama (Ver Figura 2.14)

Bytes de voz/trama = 64 kb/s * 20 ms / 8 = 160 bytes

El paquete IP (incluyendo los protocolos RTP y UDP) agrega 40 bytes adicionales

Bytes de paquete IP = 160 + 40 = 200 bytes

La trama Ethernet agrega otros 26 bytes:

Bytes de Trama Ethernet = 200 + 26 = 226 bytes

En este ejemplo, cada 20ms se generan 226 bytes que se deben enviar por la LAN. Esto equivale a un ancho de banda de 90,4 kb/s (compárese con los 64 kb/s del flujo de audio)

Ancho de banda LAN = 226 * 8 / 20 ms = 90.4 kb/s

Es de hacer notar que este cálculo fue hecho para el envío de audio en una dirección. Pueden utilizarse técnicas de "supresión de silencio", en las que no se envían paquetes cuando no hay audio. En este caso, el ancho de banda en cada dirección es poco más de la mitad del cálculo anterior.

Por lo visto anteriormente, el ancho de banda de la voz paquetizada en la LAN depende del tamaño de la "ventana" (típicamente 10, 20 o 30 ms) y el codec utilizado.

La Tabla 2.3 muestra los anchos de banda unidireccionales necesarios utilizando redes IP sobre Ethernet para algunos codecs de banda angosta.

Tipo de Codec	Duración de Trama (ms)	Bytes de voz/Trama	Bytes de paquete IP	Bytes de trama Ethernet	Ancho de Banda en LAN (kbps)
G.711 (64 kb/s)	10	80	120	146	116,8
	20	160	200	226	90,4
	30	240	280	306	81,6
G.729 (8 kb/s)	10	10	50	76	60,8
	20	20	60	86	34,4
	30	30	70	96	25,6
G.723.1 (6.3 kb/s)	30	24	64	90	23,9
G.723.1 (5.3 kb/s)	30	20	60	86	22,9

Tabla 2.3

Como se puede ver en la tabla, el ancho de banda puede variar notablemente, dependiendo del codec y la ventana seleccionada.

Análisis similares se pueden realizar para otros codecs, tanto de banda angosta, como de banda ancha, super ancha o completa.

3 Video sobre redes de datos

3.1 Aplicaciones de video

El video es utilizado en diversos tipos de aplicaciones, las que a su vez, tienen diversos requerimientos. La TV es, quizás, la aplicación de video más conocida. Sin embargo, existen en forma cada vez más difundida un nuevo conjunto de aplicaciones de video, entre las que se encuentran la video telefonía, los servicios de video conferencia, la distribución de video a demanda a través de Internet y la IP-TV, por mencionar los más relevantes. Cada una de estas aplicaciones tiene sus características propias en lo que respecta a requerimientos de calidad, velocidades, etc.

En el área corporativa, se nota una creciente importancia de la video-telefonía y las video-conferencias.

La video-telefonía es una aplicación típicamente punto a punto, con imágenes del tipo “cabeza y hombros”, y generalmente poco movimiento. Por otra parte, es una aplicación altamente interactiva, dónde los retardos punta a punta juegan un rol fundamental en la calidad conversacional percibida.

Las aplicaciones de video conferencias son típicamente punto a multi-punto. Al igual que la video telefonía, generalmente tienen poco movimiento. Además de la difusión del audio y el video es deseable en estas aplicaciones poder compartir imágenes y documentos. La interactividad también es típicamente un requisito, aunque podrían admitirse retardos punta a punta un poco mayores que en la video telefonía, ya que los participantes generalmente están dispuestos a “solicitar la palabra” en este tipo de comunicaciones.

Las aplicaciones de difusión (IP-TV o video a demanda por Internet, por ejemplo) son unidireccionales, y los retardos desde la emisión a la recepción pueden ser más grandes. Sin embargo, esto tiene impacto en eventos de tiempo real, como por ejemplo, campeonatos deportivos.

3.2 Codificación de video

Los procesos de codificación y decodificación de video, así como los estándares más utilizados, pueden ser consultados en [4]

A modo de referencia, la Tabla 3.1 resume algunos de los codecs más conocidos.

Característica	MPEG-1	MPEG-2	MPEG-4	H.264/MPEG-4 Part 10/AVC
Tamaño del macro-bloque	16x16	16x16 (frame mode) 16x8 (field mode)	16x16	16x16
Tamaño del bloque	8x8	8x 8	16x16 8x8, 16x8	8x8, 16x8, 8x16, 16x16, 4x8, 8x4, 4x4
Transformada	DCT	DCT	DCT/DWT	4x4 Integer transfor
Tamaño de la muestra para aplicar la transformada	8x8	8x8	8x8	4x4
Codificación	VLC	VLC	VLC	VLC, CAVLC, CABAC
Estimación y compensación de movimiento	Si	Si	Si	Si, con hasta 16 MV
Perfiles	No	5 perfiles, varios niveles en cada perfil	8 perfiles, varios niveles en cada perfil	3 perfiles, varios niveles en cada perfil
Tipo de cuadros	I,P,B,D	I,P,B	I,P,B	I,P,B,SI,SP
Ancho de banda	Hasta 1.5 Mbps	2 a 15 Mbps	64 kbps a 2 Mbps	64 kbps a 150 Mbps
Complejidad del codificador	Baja	Media	Media	Alta
Compatibilidad con estándares previos	Si	Si	Si	No

Tabla 3.1

3.3 Transmisión de video sobre redes de datos

3.3.1 Paquetización del video

Las secuencias (Elementary Streams) son paquetizadas en unidades llamadas PES (Packetized Elementary Streams), consistentes en un cabezal y hasta 8 kbytes de datos de secuencia. Estos PES a su vez, son paquetizados en pequeños paquetes, de 184 bytes, los que, junto a un cabezal de 4 bytes (totalizando 188 bytes) conforman el "MPEG Transport Stream" (MTS) y pueden ser transmitidos por diversos medios.

En redes IP, el transporte del video se realiza mediante los protocolos RTP y RTCP, ya descritos en 2.3. El RFC 2250 [21] establece los procedimientos para transportar video MPEG-1 y MPEG-2 sobre RTP. Varios paquetes MTS de 188 bytes pueden ser transportados en un único paquete RTP, para mejorar la eficiencia. Esto se puede observar en la siguiente figura, donde se muestra un paquete RTP con 7 paquetes MTS (ISO 13818), cada uno de ellos de 188 bytes. A su vez, cada paquete MTS contiene un cabezal de 4 bytes y 184 bytes de "contenido", con el video codificado en MPEG-2, como se ve en la segunda figura.

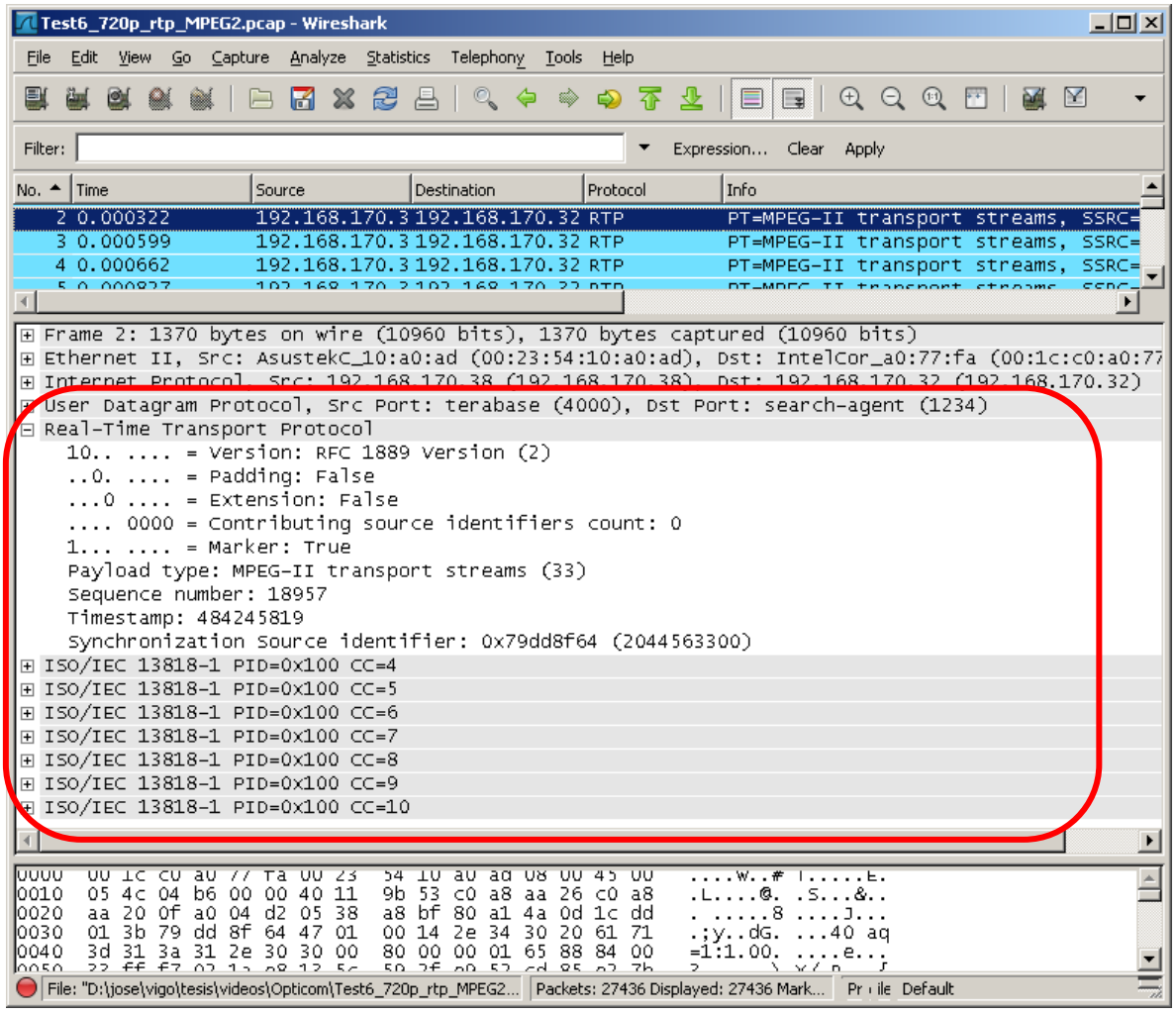
Test6_720p_rtp_MPEG2.pcap - Wireshark
File Edit View Go Capture Analyze Statistics Telephony Tools Help
Filter: Expression... Clear Apply
No. Time Source Destination Protocol Info
2 0.000322 192.168.170.3 192.168.170.32 RTP PT=MPEG-II transport streams, SSRC=
3 0.000599 192.168.170.3 192.168.170.32 RTP PT=MPEG-II transport streams, SSRC=
4 0.000662 192.168.170.3 192.168.170.32 RTP PT=MPEG-II transport streams, SSRC=
Frame 2: 1370 bytes on wire (10960 bits), 1370 bytes captured (10960 bits)
Ethernet II, Src: AsustekC_10:a0:ad (00:23:54:10:a0:ad), Dst: IntelCor_a0:77:fa (00:1c:c0:a0:77
User Datagram Protocol, Src Port: terabase (4000), Dst Port: search-agent (1234)
Real-Time Transport Protocol
10.. .... = Version: RFC 1889 Version (2)
..0. .... = Padding: False
...0 .... = Extension: False
.... 0000 = Contributing source identifiers count: 0
1... .... = Marker: True
Payload type: MPEG-II transport streams (33)
Sequence number: 18957
Timestamp: 484245819
Synchronization Source identifier: 0x79dd8f64 (2044563300)
ISO/IEC 13818-1 PID=0x100 CC=4
ISO/IEC 13818-1 PID=0x100 CC=5
ISO/IEC 13818-1 PID=0x100 CC=6
ISO/IEC 13818-1 PID=0x100 CC=7
ISO/IEC 13818-1 PID=0x100 CC=8
ISO/IEC 13818-1 PID=0x100 CC=9
ISO/IEC 13818-1 PID=0x100 CC=10
0010 05 4c 04 b6 00 00 40 11 9b 53 c0 a8 aa 26 c0 a8 .L....@. .S...&..
0020 aa 20 0f a0 04 d2 05 38 a8 bf 80 a1 4a 0d 1c dd . .....8 ....J...
0030 01 3b 79 dd 8f 64 47 01 00 14 2e 34 30 20 61 71 .;y..dG. ...40 aq
0040 3d 31 3a 31 2e 30 30 00 80 00 00 01 65 88 84 00 =1:1.00. ....e...
File: "D:\jose\vigo\tesis\videos\Opticom\Test6_720p_rtp_MPEG2...
Packets: 27436 Displayed: 27436 Mark...
Default

Figura 3.1

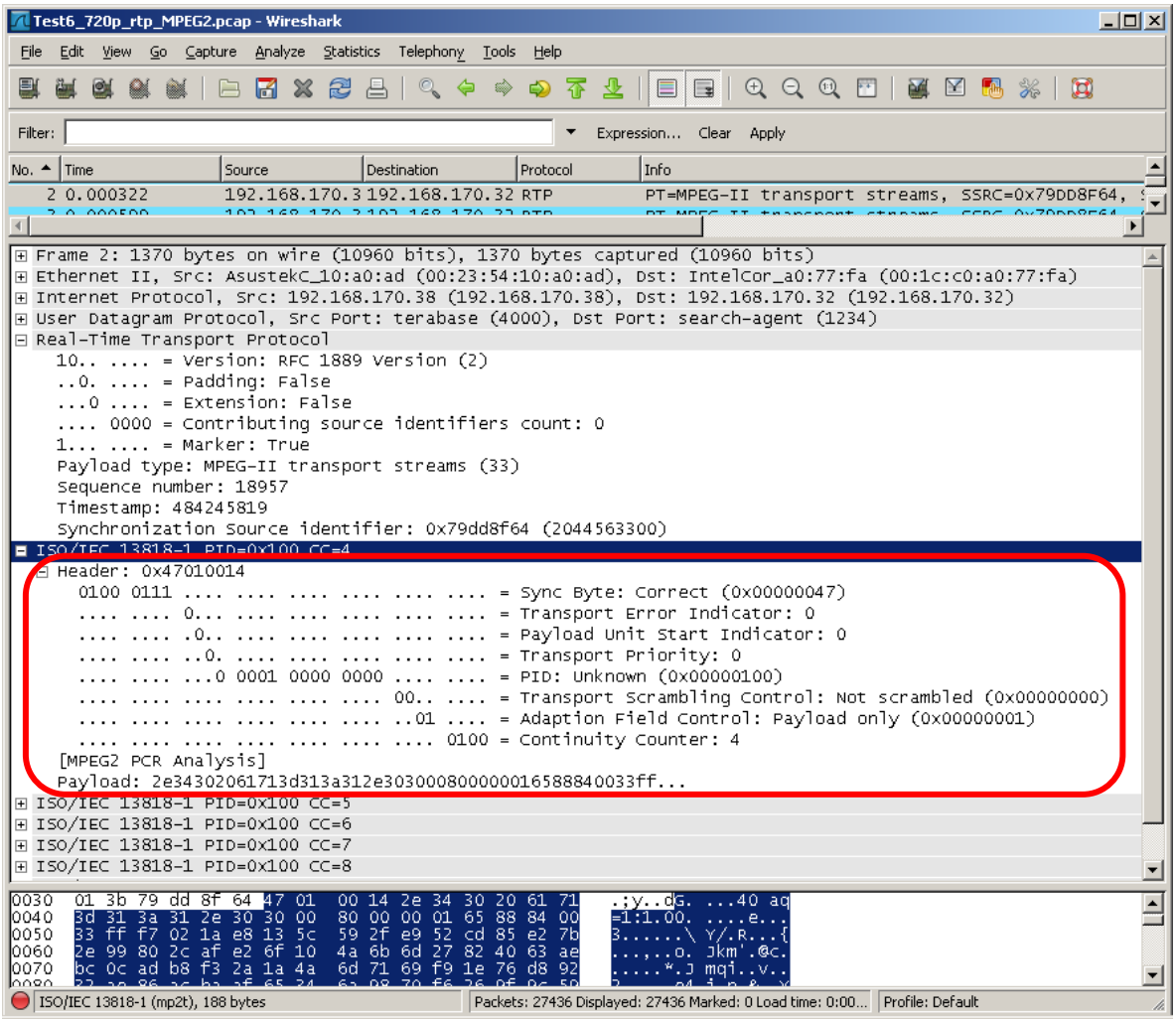

Figura 3.2

Algunos sistemas no utilizan el protocolo RTP, sino que incluyen los paquetes de MTS directamente en el paquete UDP, como se muestra en la siguiente figura. Si bien esta forma de transmisión tiene menos sobrecarga (“overhead”), ya que no se envían los bytes del protocolo RTP, el envío del video paquetizado dentro de RTP tiene varias ventajas [22] :

- El envío de video sobre dentro del protoclo RTP está estandarizado en el RFC 2250 y por el grupo DVB-IP [23], lo que asegura mejor interoperabilidad.
- El protocolo RTP utiliza el protocolo RTCP para como realimentación de la calidad. No hay mecanismos similares estandarizados sin se envía los MTS directamente sobre UDP
- RTP contiene nativamente información de reloj, que puede ser utilizado por el destino para decodificar apropiadamente los flujos multimedia.

- Muchos equipos de redes pueden darle prioridad al protocolo RTP frente a otros protocolos, gestionando de esta manera la QoS (Calidad de Servicio). Esto no es así si los MTS se encapsulan directamente debajo de UDP.

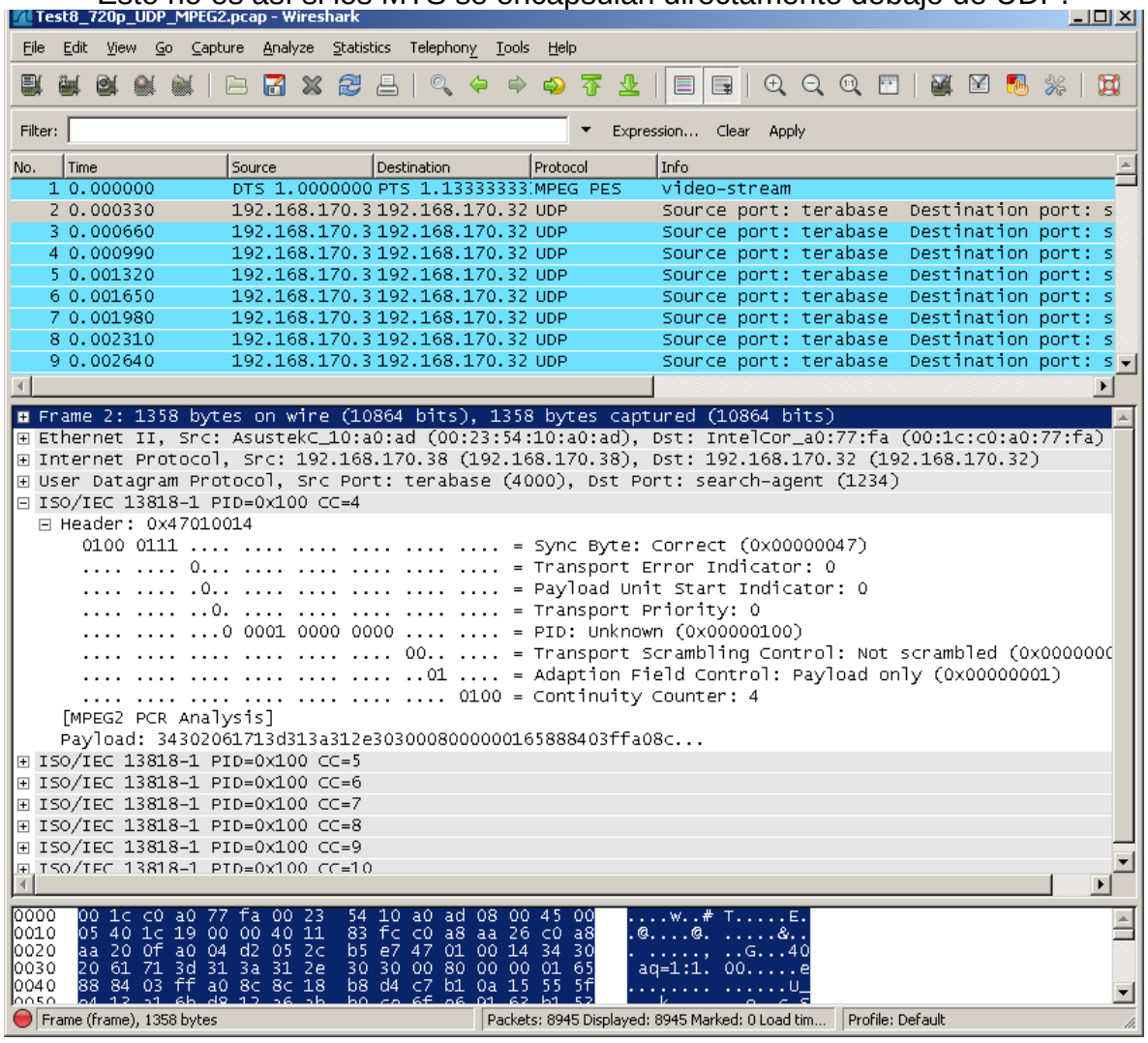

Figura 3.3

Los RFC 3016 [24] y RFC 3640 [25] establecen los procedimientos para transportar flujos de audio y video MPEG-4. El RFC 3984 [26] establece los procedimientos para transportar flujos de video codificados en H.264. El RFC 7798 [27] indica como transportar H.265 (HEVC) sobre RTP. En la siguiente figura se muestra un paquete de video codificado en H.264 dentro de RTP. En este caso se utiliza el "paylod tipe" dinámico, con el número 96. El paquete tiene 1430 bytes de "payload" con el video codificado en H.264.

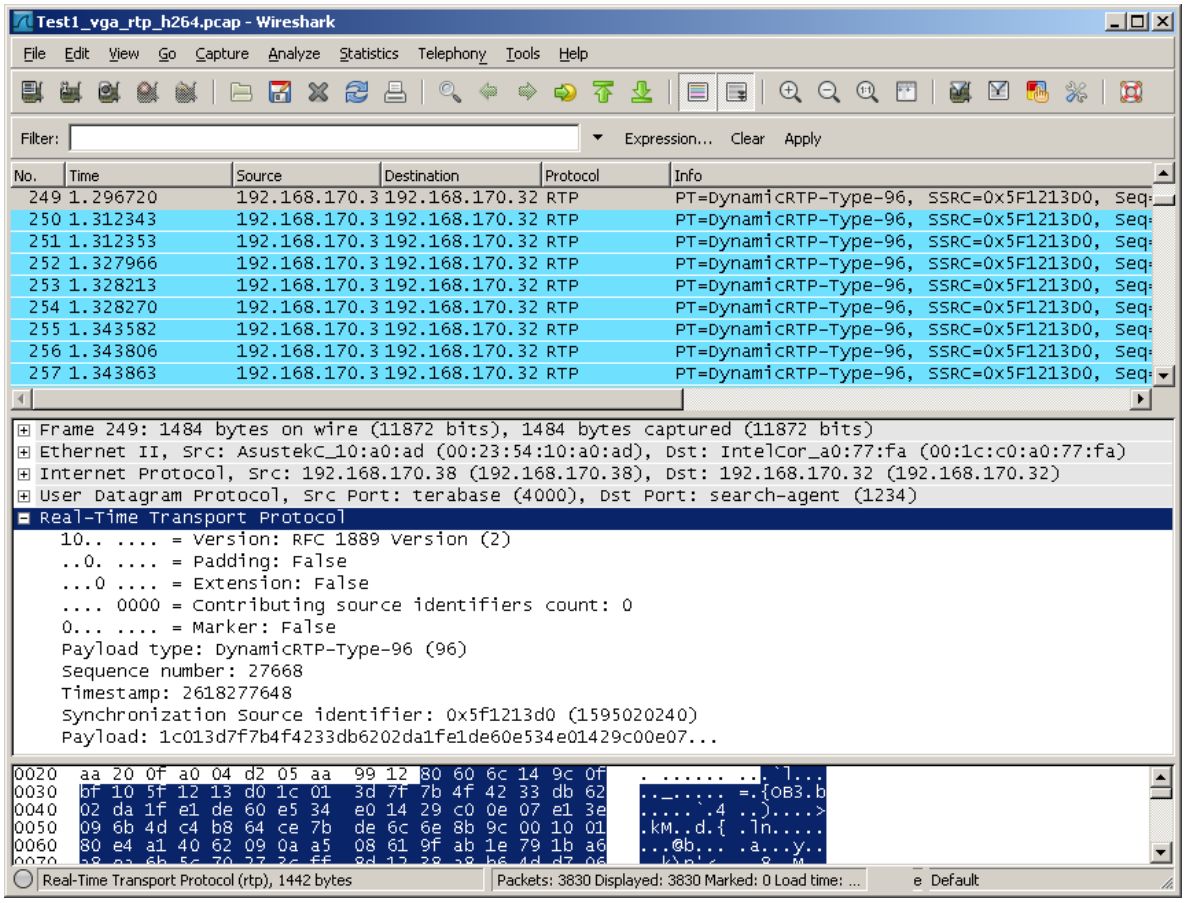

Figura 3.4

3.3.2 Ancho de banda en IP para video

Como se ha visto, la codificación digital de video utiliza algoritmos de compresión, los que generan codificación de largo variable y flujos de ancho de banda también variables. Para una aplicación determinada, el ancho de banda requerido en una red IP dependerá del tipo de codificación utilizada (MPEG-1, 2, 4, H264, etc.), del tamaño de la pantalla (SD, CIF, QCIF, etc), del tipo de cuantización seleccionado y del movimiento y textura de la imagen. Al ancho de banda propio de la señal de video se le debe sumar la sobrecarga de los paquetes IP, UDP y RTP y para la LAN, de las tramas Ethernet. A diferencia de la codificación de audio de tasa de bits constantes, donde los anchos de banda pueden calcularse en forma exacta en base únicamente al codec utilizado, la codificación de video es estadística, y depende de la imagen transmitida, por lo que los cálculos son también aproximados y estadísticos. En video, generalmente, se puede establecer la tasa de bits o ancho de banda deseado, y el codec varía dinámicamente sus parámetros de codificación para alcanzar el ancho de banda establecido, a expensas de modificar la calidad.

A modo de ejemplo, en la Figura 3.5 se muestra coma varía la calidad percibida en función del ancho de banda para diferentes secuencias de video, codificadas en MPEG-2, y en resolución CIF (352 x 288) a 25 cuadros por segundo. En esta gráfica, la calidad percibida se mide como "MOS", donde valores de 5 se corresponden a "excelente calidad" y valores de 1 se corresponden con muy "mala calidad". Cada línea se corresponde con una secuencia de video diferente, tomadas de la página del VQEG [28]. En este caso, para formatos CIF, se puede ver que para anchos de banda superiores a 1.75 Mb/s la calidad es "excelente" para cualquier secuencia de video. Sin embargo, para anchos de banda inferiores a 1 Mb/s la calidad percibida depende fuertemente del contenido del video (movimiento, textura de la imagen, etc.)

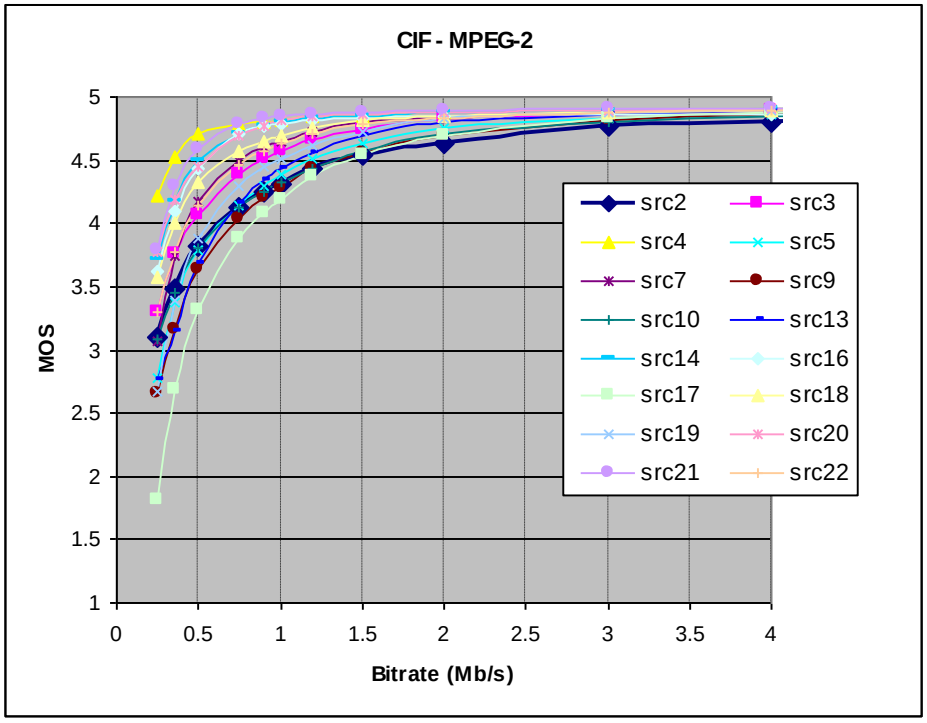

Figura 3.5

Por otra parte, en la Figura 3.6, tomada de [29], se muestra como varía el ancho de banda requerido usando diversos codificadores, en función de la calidad de la imagen, para una secuencia de video particular ("Tempete", src22), en resolución CIF a 15 Hz. Puede verse como para una misma calidad (medida en este caso como PSNR), MPEG-2 requiere de aproximadamente el doble de ancho de banda que H.264/AVC.

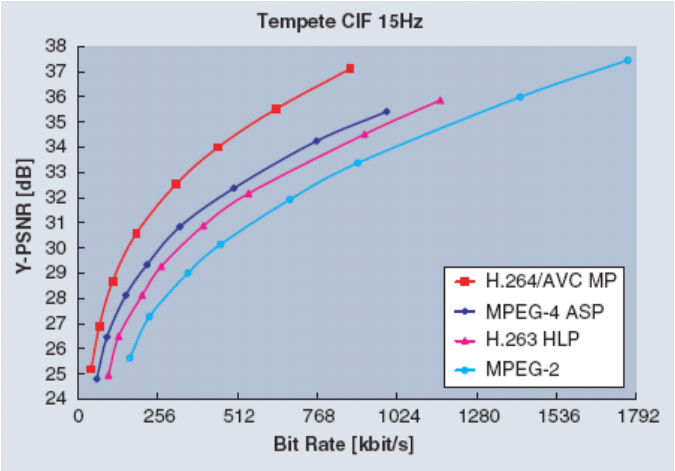

Figura 3.6

Como se puede ver, el ancho de banda de las señales de video puede variar notoriamente, desde valores cercanos a los 64 kb/s (para baja resolución de pantalla, imágenes con poco movimiento, baja cantidad de cuadros por segundo), hasta varios Mb/s (para resoluciones medias o altas).

En redes IP, el "overhead" o sobrecarga depende de la forma de encapsulado utilizada. Como se mencionó anteriormente (Ver 2.6) los procolos IP/UDP/RTP tienen 40 bytes, y Ethernet otro 26 bytes. En el caso de MPEG-2, utilizando MTS encapsultados en RTP, se pueden incluir hasta 7 paquetes MTS dentro de un mismo paquete IP. Cada MTS tiene 4 bytes de cabezal y 184 bytes de contenido. Por tanto, un paquete IP con MPEG-2 está formado como se muestra en la siguiente figura:

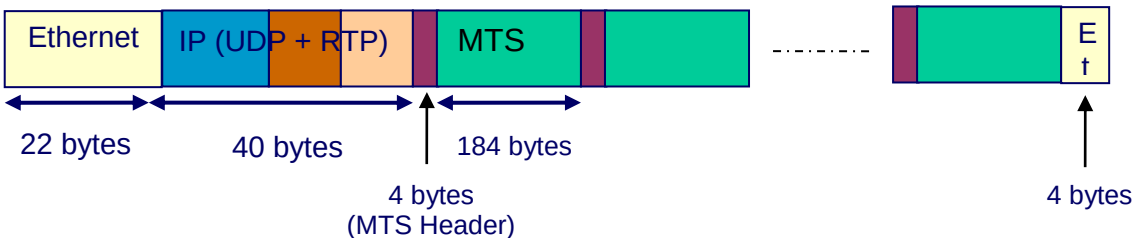

Figura 3.7

En un paquete IP se pueden incluir 7 x184 = 1288 bytes de contenido MPEG-2, y por otra parte hay 40 + 4 x 7 = 68 bytes de cabezales a nivel de capa 3 (IP) y 94 bytes de cabezales a nivel de capa 2 (en Ethernet). Por lo tanto, el ancho de banda de MPEG-2 transportado en RTP es 5.3% (68/1288) mayor que el ancho de banda propio del video en capa 3 (IP) y 7.3 % (94/1288) mayor que el ancho de banda propio del video en capa 2 (Ethernet).

En el caso de H.264 encapsulado directamente sobre RTP, sin utilizar TS, se pueden enviar hasta 1430 bytes de “payload” en un paquete IP/UDP/RTP, por lo que el ancho de banda en capa 3 es 2.8% (40/1430) mayor que el del propio video codificado y en capa 2 es 4.6% (66/1430) mayor que el del propio video codificado.

4 Seguridad en RTP

El protocolo RTP es inseguro, ya que es abierto y fácilmente decodificable. Herramientas habituales (como wireshark) permiten escuchar el audio codificado en flujos RTP de manera muy sencilla. Basta identificar el tráfico RTP, realizar un análisis del flujo, y presionar el botón “Player”.

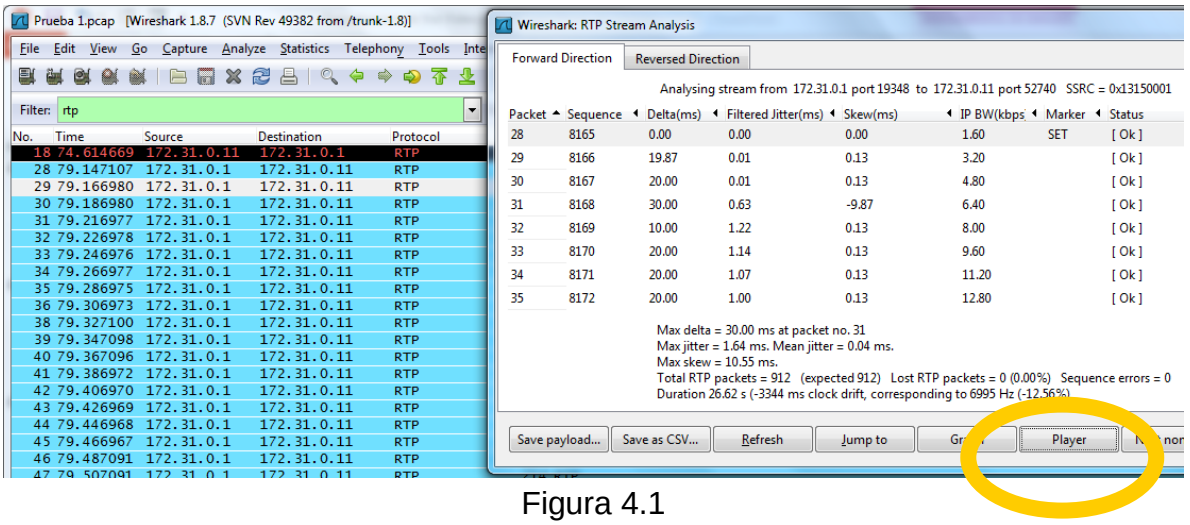

Figura 4.1

Es posible cifrar el medio, a través del protocolo SRTP (Secure RTP), y SRTCP (Secure RTCP), estandarizado en el RFC 3711 [30]. Cuando se implementa SRTP, los paquetes RTP y RTCP son cifrados en la fuente, antes de ser enviados a las capas inferiores de comunicación y descifrados en el destino antes de ser enviados a las capas superiores. Para ello se utilizan técnicas de cifrado AES.
Un paquete SRTP es muy similar al RTP, pero el “payload” se encuentra encriptado, como se muestra en la siguiente figura.

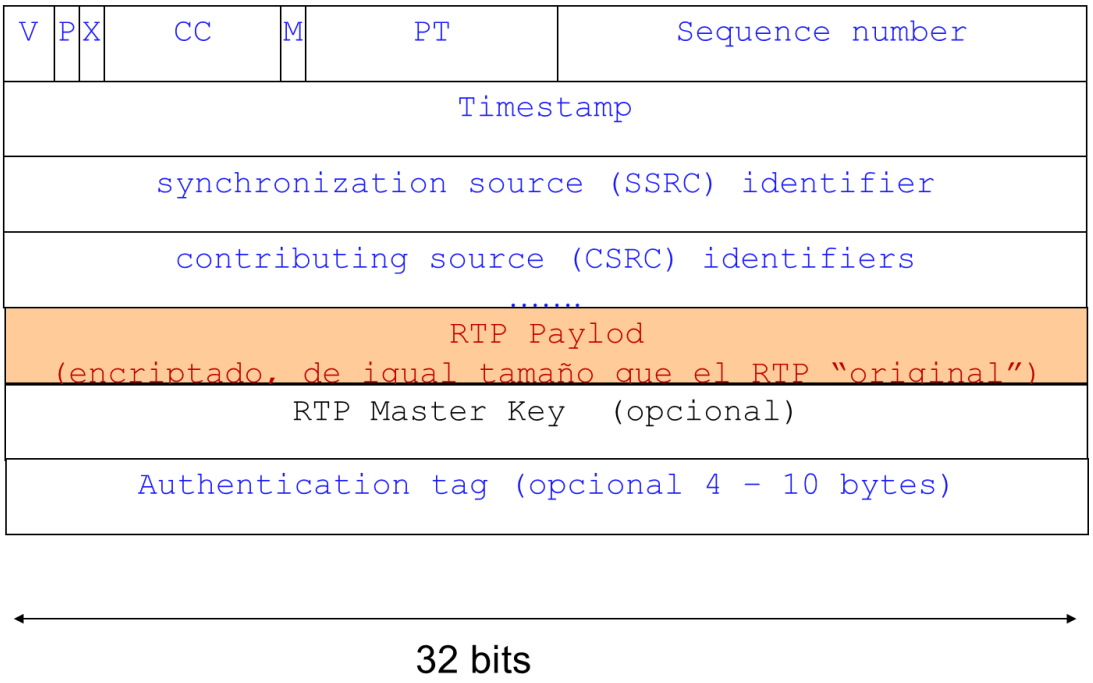

Figura 4.2

Las claves de cifrados utilizados para los protocolos RTP y RTCP son derivadas de una "clave maestra". Esta clave maestra debe ser compartida entre los usuarios, y puede ser obtenida de una entidad externa de administración de claves de cifrado, utilizando los protocolos MIKEY, ZRTP, KEYMGT entre otros.

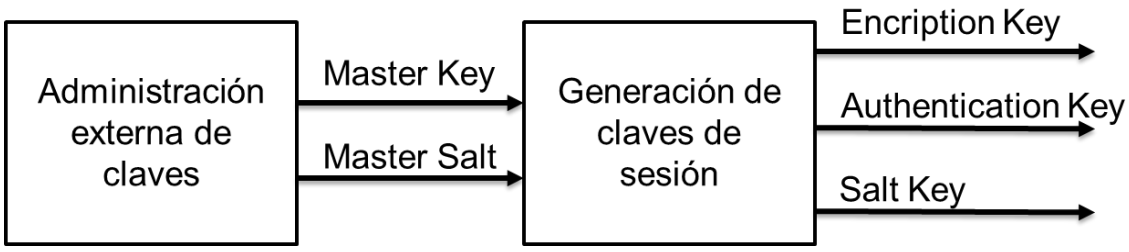

Figura 4.3

El medio (RTP) es encriptado mediante una función "XOR" con un generador de flujos AES, cuyo valor se obtiene en función de la clave de encripción y un vector de inicialización.

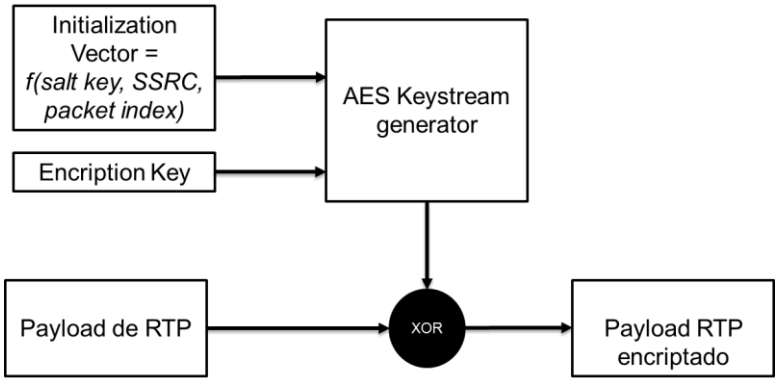

Figura 4.4

Adicionalmente, y en forma opcional, es posible autenticar los paquetes generados, utilizando la clave de encripción y algoritmos HMAC, basado en una función de Hash SHA-1 de 160 bits

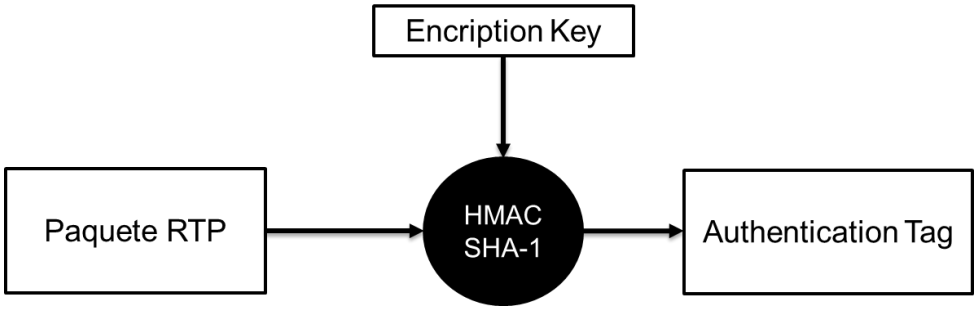

Figura 4.5

Referencias

[1] Conceptos de Telefonía Corporativa, Versión 08, José Joskowicz (Julio 2009)

[2] Redes de Datos, Versión 05, José Joskowicz (Agosto 2008)

[3] Cisco, Telegeography, SCF Associates Ltd , 2014

[4] Codificación de Voz y Video, José Joskowicz (Marzo 2011)

[5] Recommendation G.711: “Pulse Code Modulation (PCM) of voice frequencies”, CCITT, 1988.

[6] Recommendation G.723.1: “Dual Rate speech coder for multimedia communications transmitting at 5.3 and 6.3 kbit/s”, ITU-T, May 2006.

[7] Recommendation G.728: “Coding of speech at 16 kbit/s using Low-delay code excited linear prediction”, CCITT, 1992.

[8] Recommendation G.729: “Coding of speech at 8 kbits using Conjugate-Structure Algebraic-Code-Excited Linear-Prediction (CS-ACELP)”, ITU-T, Jan 2007.

[9] Adaptive Multi-Rate (AMR) speech codec, ETSI TS 126 090 V9.0.0, 2010-01

[10] Recommendation G.722: “7 kHz audio-coding within 64 kbit/s”, CCITT, 1988.

[11] Recommendation G.722.1: “Low-complexity coding at 24 and 32 kbit/s for hands-free operation in systems with low frame loss”, ITU-T, 05/2005.

[12] Recommendation G.722.2: “Wideband coding of speech at around 16 kbit/s using Adaptive Multi-Rate Wideband (AMR-WB)”, ITU-T, 07/2003.

[13] Recommendation G.711.1: “Wideband embedded extension for G.711 pulse code modulation”, ITU-T, 03/2008.

[14] Recommendation G.729.1: “G.729-based embedded variable bit-rate coder: An 8-32 kbit/s scalable wideband coder bitstream interoperable with G.729”, ITU-T, 05/2006.

[15] Overview of the Microsoft RTAudio Speech Codec, Microsoft, 2006.

[16] SILK – Super Wideband Audio Codec, https://developer.skype.com/silk

[17] Recommendation G.719: “Low-complexity, full-band audio coding for high-quality, conversational applications”, ITU-T, 06/2008.

[18] RFC 3550: “RTP: A Transport Protocol for Real-Time Applications”, H. Schulzrinne et al (July 2003)

[19] RFC 3551: “RTP Profile for Audio and Video Conferences with Minimal Control”, H. Schulzrinne et al (July 2003)

[20] RFC 2833: “Payload for DTMF Digits, Telephony Tones and Telephony Signals”, H. Schulzrinne et al (May 2000)

[21] RFC 2250 Payload Format for MPEG1/MPEG2 Video
D. Hoffman et al, January 1998

[22] WHITEPAPER – IP Streaming of MPEG-4: Native RTP vs MPEG-2 Transport Stream

Alex MacAulay, Boris Felts, Yuval Fisher, October 2005

[23] DVB IP Phase 1 handbook , ETSI TS 102 034, “Digital Video Broadcasting (DVB); Transport of MPEG-2 Based DVB Services over IP Based Networks”, March 2005.

[24] RFC 3016 RTP Payload Format for MPEG-4 Audio/Visual Streams
Y. Kikuchi et al, November 2000

[25] RFC 3640 RTP Payload Format for Transport of MPEG-4 Elementary Streams, J. van der Meer et al, November 2003

[26] RFC 3984 RTP Payload Format for H.264 Video, S. Wenger et al, febrero 2005

[27] RFC 7798 RTP Payload Format for High Efficiency Video Coding (HEVC), Y.-K. Wang et al, marzo 2016

[28] VQEG Phase I Test Sequences. [Online]. Disponibles en: ftp://vqeg.its.bldrdoc.gov/SDTV/VQEG_PhaseI/TestSequences/Reference/

[29] Video coding with H.264/AVC: Tools, Performance, and Complexity
Jörn Ostermann, Jan Bormans, Peter List, Detlev Marpe, Matthias Narroschke, Fernando Pereira, Thomas Stockhammer, and Thomas Wedi
IEEE Circuits and Systems Magazine, First Quarter 2004

[30] RFC 3711 The Secure Real-time Transport Protocol (SRTP)
M. Baugher et al. March 2004