UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T H.234

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

(11/94)

TRANSMISIÓN DE SEÑALES NO TELEFÓNICAS

SISTEMA DE GESTIÓN DE CLAVES DE CRIPTACIÓN Y DE AUTENTICACIÓN PARA SERVICIOS AUDIOVISUALES

Recomendación UIT-T H.234

(Anteriormente «Recomendación del CCITT»)

PREFACIO

El UIT-T (Sector de Normalización de las Telecomunicaciones) es un órgano permanente de la Unión Internacional de Telecomunicaciones (UIT). Este órgano estudia los aspectos técnicos, de explotación y tarifarios y publica Recomendaciones sobre los mismos, con miras a la normalización de las telecomunicaciones en el plano mundial.

La Conferencia Mundial de Normalización de las Telecomunicaciones (CMNT), que se celebra cada cuatro años, establece los temas que han de estudiar las Comisiones de Estudio del UIT-T, que a su vez producen Recomendaciones sobre dichos temas.

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución N.° 1 de la CMNT (Helsinki, 1 al 12 de marzo de 1993).

La Recomendación UIT-T H.234 ha sido preparada por la Comisión de Estudio 15 (1993-1996) del UIT-T y fue aprobada por el procedimiento de la Resolución N.° 1 de la CMNT el 1 de noviembre de 1994.

NOTA

En esta Recomendación, la expresión «Administración» se utiliza para designar, en forma abreviada, tanto una administración de telecomunicaciones como una empresa de explotación reconocida de telecomunicaciones.

ÍNDICE

Página

SUMARIO

Se describen tres métodos de gestión de claves de criptación, a saber:

- ISO 8732;
- Diffie-Hellman; y
- RSA.

Estos son aplicables a la criptación de señales audiovisuales transmitidas digitalmente empleando la estructura de trama H.221. Los mensajes de gestión descritos se transmiten por el canal de señal de control de criptación de la Recomendación H.221, cuya estructura y utilización se definen en la Recomendación H.223.

Recomendación H.234

SISTEMA DE GESTIÓN DE CLAVES DE CRIPTACIÓN Y DE AUTENTICACIÓN PARA SERVICIOS AUDIOVISUALES

(Ginebra, 1994)

1 Consideraciones generales

Un sistema de privacidad consta de dos partes: el mecanismo de confidencialidad, o proceso de criptación de los datos, y un subsistema de gestión de claves. En esta Recomendación se describen métodos de autenticación y gestión de claves de un sistema de privacidad adecuado para utilizarlo en los servicios audiovisuales de banda estrecha conformes con las Recomendaciones UIT-T H.221, H.230 y H.242. La especificación de la confidencialidad es independiente y figura en la Recomendación H.233.

La privacidad se consigue utilizando *claves secretas*. Las claves se cargan en la parte confidencialidad del sistema de privacidad y controlan la manera según la cual se criptan y descriptan los datos transmitidos. Si un tercero consigue acceder a las claves que están siendo utilizadas, el sistema de privacidad deja de ser seguro.

El mantenimiento de claves por los usuarios constituye, pues, parte importante del sistema de privacidad. En la presente Recomendación se especifican tres métodos prácticos alternativos de gestión de claves. En los casos en que no sea posible la gestión de claves automatizada puede utilizarse una alternativa no especificada, por ejemplo la gestión de claves manual.

El primer método se identifica como ISO 8732. Se basa en claves instaladas manualmente en sistemas que atribuyen físicamente a estas claves un alto grado de protección y a continuación en intercambios automatizados de claves criptadas según las claves distribuidas manualmente. Normalmente, el algoritmo utilizado para criptar las claves distribuidas automáticamente es el mismo que para criptar la propia comunicación. La seguridad de las claves distribuidas automáticamente depende, por tanto, de la seguridad de las claves distribuidas manualmente.

Las claves distribuidas automáticamente pueden utilizarse para una única sesión o para múltiples sesiones en un periodo de tiempo determinado (por ejemplo, un mes). El método ISO 8732 contiene no sólo protocolos para el intercambio automatizado de información entre los dos terminales, sino también protocolos físicos con los que se garantiza asimismo la seguridad de las claves de distribución manual.

Hay dos entornos diferentes: el entorno directo punto a punto (dos capas), en el que los dos terminales comparten una clave común, y el entorno de tres capas, en el que los dos terminales que desean comunicar no comparten una clave común, pero utilizan las facilidades de un tercer participante con el que cada uno de ellos comparte una llave común. Las interfaces con ese tercer participante quedan fuera del alcance de la presente Recomendación, pero es preciso distinguir entre ambos entornos.

Se señala que el intercambio de claves de sesión especificado en 2.3.2 está duplicado funcionalmente en X9.17, en el sentido de que las claves distribuidas automáticamente en X9.17 son lo bastante fuertes como para ser utilizadas como claves de sesión. No obstante, para seguir el modelo de esta Recomendación, se utilizará como *clave*, la *clave* de claves del 2.3.2.

El segundo método es un método sencillo, aunque seguro, conocido como método «Diffie-Hellman ampliado», que genera e intercambia claves automáticamente por conducto del propio sistema (este intercambio de claves está él mismo criptado). No requiere acción alguna por parte de los usuarios hasta que se hayan intercambiado las claves; a los usuarios se les dice a continuación que confirmen *verbalmente* que en cada terminal está disponible la misma secuencia de comprobación. El método es muy adecuado para evitar, por ejemplo, que en una comunicación audiovisual efectuada por canal de satélite haya quienes la escuchen siendo ajenos a la misma. Para burlar el sistema haría falta que el intruso interceptara por completo la comunicación bidireccional antes de que se hubiera activado la criptación y que intercambiara claves con ambos participantes, simulando ser, ante cada uno de ellos, el otro legítimo participante. El método no proporciona autenticación.

El tercer método es más complejo y proporciona un mayor grado de privacidad y también la *autenticación* de entidades de servicios audiovisuales [terminales, unidades de control multipunto (MCU, *multipoint control units*) etc.]. El «método RSA» es muy similar al de las claves públicas especificado en la Recomendación X.509 y utiliza el algoritmo RSA. Este método requiere el establecimiento de una agencia de seguridad a disposición de todo el colectivo de entidades que necesitan interconectabilidad: la certificación es efectivamente autónoma y se basa en la integridad de la agencia. Este mecanismo de autenticación hace posible que los participantes en una comunicación conferencia sean identificados a otros de manera segura y puede utilizarse tanto en llamadas multipunto como en llamadas punto a punto.

Todos los métodos precisan la utilización de un canal asociado, despejado y libre de errores. Se señala que ninguno de estos métodos proporciona control de acceso, integridad y no rechazo.

En este documento se hace referencia a un cuarto método, el de «intercambio de claves manual».

El intercambio de claves manual se define como la introducción por los usuarios de claves de criptación de claves directamente en los terminales, sin intercambio de mensajes de la Recomendación H.234. La misma clave se introduce en ambas ubicaciones. La longitud de las claves depende del algoritmo de criptación. El orden de los bits de las claves es el de bit más significativo (msb, most significant bit) introducido en primer lugar y bit menos significativo (lsb, least significant bit) introducido en último lugar. El mecanismo real de introducción de las claves en el terminal depende del terminal y queda fuera del alcance de la presente Recomendación.

A continuación se dan unos ejemplos:

- utilización de un teclado telefónico para introducir: (msb) 00111010...01110100 (lsb);
- telecargar lo mismo desde un computador;
- utilización de un teclado para introducir lo mismo como caracteres hexadecimales: (msb) 3A...74 (lsb).

La introducción manual puede efectuarse antes de iniciar la llamada o durante la misma. En este segundo caso, los participantes pueden optar por invocar la criptación mientras están en una conferencia, introducir una clave utilizando la interfaz proporcionada por el terminal e iniciar seguidamente la criptación a través de la interfaz de usuario del terminal. Cuando se solicita la criptación a través de la interfaz de usuario, se envía el código de señal de asignación de velocidad binaria (BAS, *Bit rate allocation signal*), «Encrypt-ON» (criptación activada), se abre el canal de señal de control de criptación (ECS, *encryption ccontrol signal*), se seleccionan los algoritmos de criptación, se acuerda el modo manual de gestión de claves y se intercambian las claves de sesión.

Para que un sistema de criptación se considere privado, todos los conferenciantes deben estar al corriente de quién o qué tiene acceso a datos no criptados, ya sean otros conferenciantes o equipos tales como MCU o facilidades de conversión. Para ello es preciso un periodo de establecimiento inicial, antes de que comience la conferencia, de modo que las entidades puedan autenticarse entre sí. De este modo, todas las entidades que tienen acceso a datos no criptados son identificadas de manera segura por todas las demás entidades antes del comienzo de la conferencia. El marco de autenticación proporciona también información a cualquier proveedor de red, por ejemplo la relativa a la facturación de una llamada de MCU.

Si se dispone de datos no criptados en la MCU (denominada «MCU encargada») el equipo deberá formar parte de cualquier marco de autenticación. Se deberá además informar a los usuarios de la existencia de una MCU encargada en la red.

En la cláusula 2 se examinan aspectos comunes a todos los métodos, mientras que las cláusulas 3, 4 y 5 se ocupan, respectivamente, de los métodos ISO 8732, Diffie-Hellman y RSA.

Definiciones

AVSE: Entidad de servicio audiovisual (terminales, MCU, etc.).

clave: Clave de criptación de claves.

2 Sistema de mensajes e intercambio de claves

2.1 Canal de mensajes

El sistema que se describe a continuación se compone de un cierto número de mensajes definidos, vehiculados en secuencia entre los dos extremos del enlace. El canal sin errores requerido a tal efecto se describe en la Recomendación H.233, donde se hace referencia a los bloques de intercambio de sesión (SE, *session exchange*).

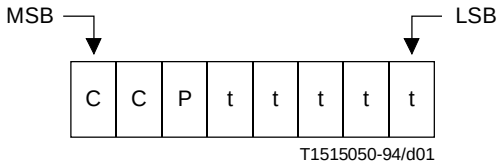
2.2 Formatos de mensaje

Los mensajes utilizados por el sistema de criptación para la distribución y autenticación de claves se formatan en la forma identificador, longitud, contenido (ILC, *identifier*, *length*, *content*) nidificada que se describe en la Recomendación X.209. La longitud puede codificarse en forma corta o forma larga. No se utilizará la forma indefinida indicada en la Recomendación X.209.

A continuación se da una breve descripción de algunas de las definiciones de la Recomendación X.209 utilizadas en esta Recomendación.

2.2.1 Identificador

Un identificador es un octeto con la siguiente estructura:



Los dos bits C, «clase de rótulo», definen el tipo de identificador, que es 10 (específico del contexto) para los identificadores definidos en esta Recomendación.

El bit de primitiva/constructor (P) indica si el contenido es una primitiva o si se compone de elementos nidificados.

El rótulo de 5-bits (ttttt) define de manera exclusiva al identificador (de acuerdo con su clase).

Así pues, todos los identificadores de esta Recomendación tienen la forma de octeto 1 0 P t_1 t_2 t_3 t_4 t_5.

2.2.2 Longitud

La longitud especifica la longitud en octetos del contenido y es de naturaleza variable.

La forma corta tiene un octeto de longitud y se utilizará con preferencia a la forma larga cuando L sea menor que 128. El bit 8 tiene el valor cero y los bits 7 a 1 codifican L como un número binario sin signo cuyos MSB y LSB son el bit 7 y el bit 1, respectivamente.

La forma larga tiene una longitud de 2 a 127 octetos si se utiliza cuando L es superior o igual a 128 y menor que 2 a la potencia 1008. El bit 8 del primer octeto tiene el valor uno. Los bits 7 a 1 del primer octeto codifican un número inferior en una unidad al tamaño de la longitud en octetos, como número binario sin signo, cuyos MSB y LSB son el bit 7 y el bit 1, respectivamente. El propio L se codifica como un número binario sin signo cuyos MSB y LSB son el bit 8 del segundo octeto y el bit 1 del último octeto, respectivamente. Este número binario será codificado con el menor número posible de octetos, sin octetos delanteros que contengan el valor 0.

2.2.3 Cadena de bits

Una cadena de bits en forma primitiva tiene los bits empaquetados a ocho en un octeto y va precedida por un octeto que codifica el número de bits no utilizados en el octeto final del contenido – de cero a siete – como número binario sin signo. Estos MSB y LSB son el bit 8 y el bit 1, respectivamente.

2.3 Comienzo del sistema de privacidad

El comienzo del sistema entraña tres mensajes, P0, P1 y P2, que se detallan más adelante. El sistema de privacidad se invoca enviando un mensaje (desde cualquier extremo) de tipo (P0). El mensaje (P0) incluye bits que describen los mecanismos, ISO 8732 y/o Diffie-Hellman y/o RSA, que el emisor puede manejar. El receptor de ese mensaje determina el mecanismo que se debe utilizar y responde con un mensaje de tipo (P0) o de tipo (P1), dependiendo del resultado.

Si los dos envían el mensaje (P0) al mismo tiempo, todavía es posible efectuar la elección comparando los campos de bits intercambiados:

- si ambos extremos soportan el mismo mecanismo, ése es el que se utiliza; si soportan más de un mecanismo, el orden de preferencia es ISO 8732, a continuación Diffie-Hellman, a continuación RSA/Recomendación X.509 y, por último, la opción no especificada a la que se hace referencia en esta Recomendación como opción «manual»;

- si no hay ninguna capacidad común, el enlace no puede ser criptado.

2.3.1 Mensajes de comienzo

Nombre del mensaje:	Petición de sistema de privacidad (P0).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000000
Significado:	El emisor del mensaje desea utilizar un sistema de criptación. Este mensaje puede utilizarse para tratar de iniciar la criptación o en respuesta a otro mensaje P0.
Contenido:	Un octeto primitivo como se muestra a continuación. El campo de bits dentro del contenido muestra el tipo de mecanismo que puede utilizarse. (msb) 0000XDRM (lsb). X se pone a "1" si se soporta ISO 8732, o a "0" si no se soporta. D se pone a "1" si se soporta Diffie-Hellman, o a "0" si no se soporta. R se pone a "1" si se soporta RSA, o a "0" si no se soporta. M se pone a "1" si existe un sistema de gestión de claves no especificado, tal como el de introducción de claves manual, o a "0" si no existe.
En la «notación de sintaxis abstracta» ASN.1 de la Recomendación X.209:	RequestEncryptionSystem ::= [0] IMPLICIT OCTET STRING
	En este mensaje el contenido tiene siempre una longitud de un octeto.

Nombre del mensaje:	No se puede criptar (P1).
Significado:	Enviado en respuesta al (P0). El emisor de este mensaje no utilizará un sistema de criptación.
Identificador del mensaje	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000001
Contenido:	Este mensaje no tiene contenido.

Nombre del mensaje:	No comienza el sistema de criptación (P2).
Significado:	El emisor de este mensaje no ha comenzado su sistema de criptación. Esto podría deberse a un fallo del intercambio de claves pero, por razones de seguridad, no se da ninguna indicación de la causa del fallo en el mensaje.
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000010
Contenido:	Este mensaje no tiene contenido.

2.3.2 Intercambio de claves de sesión

Las claves de sesión utilizadas para criptar la información se obtienen del intercambio de claves de sesión. El mensaje que contienen las claves de sesión está formatado como aquí se describe, y criptado utilizando una clave de criptación de claves (abreviado a *clave* en esta Recomendación) derivada del protocolo de autenticación o de distribución de *claves*. Conviene insistir en la diferencia entre estos dos tipos de clave. Las claves de sesión se utilizan en la criptación/descriptación de la señal audiovisual en su trama de la Recomendación H.221, mientras que la *clave* sólo se utiliza en la criptación y descriptación del intercambio de claves de sesión.

El mecanismo de criptación entraña claves de N bits de longitud. Los dos participantes implicados establecen una *clave* común, cuya longitud es también de N bits; en el caso del RSA, hay una #clave# de autenticación adicional, utilizada para deducir la *clave*.

La *clave* común se utiliza para criptar cuatro claves de N bits descritas en esta subcláusula (Figura 1). El método de criptación utilizado será el mismo que el elegido para la criptación de la señal audiovisual, lo que se indica mediante la transmisión del mensaje P9, definido a tal efecto en la Recomendación H.233.

El mensaje de intercambio de claves de sesión consta de un identificador de mensaje de 8 bits, un vector de inicialización con corrección de errores y un valor aleatorio de 4N bits. Cada extremo envía ese valor y deduce de él el conjunto de cuatro claves de sesión. La longitud de cada una de las claves es de N bits, dependiendo el valor de N del algoritmo de criptación que haya de utilizarse (por ejemplo, en el caso de B-crypt, N = 56).

Los números aleatorios transmitidos y recibidos se procesan como cuatro bloques de N bits, según se indica a continuación:

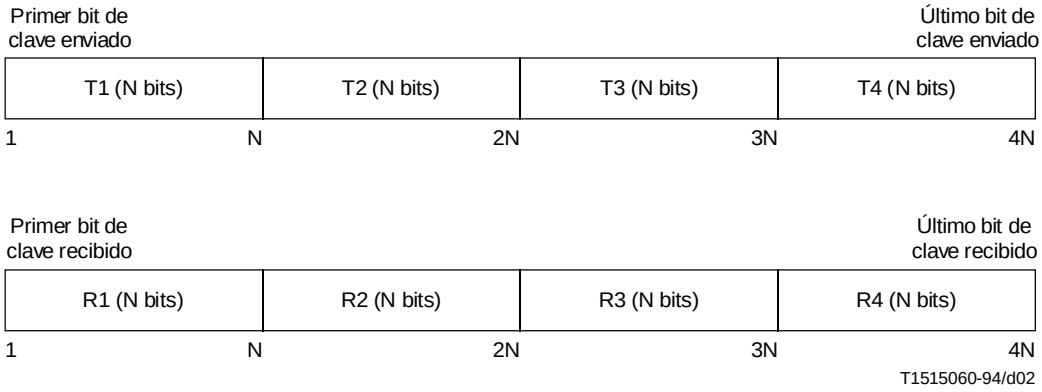


FIGURA 1/H.234

Orden de los bits de intercambio de claves de sesión

Cada una de las cuatro claves se forma mediante una O exclusiva bit por bit de un bloque transmitido y un bloque recibido, manteniendo la ordenación de los bits, esto es, el bit más significativo de la clave (es decir, el bit más significativo del primer byte o palabra de datos clave cargados en el dispositivo criptador) se forma mediante la O exclusiva bit por bit ... en los dos primeros bits de los bloques. Utilizando la ordenación de bits de la Figura 1 se obtienen las cuatro claves de la siguiente manera:

«Enviar clave de criptación #1»	formada por el bloque T1, la O exclusiva y el bloque R3
«Enviar clave de criptación #2»	formada por el bloque T2, la O exclusiva y el bloque R4
«Recibir clave de criptación #1»	formada por el bloque T3, la O exclusiva y el bloque R1
«Recibir clave de criptación #2»	formada por el bloque T4, la O exclusiva y el bloque R2

La clave de criptación #1 se utilizará para la criptación del contenido de la señal tramada especificada en A.3/H.221, «encryp-on». Cuando el protocolo multicapas (MLP, *multi-layer protocol*) está activado según una de las instrucciones BAS del Cuadro A.1/H.221 o del Cuadro A.2/H.221, la criptación del canal MLP es tal como se especifica en las normas de las Recomendaciones de la serie T.120, utilizando la misma clave #1 o la clave #2 alternativa.

Es posible que el algoritmo elegido requiera paridad en las claves; esto es un asunto local y no forma parte de la transmisión.

La única comprobación efectuada se lleva a cabo en el conjunto de los 4N bits. Si el resultado de la operación «O exclusiva» en la totalidad de los 4N bits es cero (es decir, las cuatro claves de N bits son cero), no se cargan las claves y no se invoca el sistema de privacidad.

Mensaje de intercambio de claves de sesión (P6)

El mensaje consta del identificador del mensaje, un vector de inicialización de 96 bits que incluye por defecto bits de corrección de errores y un número aleatorio de 4N bits.

Nombre del mensaje:	Esta es la información de claves de sesión (P6).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100110
Significado:	El emisor de este mensaje está intercambiando información de claves de sesión.
Contenido:	Un constructor que contiene el vector de inicialización (no criptado) utilizado para la criptación de los datos de las claves de sesión y la información de claves de sesión criptadas en el formato mostrado.
En la «notación de sintaxis abstracta» de la Recomendación X.209:	SessionKeyInformation ::= [6] IMPLICIT SEQUENCE { initialisation vector [0] IMPLICIT BIT STRING, session key information [1] IMPLICIT BIT STRING }

3 Gestión de claves de ISO 8732

3.1 Introducción

La norma de la referencia 1 proporciona un proceso uniforme de protección e intercambio de claves criptográficas para autenticación y criptación. La norma define la gestión manual y automatizada del material de generación de claves, incluyendo:

- El control durante la vida útil del material de generación de claves para evitar la revelación no autorizada de las mismas, su modificación o su sustitución.
- La distribución de material de generación de claves para permitir el interfuncionamiento entre equipos o facilidades de criptografía.
- La seguridad de la integridad del material de generación de claves durante todas las fases de su vida útil, incluyendo su generación, distribución, almacenamiento, introducción, utilización y destrucción.
- La recuperación en caso de fallo del proceso de gestión de claves o cuando se cuestione la integridad del material de generación de claves.

El algoritmo utilizado para la criptación de las claves distribuidas automáticamente suele ser el mismo que el utilizado para criptar la propia comunicación, y puede negociarse mediante intercambios de mensajes P8. Cuando se utiliza un algoritmo distinto del DES, el sistema de gestión de claves no es estrictamente conforme a la referencia 1, pero la única diferencia es este aspecto.

3.2 Arquitectura de gestión de claves

En la referencia 1 puede encontrarse una lista de requisitos del par de comunicantes. Existe una arquitectura de dos capas y una arquitectura de tres capas. Cualquiera de ellas puede utilizarse para el intercambio de claves.

3.3 Entornos de gestión de claves

Existen tres entornos para la distribución de claves:

- punto a punto;
- centro de distribución de claves (CKD, *key distribution centre*); y
- centro de traducción de claves (CKT, *key translation centre*).

En la referencia 1 pueden encontrarse los detalles relativos a estos entornos.

Punto a punto es un entorno de dos capas en el que dos terminales comparten una clave común. Se supone que esta clave común ha sido distribuida manualmente utilizando protocolos seguros y protección física según se indica en ISO 8732. El intercambio automático de claves especificado en ISO 8732 garantiza la generación de una *clave* común por un terminal, que se pasa al otro terminal de manera segura y que es la clave utilizada en la creación de las claves de sesión especificadas en 2.3.2.

Las distinciones entre un centro de distribución de claves (CKD) y un centro de traducción de claves (CKT) no son pertinentes a los efectos de la presente Recomendación, pero está especificado que la clave compartida por cada terminal con el tercer participante o centro (CKD o CKT) sea una clave de longitud doble. La manera según la cual uno de los terminales, por ejemplo el terminal A, interconecta con el centro queda también fuera de la especificación de esta Recomendación, pero al concluir el intercambio con el centro, el terminal A posee no sólo una *clave* clara sino también una *clave* criptada según la clave de longitud doble (véase la especificación del algoritmo en ISO 8732) del terminal B. Envía ésta a través del ECS al terminal B, en donde se convierte a continuación en una *clave* clara, y el protocolo de intercambio de sesión puede comenzar.

3.4 Intercambios de mensajes de servicio criptográfico

ISO 8732 emplea texto para intercambiar mensajes. El orden en el que se envían los mensajes y las circunstancias en las que éstos se envían pueden encontrarse en la referencia 1. El siguiente mensaje (P11) proporciona el mecanismo de envío de un mensaje de servicio criptográfico (CSM, *cryptographic service message*) de ISO 8732. Cada byte representa un carácter del texto.

La ordenación de bits es tal que se transmite primero el bit más significativo.

Nombre del mensaje:	Mensaje de servicio criptográfico (P11).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10101011
Significado:	El emisor de este mensaje envía un solo mensaje de servicio criptográfico.
Contenido:	Cadena de texto primitiva.
En la «notación de sintaxis abstracta» de la Recomendación X.209:	CryptographicServiceMessage := [11] IMPLICIT VisibleString

Se supone que la interfaz de usuario del terminal proporciona protocolos para identificar por su nombre claves apropiadas y otros identificadores implícitos en el protocolo de ISO 8732. Por ejemplo, en una red privada, cada par de comunicantes en un entorno de dos capas puede tener una clave compartida y denominada insertada en la unidad criptográfica del sistema, y el mecanismo para efectuar la llamada puede identificar automáticamente al subsistema criptográfico la clave compartida y denominada apropiada.

En la referencia 1 se especifican mensajes de servicio para condiciones de error y respuestas erróneas. Si dos terminales, que soportan ISO 8732, tratan de comunicar y comunican de manera puntual, cuando, de hecho, no se corresponden mutuamente en ninguno de los tres entornos, los protocolos (que implican identificadores o nombres de claves, contadores, centros, ..., conocidos comúnmente) se descompondrán y la sesión criptográfica intentada terminará con una notificación a los operadores de los terminales. Para completar una llamada que requiera criptación, los usuarios de los dos terminales deberán retroceder a otro mecanismo de intercambio de gestión de claves o establecerse ellos mismos en uno de los tres entornos (utilizando muy probablemente un tercer participante mutuo o centro).

3.5 Ejemplo de intercambio de mensajes de ISO 8732

Considérese, a título de ejemplo, la Figura 2 en la que se muestra un flujo de mensajes normal. El primer mensaje que ha de enviarse es el de petición de servicio (RSI, *request service*). En la subcláusula 8.4 de la referencia 1 se describe el formato de mensaje (CSM, *cryptographic service message*), [mensaje de servicio criptográfico] que es como sigue:

CSM(MCL/...)

en donde todos los caracteres son ASCII, los paréntesis indican el comienzo y el final del mensaje y el trazo inclinado [o barra oblicua](/) se utiliza para separar los rótulos de los campos del contenido de los mismos.

En este caso, el contenido de los campos MCL es RSI, con lo que el texto efectivamente enviado es:

CSM(MCL/RSI...)

El orden de los campos del mensaje RSI se indica en el Cuadro III de ISO 8732. Dicho orden es MCL RCV ORG SVR EDC (opcional). En este ejemplo se omite el EDC opcional.

En el Cuadro II se define con más detalle cada uno de los campos. Así pues, el mensaje enviado sería:

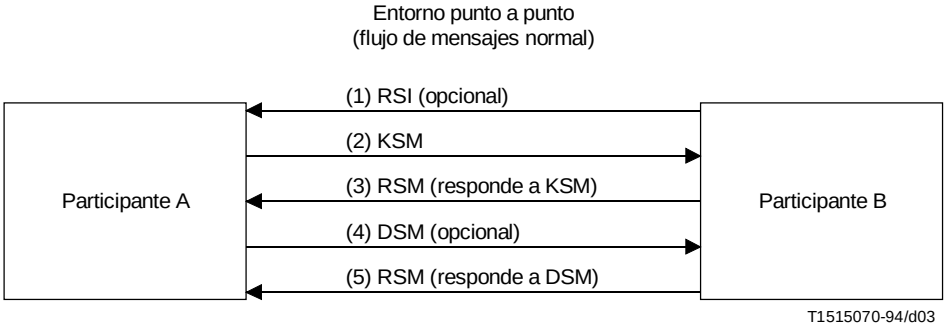
CMS (MCL/RSI"RCV/A"ORG/B"SVR/KK.KD.IV)

donde

"	Espacio en blanco insertado, utilizado como separador de campos
A	El receptor
B	El emisor
.	Separador de subcampos
SVR	Petición de servicio
KK	Petición de *claves*
KD	Petición de dos claves de datos
IV	Petición de IV (vector de inicialización, *initialisation vector*)
MCL	Clase de mensaje
RCV	Receptor
ORG	Originador

En la subcláusula 9.7 de ISO 8732 se describe con más detalle el mensaje de RSI.

El segundo mensaje sería el KSM [*key service message*, mensaje de servicio de clave], el tercero el RSM [*response service message*, mensaje de servicio de respuesta], el cuarto el DSM [*disconnected service message*, mensaje de servicio desconectado] y el quinto el RSM de nuevo.



NOTA – El proceso de desconexión (DSM) puede ser iniciado por el participante A o por el participante B; se muestra la iniciación por el participante A.

FIGURA 2/H.234

4 Distribución de claves Diffie-Hellman ampliada

4.1 Introducción

El intercambio se basa en el método Diffie-Hellman, pero ampliado para aprovechar las propiedades del enlace audiovisual a fin de proporcionar un elemento de protección frente a las derivaciones de líneas activas. El resultado del intercambio es un valor secreto compartido utilizado tanto para verificar la línea como para intercambiar claves de sesión.

El funcionamiento es como sigue (véase Apéndice I [1]:

1) el protocolo de distribución de la *clave* intercambia datos de acuerdo con el protocolo aquí descrito;
2) los datos de (1) se utilizan para intercambiar claves de sesión, que se emplean a continuación para criptar el enlace;
3) los datos de (1) se utilizan para verificar el enlace.

4.2 Protocolo básico

El protocolo básico consiste en un intercambio inicial de datos seguido de un intercambio bidireccional de los resultados intermedios, de los que se obtienen los datos compartidos.

4.2.1 Método de intercambio de la *clave*

El método utilizado es una doble versión del método Diffie-Hellman básico. El doble intercambio se utiliza para que la *clave* resultante no esté basada enteramente en un primo y una raíz primitiva elegidos en un solo terminal.

Considérense dos AVSE: A y B.

A envía a B: el primo p_A,

la raíz primitiva probabilística a_A,

el valor $c_1 = \{a_A^{a1} \text{mod } p_A\}$, donde a_1 es un número aleatorio conocido solamente por A.

B envía a A: el primo p_B,

la raíz primitiva probabilística a_B,

el valor $c_2 = \{a_B^{b1} \text{mod } p_B\}$, donde b_1 es un número aleatorio conocido solamente por B.

A envía a B: el valor $c_3 = \{a_B^{a2} \text{mod } p_B\}$, donde a_2 es un número aleatorio conocido solamente por A.

B envía a A: el valor $c_4 = \{a_A^{a2} \text{mod } p_A\}$, donde b_2 es un número aleatorio conocido solamente por B.

Calcúlese un par de resultados r_1 y r_2 para A y a continuación para B.

El AVSE A forma: $r_1 = c_4^{a1} \text{mod } p_A$ y $r_2 = c_2^{a2} \text{mod } p_B$

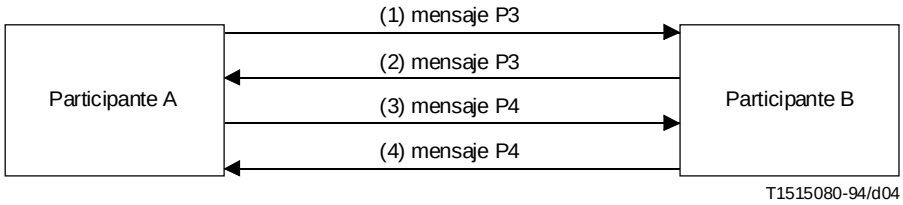
El AVSE B forma: $r_1 = c_1^{b2} \text{mod } p_A$ y $r_2 = c_3^{b1} \text{mod } p_B$

Tanto A como B están ahora en posesión de los mismos valores de resultados $r_1 = a_A^{a1.b2} \text{mod } p_A$ y $r_2 = a_B^{a2.b1} \text{mod } p_B$.

El resultado final R_{12} se obtiene mediante una «O exclusiva» bit por bit de r_1 con r_2. Si r_1 y r_2 no tienen la misma longitud y L denota la longitud del más corto, la operación O exclusiva es:

{(bits L menos significativos de r_1). O exclusivo. (bits L menos significativos de r_2)}

El intercambio Diffie-Hellman doble se ilustra en la Figura 3.



(1) p_A, a_A, ($a_A^{a1} \text{mod } p_A$) por mensaje {P3}

(2) p_B, a_B, ($a_B^{b1} \text{mod } p_B$) por mensaje {P3}

(3) $a_B^{a2} \text{mod } p_B$) por mensaje {P4}

(4) $a_A^{b2} \text{mod } p_A$) por mensaje {P4}

FIGURA 3/H.234

Intercambio Diffie-Hellman doble

4.2.2 Obtención de la *clave*

Como se expone más arriba, A y B forman $r_1 = (a_A^{a1.b2} \text{mod } p_A)$ y $r_2 = (a_B^{a1.b2} \text{mod } p_B)$, y a continuación se forma R_{12}, mediante una «O exclusiva» bit por bit de estos valores. Tanto A como B comprueban el valor del resultado y, si todos los bits son 0, se envía el mensaje «no comienza el sistema de criptación» (P2) a la otra entidad.

R_{12} es un valor de K bits disponible en cada extremo del enlace. Se emplea para obtener el código de comprobación y la *clave* utilizada para la criptación de las claves de sesión. En un mecanismo de confidencialidad de N bits, con un código de comprobación de M bits, los N bits menos significativos constituyen el código de comprobación y los siguientes N bits constituyen la *clave*. Esto es lo que se muestra en la Figura 4. El valor de M es de 64 bits. El valor de N es la longitud de la *clave* y lo determina el código de criptación que ha de utilizarse.

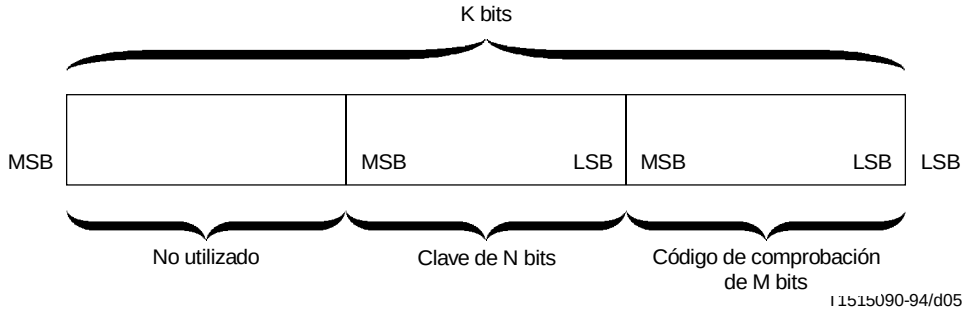


FIGURA 4/H.234

Interpretación de resultado de distribución de claves

Se señala que K debe tener una longitud superior a (M + N) bits. En el caso de un algoritmo de criptación de 64 bits y código de comprobación de 64 bits, K debe ser superior a 128 bits. En la práctica, K será bastante más largo que eso.

4.3 Mensajes Diffie-Hellman

En esta subcláusula se describe el contenido de los mensajes necesarios para comenzar el sistema de criptación y para el intercambio de la *clave* Diffie-Hellman.

4.3.1 Información de intercambio de la *clave*

Nombre del mensaje:	Esta es la información de intercambio de la *clave* (P3).
Significado:	El emisor de este mensaje envía la información de intercambio de la *clave* contenida como parte de un intercambio Diffie-Hellman doble.
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100011
Contenido:	Un constructor constituido por primitivas para raíz primitiva, primo y resultado intermedio, como se muestra más adelante. Se señala que el término raíz primitiva no tiene relación con el término primitiva utilizado en las definiciones de mensaje.
En la notación de ASN.1:	keyExchangeInformation ::= [3] IMPLICIT SEQUENCE { primitive root [0] IMPLICIT BIT STRING, prime [1] IMPLICIT BIT STRING, intermediate result [2] IMPLICIT BIT STRING } El contenido de Primitive Root (raíz primitiva) es una cadena de bits primitiva. El contenido de Prime (primo) es una cadena de bit primitiva. El contenido de Intermediate Result (resultado intermedio) es una cadena de bits primitiva que contiene uno de los resultados intermedios para el intercambio Diffie-Hellman.

4.3.2 Información de intercambio de *clave* intermedia

Nombre del mensaje:	Información de intercambio de *clave* intermedia (P4).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000100
Significado:	El emisor de este mensaje envía la información de intercambio de la *clave* contenida como parte de un intercambio Diffie-Hellman doble.
Contenido:	Una cadena de bits primitiva que contiene el resultado intermedio.
En la notación de ASN.1:	IntermediateKeyExchangeInformation ::= [4] IMPLICIT BIT STRING
	La cadena de bits del resultado intermedio contiene uno de los resultados intermedios del intercambio Diffie-Hellman. Los mensajes P3 y P4 constituyen un intercambio Diffie-Hellman doble de modo que la *clave* Diffie-Hellman final viene determinada por los dos extremos de enlace.

4.3.3 Información de código de comprobación procedente de MCU

Nombre del mensaje:	Esta es la información de código de comprobación procedente de MCU (P5).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100101
Significado:	Una MCU envía la información de código de comprobación contenida resultante de los intercambios Diffie-Hellman.
Contenido:	Un constructor para identificador de enlace y código de comprobación.
En la notación de ASN.1:	Link check code information ::= [5] IMPLICIT SEQUENCE { link identifier [0] IMPLICIT BIT STRING, check code [1] IMPLICIT BIT STRING }

Una MCU enviará un mensaje (P5) por cada uno de los enlaces que haya completado el intercambio de *clave* Diffie-Hellman.

Se señala que el identificador de enlace se utiliza para identificar el enlace de la MCU con el que está relacionado el código de comprobación. Para interpretar este identificador es preciso un conocimiento de la configuración de la MCU. (Véase también la Nota de 4.4).

4.4 Extensión para comprobaciones de líneas

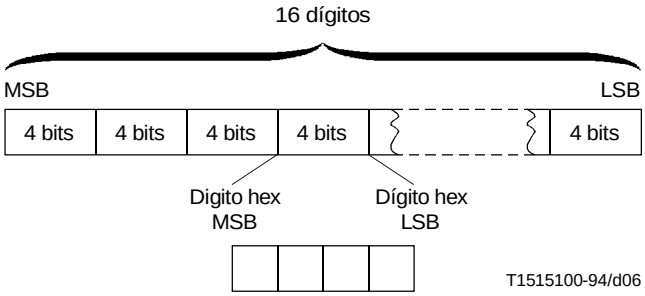
En 4.2 se obtuvo un código de comprobación de 64 bits. Dicho código será presentado por el terminal como la totalidad o parte de un número hexadecimal de 16 dígitos, cuya ordenación de bits se muestra en la Figura 5, utilizando la terminología de la Figura 4.

El valor se presenta a cada usuario tal como se muestra, es decir, el dígito situado más a la izquierda se obtiene del extremo del MSB del código de comprobación. No es necesario presentar todos los dígitos; es probable que baste con presentar los cuatro situados más a la izquierda ya que, con esa presentación, la probabilidad de no detectar un problema en la línea es de 1 en 2^{16}. El valor presentado es transferido verbalmente de un usuario a otro por el canal audiovisual criptado; el segundo usuario debe comprobar que corresponde al valor visualizado en su terminal.

NOTA – Se sugiere que la comprobación verbal se haga antes de que el audio sea criptado efectivamente; además, la temporización de este proceso y la del proceso alternativo para la situación multipunto descrita en 4.3.3 debe ser la misma.

5 Funcionamiento basado en RSA

NOTA – Todas las referencias de esta cláusula a «clave» tienen el sentido de #clave# mencionado en 2.3.2.



NOTA – Cada bloque de 4 bits del código de comprobación constituye un dígito hexadecimal que será presentado visualmente al usuario.

FIGURA 5/H.234

Ordenación de los bits para comprobaciones de líneas

5.1 Introducción

5.1.1 Consideraciones generales

En esta subcláusula se describe un marco de autenticación basado en RSA para servicios audiovisuales que incluyen conexiones punto a punto y multipunto.

Los procedimientos y las funciones de autenticación descritas se basan en la Recomendación X.509 de la UIT. En la presente Recomendación, la autenticación se establece con la utilización de uno o más niveles de las llamadas autoridades de certificación. Una autoridad de certificación (CA) emite certificados autónomos a entidades u otras CA, que dichas entidades o CA pueden utilizar para autenticarse ellas mismas ante otras entidades y CA. En el caso de los servicios audiovisuales, las entidades pueden ser terminales de usuario o MCU encargadas.

El marco de autenticación específico que aquí se describe utiliza dos niveles de CA. En el nivel más bajo, cada dominio de red, por ejemplo, un país o una compañía, tendrá su propia CA. Para hacer posibles los servicios audiovisuales entre dominios autenticados, las CA tendrán una CA común a un nivel superior para autenticarlas a ellas. Esta CA común ha de representar un punto de confianza común a todos los usuarios.

Para cuando esto no es posible, existe un esquema alternativo, más complicado, descrito brevemente en 5.5.

Ha de depositarse en las CA a nivel de dominio de red la confianza de que no repetirán los nombres de identificación en los certificados. Se supone que la propia autenticación ha de establecerse en un entorno no fiable. Además, una vez que una entidad ha sido autenticada se confía en ella (hasta que termine la llamada).

5.1.2 Abreviaturas

CA	Autoridad de certificación (*certification authority*)
CCA	Autoridad de certificación de país (*country certification authority*)
GCA	Autoridad de certificación general (*general certification authority*)
h[*]	Resultado de la función h aplicada a *
X<<Y>>	Certificado de Y generado por X
Xp	Clave pública de RSA de la entidad X
Xs	Clave secreta de RSA de la entidad X
Xp[*]	Criptación/descriptación de [*] con Xp. En caso de RSA, se efectúan por exponenciación.
Xs[*]	Criptación/descriptación de [*] con Xs. En caso de RSA, se efectúan por exponenciación.

5.2 Establecimiento del sistema

El sistema que aquí se especifica contiene una jerarquía de tres niveles. En el nivel más bajo se hallan las AVSE. Cada una de éstas tiene relación con una sola CA de nivel medio cuando comunica con otra AVSE. Las CA de este nivel sirven como autoridades de certificación de un grupo de entidades (normalmente todas ellas dentro del mismo país o dominio de red). Estas CA, a las que se hará referencia como CCA (*country certification authorities*) emiten certificados para aquellas entidades con las que están relacionadas. Al nivel más elevado existe una sola CA, llamada GCA (*general certification authority*). La GCA emite certificados para todas las CA. En la Figura 6 se muestra la jerarquía.

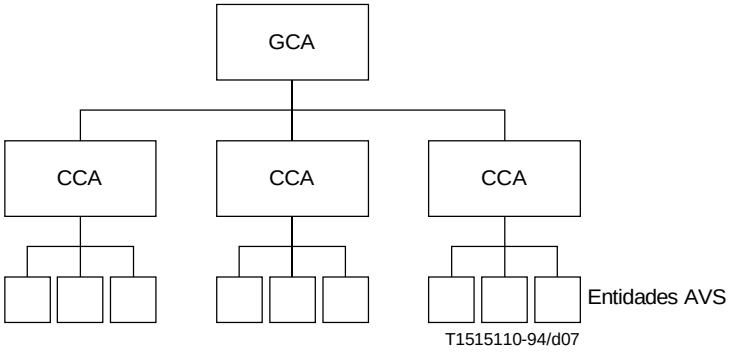


FIGURA 6/H.234

Jerarquía de autoridades de certificación

El marco de autenticación utiliza el algoritmo criptográfico RSA. Se trata de un algoritmo llamado de clave pública de usuario, en el que la clave de criptación difiere de la de descriptación. Una de estas claves puede hacerse pública mientras que la otra permanece secreta. Se les llama *clave pública* y *clave secreta*, respectivamente.

La autenticación utiliza además una función hash (desmenuzar) h(*) que proyecta una secuencia de caracteres de longitud arbitraria en una secuencia de caracteres de longitud limitada, no superior a la del módulo RSA utilizado. La función h(*) no se especifica en la presente Recomendación pero debe ser especificada por la autoridad de certificación. En el Apéndice I [3] se da un ejemplo de dicha función hash disponible públicamente.

5.3 Generación y distribución de claves de autenticación

Una clave de autenticación se compone de un par de claves secreta/pública del algoritmo RSA. Cada CA y cada entidad AVS tiene su propio par de autenticación.

La GCA genera su propia clave de autenticación, formada por una clave secreta GCA y una clave pública GCAp.

Cada CCA genera su propia clave de autenticación, formada por una clave secreta CCA y una clave pública CCAp. La CCA pone la CCAp a disposición de la GCA, la cual certifica esta clave.

La clave de autenticación de una AVSE U, formada por una clave secreta Us y una clave pública Up, es generada por su CCA. La Up y la Us se ponen a disposición de la AVSE. La CCA certifica la Up.

Ha de haber un consenso internacional sobre la generación de la clave de autenticación de la GCA y sobre la generación y distribución de las claves de autenticación de las CCA.

NOTA – La interfaz física entre autoridades de certificación y entidades de AVS queda fuera del alcance de la presente Recomendación.

5.4 Certificación

La GCA certifica una clave pública CCAp calculando un certificado, denotado como GCA <<CCA>>, que consta de la siguiente información:

GCA<<CCA>>: GCA,CCA,CCAp,T1,GCAs[h(GCA,CCA,CCAp,T1)]

donde:

GCA es la identidad de GCA

CCA es la identidad de CCA

CCAp es la clave pública de CCA

T1 es la fecha de comienzo y terminación de la validez del certificado

GCAs[*] es la criptación de * con la clave GCA

NOTA – Se ha incluido la identidad de la GCA a efectos de conformidad con la Recomendación X.509, pero en el sistema descrito la identidad de la GCA se determina de manera exclusiva.

La CCA certifica una clave pública Xp de una AVSE X calculando un certificado, denotado como CCA<<X>>, que consta de la siguiente información:

CCA<<X>>: CCA,X,Xp,T2,CCAs[h(CCA,X,Xp,T2)]

donde:

CCA es la identidad de CCA

X es la identidad de X

Xp es la clave pública de X

T2 es la fecha de comienzo y terminación de la validez del certificado

CCAs[*] es la criptación de * con la clave CCA

GCAp, GCA<<CCA>> y CCA<<X>> se ponen, junto con X, a disposición de la entidad X, por ejemplo, en forma de tarjeta con memoria o módulo incorporado en el soporte físico. X debe tener también una copia impresa de las GCAp que podrá servir como referencia en caso de duda respecto a la integridad de la GCAp.

Verificación de certificados

GCA<<CCA>> puede verificarse mediante el cálculo de h(GCA,CCA,CCAp,T1) utilizando la GCAp y comparándolo con GCAp[GCAs[h(GCA,CCA,CCAp,T1)]]; ambos valores deben ser iguales. CCA<<X>> puede verificarse mediante el cálculo de h(CCA,X,Xp,T2) utilizando la CCAp y comparándolo con CCAp,[CCAs[h(CCA,X,Xp,T2)]]; ambos valores deben ser iguales.

El esquema descrito en 5.4 se presenta de forma resumida en la Figura 7 para las AVSE X e Y con autoridades de certificación CA1 y CA2, respectivamente.

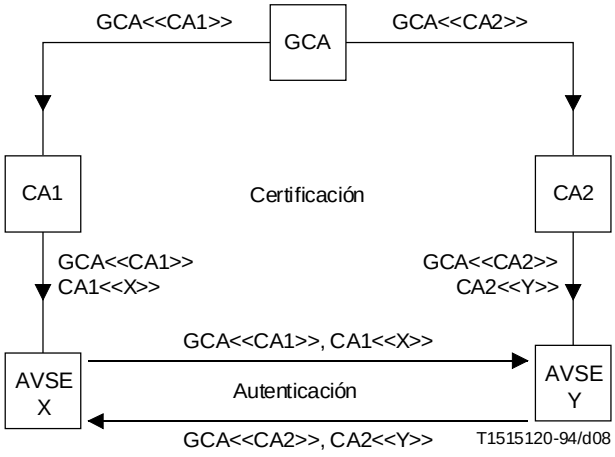


FIGURA 7/H.234

Resumen del procedimiento de certificación

5.5 Solución alternativa para certificación sin una GCA

Si dos operadores de red o compañías desean que sus AVSE se autentiquen mutuamente, sus autoridades de certificación CA1 y CA2 deben certificarse mutuamente intercambiando los certificados CA1<<CA2>> y CA2<<CA1>>. Este sistema funciona de manera compleja ya que las AVSE X e Y podrían tener que entrar en un directorio externo para obtener el CA1<<CA2>> o el CA2<<CA1>> y tener también que intercambiar de antemano las identidades de sus autoridades de certificación. Esto es lo que se detalla en la Figura 8.

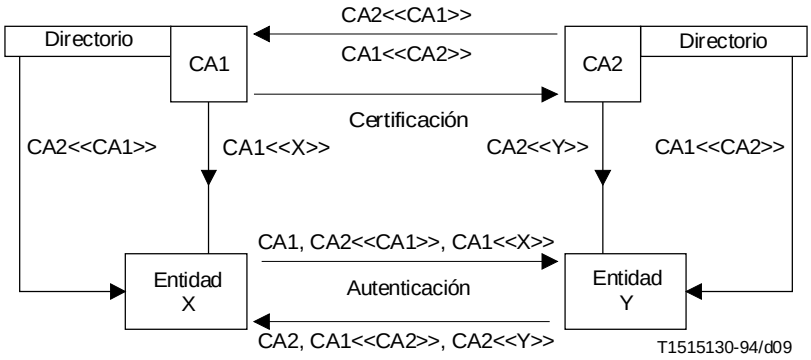


FIGURA 8/H.234

Certificación sin una autoridad de certificación superior

5.6 Autenticación de entidades

A continuación se detalla el procedimiento de autenticación, que es aplicable en todas las conexiones posibles, es decir, MCU-MCU, terminal-MCU, MCU-terminal y terminal-terminal.

El procedimiento de autenticación entre dos entidades en el establecimiento de la comunicación entraña cuatro mensajes:

- RSA.P1 – Iniciación de autenticación;
- RSA.P2 – Respuesta de autenticación;
- RSA.P3 – Autenticación completa;
- RSA.P4 – Autenticación fallada.

RSA.P1 y RSA.P3 son enviados por la entidad iniciadora, denotada por X; RSA.P2, por la entidad llamada, denotada por Y. Las CCA de X e Y se denotan por CX y CY, respectivamente.

El contenido de RSA.P1 es:

GCA<<CX>>,CX<<X>>,RX,Y,Xs[h(RX,Y)]

donde RX es un número aleatorio generado por X.

Y, ahora,

1) obtiene Xp de RSA.P1 y comprueba Xp utilizando los certificados con GCAp como punto de confianza;
2) comprueba la integridad del mensaje calculando h(RX,Y) y comparándolo con Xp[Xs[h(RX,Y)]]; ambos valores deben ser iguales;
3) comprueba las fechas de caducidad de los certificados;
4) comprueba la integridad de X.

El contenido de RSA.P2 es:

GCA<<CY>>,CY<<Y>>,RY,X,RX,Xp[KY],Ys[h(RY,X,RX,KY)]

donde RY es un número aleatorio y KY son datos de clave (véase 2.3.2), generados el uno y los otros por Y.

X, ahora,

1) obtiene Yp de RSA.P2 y comprueba Yp utilizando los certificados con GCAp como punto de confianza;

2) descripta Xp[KY], obteniendo KY;

3) comprueba la integridad del mensaje calculando h(RY,X,RX,KY) y comparándolo con Yp[Ys[h(RY,X,RX,KY)]]; ambos valores deben ser iguales;

4) comprueba las fechas de caducidad de los certificados;

5) comprueba que RX es el mismo que el enviado en RSA.P1;

6) comprueba la integridad de Y.

El contenido de RSA.P3 es:

RY,Y,Yp[KX],Xs[h(RY,Y,KX)],

donde KX son los datos de clave generados por X.

Y, ahora,

1) descripta Yp[KX], obteniendo KX;

2) comprueba la integridad del mensaje calculando h(RY,Y,KX) y comparándolo con Xp[Xs[h(RY,Y,KX)]]; ambos valores deben ser iguales;

3) comprueba que RY es el mismo que el enviado en RSA.P2;

4) comprueba la integridad de X.

Si cualquiera de las comprobaciones efectuadas en RSA.P1, RSA.P2 o RSA.P3 falla, deberá interrumpirse el establecimiento de la comunicación enviando un mensaje RSA.P4: autenticación fallada. RSA.P4 puede ser enviado tanto por X como por Y, y después de RSA.P1, RSA.P2 o RSA.P3. El envío de RSA.P4 deberá invocar la terminación del procedimiento de establecimiento de la comunicación.

NOTAS

1 Es posible acelerar los cálculos de RSA eligiendo parámetros públicos específicos.

2 El presente esquema difiere de la especificación de la Recomendación X.509 original en que KX se envía en el RSA.P3 y no en el RSA.P1. Esto tiene la ventaja de que X no tiene que obtener Yp de un directorio. Tanto para X como para Y, GCAp es el único punto de confianza: mientras se confíe en esta clave y se tenga confianza en que la información secreta de una entidad no va a ser sustraída, X e Y no necesitan acceder a directorios. Además, en el RSA.P3, se añade la identidad de Y por motivos de seguridad y en el RSA.P2 y el RSA.P3, la firma se halla en los datos de clave no criptados KY y KX, respectivamente.

5.6.1 Transmisión simultánea de mensajes RSA.P1

Si la entidad X envía a la entidad Y un mensaje iniciador:

RSA.P1(X–>Y): GCA<<CX>>,CX<<X>>,RX,Y,Xs[h(RX,Y)]

y, antes de recibir RSA.P2(Y->X), Y envía a X un mensaje iniciador:

RSA.P1(Y–>X): GCA<<CY>>,CY<<Y>>,RY,X,Ys[h(RY,X)]

entonces X e Y resolverán esta situación, *comparando RX y RY*.

Si RX>RY, el mensaje RSA.P1(Y–>X) debe ser desechado e Y debe responder con un mensaje RSA.P2.

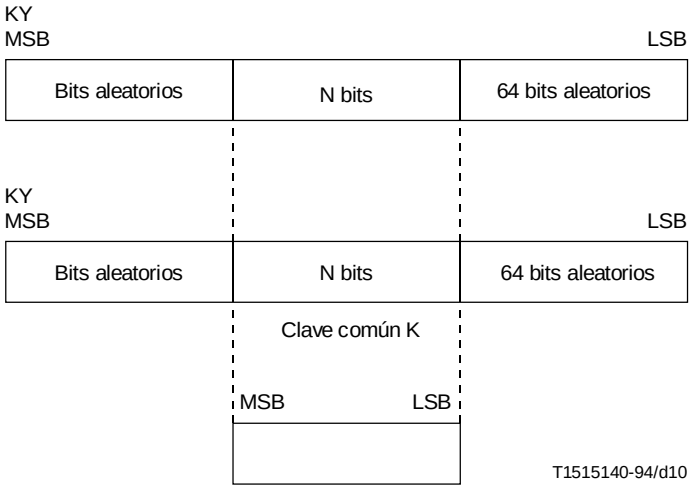
Si RY>RX, el mensaje RSA.P1(X−>Y) debe ser desechado y X debe responder con un mensaje RSA.P2.

Si se da la coincidencia de que RX = RY, deben desecharse ambos mensajes RSA.P1 y terminarse el procedimiento de autenticación con el envío de un mensaje RSA.P4 (fallo de la autenticación).

5.7 Generación de una clave para la criptación de claves de sesión

Los datos de clave KY y KX transmitidos en los mensajes RSA.P2 y RSA.P3 se utilizarán para establecer una *clave* común K, que se utilizará para criptar los mensajes de intercambio de claves de sesión, como se describe en 2.3.2. (De estos mensajes se obtiene un conjunto de cuatro claves de sesión). Si N representa la longitud de K, la clave K se forma tomando la suma en módulo 2 de los bits 64 a 64 + N – 1 de KX y de 64 a 64 + N – 1 de KY (el bit cero indica aquí el bit menos significativo de KX y KY). El bit 64 de KX y el bit 64 de KY generan juntos el bit 0 de K. El valor de N es la longitud de la *clave* y lo determina el algoritmo de criptación que ha de utilizarse.

Los bits no utilizados de KX y KY (índice 0 a 63 y 64 + N y superior) deben rellenarse con información aleatoria. La generación de la clave K común a partir de KX y KY se muestra en forma de diagrama en la Figura 9.



NOTA – Bloques de N bits de KX y KY se suman en módulo 2 para formar la clave común K.

FIGURA 9/H.234

Generación de una *clave* común

5.8 Mensajes RSA

En esta subcláusula se detalla el contenido de los mensajes requeridos en el esquema de autenticación basado en RSA que se describe en 5.6. Las descripciones se basan en la Recomendación UIT-T X.209. En 2.2 se han dado descripciones breves de algunas de las definiciones de la Recomendación X.209 utilizadas en este punto.

5.8.1 Iniciación de autenticación

Nombre del mensaje:	Iniciación de autenticación (RSA.P1).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100111
Significado:	El emisor de estos mensajes desea comenzar un procedimiento de autenticación con el receptor pretendido y envía la información necesaria para empezar el procedimiento.
Contenido:	Un constructor, que consta de dos constructores para los certificados GCA<<CX>> y CX<<X>> y tres primitivas para un número aleatorio RX, una identidad de Y y la información hashed (desmenuzada) criptada Xs[h(RX,Y)].
Notación en ASN.1:	RSA.P1 ::= [7] IMPLICIT SEQUENCE { GCA-certificate-for-CCA [0] IMPLICIT GCA-Certificate, CCA-certificate-for-entity [1] IMPLICIT CCA-Certificate, calling-entity-random-number [2] IMPLICIT BIT STRING, called-entity-identity [3] IMPLICIT BIT STRING, hashed-information-in-calling-secret-key [4] IMPLICIT BIT STRING }

El contenido de Calling-Entity-Random-number (número aleatorio de entidad que llama) es una cadena de bits primitiva.

El contenido de Called-Entity-Identity (identidad de la entidad llamada) es una cadena de bits primitiva.

El contenido de Hashed-Information-In-Calling-Secret-Key (información desmenuzada en clave secreta de llamante) es una cadena de bits primitiva.

Contenido de GCA-Certificate-For-CCA (certificado de GCA para CCA): un constructor, que consta de cinco primitivas para una identidad de GCA, una identidad de CCA, una clave pública CCAp, una gama de fechas de validez T1 y la información hashed (desmenuzada) criptada GCAs[h(GCA,CCA,CCAp,T1)].

En notación ASN.1:

GCA-Certificate ::= SEQUENCE{ GCA-identity [0] IMPLICIT BIT STRING,

CCA-identity [1] IMPLICIT BIT STRING,

CCA-public-key [2] IMPLICIT BIT STRING,

certificate-valid-date-range [3] IMPLICIT BIT STRING,

hashed-information-in-GCA-secret-key [4] IMPLICIT BIT STRING}

El contenido de GCA-Identity (identidad de GCA) es una cadena de bits primitiva.

El contenido de CCA-Identity (identidad de CCA) es una cadena de bits primitiva.

El contenido de CCA-Public-Key (clave pública de CCA) es una cadena de bits primitiva.

El contenido de Certificate-Valid-Date-Range (gama de fechas de validez de certificados) es una cadena de bits primitiva.

El contenido de Hashed-Information-In-GCA-Secret-Key (información desmenuzada en clave secreta de GCA) es una cadena de bits primitiva.

Contenido de CCA-Certificate-For-Entity (certificado de CCA para entidad): un constructor, que consta de cinco primitivas para una identidad de CCA, una identidad de entidad X, una clave pública Xp, una gama de fechas de validez T2 y la información hashed (desmenuzada) criptada CCAs[h(CCA,X,Xp,T2)].

En notación ASN.1:

```
CCA-Certificate ::= SEQUENCE{ CCA-identity [0] IMPLICIT BIT STRING,

                              entity-identity [1] IMPLICIT BIT STRING,

                              entity-public-key [2] IMPLICIT BIT STRING,

                              certificate-valid-date-range [3] IMPLICIT BIT STRING,

                              hashed-information-in-CCA-secret-key [4] IMPLICIT BIT STRING}
```

El contenido de CCA-Identity (identidad de CCA) es una cadena de bits primitiva.

El contenido de Entity-Identity (identidad de entidad) es una cadena de bits primitiva.

El contenido de Entity-Public-Key (clave pública de entidad) es una cadena de bit primitiva.

El contenido de Certificate-Valid-Date-Range (gama de fechas de validez de certificados) es una cadena de bits primitiva.

El contenido de Hashed-Information-In-CCA-Secret-Key (información desmenuzada en clave secreta de CCA) es una cadena de bits primitiva.

5.8.2 Respuesta de autenticación

Nombre del mensaje:	Respuesta de autenticación (RSA.P2).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10101000
Significado:	El emisor de este mensaje responde a una iniciación de autenticación y envía la información necesaria para el procedimiento de autenticación.
Contenido:	Un constructor, que consta de dos constructores para los certificados GCA<<CY>> y CY<<Y>> y cinco primitivas para un número aleatorio RY, una entidad de X, un número aleatorio RX, la información de clave criptada Xp[KY] y la información hashed (desmenuzada) criptada Ys[h(RY,X,RX,KY)].
Notación en ASN.1:	RSA.P2 ::= [8] IMPLICIT SEQUENCE { GCA-certificate-for-CCA [0] IMPLICIT GCA-Certificate, CCA-certificate-for-entity [1] IMPLICIT CCA-Certificate, called-entity-random-number [2] IMPLICIT BIT STRING, calling-entity-identity [3] IMPLICIT BIT STRING, calling-entity-random-number [4] IMPLICIT BIT STRING, key-information-in-calling-public-key [5] IMPLICIT BIT STRING, hashed-information-in-called-secret-key [6] IMPLICIT BIT STRING }

El contenido de Called-Entity-Random-Number (número aleatorio de entidad llamada) es una cadena de bits primitiva.

El contenido de Calling-Entity-Identity (identidad de entidad llamante) es una cadena de bits primitiva.

El contenido de Calling-Entity-Random-Number (número aleatorio de entidad que llama) es una cadena de bits primitiva.

El contenido de Key-Information-In-Calling-Public-Key (información de clave en clave pública llamante) es una cadena de bits primitiva.

El contenido de Hashed-Information-In-Called-Secret-Key (información desmenuzada en clave secreta llamada) es una cadena de bits primitiva.

Los contenidos de GCA-Certificate-For-CCA (certificado de CCA para CCA) y CCA-Certificate-For-Entity (certificado de CCA para entidad) son similares a las descripciones dadas en 5.8.1.

5.8.3 Autenticación completa

Nombre del mensaje:	Autenticación completa (RSA.P3).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10101001
Significado:	El emisor de este mensaje, que es el iniciador del procedimiento de autenticación, envía la información necesaria para completar el procedimiento de autenticación.
Contenido:	Un constructor, que consta de cuatro primitivas para un número aleatorio RY, una entidad de Y, la información de clave criptada Yp[KX] y la información hashed (desmenuzada) criptada Xs[h(RY,Y,KX)].
Notación en ASN.1:	RSA.P3 ::= [9] IMPLICIT SEQUENCE { called-entity-random-number [0] IMPLICIT BIT STRING, called-entity-identity [1] IMPLICIT BIT STRING, key-information-in-called-public-key [2] IMPLICIT BIT STRING, hashed-information-in-calling-secret-key [3] IMPLICIT BIT STRING }

El contenido de Called-Entity-Random-Number (número aleatorio de entidad llamada) es una cadena de bits primitiva.

El contenido de Called-Entity-Identity (identidad de entidad llamada) es una cadena de bits primitiva.

El contenido de Key-Information-In-Called-Public-Key (información de clave en clave pública llamada) es una cadena de bits primitiva.

El contenido de Hashed-Information-In-Calling-Secret-Key (información desmenuzada en clave secreta llamante) es una cadena de bits primitiva.

5.8.4 Autenticación fallada

Nombre del mensaje:	Autenticación fallada (RSA.P4).
Identificador del mensaje:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10001010
Significado:	El emisor de este mensaje indica que algo ha fallado durante el procedimiento de autenticación y que se va a terminar dicho procedimiento. El envío o la recepción de este mensaje debe invocar la terminación del procedimiento de establecimiento de la comunicación.
Contenido:	Este mensaje no tiene contenido.

6 Operación de MCU

En el caso de una «MCU encargada» (en que las señales son todas ellas descriptadas en las entradas a la MCU y, por consiguiente, la MCU debe estar en un sitio seguro), las comunicaciones entre cada terminal audiovisual y la MCU pueden ser criptadas como se describe en la presente Recomendación para un enlace punto a punto. Este método no es aplicable, evidentemente, a la conexión de terminales telefónicos a la conferencia por conducto de la red telefónica analógica.

No está previsto en esta Recomendación el funcionamiento de una MCU sin esa descriptación.

7 Referencias normativas

- ISO 8732, *Banking Key Management*.
- Recomendación UIT-T X.209, *Especificación de las reglas básicas de codificación de la notación de sintaxis abstracta uno*.
- Recomendación UIT-T H.233, *Sistemas con confidencialidad para servicios audiovisuales*.

– Recomendación UIT-T H.221, *Estructura de trama para un canal de 64 a 1920 kbit/s en teleservicios audiovisuales*.

– Recomendación UIT-T H.230, *Señales de control e indicación con sincronismo de trama para sistemas audiovisuales*.

– Recomendación UIT-T H.242, *Sistema de establecimiento de comunicación entre terminales audiovisuales por canales digitales de hasta 2 Mbit/s*.

– Recomendación UIT-T X.509, *La guía-marco de autenticación*.

Apéndice

Bibliografía

[1] DIFFIE (W.) y HELLMAN (M.): New directions in cryptography, *IEEE Transactions IT-22*, 6, pp. 644-654, (noviembre de 1976).

[2] RIVEST (R.L.), SHAMIR (A.) y ADLEMAN (L.): A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *Communications of the ACM*, 21, 2, pp. 120-126, (febrero de 1978).

[3] The MD4 Message Digest Algorithm, *RSA Data Security Inc*., Redwood City, California 94065.