Señalización para voz IP (VoIP)

Ferran Adelantado i Freixer

PID_00201837

Tiempo de lectura y comprensión: **4 horas**

UOC
Universitat Oberta de Catalunya

Los textos e imágenes publicados en esta obra están sujetos –excepto que se indique lo contrario– a una licencia de Reconocimiento-NoComercial-SinObraDerivada (BY-NC-ND) v.3.0 España de Creative Commons. Podéis copiarlos, distribuirlos y transmitirlos públicamente siempre que citéis el autor y la fuente (FUOC. Fundació per a la Universitat Oberta de Catalunya), no hagáis un uso comercial y no hagáis una obra derivada. La licencia completa se puede consultar en http://creativecommons.org/licenses/by-nc-nd/3.0/es/legalcode.es

Índice

Introducción 5

Objetivos 6

1. Voz IP: los fundamentos 7

2. El conjunto de protocolos *Signaling Transport* (SIGTRAN) 11
2.1. *Stream Control Transmission Protocol* (SCTP) 13
2.1.1. Protección contra ataques 15
2.1.2. Transmisión fiable 19
2.1.3. Redundancia 19
2.1.4. Bajo retardo 20
2.1.5. Resumen de las características de la SCTP 21
2.2. *User Adaptation Layers* (UA) 22
2.2.1. *MTP2-User Adaptation* (M2UA) 23
2.2.2. *MTP2-user Peer-to-peer Adaptation* (M2PA) 25
2.2.3. *MTP3-User Adaptation* (M3UA) 28
2.2.4. *SCCP User Adaptation* (SUA) 30

3. El protocolo *Session Initiation Protocol* (SIP) 32
3.1. Diálogos, transacciones y mensajes 34
3.1.1. Formato de los mensajes 34
3.2. La sesión SIP 38
3.2.1. Establecimiento de la sesión 38
3.2.2. Finalización de la sesión 41
3.2.3. Modificación de la sesión 43
3.3. Servicios ofrecidos por SIP 44

4. El futuro de la señalización en las redes IP 47

Resumen 48

Ejercicios de autoevaluación 49

Glosario 50

Bibliografía 52

Introducción

La mejora de las redes de telecomunicación –en términos de ancho de banda, retardo, fiabilidad, extensión, etc.– ha permitido en todo el mundo, pero particularmente en los países más desarrollados, la aparición y oferta de servicios de banda ancha. En este contexto, las comunicaciones de voz no han quedado al margen y la voz sobre IP (en inglés *voice over IP*, VoIP) toma fuerza en el presente y de cara al futuro.

A pesar del aumento imparable de la demanda de servicios de banda ancha, así como el despliegue constante de nuevas redes de paquetes, no hay que olvidar que las redes de conmutación de circuitos existentes (RTC o RDSI) son una parte muy importante del total de redes de acceso y de transporte actuales. En el futuro parece evidente que las redes de conmutación de circuitos perderán importancia y serán sustituidas progresivamente por redes de conmutación de paquetes (redes IP). A pesar de esta certeza, una de las necesidades principales de hoy en día es conseguir la interoperabilidad de las nuevas redes de paquetes y las viejas redes de conmutación de circuitos.

Es en el marco de esta interoperabilidad que este módulo presenta dos partes bien diferenciadas. La primera parte trata el conjunto de protocolos SIGTRAN, la solución definida por la IETF para interconectar la RTC o RDSI con la red IP. El objetivo final de este conjunto de protocolos es definir una capa de transporte que permita cumplir con los estrictos requisitos de la señalización SS7, así como establecer un grupo de capas de adaptación de usuario que permitan el intercambio de señalización entre puntos de señalización de la red SS7 y nodos de la red IP.

La segunda parte del módulo, en cambio, se centra en uno de los protocolos de señalización de las redes IP más extendidos y prometedores: el *Session Initiation Protocol* (SIP).

Objetivos

Los objetivos que debe alcanzar el estudiante una vez estudiados los materiales didácticos de este módulo son:

1. Entender el cambio de paradigma en las comunicaciones de voz y qué la ha motivado.
2. Conocer el conjunto de protocolos que permiten la interconexión de la red conmutada de telefonía y la red IP.
3. Entender los fundamentos de las capas de adaptación de usuario definidas en SIGTRAN.
4. Saber cuáles son las características que hacen del protocolo SCTP un buen candidato para el transporte de señalización.
5. Entender, de manera introductoria, el protocolo SIP y su importancia en las redes IP.

1. Voz IP: los fundamentos

Hasta hace relativamente pocos años el concepto de voz sobre IP (más conocido por el acrónimo inglés VoIP, es decir, *voice over IP*) era desconocido. Se asociaba el protocolo IP, de la capa de red, a la transmisión de datos más que a la transmisión de voz. Ahora, sin embargo, no solo se conoce el concepto sino que la voz IP es una realidad.

Hay dos motivos básicos por los que la voz IP, una tecnología que inicialmente presentaba serias dudas, ha acabado teniendo éxito:

- **La disminución de los costes**. El paso de la telefonía basada en conmutación de circuitos hacia una telefonía basada en conmutación de paquetes (telefonía IP) consigue un incremento de la eficiencia. Ya no es necesario reservar circuitos para una única llamada, sino que la transmisión por conmutación de paquetes permite maximizar la utilización de cada enlace (multiplexando varios flujos de datos o voz). En términos de señalización, además, la implantación de la voz IP supone no tener que mantener una red paralela de señalización*, con el consecuente ahorro. Fijémonos, en términos de eficiencia, que los sistemas de compresión que permiten la voz digitalizada (como por ejemplo la detección de actividad para disminuir los bits transmitidos en periodos de silencio) minimizan el tráfico transportado a través de las redes de comunicación. En cualquier caso, la reducción de costes se traduce en una disminución del precio del servicio.

*** Recordemos que la red de señalización SS7 era paralela a la red RTC.**

- **Independencia de la distancia**. La tarificación y señalización de llamadas internacionales mediante VoIP no añade dificultades adicionales respecto de las llamadas nacionales, como sí sucede en el caso de las llamadas a través de la RTC (Red Telefónica Conmutada).

- **La convergencia de las redes**. El aumento de las comunicaciones de datos durante las últimas décadas ha sido enorme. Este hecho, teniendo en cuenta que la mayoría de estas transmisiones de datos se basan en el protocolo de red IP, ha producido un aumento de la extensión de las redes de comunicaciones en todo el mundo. En este contexto, las redes de telefonía no podían quedar al margen de la tendencia a converger hacia las denominadas redes *All-IP* que han experimentado todas las redes de telecomunicaciones.

- **Nuevos servicios**. La telefonía IP, por definición, permite ofrecer tanto el servicio de datos como de vídeo y datos a través de una red IP.

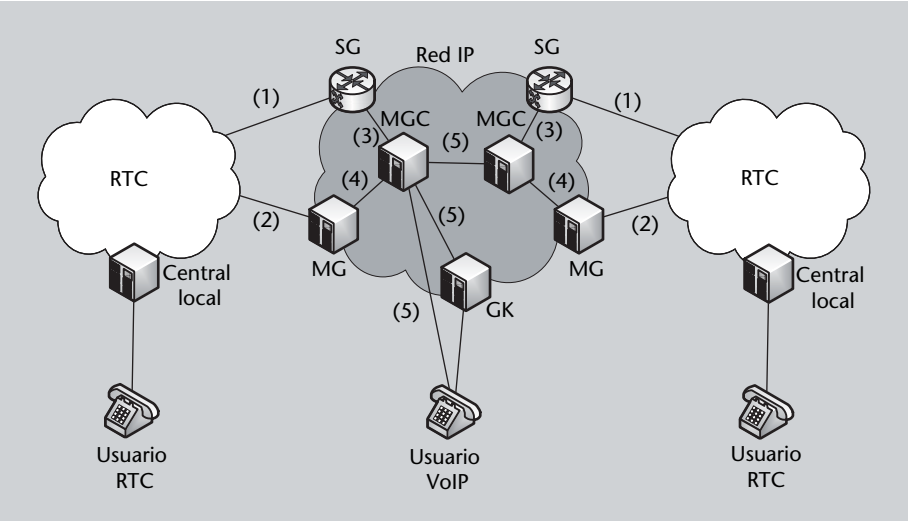
Aún así, también es bueno destacar las desventajas de la implementación de la VoIP:

- **Necesidad de ancho de banda en el bucle local**. La utilización de telefonía IP obliga a tener conexión de datos con una mínima anchura de banda en el bucle local. Pese a que este era un problema grave en el momento de la aparición de la VoIP, se trata de un problema que ha ido disminuyendo gracias a la extensión de las redes de acceso de banda ancha.
- **Llamadas de emergencia**. Las llamadas de emergencia en la RTC permiten determinar la localización del abonado, un hecho determinante para la redirección necesaria de la llamada. Esta localización del abonado no es inmediata en el caso de la VoIP, ya que la dirección IP no está necesariamente ligada a una dirección física.
- **Cortes de luz**. Cuando hay un corte del servicio eléctrico el terminal de VoIP queda fuera de servicio.
- **Calidad de servicio**. Garantizar calidad de servicio (en inglés, *Quality of Service*, QoS) mediante el acceso a Internet no es fácil. Para la voz IP se necesita garantizar, entre otras cosas, un retardo máximo para los paquetes IP.

La tecnología asociada a la VoIP y su desarrollo, sin embargo, no puede obviar la existencia –y la extensión– de la red telefónica *convencional* existente, la llamada RTC. Es por este motivo que, a pesar del prometedor futuro de la VoIP, hay que afrontar el reto de interconectar las redes IP y las redes RTC.

La figura 1 muestra de manera esquemática la interconexión de las dos redes y los nodos principales que lo permiten. En la figura se representan dos redes RTC y una red IP.

Figura 1. Conexión de las redes RTC e IP



Para tratar de facilitar su comprensión, se ha añadido un usuario final (también llamado *User Agent*, UA) a cada una de las redes y los nodos fundamentales, entre los que cabe destacar los siguientes:

- **Signalling Gateway (SG)**. Se trata de un nodo fundamental que conecta la red de señalización de la RTC con el *Media Gateway Controller* (MGC) de la red IP. Este nodo está íntimamente relacionado con la señalización SS7 que hemos detallado en el módulo anterior, ya que permite adaptar la señalización de la RTC para que se pueda transportar a través de la red IP. Conviene darse cuenta, de acuerdo con esta descripción, que los enlaces marcados con un (1) en la figura 1 son enlaces SS7. En la salida del SG el conjunto de protocolos utilizados, ya en la red IP, es el llamado SIGTRAN, marcado con un (3) en la figura. Este conjunto de protocolos son objeto de estudio en los apartados próximos.

SIGTRAN

Esta es la denominación del conjunto de protocolos definidos con el objetivo de adaptar los mensajes recibidos en un formato SS7 para transportarlos a través de una red IP. El marco general para este protocolo se puede consultar en el RFC 2719 de la IETF.

- **Media Gateway (MG)**. Es el nodo que convierte la información que le llega de los circuitos de voz de la RTC a un formato adecuado para la red IP. Por tanto, y de acuerdo con la nomenclatura de la figura, los enlaces marcados con un (2) son enlaces de voz de la RTC. El nodo convierte las conexiones de voz para transferir la información en formatos adecuados para la red IP, como por ejemplo el RTP (*Real Time Transport Protocol*). Este protocolo, a pesar de su importancia, queda fuera del alcance de esta asignatura, y se trabaja en otras asignaturas del plan de estudios. La relación entre el MG y el MGC se hace a través del protocolo definido por la ITU-T, el H.248.1* (con un (4) en la figura).

Protocolo H.248.1

El H.248.1 es una Recomendación de la ITU-T. Se trata de una nueva versión de un protocolo inicialmente definido por la IETF con el nombre de MEGACO e incluido en la RFC 3525.

- **Media Gateway Controller (MGC)**. Este nodo controla tanto el SG como el MG. Así, establece comunicación entre ambos nodos para procesar la información de la llamada. Tendría funciones similares a las de un conmutador en una RTC. También se conoce con el nombre de *Call agent*. Por otro lado, también se comunica y controla un conjunto de *gatekeepers* (GK).

- **Gatekeeper (GK)**. Las funciones básicas de este nodo son el control de acceso a la red y la traducción de direcciones IP. La VoIP basa su direccionamiento, como cualquier servicio sobre IP, en la dirección IP. Aún así, el servicio de telefonía se basa en un sistema de numeración diferente (definido en la Recomendación E.164 de la ITU-T). La traducción de un sistema a otro, o la decisión sobre el direccionamiento de los paquetes, la lleva a cabo el GK. Como los *gatekeepers* habitualmente realizan sus funciones para un conjunto de terminales/usuarios de una zona, pueden establecer comunicación con otros *gatekeepers* para hacer la traducción de la dirección y el direccionamiento. La comunicación entre dos GK, con el MGC o con el usuario se realiza de acuerdo con protocolos como el *Session Initiation Protocol* (SIP) o el H.323. En la figura se señala con el (5).

SIP y H.323

Son dos de los protocolos de señalización más importantes para VoIP. En el caso del SIP, se trata de un protocolo definido por la IETF, mientras que el H.323 ha sido definido por la ITU-T.

Para hacer un poco más comprensible la estructura de la figura 1, proponemos dos ejemplos sencillos que deben permitir visualizar los procesos que ocurren en la red.

Ejemplo 1

Imaginemos una llamada generada por un usuario conectado a la RTC hacia un usuario conectado, directamente, a la red IP. La señalización llega al SG desde la red SS7. Se establece una conexión entre el SG y el MGC (con los protocolos SIGTRAN). A partir de este momento el MGC establece una conexión de señalización entre el MGC y el MG, utilizando el protocolo H.248, para que se establezca una conexión de datos entre el usuario final y el MG. Esta conexión de datos sería una conexión RTP sobre IP.

Ejemplo 2

El segundo ejemplo es el caso contrario, es decir, la llamada es generada por un usuario conectado a la red IP y el usuario final está conectado a la RTC. En este caso, en primer lugar el usuario se comunica con el GK tanto para que traduzca el número de teléfono del usuario final en una dirección IP, como para que determine el direccionamiento apropiado para llegar al usuario final (conectado a la RTC). Una vez obtenida esta información del GK, el usuario inicial establece un canal de señalización con el MGC que, a su vez, establece un canal de señalización entre el MGC y el MG, y entre el MGC y el SG. Luego, el MG establece la conexión de datos entre el usuario y el MG.

De lo que hemos visto hasta ahora debemos tener en mente las siguientes conclusiones:

- La señalización y el circuito de voz, en una RTC, utilizan redes diferentes y, por tanto, llegan a la red IP a través de diferentes nodos (la señalización a través del SG y la voz a través del MG).

- Como tanto la voz como la señalización tienen formatos diferentes en la RTC (o RDSI) y en la red IP, se necesitan protocolos que permitan la adaptación.

Este módulo tiene un doble objetivo. Por un lado profundizar en la adaptación de la señalización de la red SS7 para que pueda ser transportada a través de la red IP. Por otra parte, introducir el protocolo de señalización más utilizado para conexiones de voz en una red IP, el SIP.

2. El conjunto de protocolos *Signaling Transport* (SIGTRAN)

El conjunto de protocolos del SS7 presenta varias características que dificultan su adaptación a las redes IP. En concreto, la pila de protocolos SS7 define niveles/capas equivalentes a las tres capas bajas del modelo OSI (desde la capa MTP1, considerada la capa física, hasta la capa MTP3, que sería la capa de red). Conviene recordar, además, que el protocolo IP es un protocolo de la capa de red (capa 3), y que por este motivo es difícil adaptar las dos pilas de protocolos.

El conjunto de protocolos denominados con el nombre de SIGTRAN tiene la intención de permitir el transporte, a través de la red IP, de la señalización SS7.

Los protocolos de transporte más habituales para IP son el protocolo TCP (*Transmission Control Protocol*) y el protocolo UDP (*User Datagram Protocol*). En cuanto a la conexión de voz en tiempo real, las redes IP usan el protocolo de transporte RTP (*Real Time Transport Protocol*). Ahora bien, para la conexión de señalización SS7, la IETF, durante la definición del conjunto de protocolos SIGTRAN, tuvo que decidir qué protocolo de transporte se utilizaba. Los requisitos que debía cumplir cualquier protocolo de transporte eran los siguientes:

- **Transporte fiable de la información de señalización**. Tal como se ha estudiado en el módulo anterior, la información de señalización es muy sensible y es por este motivo que la capa MTP implementa mecanismos de corrección de errores.

- **Retardos bajos**. El retardo máximo permitido para la información de señalización está limitado en la red SS7 por los requisitos que impone el servicio de telefonía ofrecido por la RTC y/o la RDSI.

- **Redundancia**. La red SS7 está definida de forma que garantiza redundancia gracias a las múltiples rutas posibles. Es por este motivo que los nodos STP están desplegados por parejas y con enlaces redundantes.

Véase también

En el módulo "Un sistema universal de señalización: *Signalling System No. 7 (SS7)*" de esta asignatura se explica que la capa MTP implementa mecanismos para una transmisión/recepción fiable y manteniendo el orden de los mensajes.

Para hacer frente a estos requisitos, el protocolo de transporte utilizado para la señalización SS7 no puede ser el UDP. Como es sabido, el protocolo de transporte UDP no implementa control de flujo ni corrección de errores. Así pues, la utilización de UDP como protocolo de transporte no permitiría garantizar ni el transporte fiable de la información ni la redundancia. En cambio, sí que permitiría tener retardos bajos.

Lectura complementaria

En caso de que queráis profundizar en el conocimiento de los protocolos TCP y UDP, podéis leer el libro *The TCP/IP guide: A comprehensive, illustrated Internet Protocol Reference*.

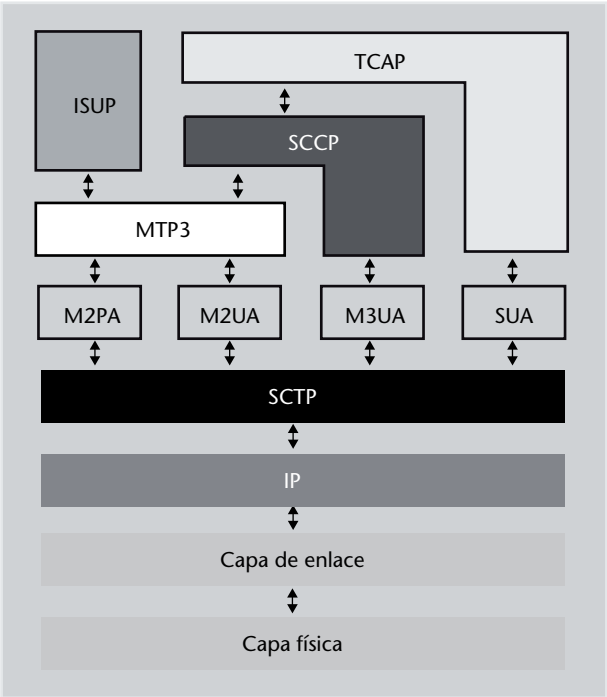
En cuanto al protocolo TCP, ofrece un servicio de transporte fiable y ordenado de la información, pero en cambio no es apropiado para garantizar retardos bajos. El protocolo, por definición, implementa un mecanismo de control de flujo con una ventana deslizante y requiere la notificación positiva de recepción de los paquetes transmitidos. El problema de TCP es que los errores de transmisión de un paquete retrasan la transmisión de los paquetes siguientes y, por tanto, afectan negativamente al retardo. Además, TCP presenta problemas ante los llamados ataques DoS (*Denial of Service*).

Los protocolos de la capa de transporte TCP y UDP no son el objeto de estudio de esta asignatura. A pesar de todo, hay numerosas referencias que permiten profundizar en el caso de que se quieran conocer mejor.

Así pues, el conjunto de protocolos SIGTRAN define un nuevo protocolo de transporte sobre IP que permite garantizar los requisitos exigidos para la señalización SS7: el *Stream Control Transmission Protocol* (SCTP). Una vez definido el protocolo de la capa de transporte, SIGTRAN define un conjunto de capas de adaptación de usuario entre los protocolos definidos en el SS7 y el protocolo de transporte SCTP. Estas capas de adaptación de usuario (en inglés *User Adaptation layer*, UA) son las cuatro que se muestran en la figura 2: *MTP2-user Peer-to-peer Adaptation* (M2PA), *MTP2-User Adaptation* (M2UA), *MTP3-User Adaptation* (M3UA), y *SCCP User Adaptation* (SUA)*.

*** SIGTRAN también define capas de adaptación de usuario para la señalización de acceso, como por ejemplo V5UA y IUA, pero su estudio no es el objetivo de esta asignatura.**

Figura 2. Protocolos definidos por SIGTRAN



Tal como se puede observar en la figura 2, tenemos una capa de red basada en IP y una red de transporte basada en el protocolo SCTP. Por encima, una capa de adaptación de usuario compuesta por cuatro protocolos. Cada uno tiene la función de adaptar un tipo diferente de usuario haciendo transparente a los usuarios de capas superiores qué capas hay por debajo (en este caso SCTP e IP).

Es decir, las capas M2UA y M2PA deben conseguir que, para la capa MTP3, no haya diferencia entre transportar los mensajes con SCTP o hacerlo con MTP2. Lo mismo se puede decir de la M3UA, que debe permitir al SCCP funcionar correctamente, o del SUA, que debe hacer lo mismo con la TCAP.

En este apartado nos centraremos en aquellos protocolos que componen el SIGTRAN: SCTP, M2UA, M2PA, M3UA y SUA.

2.1. *Stream Control Transmission Protocol* (SCTP)

El protocolo SCTP tiene el objetivo de hacer frente a los requisitos que plantea la señalización SS7 y que ya hemos señalado. En general el protocolo SCTP presenta muchas semejanzas con el protocolo TCP e introduce algunas mejoras. Se trata, pues, de un protocolo de transporte fiable (hay notificaciones y retransmisiones), en ambas direcciones (es decir, *full-duplex*), y que implementa un control de congestión y de flujo muy parecido al implementado en TCP y basado en el concepto de *Additive-Increase Multiplicative-Decrease* (AIMD)*.

*** Este es uno de los puntos clave en el protocolo y se implementa mediante los mecanismos de *slow start* y *congestion avoidance*.**

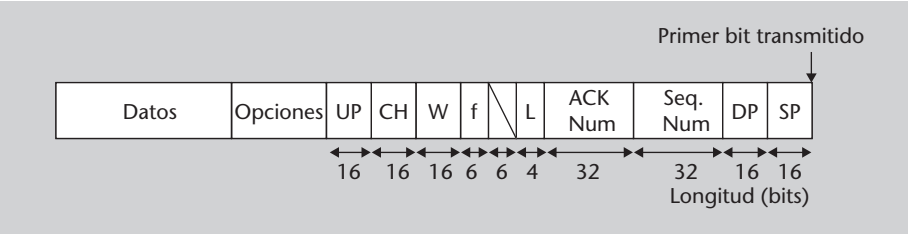
Tal como es sabido, el control de congestión y de flujo TCP (de la misma manera que ocurre en SCTP) se basa en la recepción de confirmaciones positivas de recepción de paquetes y en el llamada ventana deslizante *sliding window*. El receptor no necesariamente debe notificar la recepción correcta de cada uno de los mensajes enviados por TCP, sino que puede reconocer un conjunto de mensajes para disminuir el número de mensajes de notificación que hay que enviar**.

**** Este mecanismo de notificación es muy similar al mecanismo llamado *go-back N* pero permite descartar mensajes que llegan fuera de secuencia.**

Otra de las características del protocolo SCTP que lo diferencia del protocolo TCP es que está orientado a mensaje y no a byte. De hecho, este es un aspecto que se puede observar fácilmente si miramos el formato de un segmento TCP en la figura 3 y de una *Protocolo Data Unit* (PDU) de SCTP en la figura 4.

Es conveniente constatar que, en esencia, los segmentos de TCP y las PDU de SCTP tienen estructuras similares. Ahora bien, la PDU transporta la información dividida en *chunks*, mientras que el segmento TCP no divide los datos en unidades más pequeñas. Este es un punto clave para superar el problema del retardo excesivo, tal como se explicará a continuación.

Figura 3. Segmento del protocolo TCP



Los parámetros del segmento TCP son los siguientes:

Protocolo TCP

Si queréis consultar con más detalle la estructura del segmento TCP, podéis consultar la RFC 793.

- **Puerto de origen (*Source Port*, SP) y Puerto de destino (*Destination Port*, DP)**. Los puertos de destino y de origen permiten determinar las aplicaciones de origen y de destino. Cada uno tiene una longitud de 16 bits.

- **Número de secuencia (*Sequence Number*)**. Campo que permite identificar el primer byte del campo de datos. Es a partir de este valor que se realiza la notificación de recepción correcta. Tiene una longitud de 32 bits.

- **Número de notificación (*Acknowledgement number*)**. Este campo de 32 bits es utilizado para notificar la recepción correcta de un segmento anterior. El valor de este campo notifica la recepción correcta de todos los bytes que tienen un número de secuencia inferior.

- **Longitud de la cabecera (*Header length*, L)**. El valor de la longitud se expresa en múltiplos de 32 bits (es decir 4 bytes).

- ***Flags* (f)**. Seis campos, de un bit de longitud cada uno, que determinan si el segmento contiene el puntero a datos urgentes (UP), si el campo de *acknowledgement number* tiene información válida, el tipo de gestión de la memoria interna del receptor, el aborto de la conexión, si se trata de un mensaje *Synchronize* (inicio de conexión) o *Finalize* (Final de la conexión). Todos juntos tienen una longitud de 6 bits.

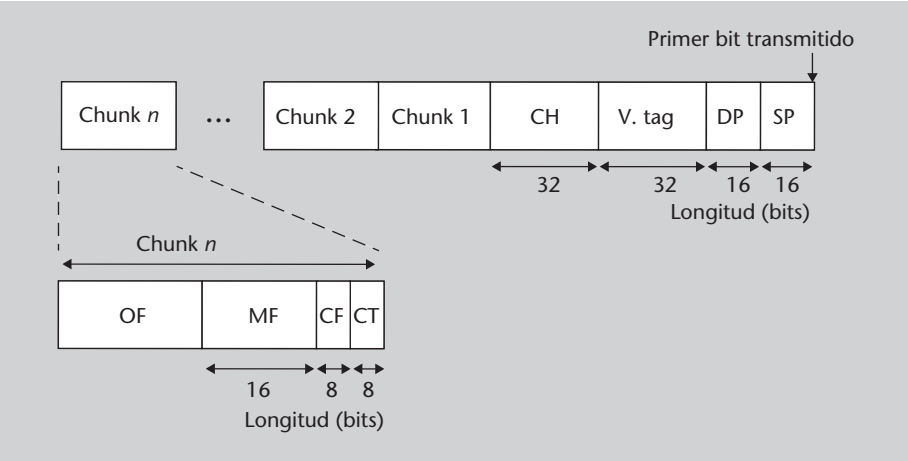
- **Ventana de recepción (*Receive window*, W)**. Tamaño de la ventana que el receptor comunica al transmisor. Tiene una longitud de 16 bits.

- **Detector de errores (*Checksum*, CH)**. Tiene una longitud de 12 bits.

- **Puntero a datos urgentes (*urgent data pointer*, UP)**. Campo de 16 bits que permite indicar la localización, dentro de los datos del segmento, de los datos urgentes (en caso de que existan).

- **Opciones**. Conjunto de campos opcionales.

Figura 4. PDU del protocolo SCTP



Por su parte, los parámetros de las PDU de SCTP son los que se detallan a continuación:

Protocolo SCTP

Si queréis consultar con más detalle la estructura de la PDU del SCTP, podéis consultar la RFC 4960.

- **Puerto de origen (*Source Port*, SP) y Puerto de destino (*Destination Port*, DP)**. Estos campos tienen la misma función que los segmentos TCP. Cada uno tiene una longitud de 16 bits.
- **Verificación (*Verification tag*, V. tag)**. Campo de 32 bits que se utiliza para verificar la identidad del transmisor.
- **Detector de errores (*Checksum*, CH)**. Tiene una longitud de 32 bits y realiza la misma función que los segmentos TCP.
- ***Chunk***. El *chunk* es la unidad de transporte de datos o de señalización. El *chunk* tiene una longitud variable, en función del tipo de *chunk*. En cuanto a la estructura interna de cada uno de los *chunks*, la figura 4 muestra que consta del tipo de *chunks*, los *flags* (no definidos en el estándar y, por tanto, no se utilizan), la longitud de los *chunks*, y, finalmente, los datos.
- **Tipo de *chunk* (*Chunk Type*, CT)**. Campo de 8 bits de longitud. Los valores más destacados que puede tomar son DATA (*payload data*), INIT (*Initiation*), INIT ACK (*Initiation acknowledgement*), SACK (*selective acknowledgement*), ABORT, SHUTDOWN, SHUTDOWN ACK, o SHUTDOWN COMPLETE.
- ***Flags* del *chunk* (*Chunk flags*, CF)**. Campo de 8 bits no utilizado actualmente.
- **Campos obligatorios (*Mandatory Fields*, MF)**. El número de campos obligatorios y su longitud depende del tipo de *chunk*.
- **Campos opcionales (*Optional Fields*, OF)**. El número de campos opcionales es variable.

Una vez descrita la estructura de los segmentos TCP y de las PDU de SCTP, pasamos a describir cómo el nuevo protocolo de transporte hace frente a las necesidades de la señalización SS7. Para hacerlo, describiremos los problemas que presenta TCP y la solución que presenta la propuesta de SCTP.

Terminología

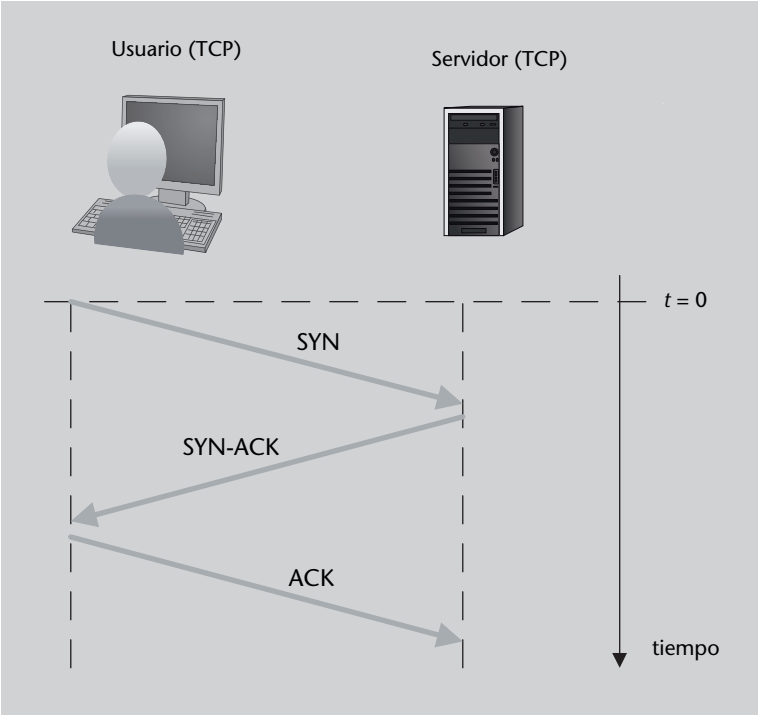
La nomenclatura utilizada por el SCTP varía ligeramente respecto de la utilizada por el TCP. Mientras que la relación establecida entre dos *endpoints* TCP se denomina **conexión**, los dos *endpoints* SCTP establecen una **asociación**.

2.1.1. Protección contra ataques

El protocolo TCP es vulnerable al ataque llamado *Denial of Service* (DoS). Para entender este ataque hay que tener claro cuál es el procedimiento de establecimiento de una conexión TCP. La conexión entre dos nodos TCP se hace mediante el llamado *three-way handshake*. Este procedimiento consiste, como muestra la figura 5, en el proceso siguiente:

1) Inicialmente el nodo que inicia el establecimiento de la conexión transmite un segmento con uno de los bits del campo de *flags* igual a 1. Este bit es el que llamamos bit de *synchronize* o de inicio de conexión y suele denominarse con SYN. Otro de los bits del campo de *flags*, el bit de *acknowledgement* (ACK), se coloca a 0. Pese a que durante este proceso de establecimiento se pueden negociar varios parámetros de la conexión, los campos que hay que incluir obligatoriamente son el puerto de destino (que es conocido), el puerto

Figura 5. Establecimiento de una conexión TCP



de origen (que es seleccionado por el transmisor) y el número de secuencia. Como se trata del primer segmento de la conexión, el valor se puede escoger aleatoriamente.

2) Al recibir el segmento SYN, el receptor responde con un segmento con los bits del campo de *flags* de *synchronize* y de *acknowledgement* (SYN+ACK) iguales a 1. En cuanto al número de secuencia, se utiliza un número aleatorio seleccionado por el receptor. Como el objetivo de este segmento es confirmar la correcta recepción del segmento previo, el campo *acknowledgement number* debe ser igual al número de secuencia del mensaje SYN más 1*.

*** El valor de la *acknowledgement number* siempre debe ser igual al valor del primer segmento que aún no ha sido confirmado. Es por este motivo que se suma 1 al valor del número de secuencia del segmento que se quiere confirmar.**

3) Una vez recibido el segmento (SYN+ACK), el transmisor responde con un segmento de confirmación (*acknowledgement*, ACK). El número de notificación que hay que enviar en esta ocasión es igual al número de secuencia enviado por el receptor en el segmento (SYN+ACK) más 1.

El nodo receptor (aquel que no inicia el establecimiento) puede sufrir un ataque DoS si recibe una gran cantidad de segmentos SYN y, posteriormente, después de responder con un segmento (SYN+ACK) para cada segmento recibido, no recibe el segmento ACK que finalice el proceso de establecimiento. Con un número suficientemente grande de intentos de establecimiento sin finalizar, el nodo receptor puede tener problemas de memoria.

El protocolo SCTP soluciona este problema mediante un mecanismo de establecimiento de asociación en cuatro etapas (*four-way handshake*), a menudo también conocido como *cookie mechanism*.

El establecimiento de asociación de SCTP sigue los siguientes pasos (figura 6):

1) El SCTP que inicia el establecimiento envía una PDU con el campo *Chunk Type* igual a INIT*. El campo *verification tag* permite identificar la identidad de los otros nodos. Antes de enviar la PDU, el transmisor entra en un estado llamado *cookie wait*. Este estado se mantiene durante un tiempo máximo. En caso de no haber recibido una respuesta con la *cookie*, el transmisor abandona este estado y el establecimiento de asociación queda abortado.

*** Las PDU que contengan un *chunk* igual a INIT o INIT ACK solo pueden tener este *chunk*. No es posible utilizar una misma PDU para transmitir un INIT o INIT ACK con cualquier otro *chunk*.**

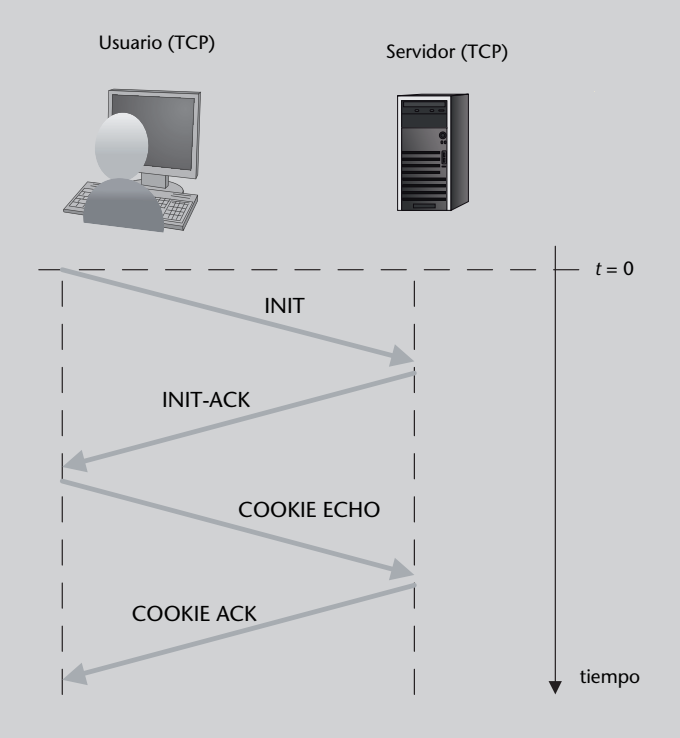
2) El receptor, al recibir la PDU de inicio de asociación, responde con una PDU de notificación de inicio de asociación correcta (con un *chunk type* igual a INIT ACK*). El receptor necesita incluir en la PDU el parámetro llamado *state cookie***.

**** Podéis encontrar la descripción de cómo se genera la *state cookie* en la RFC 4960 de la IETF.**

3) La recepción por parte del transmisor de la *state cookie* hace que abandone el estado *cookie wait*. En este instante, el transmisor envía hacia el receptor una PDU con un *chunk type* igual a COOKIE ECHO y que contiene el *cookie state* recibido. Seguidamente entra en el estado llamado *cookie echoed* y inicializa el tiempo máximo que el *endpoint* puede permanecer en este estado.

4) El receptor, por último, pasa al estado *established* y transmite la última PDU del establecimiento: una PDU con un *chunk type* igual a COOKIE ACK. El transmisor, al recibir esta PDU, también pasa al estado *established* y la asociación se da por establecida.

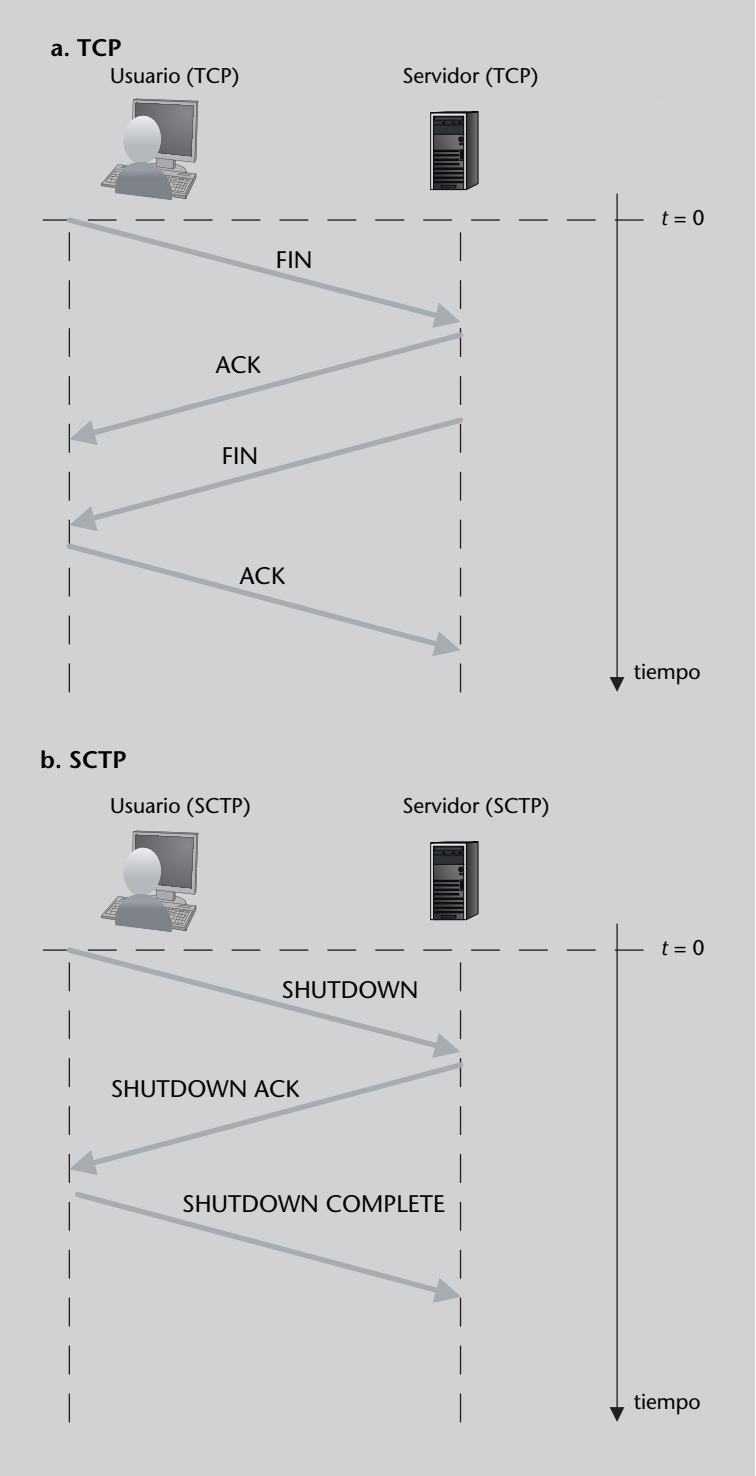
Figura 6. Establecimiento de una asociación SCTP



Tal como se ha podido comprobar, gracias a la existencia de temporizadores y de un establecimiento de asociación estructurado en cuatro fases, el protocolo SCTP es capaz de hacer frente a los ataques DoS.

El intercambio de datos sigue el mismo procedimiento (con ligeras diferencias, ya que la notificación de recepción en SCTP se hace mediante un *chunk* igual a *selective acknowledgement*, SACK) tanto para TCP como para SCTP, pero la finalización presenta ciertas diferencias. La diferencia básica es que en el protocolo TCP cada uno de los *endpoints* libera la conexión enviando un segmento de *Finalize* (FIN) y recibiendo un segmento de *acknowledgement* (ACK), como muestra la figura 7 (a). En cambio, el protocolo SCTP hace una liberación de la asociación mediante un procedimiento en tres etapas, tal como muestra la figura 7 (b).

Figura 7. Liberación de una (a) conexión TCP y de una (b) asociación SCTP



La liberación de la asociación SCTP en tres fases asegura que la asociación queda definitivamente cerrada. En el caso de TCP, en cambio, como la finalización se hará desde los dos extremos de la conexión, si uno de los extremos no transmite el segmento FIN, se dice que la conexión queda medio cerrada (en inglés, *half closed connection*).

2.1.2. Transmisión fiable

La transmisión fiable de la información (en este caso, y para esta aplicación en concreto, la señalización SS7) hace referencia tanto a la detección de errores y la posterior retransmisión de la información errónea, como a la entrega ordenada de las PDU.

Tal como ya se ha destacado, el control de flujo y de congestión de SCTP está completamente basado en los mismos mecanismos definidos para el protocolo TCP. Este hecho garantiza que el SCTP ofrezca un transporte de información fiable.

2.1.3. Redundancia

La redundancia, en el contexto en el que nos encontramos, se puede definir como la capacidad de una red de tener rutas alternativas a través de las cuales direccionar los paquetes IP en caso de fallo de alguna de las rutas. Si recordamos la estructura de la red SS7, uno de los puntos clave es conseguir redundancia. Para conseguirlo, la red SS7 duplica tanto los nodos (por ejemplo, las parejas de STP) como los enlaces.

El objetivo, pues, del protocolo SCTP es conseguir redundancia entre dos *endpoints* SCTP, y lo consigue incorporando una de las características más destacadas respecto al protocolo TCP: **el *multi-homing*.**

Para entender el concepto de *multi-homing* conviene recordar que la dirección de transporte de TCP es el resultado de una dirección de red (en este caso una dirección IP) y un puerto. Por tanto, una conexión TCP se establece entre una dirección IP más un puerto, y otra dirección IP más un puerto. En el momento en el que una de las direcciones IP se convierte en inaccesible, la conexión no es posible. **No hay redundancia**.

Para hacer frente a la falta de redundancia del protocolo de transporte TCP, el protocolo SCTP define el *multi-homing*. En este protocolo de transporte, la dirección de transporte de cada uno de los *hosts* se expresa como un conjunto de direcciones IP y un único puerto. Así pues, un *host/endpoint* SCTP puede dirigirse mediante un único puerto pero mediante varias direcciones IP (tantas direcciones IP como interfaces tenga el nodo).

A pesar del hecho de poder establecer una asociación SCTP a través de varias direcciones IP, una de las direcciones IP es considerada la principal/primaria, y el resto son consideradas direcciones alternativas. Es en el caso de que la dirección IP principal sea inaccesible que se utilizan las direcciones alternativas. Las direcciones alternativas también son utilizadas para las retransmisiones en caso de error.

Aunque se podría pensar que el *multi-homing* permite distribuir el tráfico entre las diversas interfaces del nodo SCTP, tal como sucede con el mecanismo basado en el SLS definido en la capa MTP3, el *multi-homing* ha sido definido con el único propósito de conseguir tener redundancia, y no para distribuir la carga de tráfico.

Los dos nodos de una asociación SCTP intercambian la lista de direcciones IP alternativas de cada uno durante el proceso de establecimiento.

El funcionamiento correcto del *multi-homing* requiere la detección de los fallos. Hay dos tipos de fallos posibles: o bien que el *endpoint* sea inaccesible (fallo de la red), o bien que el *endpoint* falle. En cualquiera de los dos casos el efecto es el mismo para el mecanismo de *multi-homing*, y por tanto, es necesario que el SCTP establezca un mecanismo de monitorización de las direcciones de la asociación. Este mecanismo recibe el nombre de *heartbeat mechanism*.

El mecanismo de *heartbeat* consiste en enviar PDU con un *chunk type* igual a HEARTBEAT hacia aquellas direcciones que han permanecido inactivas* durante un cierto período. Una vez transmitido un HEARTBEAT, el transmisor establece un tiempo máximo para recibir una notificación desde el otro extremo de la asociación (HEARTBEAT ACK). En caso de no recibir la notificación de recepción, el SCTP considera que la dirección en cuestión es inaccesible.

*** Una dirección se considera inactiva si no se reciben o bien datos o bien un *heartbeat* durante un período determinado.**

Pese a considerar que una dirección es inaccesible, SCTP continúa enviando periódicamente *chunks* HEARTBEAT para detectar si, en algún momento, la dirección vuelve a ser accesible.

2.1.4. Bajo retardo

La señalización SS7 es muy sensible al retardo. Es por este motivo que el protocolo de transporte definido por SIGTRAN, el SCTP, debe garantizar retardos bajos. El protocolo TCP presenta un problema denominado *head-of-line blocking***. Cuando un segmento TCP no se recibe correctamente, la recepción de cualquier segmento transmitido posteriormente no se puede notificar hasta que el segmento perdido/erróneo no se retransmite. Aunque esta situación no es un problema para muchas aplicaciones, en el caso de la señalización puede causar retardos superiores a sus restrictivos límites de retardo.

**** El término inglés *head-of-line blocking* podría ser traducido como *bloqueo del primero de la fila*.**

SCTP hace frente a este problema definiendo *streams* (o flujos). El número de flujos es determinado durante el establecimiento de la asociación y se mantiene durante el tiempo que se prolongue en esta asociación. A cada uno de

los flujos se le asigna un identificador (*Stream Identifier*, SI), y por tanto, todos los *chunks* transmitidos contienen el identificador de flujo y un número de secuencia de flujo (*Stream Sequence Number*, SSN), y esto hace que la entrega ordenada de *chunks* se haga a nivel de flujo. Por ejemplo, cuando un *chunk* es recibido con errores, solo la información que pertenece al mismo flujo que la información perdida/errónea sufre un retardo adicional por el hecho de tener que esperar su retransmisión. Así pues, SCTP solo garantiza la entrega ordenada dentro de cada flujo, pero no entre flujos.

Este mecanismo se denomina multi-streaming y reduce el retardo global de la asociación, ya que el retardo debido a las retransmisiones solo afecta a aquellos flujos en los que ha habido errores, pero no toda la información, como sucede en las conexiones TCP.

2.1.5. Resumen de las características de la SCTP

Las principales características del protocolo SCTP, algunas de las cuales son comunes a TCP, son las siguientes:

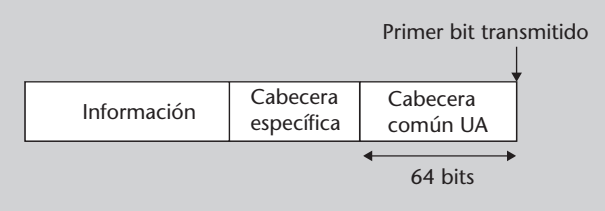
- **Conserva la estructura de los mensajes de capas superiores**. SCTP permite la segmentación de los datos de capas superiores y, por tanto, un mensaje se puede segmentar en varias PDU o bien varios mensajes se pueden incluir en diferentes *chunks* de una única PDU.
- **Evita el problema del llamado bloqueo *head-of-line***. Esta es una característica básica para que el protocolo SCTP pueda ser el protocolo de transporte para la señalización SS7.
- **Diferentes modos de entrega de la información**. Aunque en el caso de la señalización estamos interesados en una entrega secuencial y ordenada de la información (como TCP), el protocolo SCTP también puede implementar modos similares a UDP.
- **Multi-homing**. Tal como se ha explicado, SCTP permite redireccionar las PDU hacia direcciones IP alternativas cuando la dirección principal falla.
- **Control de flujo como el de TCP**. El SCTP implementa mecanismos de control de flujo basados en TCP, como por ejemplo *slow-start* y *congestion avoidance*.
- **Notificación de recepción correcta**. Tal como hace TCP, se notifica la recepción correcta de las PDU mediante un *chunk* llamado *Selective acknowledgement* (SACK).
- **Mecanismo *heartbeat***. Gracias a este mecanismo, el protocolo SCTP es capaz de detectar el fallo de una dirección IP.
- **Ofrece protección contra el ataque DoS**. Gracias al *cookie mechanism* es posible evitar este tipo de ataques.

2.2. *User Adaptation Layers* (UA)

Las capas de adaptación de usuario tienen, en el contexto de SIGTRAN, la función de adaptar la señalización proveniente de alguno de los niveles de la SS7 y la transmisión de paquetes en una red IP. Una vez definido el protocolo de transporte que cumple con los requisitos exigibles en términos de fiabilidad, redundancia y de retardo, las diversas UA acomodan la información en formato SS7 al formato impuesto por el SCTP.

La información SS7 es encapsulada/desencapsulada en el mensaje propio de cada una de las capas de adaptación específicas. El formato de los mensajes de estas capas de adaptación sigue un mismo formato genérico (figura 8). Así, como se puede ver, en primer lugar hay una cabecera común, seguida de una cabecera particular para cada capa de adaptación, y finalmente la información.

Figura 8. Formato general de un mensaje de las capas UA



En cuanto a la cabecera común, esta consta de un campo de versión del protocolo, 8 bits no utilizados, el campo de la clase del mensaje, el tipo de mensaje, y, finalmente, la longitud de todo el mensaje.

- **Versión del protocolo**. Especifica la versión del protocolo de la capa de adaptación que ejecuta el nodo que genera el mensaje (8 bits).
- **Campo no utilizado**. Tiene una longitud de 8 bits y se reserva para posibles extensiones de los protocolos en el futuro (8 bits).
- **Clase**. Este campo especifica cuál de los protocolos de adaptación de usuario (M2UA, M2PA, M3UA, etc.) ha generado el mensaje (8 bits).
- **Tipo de mensaje**. Dentro de cada clase de mensaje, la IETF define un conjunto de tipos de mensaje, como por ejemplo los mensajes de establecimiento, de finalización, de datos, etc. (8 bits).
- **Longitud del mensaje**. Contiene la longitud total del mensaje (datos más cabeceras) expresada en octetos/bytes (32 bits).

Tal como se verá a continuación, los mensajes de tamaño variable empleados en todas las capas de adaptación tienen la misma estructura: una etiqueta de 16 bits, un campo de longitud de 16 bits, y para terminar el valor del parámetro, que tiene una longitud variable.

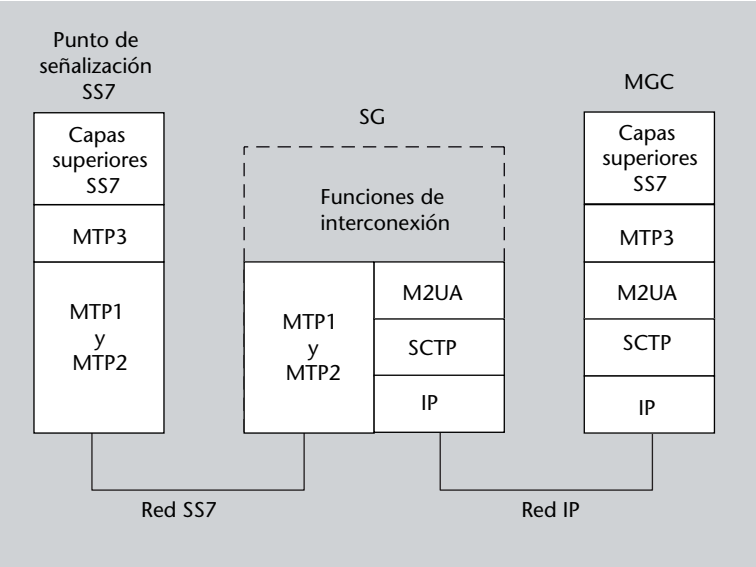
2.2.1. *MTP2-User Adaptation* (M2UA)

M2UA

La capa M2UA está descrita en el RFC 3331 de la IETF.

Tal como se ha explicado hasta ahora, la interconexión de las redes de conmutación de circuitos y de las redes IP requiere, en términos de señalización, la inclusión de un nodo intermedio llamado *Service Gateway* (SG). Este nodo, el SG, tiene, por un lado, una conexión a la red SS7 a través de las capas MTP1 y MTP2. Por otro, una conexión a la red IP a través de SCTP/IP. El esquema de la figura 9 muestra la situación descrita en el RFC 3331 para el uso de la M2UA.

Figura 9. Arquitectura de referencia para M2UA



Fuente: RFC 3331. SS7 MTP2 User Adaptation Layer.

En esta arquitectura hay ciertos de aspectos que tienen especial relevancia:

- El código de punto, dirección mediante la cual se encaminan los mensajes a través de la red SS7, es necesario para todos aquellos nodos que implementan la capa MTP3. Como se puede ver en la figura, la SG no implementa la capa MTP3 y, por tanto, no tiene un código de punto (sí tiene una dirección IP en la red IP). En cuanto a la MGC, en cambio, sí que dispone tanto de dirección IP como de código de punto, ya que establece una asociación con el SG mediante SCTP/IP y una conexión con el punto de señalización SS7 en la capa MTP3.
- Para la capa MTP3 del MGC y del punto de señalización SS7, la SG es transparente.

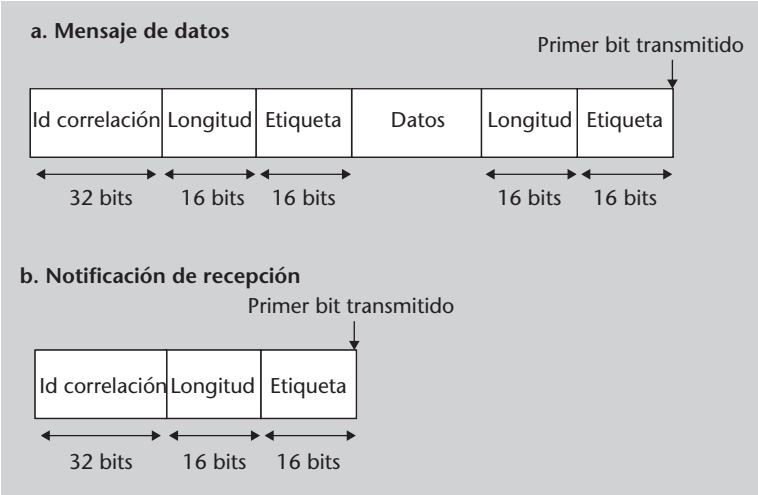
Los mensajes utilizados por el protocolo M2UA tienen el formato descrito en la figura 8, y por lo tanto contienen una cabecera genérica, una cabecera específica de la M2UA y los datos para transmitir. En cuanto a la cabecera específica, consta de tres campos: la etiqueta o *tag* (16 bits), la longitud (16 bits), y el identificador de interfaz.

- **Etiqueta**. Indica qué información transporta el mensaje, es decir, si contiene datos, una notificación (ACK), una petición de establecimiento, una solicitud, etc.

- **Longitud**. Para preservar la estructura general de los parámetros, la cabecera incluye el campo de longitud. Sin embargo, no desarrolla ninguna función específica y su valor siempre es igual a 8.
- **Identificador de interfaz**. Las redes IP pueden tener capas física y de enlace diversas. Este campo describe cuál es el interfaz al cual está conectado el nodo.

Los mensajes, además de las cabeceras común y específica, incluyen el campo de información. En el caso del M2UA, el campo de información puede contener datos, notificaciones de recepción de datos, peticiones y confirmaciones de establecimiento, peticiones y confirmaciones de liberación, notificaciones de estado y de congestión, etc. En general los mensajes transportados en el campo de información mantienen la estructura de etiqueta, longitud y valor. Aún así, conviene destacar el formato y el funcionamiento del intercambio de datos (figura 10).

Figura 10. Formato de la parte de información de los mensajes utilizados por el M2UA en el intercambio de datos



Cuando las capas M2UA de dos nodos se intercambian datos, es necesario que haya una confirmación de recepción. Para hacerlo, la estructura del mensaje de datos es la que muestra la figura 10 (a). El primero de los parámetros es el que contiene los datos. Así, el primer campo de etiqueta y de longitud permite que la capa M2UA reconozca los datos y su longitud. Estos datos, sin embargo, van acompañados de un segundo parámetro (también con la estructura etiqueta-longitud*-valor): el identificador de correlación. Este parámetro permite que la notificación de recepción del mensaje se lleve a cabo. El receptor, al recibir el mensaje de datos, responde con un mensaje de notificación como el de la figura 10 (b), con el mismo identificador de correlación.

*** El campo de longitud asociado al identificador de correlación toma siempre un valor igual a 8 porque la longitud del campo es conocida a priori.**

Es importante hacer notar que la recepción correcta de los mensajes de datos del M2UA debe ser notificada para que, en caso de no hacerlo, para el enlace SS7 equivaldría a la falta de notificación de las MSU transmitidas.

2.2.2. *MTP2-user Peer-to-peer Adaptation* (M2PA)

El objetivo principal de la capa M2PA es conseguir una verdadera convergencia entre las redes de señalización SS7 y las redes IP. Para hacerlo es necesario que los puntos de señalización, sea cual sea la red a la que se encuentran conectados, puedan comunicarse entre ellos de manera transparente.

M2PA

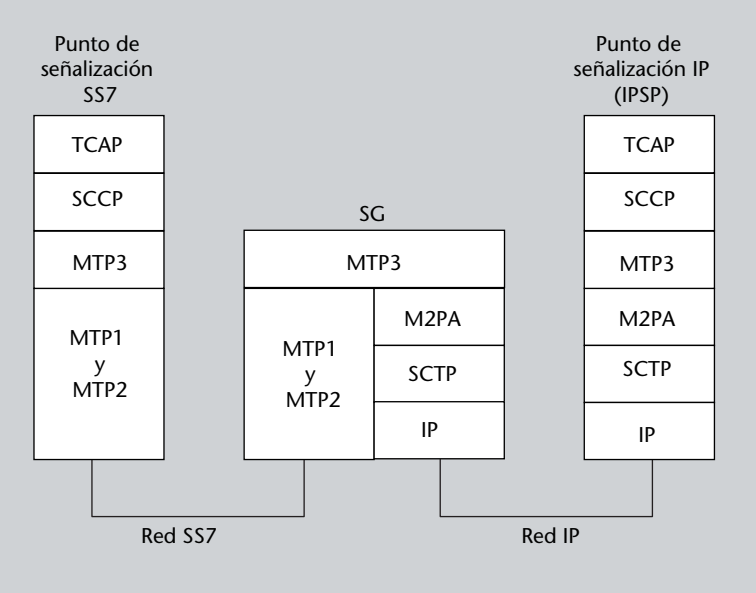
La capa M2PA está descrita en el RFC 4165 de la IETF.

El RFC 4165, que define esta capa de adaptación de usuario, propone dos arquitecturas en las que se puede utilizar el M2PA (figura 11 y figura 12). La primera de las arquitecturas es la conexión de dos puntos de señalización, uno conectado a la red SS7 y el otro conectado a la red IP (figura 11). De la misma manera que sucede en el M2UA, el intercambio de señalización se realiza a través de un SG, pero esta vez el SG implementa la capa MTP3. Por tanto, en esta arquitectura el SG sí tiene una dirección IP y un Código de Punto SS7.

IPSP

Los puntos de señalización IP (denominados como *IP Signalling Point*, IPSP) son puntos de señalización conectados a la red IP que implementan las capas altas de la SS7, a partir de la MTP3.

Figura 11. Primera arquitectura de referencia para M2PA

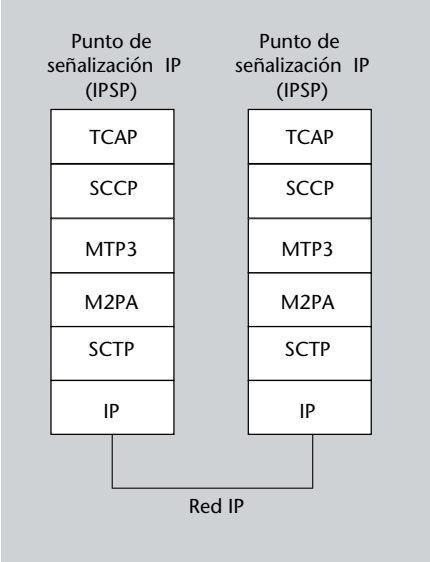


Fuente: RFC 4165. SS7 MTP2 User Peer-to-Peer Adaptation Layer.

Desde el punto de vista del punto de señalización SS7, el SG es visto como un punto de señalización STP. Gracias a la M2PA y la primera arquitectura presentada, es posible que los puntos de señalización de la red SS7 puedan acceder a nodos de la red IP (como, por ejemplo, bases de datos) y que los nodos de la red IP puedan utilizar los servicios ofrecidos por la red SS7.

La segunda arquitectura especificada en el RFC 4165 es la que se muestra en la figura 12. Se trata, tal como se puede observar, de una situación de convergencia de las dos redes, en las que los puntos de señalización SS7 pueden estar conectados directamente a la red IP.

Figura 12. Segunda arquitectura de referencia para M2PA



Fuente: RFC 4165. SS7 MTP2 User Peer-to-Peer Adaptation Layer.

Las arquitecturas propuestas para la M2PA permiten observar que en todos los puntos de señalización hay, por encima de la M2PA, la capa MTP3. Por tanto, la M2PA debe ser vista por el MTP3 de la misma manera que sería vista la MTP2 (el intercambio de mensajes entre MTP3 y M2PA debe ser el mismo que el intercambio de mensajes entre MTP3 y MTP2). Si recordamos el funcionamiento de la capa MTP del estándar SS7, MTP2 transmite/recibe tres tipos de unidades de señalización: las MSU, las LSSU y las FISU. De los tres tipos de unidades de señalización solo las MSU son generadas por la capa MTP3. Así pues, las capas MTP3 y M2PA intercambian MSU.

Los otros dos tipos de unidades de señalización son intercambiados entre las capas MTP2 de los dos puntos de señalización pero no son transferidos hacia la capa MTP3. En cuanto a las LSSU, el estándar SS7 las utiliza para intercambiar información del estado del enlace. Por tanto, a pesar de que la capa M2PA no transmite/recibe LSSU, sí que define mensajes con una función análoga (intercambio de información de estado).

El caso de las FISU es diferente. En la capa MTP2 estas unidades de señalización son transmitidas siempre que no hay MSU y/o LSSU disponibles. Esta función, que tiene sentido en la red SS7, no tiene sentido en una red IP. Por tanto, la capa M2PA no define ningún mensaje que haga esta función.

Las capas M2UA y M2PA pueden parecer similares, pero presentan diferencias importantes. Las semejanzas principales entre ambas capas son las siguientes:

Semejanzas:

- Ambas capas transportan mensajes con datos del MTP3.
- Ambas capas tienen la capa MTP3 por encima y el SCTP por debajo.

Diferencias:

- Cuando hay la capa M2PA, la conexión establecida entre el SG y el punto de señalización IP es una conexión SS7*. En cambio, con la capa M2UA no se establece una conexión SS7 entre el SG y el MGC. De hecho, en este último caso, el SG es transparente a la conexión SS7.
- El SG con el M2PA también implementa el MTP3, mientras que con la M2UA no. En este caso, el SG tiene asignada un código de punto en el primer caso y no lo tiene en el segundo caso.
- La capa M2PA no define funciones de gestión de la red, ya que confía estas funciones en la capa MTP3. El M2UA, en cambio, define funciones de gestión*.
- La capa M2UA siempre tiene la capa MTP3 por encima, mientras que no sucede lo mismo en el caso del M2UA*.

*** Recordemos que un nodo con el M2UA, como por ejemplo un SG, no necesariamente tiene la capa MTP3 por encima.**

El formato de los mensajes M2PA sigue la estructura de todos los mensajes de las capas UA: cabecera común, cabecera específica para M2PA, y campo de información. Ya se ha explicado que el M2PA lleva a cabo las funciones de la MTP2 y que por encima siempre tiene la MTP3. Por tanto, la cabecera específica está muy relacionada con las unidades de señalización que en el estándar SS7 transmite/recibe el MTP2. En concreto, la cabecera específica consta de los siguientes campos (ordenados):

- **Vacío**. Campo de 8 bits de longitud reservado para futuros usos.
- ***Backward Sequence Number* (BSN)**. Campo que contiene el BSN, que, en el estándar SS7, incorpora la cabecera de las unidades de señalización (24 bits). El valor de este campo se extrae de la cabecera de la unidad de señalización.
- **Vacío**. Campo de 8 bits de longitud reservado para usos futuros.
- ***Forward Sequence Number* (FSN)**. Campo que contiene el FSN que, en el estándar SS7, incorpora la cabecera de las unidades de señalización (24 bits). El valor de este campo se extrae de la cabecera de la unidad de señalización.

Los mensajes del M2PA, tal como ya se ha apuntado anteriormente, pueden ser de dos tipos: de datos o de estado. El campo de información de los mensajes de datos es, directamente, los mensajes generados por el MTP3. Se trata simplemente de encapsular la información generada en el MTP3 tal como lo haría la capa MTP2 pero en un formato diferente.

El protocolo M2PA define dos tipos de mensajes relacionados con el estado del enlace: el mensaje de prueba de estado (*Link Status Proving*) y el mensaje de estado (*Link Status*). La función que tienen estos mensajes es análoga a la que hacen las LSSU, es decir, comprobar el estado del enlace y comunicar el resultado. En ambos casos, el campo de información contiene exclusivamente

el estado del enlace, que puede tomar valores como *Alignment*, *Proving Normal*, *Proving Emergency*, *Ready*, *Processor outage*, *Processor Recovered*, *Busy*, *Busy Ended* o *Out of Service*.

2.2.3. *MTP3-User Adaptation* (M3UA)

La capa M3UA es análoga a la M2UA, pero en este caso da servicio a los usuarios MTP3, es decir, tiene la función de hacer la adaptación entre la SCCP y el protocolo de transporte de la red IP, el SCTP.

M3UA

La capa M3UA está descrita en el RFC 4666 de la IETF.

El protocolo M3UA está definido para utilizarlo en tres situaciones diferentes:

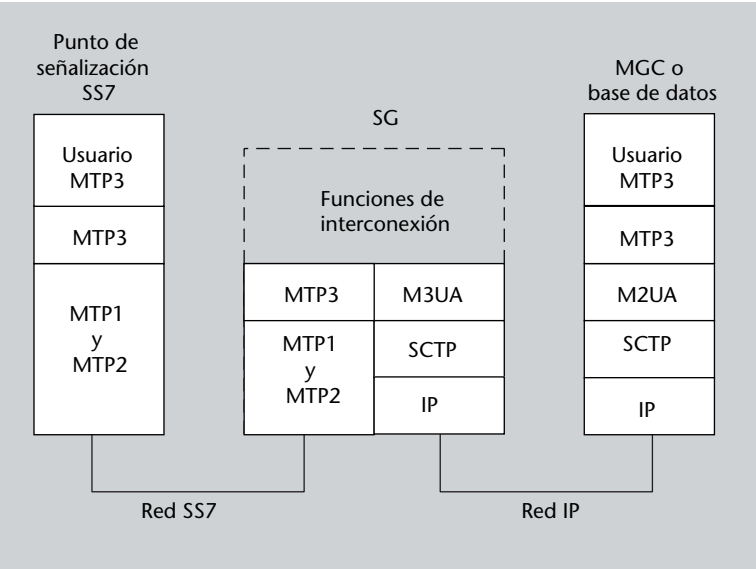
1) Entre un SG y el MGC para comunicar los usuarios MTP3 de un punto de señalización de la red SS7 y de un MGC de la red IP.

2) Permitir el acceso de un punto de señalización SS7 en una base de datos (a través de un SG) conectada a una red IP.

3) Entre dos aplicaciones IP, cuyos nodos se encuentran conectados a la red IP.

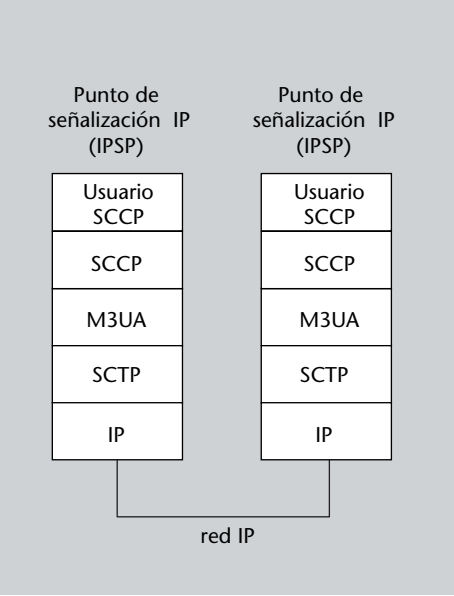
Estas tres situaciones se muestran en las figuras 13, 14 y 15. Las tres situaciones son análogas a las situaciones mostradas por los protocolos M2UA y M2PA, por tanto, no las describiremos en detalle. La más interesante de las tres arquitecturas propuestas en el RFC 4666, sin embargo, es la que muestra la figura 15. Las arquitecturas de la figura 13 y de la figura 15 son muy parecidas y solo se diferencian en el SG. En la primera (figura 13), el SG no tiene ninguna instancia de las capas superiores a MTP3, mientras que en la segunda (figura 15) hay una instancia SCCP. Esta última situación, que es muy concreta, ha sido definida porque el SG puede llevar a cabo la traducción de título global que define el SS7 (*Global Title Translation*, GTT).

Figura 13. Primera arquitectura de referencia para M3UA



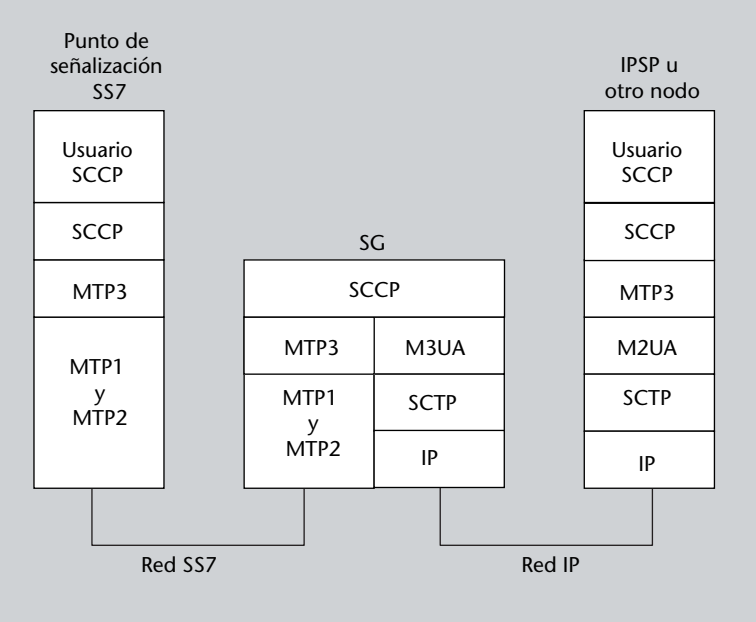
Fuente: RFC 4666. SS7 MTP3 User Adaptation Layer.

Figura 14. Segunda arquitectura de referencia para M3UA



Fuente: RFC 4666. SS7 MTP3 User Adaptation Layer.

Figura 15. Tercera arquitectura de referencia para M3UA



Fuente: RFC 4666. SS7 MTP3 User Adaptation Layer.

En la red SS7, la capa MTP3 tiene la responsabilidad de direccionar los mensajes de señalización en función del estado de los enlaces y del estado de los puntos de señalización distantes. En el contexto de SIGTRAN, la capa M3UA debe llevar a cabo las mismas funciones. Este protocolo por un lado define funciones de gestión nativas (independientes de la capa MTP3) y por el otro define funciones que permiten adaptar las funciones desarrolladas por el MTP3 a la red IP.

Fijémonos en la figura 14. En esta situación las funciones de gestión del M3UA, como por ejemplo notificar errores o direccionar los mensajes, no están relacionadas con la capa MTP3. En escenarios como este M3UA ejecuta funciones

de gestión definidas específicamente para este protocolo. Si, en cambio, observamos la figura 15, las funciones de gestión del M3UA adaptan las funciones definidas para el protocolo MTP3 de la ITU-T.

Los tipos de mensajes del M3UA pueden ser de varias clases, pero en esencia podemos decir que existen los mensajes de transferencia y los mensajes de gestión (*SS7 Signalling Network Management*, *ASP state management*, etc). La estructura de los mensajes sigue la que hemos visto para las otras capas de adaptación de usuario pero con una diferencia: **no hay una cabecera específica**. Así, el formato de los mensajes consta de la cabecera común y del campo de información (un conjunto de parámetros con la estructura etiqueta-longitud-valor).

En cuanto a los mensajes de transferencia, los parámetros que incluye son los siguientes:

- **Apariencia de la red**. Este parámetro incluye el contexto de la red SS7, la información sobre el usuario MTP3 y sobre el protocolo MTP3, así como otras informaciones de interés (32 bits).

- **Contexto de direccionamiento**. Describe el contexto de direccionamiento y las claves de direccionamiento de los mensajes (32 bits).

- **Datos del protocolo**. El campo de datos del protocolo tiene una longitud variable, y contiene el mensaje original de la capa MTP3. Por tanto, dentro de este campo se pueden encontrar los campos que contiene habitualmente un mensaje MTP3, como las direcciones de los puntos de señalización de origen y de destino (OPC y DPC), el *Service Indicator* (SI), el *Signalling Link Selector* (SLS), el *Network Indicator* (NI), el *Message Priority* (MP) y los datos.

- **Identificador de correlación**. Campo que identifica el MSU transportado en el campo de datos del protocolo (32 bits).

El resto de mensajes del M3UA son similares pero con algunos cambios. La estructura es la misma que la de los mensajes de transferencia, pero sin el campo de datos. En lugar del campo de datos incluyen una lista de puntos de señalización afectados por la situación que notifica el mensaje (congestión, accesibilidad o inaccesibilidad, etc.) y, finalmente, un campo de información adicional.

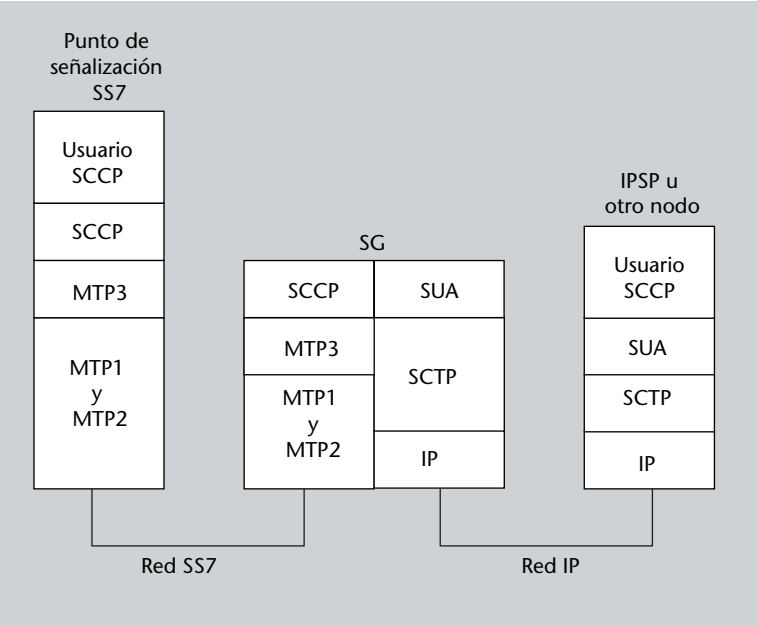
2.2.4. *SCCP User Adaptation* (SUA)

El último de los protocolos de adaptación de usuario es el SUA. Se trata de una capa análoga a la M2UA y el M3UA. La figura 16 y la figura 17 muestran las dos arquitecturas propuestas por la IETF. Ambas son muy similares a las que hemos visto hasta ahora.

SUA

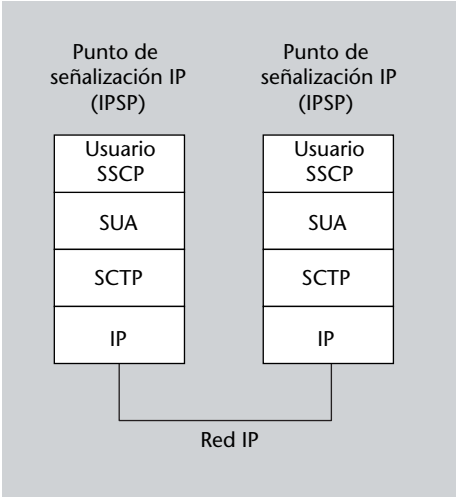
La capa SUA está descrita en el RFC 3868 de la IETF.

Figura 16. Primera arquitectura de referencia para SUA



Fuente: RFC 3868. SS7 SCCP User Adaptation Layer.

Figura 17. Segunda arquitectura de referencia para SUA



Fuente: RFC 3868. SS7 SCCP User Adaptation Layer.

Como la definición es muy similar a la de estas dos capas, no profundizaremos en los campos de cada mensaje. Aún así, cabe destacar que, como sucede en el M3UA, los mensajes incluyen la cabecera común pero no hay cabecera específica para este protocolo.

Si recordáis, la capa SCCP permite transferencia de datos no orientada a conexión (clases 0 y 1) y transferencia de datos orientada a conexión (clases 2 y 3). Es por este motivo que el protocolo SUA debe soportar la transferencia de datos para cualquiera de las cuatro clases.

3. El protocolo *Session Initiation Protocol* (SIP)

Hasta ahora hemos visto un conjunto de protocolos que tienen el objetivo de conseguir la interconexión de la red de señalización SS7 y la red IP. A partir de ahora nos centraremos en un protocolo de señalización definido para la red IP. Hay dos protocolos que por sus características son los más extendidos: el H.323 (ITU-T) y el *Session Initiation Protocol* (SIP).

El protocolo H.323 fue el primer estándar que daba solución a la voz sobre IP de una manera adecuada. Se trata de un protocolo que desarrolla la función de paraguas agrupando otras fuentes de estandarización. La Recomendación de la ITU-T que lo define hace referencia a varios estándares, tales como los protocolos para servicios en tiempo real RTP y RTCP, o las Recomendaciones H.225, Q.931 y H.245. A pesar de ser un estándar completo que permite ofrecer mejores funcionalidades, interoperabilidad e interconexión que, por ejemplo, el protocolo SIP, a menudo su complejidad ha limitado su desarrollo.

La otra opción, el SIP, es un protocolo simple y flexible que, pese a no ofrecer las mismas prestaciones que el H.323, permite establecer señalización entre nodos de la red IP de una manera sencilla y con un bajo coste. Además, gracias a 3GPP y la introducción de la arquitectura IMS en las redes troncales, el protocolo SIP ha adquirido más importancia. La arquitectura IMS es una propuesta para las futuras redes IP, y su definición está basada en el protocolo de señalización SIP.

Es por este motivo y por su aceptación que en este módulo nos centraremos en el protocolo SIP y dejaremos de lado el H.323.

El *Session Initiation Protocol* (SIP) es un protocolo de la capa de aplicación que permite el establecimiento, modificación y finalización de una sesión en una red IP de paquetes.

Estandarización del SIP

A mediados de la década de los 90, investigadores de la Universidad de Columbia establecieron por primera vez las bases del protocolo SIP. La IETF se hizo cargo de su estandarización y creó el RFC 2543, que posteriormente fue sustituido por el RFC 3261.

Se trata de un protocolo creado y estandarizado por la IETF y que, actualmente, después de una evolución histórica, ha quedado fijado en el RFC 3261, si bien es cierto que hay varias RFC que amplían algunos aspectos del protocolo.

El SIP no es por sí mismo un protocolo que transporte información multimedia, sino que se trata de un protocolo que establece y gestiona las sesiones entre dos usuarios. Una vez establecida la sesión, la información multimedia es transportada mediante otros protocolos de transporte, como por ejemplo el popular *Real-Time Transport Protocol* (RTP). Es precisamente esta flexibilidad –establecer, mantener y liberar sesiones genéricas sin imponer cómo se trans-

porta la información– la que lo ha convertido en un protocolo extendido y popular para servicios como la VoIP, el *streaming* de vídeo o la mensajería instantánea en redes de paquetes.

En general se podría decir que el protocolo SIP presenta las siguientes características:

- Es capaz de determinar la **localización del usuario** que participa en la comunicación a partir de parámetros como, por ejemplo, la dirección IP (o un número de teléfono).
- Permite que el usuario establezca, de manera previa al establecimiento de una sesión, la accesibilidad y la **disponibilidad*** de un determinado usuario.

*** La disponibilidad de un usuario la puede establecer el mismo usuario.**

- **Establece una sesión** entre dos usuarios. En cada sesión se pueden establecer los parámetros que regirán esa sesión.
- **Gestiona la sesión establecida**, y permite modificar los parámetros de la sesión durante su transcurso o finalizarla.

Gracias a estas características, que lo convierten en un estándar muy abierto y flexible para gestionar las sesiones de todo tipo de servicios, el SIP se ha convertido en uno de los más populares –si no el más popular– de los protocolos de señalización para redes de paquetes. De hecho, su flexibilidad prevé, por ejemplo, que el protocolo de transporte utilizado para transportar los mensajes de SIP sea tanto TCP como UDP, los dos protocolos de transporte más habituales en redes IP.

Fijémonos que hasta ahora, en este módulo, nos hemos centrado en los protocolos que conforman SIGTRAN. SIGTRAN es un conjunto de protocolos con el objetivo de alcanzar interoperabilidad entre las nuevas redes IP y las antiguas, pero muy extendidas y utilizadas, redes conmutadas de telefonía (y RDSI). Ahora, en cambio, presentamos un protocolo pensado explícitamente para usuarios conectados a una red IP. Aunque dos usuarios pueden establecer y gestionar una sesión SIP sin la necesidad de nodos adicionales, a medida que la dimensión y la complejidad de la red crece el protocolo SIP incluye nodos adicionales que intervienen de diferente manera en la sesión.

Para definir cuáles son estos nodos y, al mismo tiempo, establecer la nomenclatura utilizada en el contexto del SIP, a continuación hacemos una breve descripción:

- **El usuario (*User Agent*, UA)**. Se trata del usuario final de una sesión SIP, tanto el usuario que inicia la sesión como el usuario que la recibe. Cuando se habla de los usuarios se diferencia entre el que ha iniciado el establecimiento de la sesión, que recibe la denominación de *client*, y el que recibe la petición de establecimiento, que es denominado *server*.

Cliente y servidor

La denominación completa del cliente y el servidor es *User Agent Client* (UAC) y *User Agent Server* (UAS).

- **El servidor *proxy***. Este nodo apoya, en caso de ser requerido mediante el envío de una petición, a la determinación del direccionamiento entre el UAC y el UAS. De la misma manera, puede desarrollar tareas de autenticación de los usuarios.

- **Servidor de redireccionamiento (*Redirect Server*)**. Es un nodo con la función de responder a la petición de un usuario o de otro servidor entregando un conjunto de direcciones. Como indica su nombre, desarrolla sus funciones en el contexto del direccionamiento de los mensajes SIP, pero su única participación es esta: proveer al nodo que ha generado la petición con las direcciones requeridas.

- **Servidor de registro (*Register Server*)**. Por motivos de movilidad que serán explicados más adelante, es necesario que los usuarios puedan registrar su posición en este servidor.

- **Servidor de localización (*Location Server*)**. La movilidad de los usuarios es uno de los retos al que se enfrenta cualquier protocolo de señalización de redes IP. Este servidor es el que hace posible, gracias a su interacción con el servidor de registro, conocer la localización de los diferentes usuarios.

Todos estos nodos descritos pueden intervenir en el establecimiento, gestión y finalización de una sesión SIP, pero no son necesarios si los dos usuarios (cliente y servidor) son capaces de comunicarse directamente.

3.1. Diálogos, transacciones y mensajes

El protocolo SIP se basa en mensajes, transacciones y diálogos. Los mensajes pueden ser de dos tipos: Peticiones (en inglés, *Request*) y Respuestas (*Response*). Una transacción es el conjunto de mensajes que incluyen una petición y una o varias respuestas. El diálogo, en cambio, es el conjunto de transacciones que intercambian cliente y servidor entre el establecimiento y liberación de la sesión (ambas transacciones –establecimiento y liberación– incluidas).

Bloques básicos de SIP

Mensajes, transacciones y diálogos son los bloques básicos sobre los que se fundamenta SIP.

Para analizar el funcionamiento de los diálogos y cómo se puede establecer, gestionar y liberar una sesión, analizaremos el formato de los mensajes.

3.1.1. Formato de los mensajes

El protocolo SIP se basa en conceptos muy similares al protocolo HTTP, y el formato del mensaje es un buen ejemplo de ello. Los mensajes SIP son un conjunto de líneas de texto separadas entre ellas por un salto de línea. Aunque, tal como se ha hecho notar, hay dos tipos de mensajes (peticiones y respuestas), el formato de los mensajes es independiente del tipo de mensaje y presenta un único formato:

1) Línea inicial.

2) Cabeceras.

3) Línea en blanco.

4) Cuerpo del mensaje.

La **línea inicial** del mensaje presenta una estructura algo diferente en función de si se trata de una petición o de una respuesta. La forma general es la siguiente:

Method URI-Request SIP-Version **(Petición)**
SIP-Version Status-Code Reason-Phrase **(Respuesta)**

En una petición SIP el método (*Method*) es el parámetro que identifica el motivo de la petición. El número de métodos de SIP es reducido y el RFC 3261 define seis: INVITE, ACK, OPTIONS, BYE, CANCEL y REGISTER. Posteriormente, el número de métodos se ha ampliado en diferentes RFC incluyendo los métodos INFO (RFC 2976), MESSAGE (RFC 3428), NOTIFY (RFC 3265), PRACK (RFC 3263), REFER (RFC 3515), SUBSCRIBE (RFC 3265) y UPDATE (RFC 3311). En la tabla 1 se definen los métodos más importantes incluidos en el RFC 3261.

Tabla 1. Métodos de los mensajes SIP

Mètode	Descripción
INVITE	Inicia la sesión
ACK	Notifica que la sesión se ha establecido correctamente
OPTIONS	Permite solicitar información de las capacidades del destino
BYE	Finaliza la sesión
CANCEL	Cancela una sesión abierta
REGISTER	Envía la información para que el usuario sea registrado en el servidor de registro

El parámetro *URI-Request* hace referencia a la dirección del nodo al que se envía la petición. Las direcciones SIP pueden tener diferentes formatos. Lo más habitual es que tenga la forma **sip:usuario@dominio**, aunque también puede incluir parámetros como una contraseña, el puerto, etc. Aparte de este formato, la dirección URI (*Uniform Resource Identifier*) también puede incluir el número de teléfono en vez de la dirección. En este caso, la dirección presentaría una forma **sip:+teléfono@dominio;user=phone**.

En cualquier caso, el URI puede ir precedido de un descriptor entre comillas y presentar un aspecto como "**Descriptor**"**<sip:usuario@dominio>**.

Ejemplo

Imaginemos un usuario conectado al dominio *uoc.edu*, con nombre de usuario *consultor*. La dirección URI del usuario sería sip:consultor@uoc.edu

Si, para este mismo usuario, el puerto utilizado por la aplicación basada en el protocolo SIP utilizara el puerto de TCP 5060, entonces la dirección se podría escribir de la siguiente manera: sip:consultor@uoc.edu: 5060

En caso de que, para acceder a la red, el usuario que genera la petición deba proporcionar una contraseña, el URI podría añadirla: sip:consultor:contraseña@uoc.edu:5060

Sí, en cambio, el usuario tiene un número de teléfono +34933263535, entonces el URI podría tener el siguiente aspecto: sip:+34-933263535@uoc.edu;user=phone

En cuanto al parámetro *SIP-Version*, tiene el único objetivo de notificar al destinatario del mensaje cuál es la versión del protocolo SIP que se ejecuta en el origen.

Finalmente, en el mensaje de petición aparecen los parámetros *Status-Code* y *Reason-Phrase*. Los dos parámetros están muy relacionados entre sí, ya que, pese a que el primero es un campo numérico y el segundo no, ambos notifican el resultado de la petición. Cada valor del *Status-Code* se corresponde con un valor de *Reason-Phrase*. El código de estado es un entero de 3 dígitos. El primero de los tres dígitos designa la clase de respuesta, y los dos últimos dígitos determinan la respuesta concreta. Hay seis clases de respuestas:

La descripción de todas las respuestas posibles se puede encontrar en la sección 21 del RFC 3261.

- **Provisional**: tienen un *Code-Status* con el primer dígito igual a 1 (1xx). Las respuestas de esta familia indican al cliente que el servidor está llevando a cabo alguna tarea y que tardará más de 200 ms. Para evitar retransmisiones innecesarias, las respuestas de esta clase alertan al cliente que la respuesta tardará, pero que está siendo servida. Algunas de las respuestas más importantes que pertenecen a esta clase son *Code-Status*=100 y *Reason-Phrase*=*Trying* (la petición ha sido recibida y se está procesando) o *Code-Status*=180 y *Reason-Phrase*=*Ringing* (el servidor ha sido alertado de la petición entrante).
- **Éxito**. Consta de una única respuesta para indicar que la petición ha sido satisfactoria. La respuesta es *Code-Status*=200 y *Reason-Phrase*=*OK*.
- **Redireccionamiento**. Tienen un *Code-Status* con el primer dígito igual a 3 (3xx). Las respuestas de esta clase informan sobre la nueva localización de un usuario o de los nuevos servicios disponibles.
- **Petición errónea**. Tienen un *Code-Status* con el primer dígito igual a 4 (4xx). Indican que la petición recibida en el servidor es errónea y, por tanto, no puede ser reenviada sin modificaciones.
- **Error del servidor**. Tienen un *Code-Status* con el primer dígito igual a 5 (5xx). Las respuestas que pertenecen a esta clase son empleadas por el servidor cuando el error es del servidor, y no de la petición recibida.
- **Fallo global**. Tienen un *Code-Status* con el primer dígito igual a 6 (6xx). Estas respuestas ofrecen información definitiva sobre un usuario, no sobre la petición en concreto, como por ejemplo la declinación de sesión por parte del servidor, la inexistencia del usuario, etc.

El siguiente campo del mensaje SIP son las **cabeceras**. Hay dos clases de cabeceras: las cabeceras generales y las cabeceras específicas. En cuanto a las generales, hay cinco cabeceras que son obligatorias tanto para las peticiones como para las respuestas (**Call-ID**, **To**, **From**, **Cseq** y **Via**). Además, las peticio-

nes incluyen una sexta cabecera obligatoria (**Max-Forwards**) que, pese a no ser obligatoria en las respuestas, puede estar incluida opcionalmente.

- **Call-ID**. Esta cabecera identifica todos los mensajes involucrados en una sesión. Antes de iniciar el establecimiento de una sesión, el UAC determina un valor de Call-ID que se mantiene hasta el final de la sesión. En caso de que en la sesión intervenga más de un cliente, todos ellos utilizan el mismo valor para el Call-ID. El Call-ID es una cadena de caracteres aleatorios, pero en algunas versiones antiguas del protocolo SIP presentaba un formato número@dominio.

- **To**. Se trata de la dirección de destino del mensaje. Para identificar el destino se utiliza el formato del URI, tal como se ha hecho en la línea inicial. Esta cabecera, no obstante, va acompañada de un *tag* o etiqueta. Esta etiqueta es útil para poder identificar, junto con las cabeceras Call-ID y From, la sesión. Esta etiqueta es obligatoria y en caso de no estar disponible se interpreta que es igual a 0. Por ejemplo, To: sip_consultor@uoc.edu; tag=381492.

  Tal como se ha explicado, la cabecera To determina el URI de destino. Esto, de hecho, solo es cierto para las peticiones (Request). Para las respuestas (Response) sucede lo contrario, y la cabecera To incluye el URI de origen de la respuesta. En definitiva, las respuestas utilizan el mismo valor para To que el utilizado en la petición correspondiente. Como se verá, esto también sucede con la cabecera From.

  Para las peticiones de REQUEST, la cabecera To no incluye el destino (ya incorporada a la línea inicial), sino que toma el valor de la dirección que debe ser registrada en el servidor de registro.

- **From**. Es una cabecera completamente análoga a la cabecera To, pero para la dirección de origen. Como sucede con la cabecera To, la cabecera From de una respuesta contiene el URI de origen de la petición correspondiente, no el del origen de la respuesta.

- **Cseq**. Esta cabecera toma un valor entero inicialmente aleatorio. Una vez establecido, y dentro de una sesión, cada vez que se finaliza una transacción exitosa, Cseq incrementa en una unidad. Dentro de cada transacción, tanto la petición como las respuestas utilizan el mismo valor. Además del número de secuencia, también se incluye el método. Por ejemplo, en una petición INVITE, la cabecera tomaría un valor, por ejemplo, igual a Cseq: 2837 INVITE.

- **Via**. Las peticiones SIP pueden pasar por varios servidores *proxy* entre el usuario cliente y el usuario servidor. Cada uno de los servidores que direccionan la petición incluye su dirección para que, en recepción, la respuesta pueda rehacer la ruta de la petición. El formato de cada cabecera Via es el siguiente: *Protocol/Version/Transport host; port; Via-parameter comment*

El *Protocol* toma el valor de SIP, la *Version* toma el valor que corresponda a la versión del protocolo, *Transport* hace referencia al protocolo de la capa de transporte utilizado, que habitualmente es UDP, TCP, TLS o SCTP, y el *host* y el *port* son la dirección del servidor que lo retransmite. Adicionalmente se pueden añadir más parámetros, pero solo hay uno obligatorio: el llamado *branch*. El valor de *branch* es igual al identificador de la transacción precedido por el valor z9hG4bk. Por ejemplo, si una petición es retransmitida por un servidor proxy.uoc.edu, a través del puerto 5060 y SIP utiliza el protocolo UDP, la cabecera introducida por este servidor podría tener la siguiente estructura: Via: SIP/2.0/UDP proxy.uoc.edu:5060; branch=z9hG4bk17321.

- **Max-Forwards**. El cliente, al enviar una petición, puede establecer un número máximo de saltos. En concreto, si la petición no ha llegado al servidor en un número máximo de saltos, se produce un error. Tal como ya se ha hecho notar, esta cabecera solo es obligatoria para las peticiones y no se incluye en la cabecera de las respuestas.

Las cabeceras específicas no son obligatorias y están definidas para poder transportar información adicional. Las más importantes son aquellas cabeceras que incluyen, por ejemplo, información de autenticación o de registro.

El **cuerpo del mensaje** contiene la información que intercambian los dos usuarios. Es precisamente en esta parte del mensaje en la que se incluye la información de negociación de los parámetros del servicio. Para esta negociación se utiliza el *Session Description Protocol* (SDP), definido en el RFC 4566.

3.2. La sesión SIP

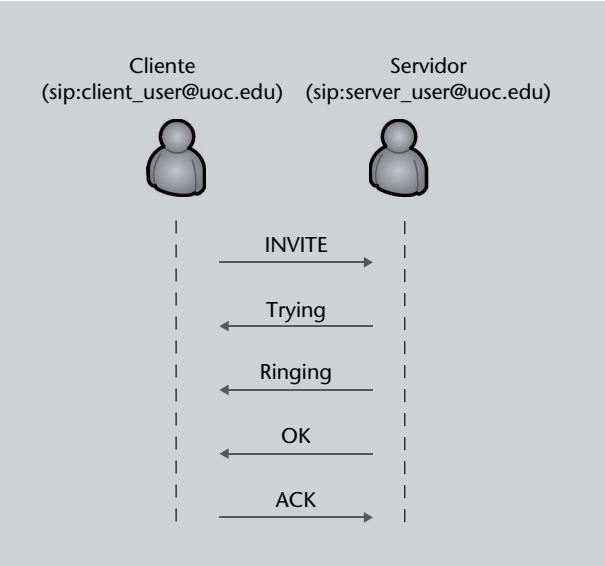
Las tres transacciones básicas llevadas a cabo por el protocolo SIP son el establecimiento de la sesión, la finalización de la sesión, y la modificación de la sesión existente. A continuación se explica cada una de estas transacciones.

3.2.1. Establecimiento de la sesión

El establecimiento de una sesión se divide en tres partes. La primera parte es aquella en la que el cliente envía una petición INVITE para iniciar el proceso de establecimiento de la sesión. En una segunda parte, el servidor responde afirmativamente o negativamente. Finalmente, el cliente, en caso de recibir una respuesta positiva, notifica positivamente la recepción.

La mejor manera de verlo es mediante un ejemplo. Imaginemos la situación de la figura 18, donde un cliente (con una URI sip:client_user@uoc.edu) se dispone a iniciar una sesión con un servidor (con una URI sip:server_user@uoc.edu). La versión del protocolo SIP es la 2.0, la capa de transporte es UDP y el puerto utilizado el 5060.

Figura 18. Establecimiento de una sesión SIP



Inicialmente el cliente envía la petición siguiente:

```
INVITE sip:server_user@uoc.edu:5060 SIP/2.0
Via: SIP/2.0/UDP uoc.edu; branch=z9hG4bk4582
Max-Forwards: 70
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060
Call-ID: 257284
Cseq: 1 INVITE
Contact: <sip:client_user@uoc.edu:5060;transport=UDP >
Content-Type: application/sdp
Content-Length: 151
— Información de negociación (SDP) —
```

Es importante destacar que el cliente ha seleccionado, aleatoriamente, el *tag* para la cabecera From. Este parámetro es elegido localmente. En cuanto al número de secuencia, el valor que toma también ha sido seleccionado por el cliente. Supongamos que es 1 porque se trata de la primera secuencia. La cabecera Contact hace referencia a la dirección URI donde se quieren recibir las peticiones, mientras que Content-Type determina qué tipo de contenido incluye el cuerpo del mensaje. En este caso, se trata de un inicio de sesión y, en consecuencia, de una negociación de parámetros mediante el protocolo SDP. El cuerpo del mensaje contiene información de la negociación de los parámetros de la sesión, pero no profundizaremos en ello porque queda fuera del objetivo del apartado.

Al recibir la petición, el servidor debe responder indicando si se ha podido alertar al usuario servidor o no. En el ejemplo de la figura 18 se incluye una

respuesta Trying. Esta respuesta es enviada por el servidor si, iniciado el procesamiento de la Petición, prevé un retardo excesivo que podría hacer pensar al cliente que la petición se ha perdido. Si el servidor tiene un tiempo de respuesta bajo, no habría que enviar ninguna respuesta Trying y se enviaría directamente la Respuesta Ringing (indicadora de la alerta en el servidor).

```
SIP/2.0 100 Trying
Via: SIP/2.0/UDP sip:server_user@uoc.edu:5060;branch=z9hG4bk4582
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060;tag=187272
Call-ID: 257284
CSeq: 1 INVITE
Contact: <sip:server_user@uoc.edu:5060;transport=UDP >
Content-Length: 0
```

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP sip:server_user@uoc.edu:5060;branch=z9hG4bk4582
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060;tag=187272
Call-ID: 257284
CSeq: 1 INVITE
Contact: <sip:server_user@uoc.edu:5060;transport=UDP >
Content-Length: 0
```

Observemos que el servidor selecciona un *tag* localmente y se incluye en la cabecera To. Además, en ambas respuestas se conserva un único identificador de diálogo (Call-ID) y, como se trata de dos respuestas que pertenecen a la misma transacción, la cabecera CSeq se mantiene. En el momento en el que el servidor acepta el establecimiento de la sesión, la respuesta es un OK que incluye la información de los parámetros negociados de la sesión.

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP sip:server_user@uoc.edu:5060;branch=z9hG4bk4582
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060;tag=187272
Call-ID: 257284
CSeq: 1 INVITE
Contact: <sip:server_user@uoc.edu:5060;transport=UDP >
Content-Type: application/sdp
Content-Length: 147
— Información de negociación (SDP) —
```

La respuesta OK supone la confirmación del establecimiento por el usuario servidor y, por tanto, hay una notificación positiva como la que se muestra:

```
ACK sip:server_user@uoc.edu:5060 SIP/2.0
Via: SIP/2.0/UDP uoc.edu; branch=z9hG4bk4582
Max-Forwards: 70
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060
Call-ID: 257284
Cseq: 1 ACK
Content-Length: 0
```

En este momento la sesión ha quedado establecida y la información multimedia, como por ejemplo la VoIP, se comienza a transferir entre los dos usuarios mediante protocolos como el RTP.

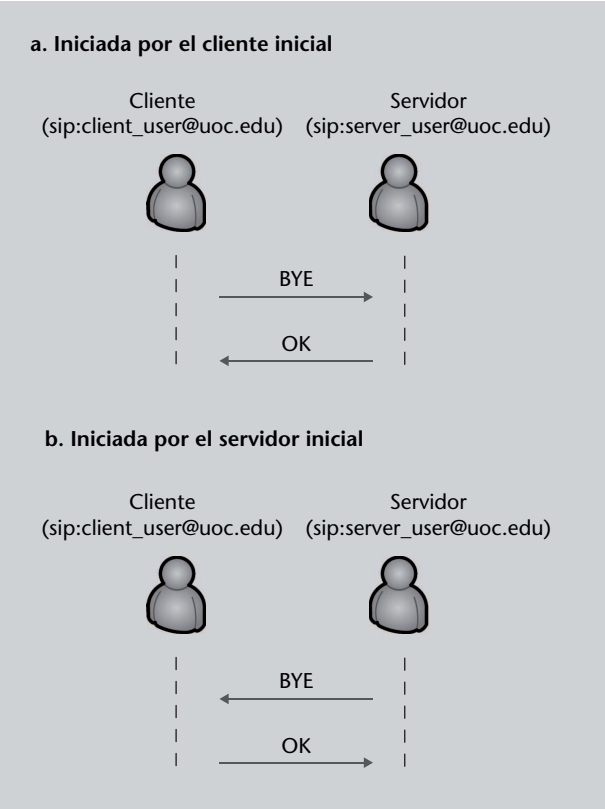
Este ejemplo es el de un establecimiento satisfactorio, pero también podría haber problemas. Si el usuario servidor, después de enviar la Respuesta Ringing, no responde con una respuesta OK antes de un tiempo máximo, el usuario cliente enviaría una petición CANCEL para cancelar el establecimiento de sesión en vez de una respuesta OK. Otra posibilidad sería que el usuario servidor esté ocupado. En este caso, la respuesta del usuario servidor sería una alerta de ocupación como la siguiente:

```
SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP sip:server_user@uoc.edu:5060;branch=z9hG4bk4582
From: sip:server_user@uoc.edu:5060;tag=187272
To: sip:client_user@uoc.edu:5060; tag=354828
Call-ID: 257284
CSeq: 1 INVITE
Contact: <sip:server_user@uoc.edu:5060;transport=UDP >
Content-Length: 0
```

3.2.2. Finalización de la sesión

La finalización de la sesión es un proceso que consta de una petición y la respuesta correspondiente. Aprovechando el ejemplo anterior, explicamos la liberación.

Figura 19. Liberación de una sesión SIP



El usuario que inicia la liberación con una petición BYE puede ser tanto el que inicialmente era el cliente como el que inicialmente era el servidor. Hay que destacar que el rol de cliente o servidor se adquiere al iniciar una transacción. Por tanto, en la figura 19 (a) el cliente es el usuario de la izquierda (client_user@uoc.edu) y en la figura 19 (b) el cliente es el usuario de la derecha (server_user@uoc.edu). En ambos casos la petición es igual, pero hay que tener en cuenta lo siguiente:

- En la figura 19 (a) la cabecera Cseq debe tomar el mismo valor que en el establecimiento más 1, suponiendo que no ha habido otras transacciones entre establecimiento y finalización (se trata de una nueva transacción).

```
BYE sip:server_user@uoc.edu:5060 SIP/2.0
Via: SIP/2.0/UDP uoc.edu; branch=z9hG4bk4582
Max-Forwards: 70
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060;tag=187272
Call-ID: 257284
CSeq: 2 BYE
Content-Length: 0
```

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP sip:client_user@uoc.edu:5060; branch=z9hG4bk4582
From: sip:client_user@uoc.edu:5060; tag=354828
To: sip:server_user@uoc.edu:5060;tag=187272
Call-ID: 257284
CSeq: 2 BYE
Content-Length: 0
```

- Si la finalización se inicia como en la figura 19 (b), el número de secuencia Cseq toma el valor de la última transacción iniciada por el usuario de la derecha. En este caso hemos considerado que Cseq vale 5, pero realmente depende del valor que hubiera tomado en la última petición.

```
BYE sip:client_user@uoc.edu:5060 SIP/2.0
Via: SIP/2.0/UDP uoc.edu; branch=z9hG4bk4582
Max-Forwards: 70
From: sip:server_user@uoc.edu:5060;tag=187272
To: sip:client_user@uoc.edu:5060; tag=354828
Call-ID: 257284
CSeq: 5 BYE
Content-Length: 0
```

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP sip:server_user@uoc.edu:5060; branch=z9hG4bk4582
From: sip:server_user@uoc.edu:5060;tag=187272
To: sip:client_user@uoc.edu:5060; tag=354828
Call-ID: 257284
CSeq: 5 BYE
Content-Length: 0
```

3.2.3. Modificación de la sesión

El protocolo SIP permite modificar los parámetros de la sesión sin necesidad de finalizarla. De la misma manera que la negociación inicial llevada a cabo durante el establecimiento se hace mediante el protocolo SDP, y unas instrucciones similares a las instrucciones básicas de establecimiento y finalización.

3.3. Servicios ofrecidos por SIP

Hasta ahora hemos analizado las sesiones SIP pero, más allá de los servidores *proxy* que reenvían las peticiones y las respuestas, no se han explicado los servicios ofrecidos por el resto de nodos: el servidor de registro, el servidor de redireccionamiento y el servidor de localización.

SIP es un protocolo de señalización en redes de paquetes diseñado para hacer frente a la movilidad de los usuarios. Dado este requisito, es necesario que cualquier usuario pueda ser localizado en cualquier momento.

La dificultad asociada a la localización es que en las redes IP la dirección URI del usuario puede cambiar a lo largo del tiempo. La red a través de la que se conecta el usuario tiene efectos sobre su dirección y, en consecuencia, sobre la posibilidad de localizarlo. El protocolo SIP define el servidor de registro como el nodo que almacena periódicamente la relación entre el URI del usuario y la identidad del usuario. Para hacerlo, se necesitan dos condiciones:

1) Que cada usuario disponga de un identificador global.

2) Que el servidor de registro almacene en cada momento cuál es la relación entre la dirección global y el URI.

La dirección global para un usuario se llama, según la terminología propia de la IETF, *Address of Record* (AOR)*. Esta dirección no cambia independientemente de la red desde la que se conecta. El proceso de registro se realiza mediante una petición REGISTER. Para estas peticiones, la línea inicial incorpora la dirección del servidor de registro, mientras que la cabecera To incluye el AOR del usuario.

*** En la terminología de 3GPP, el AOR se denomina *Public identity*.**

El problema de la movilidad, una vez registrada la relación entre la localización de un usuario y su identidad, requiere poder consultar la información almacenada en el servidor de registro. La estructura definida por el protocolo SIP no prevé la posibilidad de que el usuario haga peticiones directamente al servidor de registro; el servidor de registro recibe exclusivamente peticiones de registro y confirma o no la recepción correcta, pero no provee al usuario con la información que almacena.

Podéis consultar el RTC 3261 para profundizar en el formato y los parámetros de la petición de registro (REGISTER).

Para conseguir implementar movilidad hace falta la participación de otro nodo: el servidor de redireccionamiento. El servidor de redireccionamiento tiene la función de recibir las peticiones de los clientes y, gracias a la comunicación con el servidor de registro, contestar la petición con una respuesta que contiene la dirección de la UAS. Así, gracias a los servidores de registro y de redireccionamiento es posible implementar una solución que soporta movilidad de los usuarios.

En una red con cierta complejidad difícilmente los usuarios cliente y servidor pueden comunicarse directamente sin la ayuda de un servidor *proxy*. La función del **servidor proxy** es la de recibir mensajes (peticiones y respuestas) y redireccionarlos hacia el nodo siguiente. Fijémonos, pues, que, si nos centramos en un proxy determinado y para un mismo mensaje, ejerce la función de servidor cuando recibe el mensaje proveniente de otro nodo y, al mismo tiempo, ejerce la función de cliente cuando lo retransmite hacia el nodo siguiente.

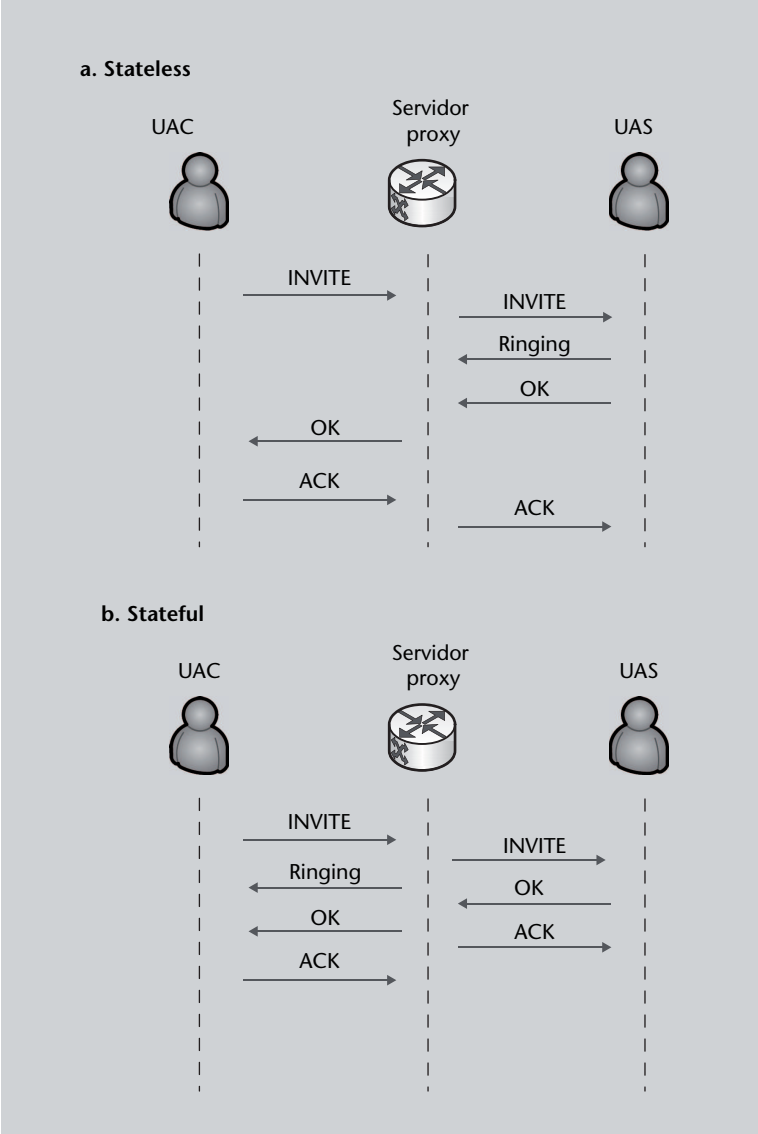
Por tanto, la función básica del servidor *proxy* es la de retransmitir los mensajes recibidos hacia el nodo apropiado. Es importante darse cuenta de que en este proceso el servidor *proxy* puede seguir dos políticas: intervenir en la comunicación de manera activa o, por el contrario, ejercer solo la función de retransmisión. Los servidores *proxy*, en función de su estrategia, se denominan *stateless proxy* o *stateful proxy*.

- Los llamados ***stateless proxys*** son aquellos servidores que no llevan a cabo ningún procesamiento de los mensajes. El objetivo es que este tipo de servidor sea transparente a los usuarios finales. El hecho de que las cabeceras de los mensajes incorporen la dirección de los diferentes servidores por donde ha transcurrido el mensaje permite que el servidor *proxy* no tenga que guardar en memoria ninguna información sobre los mensajes. Por ejemplo, este tipo de servidor nunca retransmitiría un mensaje al detectar una pérdida, ya que no utiliza ningún temporizador para controlarlo.

- Los ***stateful proxys***, en cambio, incorporan muchas más funcionalidades. Estos servidores gestionan las transacciones y son capaces de responder a determinadas peticiones antes de reenviar los mensajes. Son, pues, servidores que intervienen en la comunicación de manera activa y que, por tanto, no son transparentes a los usuarios. Por ejemplo, si después de reenviar un mensaje no se recibe el mensaje esperado antes de un período determinado, estos servidores entienden que ha habido una pérdida y vuelven a retransmitir el mensaje.

La figura 20 muestra el esquema del establecimiento de una sesión con la intervención de un servidor *proxy*. En la figura 20 (a) el servidor es *stateless* y en la figura 20 (b) el servidor es *stateful*. Como se puede observar, el servidor *stateless* solo reenvía los mensajes recibidos, mientras que el servidor *stateful* reenvía mensajes pero también genera respuestas.

Figura 20. Establecimiento de sesión con *proxy*



4. El futuro de la señalización en las redes IP

Se hace difícil predecir el futuro de las redes de telecomunicación con precisión, pero parece una evidencia que las redes del futuro, las redes denominadas de nueva generación (*Next Generation Networks*, NGN), convergen hacia las redes IP. La convergencia tanto de la red telefónica como de las redes de telefonía móvil hacia redes troncales de paquetes basadas en el protocolo IP es un hecho hoy en día.

Arquitecturas como la *IP Multimedia Subsystem* (IMS), inicialmente pensada para la red troncal de comunicaciones móviles pero posteriormente entendida como una posibilidad para convertirse en la arquitectura de las redes de nueva generación, dejan intuir que en el campo de la señalización protocolos como el SIP tienen un presente y un futuro prometedor*.

*** El IMS fue definido según las especificaciones del protocolo SIP.**

Sin embargo, no se pueden obviar las incertidumbres. Los esfuerzos ingentes de los diferentes actores del mundo de las telecomunicaciones para lograr definir estándares globales se ha demostrado, casi siempre, imposible. Ejemplos como la dicotomía entre los protocolos H.323 y SIP son una prueba de ello.

Lo que sí parece evidente es que los miles de kilómetros de enlaces y de nodos de red desplegados en todo el mundo no pueden ser sustituidos de un día para otro. La interoperabilidad entre redes, y concretamente entre sus protocolos de señalización, es una necesidad hoy y lo será en el futuro. Es en este contexto que podemos asegurar que las redes del futuro serán el resultado de un conjunto heterogéneo de redes con características diferentes, con protocolos de señalización diversos, y con necesidades particulares. Es necesario, pues, tener claro que en un entorno como el que se plantea, la señalización seguirá siendo, en esencia, el campo fundamental de las telecomunicaciones que permitirá llegar a ofrecer servicios globales.

Resumen

Este módulo nos ha permitido ver cuáles son las problemáticas en la interconexión de redes telefónicas de conmutación de circuitos y redes IP, así como la solución aportada por el conjunto de protocolos de adaptación de SIGTRAN. Se ha descrito en detalle cuáles son los protocolos que lo permiten y cómo, con los clásicos protocolos de la capa de transporte (TCP y UDP), no es posible alcanzar los requisitos de la señalización SS7.

SIGTRAN examina las carencias de los protocolos UDP y TCP y soluciona las desventajas existentes basándose, principalmente, en el protocolo TCP. Características como el *multi-homing* o *el multi-streaming* son las grandes aportaciones del protocolo de la capa de transporte definido: el SCTP.

En cuanto a la señalización entre usuarios de VoIP, este módulo también nos ha permitido adentrarnos en los conceptos básicos del protocolo SIP y en las características de sus mensajes. Hemos podido comprobar que la gran flexibilidad y sencillez (al menos en las primeras versiones de SIP) han hecho que se imponga por encima de otras alternativas, como por ejemplo el H.323, con vistas a las futuras redes *All-IP*.

Ejercicios de autoevaluación

1. Explicad los motivos por los que la señalización SS7 no se puede transportar con UDP o TCP. ¿Cuáles son las características principales que permiten que el SCTP sea apropiado para hacerlo?

2. ¿En qué consiste el *cookie mechanism* de SCTP?

3. En una PDU de SCTP, ¿se pueden incluir más de un *chunk* de datos? ¿Y un *chunk* de datos con un *chunk* INIT ACK? Razonad la respuesta.

4. Señalad la respuesta falsa y razonadlo. Un *Signalling Gateway* que conecta una red SS7 y una red IP:

a) Siempre tiene una dirección IP para la red IP y un código de punto para la red SS7.
b) No siempre tiene un código de punto; solo si tiene una instancia de la capa MTP3.
c) Si utiliza la capa de adaptación M2UA solo tiene dirección IP, pero si usa M2PA tiene dirección IP y código de punto.
d) Cuando tiene una instancia de M3UA o de SUA tiene tanto dirección IP como Código de Punto.

5. Explicad por qué el protocolo M3UA implementa funciones de direccionamiento independientes de la capa MTP3.

6. Señalad la afirmación incorrecta.

a) SIP solo define el establecimiento, finalización y gestión de una sesión, pero no es responsable del transporte de datos multimedia.
b) El protocolo H.323 es más completo que el SIP pero es menos flexible.
c) En cada transacción, el *Call-ID* presente tanto en peticiones como en respuestas se incrementará en una unidad.
d) Los *tags* de las cabeceras *From* y *To* son seleccionadas localmente.

7. Suponed la siguiente petición y respuesta entre dos nodos SIP. Sustituid los valores marcados con una X por los valores que correspondan.

```
INVITE sip:david@in3.edu SIP/2.0
Via: SIP/2.0/X in3.edu; branch=X
Max-Forwards: 70
From: sip:rosa@uoc.edu; tag=X
To: sip:david@in3.edu
Call-ID: 82283
Cseq: X INVITE
Contact: <sip:david@in3.edu;transport=TCP >
Content-Type: application/sdp
Content-Length: 149
— Información de negociación (SDP) —
```

```
SIP/2.0 X Ringing
Via: SIP/2.0/X sip:david@in3.edu;branch=z9hG4bk431ssl
From: sip:rosa@uoc.edu; tag=22164
To: sip:david@in3.edu
Call-ID: X
CSeq: 163 X
Contact: <david@in3.edu;transport=TCP >
Content-Length: 0
```

Glosario

3GPP 3rd Generation Partnership Project. Organismo de estandarización de las comunicaciones móviles.

AIMD Additive-Increase Multiplicative-Decrease. Mecanismo de control de congestión y de flujo implementado tanto en TCP como SCTP.

All-IP Todo IP. Situación en la que todas las redes convergen hacia redes IP (todo el tráfico es IP).

AoR Address of Record. Identificador global de un usuario SIP almacenado en el servidor de registro.

BSN Backward Sequence Number. Campo utilizado en el mecanismo de corrección de errores en la capa MTP2.

DoS Denial of Service. Tipo de ataque contra TCP.

DP Destination Port. Campo del puerto de destino en una PDU de SCTP.

DPC Destination Point Code. Campo del *routing label* del mensaje MTP3.

FISU Fill-In Signal Unit. Uno de los tres tipos de unidades de señalización.

FSN Forward Sequence Number. Campo utilizado en el mecanismo de corrección de errores en la capa MTP2.

GK Gatekeeper. Nodo de la red IP.

GTT Global Title Translation. Mecanismo que permite convertir una dirección en formato GT a un formato PC+SSN.

HTTP Hypertext Transfer Protocol. Protocolo de aplicación utilizado en la WWW.

IETF Internet Engineering Task Force. Organización estandarizadora en Internet.

IMS IP Multimedia Subsystem. Arquitectura de red para servicios basados en IP.

IP Internet Protocol.

IPSP IP Signalling Point. Punto de señalización IP conectado a la red IP que implementa las capas altas de la SS7.

ISUP Integrated Services User Part. Protocolo del nivel 4 del SS7.

ITU-T ITU - Telecommunications Standardization Sector. Se trata de una de las tres divisiones de la ITU.

IUA ISDN Q.921-User Adaptation Layer.

LSSU Link Status Signal Unit. Uno de los tres tipos de unidades de señalización.

M2PA MTP2-user Peer-to-peer Adaptation. Protocolo de SIGTRAN.

M2UA MTP2-User Adaptation. Protocolo de SIGTRAN.

M3UA MTP3-User Adaptation. Protocolo de SIGTRAN.

MEGACO Media Gateway Controller. Protocolo de señalización usado entre un Media Gateway y un Media Gateway Controller. También conocido como H.248.

MG Media Gateway. Nodo de la red IP.

MGC Media Gateway Controller. Nodo de la red IP.

MP Message Priority. Parámetro del mensaje M3UA.

MSU Message Signal Unit. Uno de los tres tipos de unidades de señalización.

MTP Message Transfer Part. Protocolos MTP1, MTP2 y MTP3 de los niveles 1, 2 y 3 de SS7, respectivamente.

NGN Next Generation Network. Arquitectura de red para las redes All-IP.

NI Network Indicator. Parámetro del mensaje M3UA.

OPC Originating Point Code. Campo del *Routing label* del mensaje MTP3.

OSI Open System Interconnection. Modelo de red dividido en siete capas.

PDU Protocol Data Unit. Unidad de transferencia de datos de SCTP.

QoS Quality of Service.

RDSI Red Digital de Servicios Integrados. Es la denominación castellana para el acrónimo inglés ISDN.

RFC Request for Comments.

RTC Red Telefónica Conmutada. Es la denominación castellana para el acrónimo inglés PSTN.

RTCP Real-Time Control Protocol. Véase RTP.

RTP Real-Time Transport Protocol. Protocolo de aplicación utilizado para la transmisión de información en tiempo real.

SCCP Signalling Connection Control Part. Protocolo del nivel 4 del SS7.

SCTP Stream Control Transmission Protocol. Protocolo de la capa de transporte.

SDP Session Description Protocol. Protocolo de descripción de parámetros de datos multimedia.

SG Service Gateway. Nodo de la red IP.

SI Service Indicator. Es uno de los dos subcampos del SIE del mensaje MTP3.

SIGTRAN Signaling Transport. Conjunto de protocolos definidos por la IETF.

SIP Session Initiation Protocol. Protocolo de señalización IP.

SLS Signalling Link Selector. Campo del *routing label* del mensaje MTP3 de las funciones SMH.

SP Source Port. Campo del puerto de origen en una PDU de SCTP.

SS7 Signalling System No 7. Estándar de señalización desarrollado por la ITU-T.

SSN Stream Sequence Number. Campo de la PDU de SCTP.

STP Signal Transfer Point. Punto de señalización de la red SS7.

SUA SCCP User Adaptation. Protocolo de SIGTRAN.

TCAP Transaction Capabilities Application Part. Protocolo del nivel 4 del SS7.

TCP Transmission Control Protocol. Protocolo de la capa de transporte.

TUP Telephone User Part. Protocolo del nivel 4 del SS7.

UA User Adaptation layer. Conjunto de protocolos que junto con el SCTP conforman SIGTRAN.

UAC User Agent Client. Usuario que transmite una petición SIP.

UAS User Agent Server. Usuario que transmite una respuesta SIP.

UDP User Datagram Protocol. Protocolo de la capa de transporte.

URI Uniform Resource Identifier.

V5UA V5.2-User Adaptation Layer. Protocolo de SIGTRAN.

VoIP Voice over IP. Conjunto de normas, dispositivos y protocolos para comunicaciones de voz sobre el protocolo de red IP.

Bibliografía

Agbinya, J. I. (2004). *IP Communications and Services for NGN*. Boca Raton (EUA): CRC Press.

van Bosse, J. G.; F. U. Devetak (2007). *Signaling in Telecommunication Networks* (2a ed.). Nueva Jersey (EUA): John Wiley & Sons, Inc.

Davidson, J.; J. Peters (2000). *Voice over IP fundamentals*. Indianapolis (EUA): Cisco Press.

Dryburgh, L.; J. Hewett (2004). *Signaling Systems No. 7 (SS7/C7). Protocol, Architecture and Services*. Indianapolis (EUA): Cisco Press.

Hersent, O. (2011). *IP Telephony. Deploying VoIP Protocols and IMS Infrastructure, Second Edition*. West Sussex (Reino Unido): John Wiley & Sons, Inc.

IETF *RFC 4960. Stream Control Transmission Protocol*.

IETF *RFC 3331. Signaling System 7 (SS7) Message Transfer Part 2 (MTP2) - User Adaptation Layer*.

IETF *RFC 4165. Signaling System 7 (SS7) Message Transfer Part 2 (MTP2) - User Peer-to-Peer Adaptation Layer (M2PA)*.

IETF *RFC 4666. Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)*.

IETF *RFC 3868. Signalling Connection Control Part User Adaptation Layer (SUA)*.

IETF *RFC 3261. SIP: Session Initiation Protocol*.

ITU-T *Recomendación H.323: Packet-based multimedia communications systems*.

Johnston, A. B. (2009). *Sip: Understanding the Session Initiation Protocol*. Norwood (EUA): Artech House.

Kozierok, C. H. (2004). *The TCP/IP guide: A comprehensive, illustrated Internet Protocol Reference*. San Francisco (EUA): William Pollock.

Poikselka, M.; G. Mayer; H. Khartabil; A. Niemi (2004). *The IMS. IP Multimedia Concepts and Services in the Mobile Domain*. West Sussex (Reino Unido): John Wiley & Sons, Inc.

Russell, T. (2006). *Signaling Systems 7. Fifth Edition.*. Nueva York (EUA): McGraw-Hill Professional.

Stewart, R. R.; Q. Xie (2001). *Stream Control Transmission Protocol (SCTP): A Reference Guide*. Addison-Wesley.