UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T Q.814

(02/2000)

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

SERIE Q: CONMUTACIÓN Y SEÑALIZACIÓN

Especificaciones del sistema de señalización N.° 7 – Interfaz Q3

Especificación de un agente interactivo de intercambio electrónico de datos

Recomendación UIT-T Q.814

(Anteriormente Recomendación del CCITT)

RECOMENDACIONES UIT-T DE LA SERIE Q

CONMUTACIÓN Y SEÑALIZACIÓN

SEÑALIZACIÓN EN EL SERVICIO MANUAL INTERNACIONAL	Q.1–Q.3
EXPLOTACIÓN INTERNACIONAL SEMIAUTOMÁTICA Y AUTOMÁTICA	Q.4–Q.59
FUNCIONES Y FLUJOS DE INFORMACIÓN PARA SERVICIOS DE LA RDSI	Q.60–Q.99
CLÁUSULAS APLICABLES A TODOS LOS SISTEMAS NORMALIZADOS DEL UIT-T	Q.100–Q.119
ESPECIFICACIONES DE LOS SISTEMAS DE SEÑALIZACIÓN N.° 4 Y N.° 5	Q.120–Q.249
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN N.° 6	Q.250–Q.309
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN R1	Q.310–Q.399
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN R2	Q.400–Q.499
CENTRALES DIGITALES	Q.500–Q.599
INTERFUNCIONAMIENTO DE LOS SISTEMAS DE SEÑALIZACIÓN	Q.600–Q.699
ESPECIFICACIONES DEL SISTEMA DE SEÑALIZACIÓN N.° 7	Q.700–Q.849
Generalidades	Q.700
Parte transferencia de mensajes	Q.701–Q.709
Parte control de la conexión de señalización	Q.711–Q.719
Parte usuario de telefonía	Q.720–Q.729
Servicios suplementarios de la RDSI	Q.730–Q.739
Parte usuario de datos	Q.740–Q.749
Gestión del sistema de señalización N.° 7	Q.750–Q.759
Parte usuario de la RDSI	Q.760–Q.769
Parte aplicación de capacidades de transacción	Q.770–Q.779
Especificaciones de las pruebas	Q.780–Q.799
Interfaz Q3	**Q.800–Q.849**
SISTEMA DE SEÑALIZACIÓN DIGITAL DE ABONADO N.° 1	Q.850–Q.999
RED MÓVIL TERRESTRE PÚBLICA	Q.1000–Q.1099
INTERFUNCIONAMIENTO CON SISTEMAS MÓVILES POR SATÉLITE	Q.1100–Q.1199
RED INTELIGENTE	Q.1200–Q.1699
REQUISITOS Y PROTOCOLOS DE SEÑALIZACIÓN PARA IMT-2000	Q.1700–Q.1799
RED DIGITAL DE SERVICIOS INTEGRADOS DE BANDA ANCHA (RDSI-BA)	Q.2000–Q.2999

Para más información, véase la Lista de Recomendaciones del UIT-T.

Recomendación UIT-T Q.814

Especificación de un agente interactivo de intercambio electrónico de datos

Resumen

Esta Recomendación UIT-T presenta la especificación técnica de un módulo de protocolo de capa de sesión denominado agente interactivo de comunicaciones electrónicas. Puede utilizarse como un punto de referencia de interfaz en un modelo de red de gestión de las telecomunicaciones para el intercambio asíncrono de datos entre entidades de aplicación pares. El agente interactivo (IA) soporta el intercambio de transacciones de intercambio de datos electrónicos en tiempo cuasi real (EDIFACT o ASC X12 EDI). Además, esta Recomendación UIT-T define la arquitectura, el diseño, la estructura, y el proceso-flujo de las funciones comerciales de *prioridad normal*[1] y *alta prioridad*[2] utilizando la seguridad de la capa de transporte (TLS).

Orígenes

La Recomendación UIT-T Q.814, preparada por la Comisión de Estudio 4 (1997-2000) del UIT-T, fue aprobada por el procedimiento de la Resolución 1 de la CMNT el 4 de febrero de 2000.

---

[1] *Prioridad normal* – Un ejemplo de función comercial de prioridad normal es una transacción de solicitud de un pedido.

[2] *Alta prioridad* – Un ejemplo de función comercial de alta prioridad es una transacción de interrogación interactiva.

PREFACIO

La UIT (Unión Internacional de Telecomunicaciones) es el organismo especializado de las Naciones Unidas en el campo de las telecomunicaciones. El UIT-T (Sector de Normalización de las Telecomunicaciones de la UIT) es un órgano permanente de la 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 1 de la CMNT.

En ciertos sectores de la tecnología de la información que corresponden a la esfera de competencia del UIT-T, se preparan las normas necesarias en colaboración con la ISO y la CEI.

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.

PROPIEDAD INTELECTUAL

La UIT señala a la atención la posibilidad de que la utilización o aplicación de la presente Recomendación suponga el empleo de un derecho de propiedad intelectual reivindicado. La UIT no adopta ninguna posición en cuanto a la demostración, validez o aplicabilidad de los derechos de propiedad intelectual reivindicados, ya sea por los miembros de la UIT o por terceros ajenos al proceso de elaboración de Recomendaciones.

En la fecha de aprobación de la presente Recomendación, la UIT no ha recibido notificación de propiedad intelectual, protegida por patente, que puede ser necesaria para aplicar esta Recomendación. Sin embargo, debe señalarse a los usuarios que puede que esta información no se encuentre totalmente actualizada al respecto, por lo que se les insta encarecidamente a consultar la base de datos sobre patentes de la TSB.

ÍNDICE

Introducción

Esta Recomendación UIT-T define las especificaciones de un agente interactivo (IA) de intercambio electrónico de datos. El IA sustenta el intercambio de transacciones de intercambio electrónico de datos entre entidades pares. Hace corresponder las transacciones EDI con la capa de transporte. Más concretamente, se pone en interfaz con la seguridad de la capa de transporte (TLS) para solicitar el establecimiento y la terminación de sesiones TCP seguras (es decir, autenticación de entidad par soporte, integridad y privacidad) y transporte seguro de mensajes EDI. El IA proporciona además funcionalidad de control de flujo básico.

Recomendación UIT-T Q.814

Especificación de un agente interactivo de intercambio electrónico de datos

1 Alcance

Esta Recomendación UIT-T presenta una especificación del agente interactivo (IA) de intercambio electrónico de datos. El IA soporta el intercambio de transacciones de intercambio electrónico de datos (EDIFACT/ASC X12 EDI) por una de red protocolo de control de transmisión/protocolo Internet (TCP/IP) utilizando seguridad de la capa de transporte (TLS). Esta Recomendación UIT-T especifica la arquitectura general del IA, la sintaxis de los formatos de mensaje a utilizar, las reglas de codificación de los mensajes y las transformaciones de seguridad aplicables.

2 Referencias

Las siguientes Recomendaciones del UIT-T y otras referencias contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones y otras referencias son objeto de revisiones por lo que se preconiza que los usuarios de esta Recomendación investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones y otras referencias citadas a continuación. Se publica periódicamente una lista de las Recomendaciones UIT-T actualmente vigentes.

2.1 Referencias normativas

– Recomendación UIT-T Q.815 (2000), *Especificación de un modelo de seguridad para la protección del mensaje completo*.

– Recomendación UIT-T X.509 (1997) | ISO/CEI 9495-8:1998, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Marco de autenticación*.

– Recomendación UIT-T X.680 (1997) | ISO/CEI 8824-1:1998, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica*.

– Recomendación UIT-T X.681 (1997) | ISO/CEI 8824-2:1998, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de objetos de información*.

– Recomendación UIT-T X.682 (1997) | ISO/CEI 8824-3:1998, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de constricciones*.

– Recomendación UIT-T X.683 (1997) | ISO/CEI 8824-4:1998, *Tecnología de la información – Notación de sintaxis abstracta uno: Parametrización de especificaciones de notación de sintaxis abstracta uno*.

– Recomendación UIT-T X.690 (1997) | ISO/CEI 8825-1:1998, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Especificación de las reglas de codificación básicas, de las reglas de codificación canónica y de las reglas de codificación distinguida*.

Internet Society/Internet Engineering Task Force:

– RFC 2246, *The TLS Protocol Version 1.0.*

2.2 Referencias informativas

– Directory Implementors Guide (Version 12) (1999).

3 Definiciones

En esta Recomendación UIT-T se definen los términos siguientes.

3.1 protocolo de transferencia de agente interactivo (IATP, *interactive agent transfer protocol*): Este protocolo se utiliza entre agentes interactivos pares que desean intercambiar transacciones/mensajes de intercambio electrónico de datos vía protocolo de control de transmisión/protocolo Internet utilizando seguridad de capa de transporte.

3.2 traductor de intercambio electrónico de datos (EDI, *electronic data interchange*): Un traductor EDI suele ser un módulo o programa de soporte lógico que traduce formatos y representaciones de datos privados hacia/desde formatos normalizados y representaciones de datos normalizadas tales como las especificadas por ISO 9735 o ANSI ASC X.12.

3.3 agente interactivo (IA, *interactive agent*): El IA soporta el intercambio de transacciones de intercambio electrónico de datos (UN/EDIFACT o ASC X12 EDI) entre entidades pares. El IA funciona como una interfaz entre su usuario directo (normalmente un traductor EDIFACT/ASC X12 EDI o un módulo de seguridad) y la seguridad de la capa de transporte. Pueden adoptarse diversos criterios de implementación, desde una simple interfaz de programa de aplicación (API) hasta un programa autónomo. El IA se describe en esta Recomendación UIT-T y el módulo de seguridad en la Recomendación UIT-T Q.815.

3.4 seguridad de la capa de transporte (TLS, *transport layer security*): El protocolo TLS proporciona opcionalmente privacidad en las comunicaciones. El protocolo permite aplicaciones de cliente/servidor para comunicar de manera concebida para evitar escucha indiscreta, maniobras fraudulentas e intrusión. El protocolo TLS también proporciona autenticación de pares rigurosa e integridad del flujo de datos.

4 Abreviaturas

En esta Recomendación UIT-T se utilizan las siguientes siglas.

IA Agente interactivo (*interactive agent*)

IATP Protocolo de transferencia de agente interactivo (*interactive agent transfer protocol*)

MD Digesto del mensaje (*message digest*)

SHA1 Algoritmo de troceo seguro, revisión 1 (*secure hashing algorithm, revision 1*)

SM Módulo de seguridad (*security module*)

TLS Seguridad de la capa de transporte (*transport layer security*)

WAN Red de área extensa (*wide area network*)

5 Convenios

En esta Recomendación UIT-T se utilizan los siguientes convenios:

El término *EDI*, utilizado en esta Recomendación UIT-T, designa parte o la totalidad de lo siguiente:

- UN/EDIFACT definido por la UN/ECE Trade Division y adoptado por ISO/TC 154.
- EDIFACT definido por ISO 9735.

NOTA – También incluye el EDI definido por ANSI ASC X12.

El cuadro 1 de 6.2.2 utiliza los siguientes convenios:

M Obligatorio (*mandatory*)

O Opcional

Todas las veces que aparece el código *C Language* en esta Recomendación UIT-T es exclusivamente con fines informativos.

6 Arquitectura y características del servicio

6.1 Arquitectura

Las funciones IA como interfaz entre su usuario directo (normalmente una aplicación tal como traductor EDIFACT o ASC X12 EDI) y la capa de transporte (véase la figura 1). La seguridad básica de las transacciones EDI la proporciona la TLS. Las capacidades de seguridad adicionales (por ejemplo, no repudio) pueden ser proporcionadas por un módulo de seguridad separado que efectúe transformaciones de seguridad en los mensajes EDI completos. Dicho módulo de seguridad puede también ser un usuario directo del IA, como se ilustra en la figura 2.

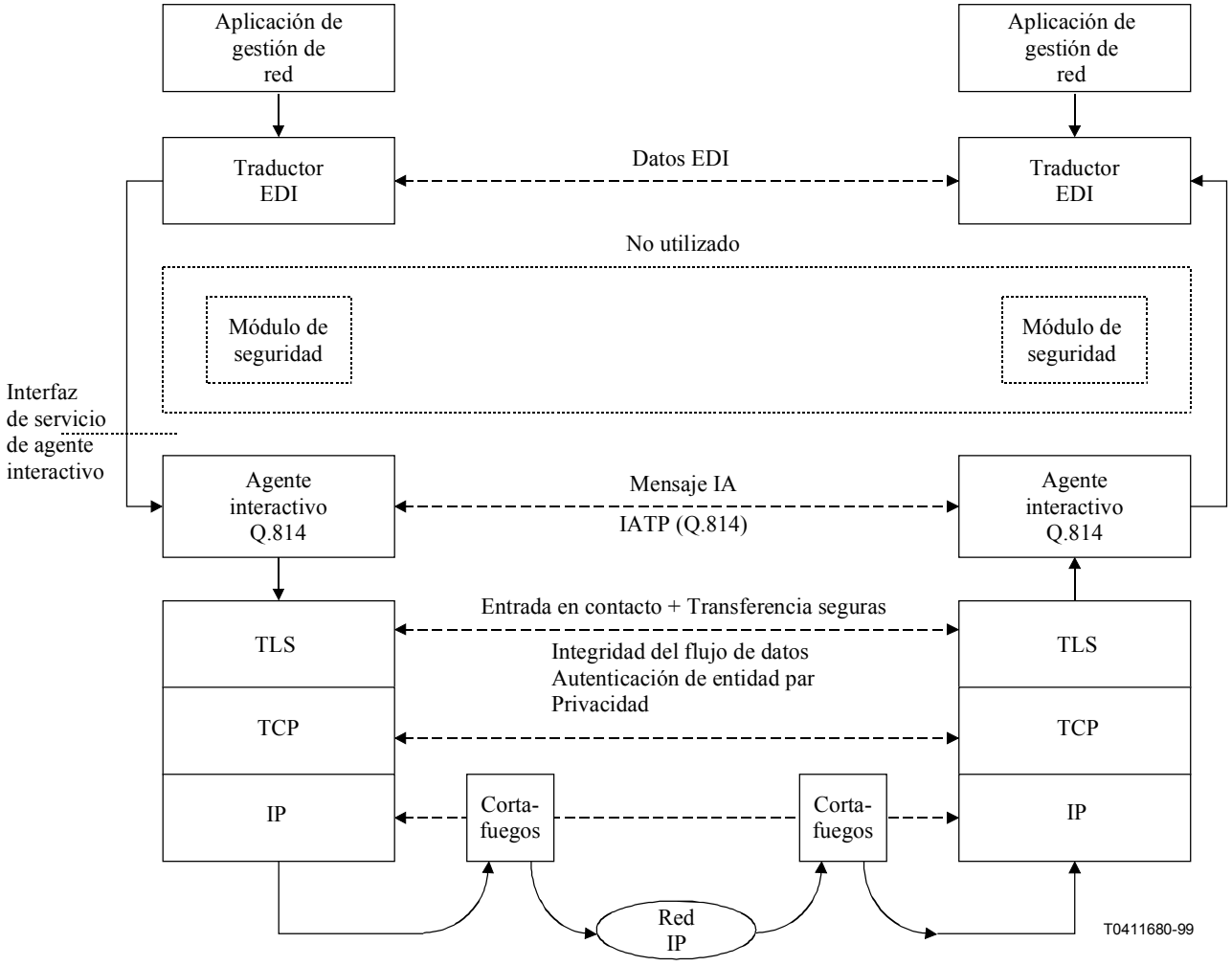

Figura 1/Q.814 – Relación de los flujos de mensajes (sin módulo de seguridad)

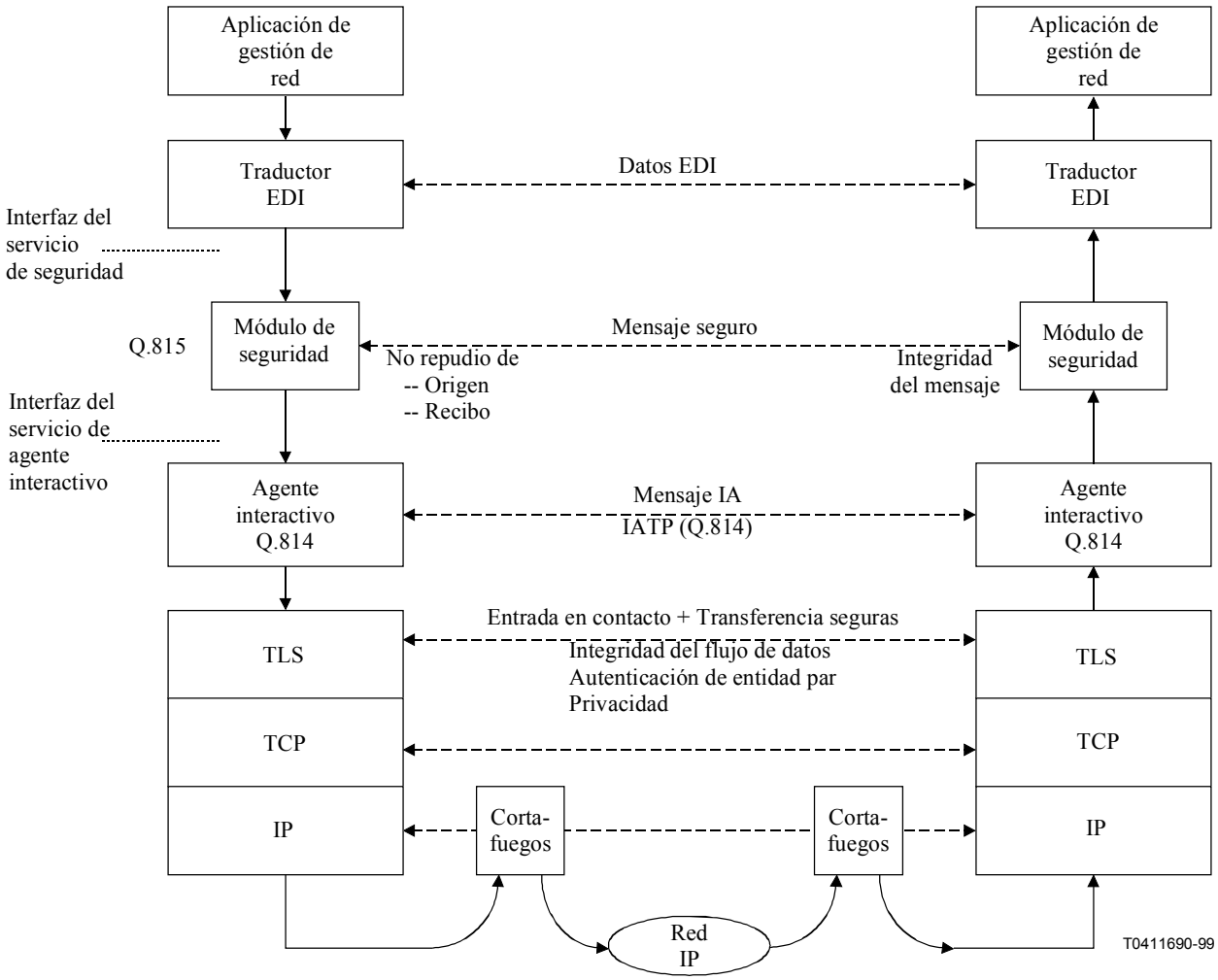

Figura 2/Q.814 – Relación de los flujos de mensajes (con servicios de seguridad del mensaje)

La estructura subyacente del IA es una configuración cliente/servidor simétrica en la que las funciones de cliente y de servidor se requieren en cada implementación. Véase la figura 3.

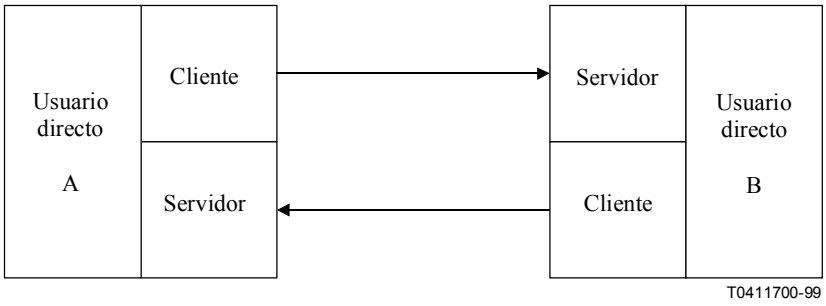

Figura 3/Q.814 – Arquitectura de agente interactivo

NOTA – La arquitectura cliente/servidor simétrica puede no ser apropiada para algunos tipos de aplicaciones en *tiempo cuasi real*.

El IA efectúa las funciones de capa de sesión del modelo de siete capas de OSI. Entre las funciones de la capa de sesión proporcionadas por el IA se hallan el establecimiento, gestión y cierre de sesiones de comunicaciones entre entidades pares. El IA puede también efectuar la conversión de identidades EDIFACT/ASC X12 EDI en direcciones de red y gestionar la sesión de la capa de transporte. Al concluir una sesión, el IA determinará si se cierra una sesión o se la suspende en un estado en el que pueda ser *reanudada*.

6.2 Características de los servicios

El IA puede exponer a su usuario directo de los IA pares los siguientes servicios:

- nombre del destinatario del IA,
- prioridad del IA,
- mensaje básico del IA,
- mensaje mejorado del IA,
- control del IA,
- registro cronológico del IA.

6.2.1 Elementos del servicio

6.2.1.1 Nombre del destinatario del IA

El usuario directo suministra al IA la identidad del destinatario del mensaje que se transfiere (por ejemplo, identidad del asociado comercial en un mensaje ASC X12 EDI). El IA convertirá este valor en una dirección de capa de transporte.

6.2.1.2 Prioridad del IA

El usuario directo suministra al IA un indicador de la prioridad del mensaje. El IA admite mensajes de prioridad *normal* y *alta*. El indicador de prioridad se convierte en una dirección de puerto de destino y se combina con la dirección de capa de transporte para su utilización al establecer una petición de conexión de capa de sesión.

6.2.1.3 Mensaje básico del IA

El usuario directo suministra el contenido de un mensaje que contiene datos no mejorados al IA.

6.2.1.4 Control del IA

El usuario directo solicita al IA que transmita información de control de sesión a la entidad par.

6.2.1.5 Mensaje mejorado del IA

El usuario directo suministra el contenido de un mensaje que contiene datos al IA.

6.2.1.6 Registro cronológico (asunto local)

El usuario directo puede especificar el mecanismo, las clasificaciones y el contenido de los datos que hay que registrar.

6.2.2 Clasificaciones de los elementos del servicio

Véase el cuadro 1.

Cuadro 1/Q.814 – Clasificación de los elementos del servicio

Clase de servicio	Origen	Recepción
Nombre del destinatario del IA	M	M
Prioridad del IA	O	M
Mensaje básico del IA	O[a)]	M
Mensaje mejorado del IA	O[a)]	M
Control del IA	O	M
Registro cronológico del IA (asunto local)	O	O
[a)] Se elegirá al menos uno.		

7 Flujo de datos

Esta cláusula ilustra el establecimiento de una sesión entre un cliente y un servidor. La figura 4 muestra el establecimiento de una sesión TLS ordinaria. La figura 5 muestra el establecimiento de una sesión abreviada, conocida también como sesión "reanudada".

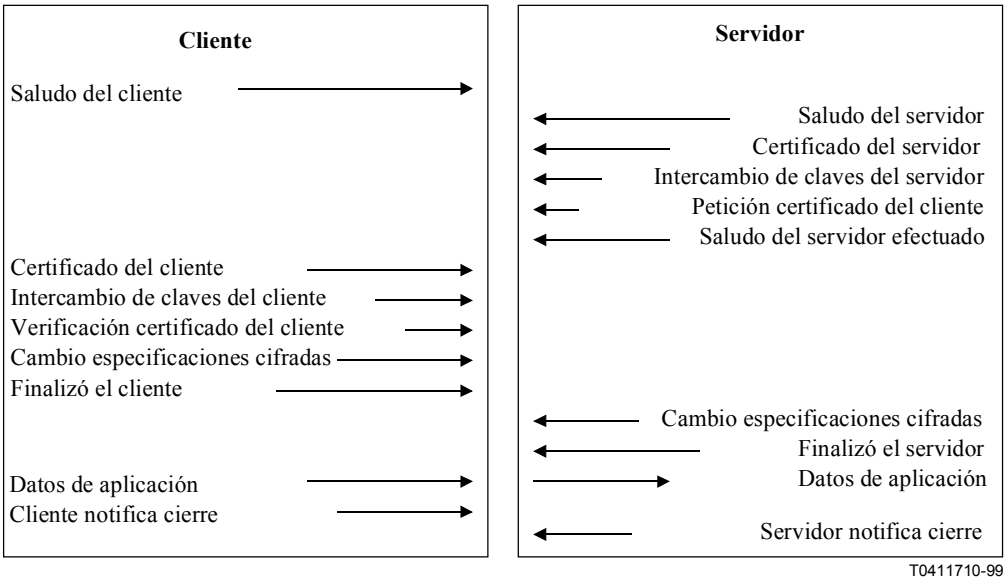

Figura 4/Q.814 – Flujo de mensajes IA – Entrada en contacto TSL normal

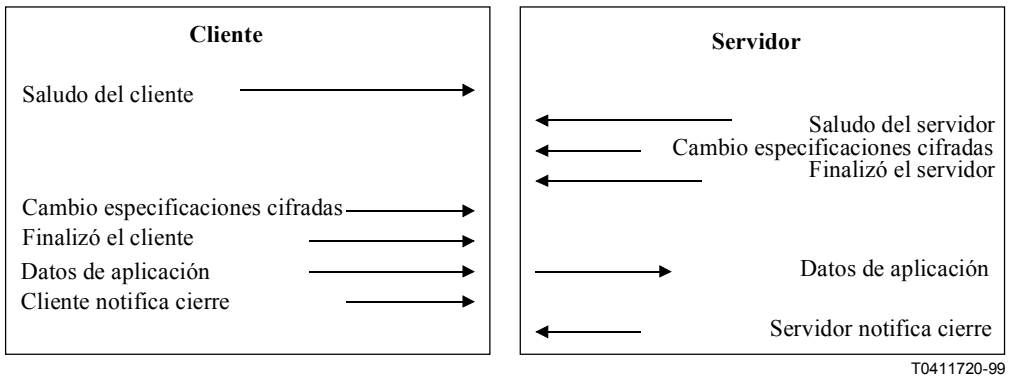

Figura 5/Q.814 – Flujo de mensajes IA – Entrada en contacto TSL abreviada

8 Mensajes IA

A fin de satisfacer las necesidades comerciales y de seguridad de una entidad, se han definido dos tipos de mensajes, que son:

- Mensaje básico (integridad del flujo de datos y/o encriptación de privacidad)
- Mensaje mejorado (integridad del mensaje completo o no repudio)

8.1 Definiciones de formato del mensaje

Los distintos requisitos de formatación del mensaje están asociados con cada uno de los dos formatos de mensaje antes definidos. Además, se ha definido un tercer formato para la comunicación de situación/control entre los agentes interactivos, a saber el mensaje de situación del IA. Cada uno de éstos se codifica de acuerdo con la correspondiente definición de notación de sintaxis abstracta uno (ASN.1).

Se utilizarán las reglas de codificación distinguida (DER) de la ASN.1 para codificar los mensajes IA. Sólo se utilizará el método de longitud definida de codificar octetos de longitud.

8.2 Definiciones de sintaxis del mensaje

El agente interactivo puede ser utilizado para transportar EDIFACT, ASC X12 EDI y/o otra forma de datos de cadena de caracteres. Las definiciones ASN.1 para cada uno de los tipos de mensajes admitidos puede verse en esta subcláusula.

La sintaxis general es la siguiente:

```
IaMessage ::= CHOICE {
        basicMessage        BasicMessage,        -- Basic Message
        iaStatusMessage     IaStatusMessage,     -- IA Status/Control Message
        enhancedMessage     EnhancedMessage      -- Security Module Enhanced Message
}
```

8.2.1 Mensaje básico

El mensaje básico se elige de manera que transporte el texto de mensajes IA en uno de dos repertorios de caracteres.

```
BasicMessage ::= CHOICE {
basicMessage1     GeneralString,
basicMessage2      IA5String
}
```

8.2.2 Mensaje de situación/control del IA

El mensaje de situación puede utilizarse para intercambiar control de flujo o condiciones de error entre los IA (véase 8.3).

IaStatusMessage ::= BIT STRING (SIZE (32))

8.2.3 Mensaje mejorado

La sintaxis de mensaje mejorado se elige de manera que transporte transparentemente mensajes que han sido codificados por una función de seguridad de capa superior. Normalmente este servicio se proporcionará con arreglo a la Recomendación UIT-T Q.815.

EnhancedMessage ::= OCTET STRING -- *Security Module Enhanced Message*

8.3 Formato detallado del mensaje de situación del IA

El mensaje de situación del IA consta de cuatro octetos que contienen información de situación. En el cuadro 2 puede verse un resumen de los valores del mensaje de situación del IA. Los cuatro octetos que transportan la situación se estructuran como sigue:

8.3.1 Primer octeto

Utilizado para condiciones de situación correspondientes a la interfaz de área extensa del IA (es decir, TCP/IP y TLS).

Estructura de codificación

Este octeto debe interpretarse como dos valores hexadecimales de la gama 00 a FF.

Los valores de este octeto deben definirse únicamente en la presente Recomendación UIT-T.

Los valores actualmente definidos son:

00 = Ignorado

08 = Petición de que el par cese las transmisiones en el tren de datos de entrada debido a problemas con la interfaz WAN

Todos los demás valores están actualmente sin definir.

8.3.2 Segundo octeto

Utilizado para condiciones de situación correspondientes a los sistemas descendentes en la interfaz local (es decir, el traductor EDI u otro correspondiente, reciba al usuario directo del IA).

Estructura de codificación

Este octeto debe interpretarse como dos valores hexadecimales de la gama 00 a FF.

Los valores de este octeto deben definirse únicamente en la presente Recomendación UIT-T.

Los valores actualmente definidos son:

00 = Ignorado

08 = Petición de que el par cese las transmisiones en el tren de datos de entrada debido a problemas con el IA que comunica con el traductor EDI u otros sistemas de procesamiento descendente

Todos los demás valores están actualmente sin definir.

8.3.3 Tercero y cuarto octetos

Cuando los octetos 1, 2, 3 y 4 son todos ceros, significará "mensaje de prueba".

Cuando los octetos 1 y 2 son todos ceros, los octetos 3 y 4 están disponibles para que las entidades pares los definan. Las entidades pares pueden definir los valores de estos octetos en cualquier manera que decidan. Se recomienda que los valores se utilicen como dos valores hexadecimales por octeto.

Cuando el octeto 1 o el 2 es distinto de cero, los valores de los octetos 3 y 4 se reservan para normalización posterior.

8.3.4 Mensaje de prueba especial

Un mensaje que contiene cuatro octetos de 00 00 00 00 hexadecimales se define como un mensaje NULO que puede utilizarse como mensaje de prueba. No se emprenderán acciones en respuesta al recibo de dicho mensaje.

8.3.5 Mensaje inválido

El siguiente mensaje es inválido y será ignorado si se recibe:

08 08 xx xx

Cuadro 2/Q.814 – Valores de mensaje de situación del IA

Byte 1	Byte 2	Byte 3	Byte 4	Utilización
00	00	00	00	Mensaje de prueba
08	00	00	00	Congestión capas inferiores
00	08	00	00	Congestión capas superiores
08	00	xx	xx	Reservado uso futuro
00	08	xx	xx	Reservado uso futuro
00	00	yy	yy	Definido por el usuario
08	08	xx	xx	Mensaje inválido ignorado
yy	00	xx	xx	Reservado
00	yy	xx	xx	Reservado
NOTA 1 – xx = 00-FF. NOTA 2 – yy = 01-FF.				

9 Especificaciones del cliente

Esta cláusula define el procesamiento asociado con el proceso de cliente de agente interactivo.

Al recibo de datos de su usuario directo a través de la interfaz del servicio IA (por ejemplo, el traductor EDIFACT/ASC X12 EDI o el módulo de seguridad), el cliente efectúa los siguientes pasos. Sin embargo, antes de que pueda seguirse este proceso, el IA determina si se ha recibido una petición de cese de transmisión pendiente; véase 11.3.

9.1 Determinar dirección de destino IP

Basándose en la entidad de la entidad par y otra información pertinente, se determina la dirección IP y el número de puerto correspondientes para el encaminamiento de los datos al servidor apropiado.

NOTA – La selección del *número de puerto* determina cuál de dos niveles de prioridad será utilizado. Véase la cláusula 12 al respecto de las asignaciones de puerto.

9.2 Conectarse al servidor

La secuencia siguiente de pasos resume esta operación. Los detalles de los pasos se indican a continuación de este resumen:

- Asignar estructura de datos y memoria TLS, si es necesario.
- Abrir zócalo.
- Enviar saludo del cliente TLS.
- Enviar certificado del cliente al servidor.
- Intercambio de claves del cliente.
- Enviar verificación certificado del cliente.
- Ejecutar cambio especificaciones cifradas.
- Enviar finalizó el cliente.

Se dan los siguientes detalles para cada uno de los pasos antes identificados:

9.2.1 Asignar estructura de datos y memoria TLS

Una estructura de datos se utiliza para almacenar los datos criptográficos asociados con una conexión individual, incluidos certificados, referencias a llamadas de vuelta y diversos otros datos. Debe asignarse una estructura de datos independiente, y mantenerse para conexión.

9.2.2 Abrir zócalo

Una aplicación de cliente crea un zócalo y a continuación emite una conexión a un servicio especificado en una estructura (por ejemplo, *sockaddr_in*). Sigue a continuación un ejemplo de operación abrir zócalo en "C":

```
int     tcpopen(host,service)
char   *service, *host;
{  int    unit;
   struct  sockaddr_in    sin;
   struct  servent        *sp;
   struct  hostent        *hp;
   if ((sp=getservbyname(service,"tcp")) == NULL) then error…
   if ((hp=gethostbyname(host)) == NULL) then error...
   bzero((char *)&sin, sizeof(sin))
       etc...
   if ((unit=socket(AF_INET,SOCK_STREAM,0)) < 0) then error...
   if (connect(unit,&sin,sizeof(sin)) < 0) then error...
   return(unit);
}
```

El resultado retornado es un *descriptor de fichero* que está conectado a un proceso de servidor. Éste es un canal de comunicaciones por el cual se puede realizar un protocolo específico de aplicación.

9.2.3 Enviar saludo del cliente TLS

La función envía la fecha/hora actual del cliente, identificador de sesión, lista de sucesiones cifradas, lista de algoritmos de compresión, una estructura de datos aleatorios y un parámetro de versión de cliente. Este parámetro especifica qué versiones del protocolo TLS pueden utilizarse para la conexión. Ésta se pondría a un valor de {3.1} para la TLS. Tras enviar un *saludo del cliente* (*client hello*), el cliente debe esperar hasta que recibe un *saludo del servidor* (*server hello*) en respuesta.

9.2.4 Enviar certificado del cliente al servidor

El cliente envía su certificado digital al servidor.

9.2.5 Intercambio de claves del cliente

Este mensaje fija el secreto maestro previo de 48 bytes, lo encripta con la clave pública del servidor y envía los resultados en un mensaje secreto maestro previo encriptado.

9.2.6 Enviar verificación certificado del cliente

Este mensaje se utiliza para proporcionar verificación explícita del certificado del cliente. Este mensaje sólo se envía si se ha enviado un certificado de cliente que tiene capacidad de firma. Seguirá inmediatamente al mensaje *intercambio de claves de cliente* (*client key exchange*).

9.2.7 Cambiar especificaciones cifradas

Este mensaje se envía para notificar al sistema par que los registros subsiguientes estarán protegidos por las *especificaciones cifradas* (*cipher spec*) y claves recién negociadas.

9.2.8 Enviar finalizó el cliente

El cliente enviará entonces el mensaje *finalizó el cliente* (*client finished*) inmediatamente después de un mensaje *cambiar especificaciones cifradas* para verificar que los procesos de intercambio de claves y de autenticación tuvieron éxito. Este mensaje es el primer mensaje enviado protegido con los algoritmos, claves y secretos recién negociados. El cliente está listo para comenzar el envío de datos de aplicación.

9.3 Enviar datos de aplicación al servidor

Después de que se ha establecido la conexión con el servidor, el cliente codificará el mensaje utilizando reglas de codificación distinguidas (DER) y lo transmitirá al servidor como un tren de datos. Este proceso se realiza utilizando la función escritura de TLS, especificada por el juego de herramientas o biblioteca de TLS.

9.4 Registro cronológico de transmisión

Se crean entradas de registro cronológico para registrar la transmisión de datos al servidor. El conjunto mínimo de datos que se registra es el siguiente:

- Fecha/hora.
- Un identificador de mensaje único (por ejemplo, el segmento ISA si ASC X12 EDI).
- Dirección IP distante y número de puerto del par.
- Indicador de éxito/fracaso con las interfaces.

9.5 Desconecta el cliente

El cliente y el servidor deben compartir el conocimiento de que la conexión está finalizando. El cliente iniciará la siguiente secuencia una vez concluida la escritura al par de todos los datos:

- Ejecutar la notificación de cierre TLS:

  Esta función cierra la sesión TLS con el par. No han de esperarse transmisiones posteriores. El cliente debe esperar a que el servidor responda antes de proceder con los casos siguientes de esta secuencia.

- Ejecutar el cierre del zócalo:

  El cliente cierra la conexión por zócalo con la función *cierre()* (*close()*), pasando el descriptor de zócalo como parámetro.

- Ejecutar la supresión de estructura de datos TLS:

  Normalmente esta función libera los recursos utilizados por la conexión TLS.

  Esto sólo se realiza si la sesión ha de ser terminada y no se efectúa en las sesiones que puedan reanudarse más tarde. Según la implementación, la memoria utilizada por la estructura de datos puede tener que ser desasignada tras su utilización.

Si no se recibe la *notificación de cierre* (*close notify*) del servidor, el cliente reconocerá una condición de error y efectuará los pasos siguientes:

- Ejecutar el cierre del zócalo.
- Ejecutar la supresión de la estructura de datos TLS y marcar la sesión como no reanudable.
- Notificar la aplicación al usuario de un fallo de comunicación.

10 Especificaciones del servidor

Esta cláusula define el procesamiento asociado con el proceso de servidor de agente interactivo.

Al arrancar el proceso del servidor, el servidor efectúa la siguiente secuencia de pasos:

10.1 Inicializar el servidor

Para aceptar condiciones, se crea un zócalo y se vincula a un puerto del servicio. Se especifica una cola para conexiones entrantes y se aceptan las conexiones como se indica en este fragmento de código de lenguaje C:

```
struct servent          *sp;
struct sockaddr_in sin,from;
if ((sp=getservbyname(service,"tcp")) == NULL) then error...
sin.sin_family=etc...
if ((s=socket(AF_INET,SOCK_STREAM,0)) < 0) then error...
if (bind(s, &sin, sizeof(sin)) < 0) then error...
if (listen(s,QUELEN) < 0) then error...
for (;;) {
   if ((g=accept(f,&from,&len)) < 0) then error...
   if (!fork()) {
      child handles request...
         ...and exits
      exit(0);
   }
   close(g);   /* parent releases file */
}
```

Cuando el proceso y el puerto están vinculados, el servidor escuchará las peticiones de conexión del puerto. Una vez recibida una petición de conexión, se crea un zócalo. Por ejemplo, cuando se utiliza un esquema multiprocesamiento, se efectúan las conexiones y el proceso toma un *proceso vástago* (*child process*) para tratar esa petición de servicio. El *proceso progenitor* (*parent process*) continúa escuchando y aceptando peticiones de servicio posteriores.

10.2 Aceptar la conexión del cliente

Basándose en la identidad de la entidad par y en otra información pertinente (que podría incluir la dirección IP distante y el número de puerto local), el encaminamiento de los datos normalmente será determinado al traductor EDIFACT/ASC X12 EDI apropiado o el módulo de seguridad.

10.3 Establecimiento de lectura del mensaje

La siguiente secuencia de pasos resume esta operación. Los detalles de los pasos se indican a continuación de este resumen:

- Asignar estructura de datos y memoria TLS.
- Vincular estructura de datos TLS al zócalo.
- Enviar saludo del servidor TLS.
- Enviar certificado del servidor al cliente.
- Intercambio de claves del servidor.
- Enviar petición de certificado del cliente.
- Enviar saludo del servidor efectuado.
- Ejecutar cambio especificaciones cifradas.
- Enviar finalizó el servidor.

Cuando se recibe una petición de conexión de entrada, el proceso de *escucha* del servidor pasará la petición de conexión al soporte lógico de *procesamiento* de la conexión. Este soporte lógico hará lo siguiente:

10.3.1 Asignar estructura de datos y memoria TLS

Una estructura de datos se utiliza para almacenar los datos criptográficos asociados con una conexión individual, incluidos certificados, referencias a llamadas de vuelta (*callbacks*) y diversos otros datos. Debe asignarse y mantenerse para toda conexión una estructura de datos independiente.

10.3.2 Vincular estructura de datos TLS al zócalo

Esta acción conecta lógicamente la estructura de datos TLS con el zócalo en el que está pendiente la petición de conexión de entrada. Los pasos reales efectuados para conseguirlo variarán según el juego de herramientas TLS y el soporte lógico de gestión de zócalo utilizado.

10.3.3 Enviar saludo del servidor TLS

Esta función envía el identificador de sesión (o uno nuevo o el valor enviado por el cliente en el caso de una sesión reanudada), una única sucesión cifrada seleccionada, un único algoritmo de compresión seleccionado, una estructura de datos aleatorios (diferente de la cliente) y un parámetro de versión de servidor. Este parámetro especifica qué versión del protocolo TLS se utilizará para la conexión. Éste debe ser un valor de {3.1} para TLS. Este mensaje se envía como respuesta a un *saludo del cliente* (*client hello*).

10.3.4 Enviar certificado del servidor al cliente

El servidor envía al cliente el contenido de su certificado X.509 versión 3.

10.3.5 Intercambio de claves del servidor

Este mensaje debe enviarse inmediatamente después del mensaje de certificado de servidor, si se requiere. El mensaje de intercambio de claves del servidor sólo se envía en casos en los que el mensaje de certificado del servidor no contiene suficientes datos para permitir al cliente intercambiar un secreto maestro previo. Esto será aplicable si se ha seleccionado un método de intercambio de claves RSA_EXPORT y la clave pública en el certificado del servidor es de longitud mayor que 512 bits.

10.3.6 Enviar petición de certificado del cliente

Esta función pide un certificado al cliente. El mensaje debe seguir inmediatamente al mensaje de intercambio de claves de servidor (si se envía), o en otro caso al mensaje de certificado del servidor.

10.3.7 Enviar saludo del servidor efectuado

Esta función notifica al cliente que el servidor ha concluido el envío del saludo del servidor (*server hello*) y sus mensajes asociados. Tras el envío de este mensaje, el servidor esperará una respuesta del cliente.

10.3.8 Ejecutar cambio especificaciones cifradas

Este mensaje encarga al sistema par que active el conjunto de parámetros criptográficos más recientemente recibido.

10.3.9 Enviar finalizó el servidor

El servidor enviará entonces el mensaje finalizó el servidor inmediatamente después de un mensaje cambio especificaciones cifradas, para verificar que los procesos de intercambio de claves y de autenticación tuvieron éxito. Este mensaje es el primer mensaje enviado protegido con los algoritmos, claves y secretos recién negociados. El lado servidor de la conexión está ahora listo para recibir datos.

10.4 Procesamiento de lectura TLS

Este proceso exige que se efectúe una *lectura* (*read*) TLS inicial. El primero de estos octetos contendrá la representación DER de una etiqueta ASN.1 que identifique el tipo de mensaje IA especificado en la cláusula 8. El segundo octeto será o bien una longitud definida de forma corta DER o el primer octeto de una longitud definida de forma larga DER, enumerando el número de octetos contenidos en el resto del campo de longitud.

Si el segundo octeto es una longitud DER de forma corta, representa la longitud del resto del actual mensaje IA.

Si el segundo octeto es una longitud DER de forma larga, se requiere una segunda *lectura* (*read*) del número indicado de octetos para determinar el número de octetos restantes que comprenden el campo de longitud definida de forma larga. Estos objetos restantes enumeran la longitud total del resto del mensaje IA.

Debe entonces iniciarse una *lectura* final para *leer* todos los octetos que comprenden el resto del mensaje IA. Esta tercera lectura puede efectuarse como una única operación *lectura* o como una serie de *lecturas* parciales cuya suma sea igual al valor total. Todas las *lecturas* de esta subcláusula se realizan utilizando la función *lectura* TLS, especificada por el juego de herramientas o biblioteca de TLS que se utilice.

Si la *lectura* TLS falla debido a recursos insuficientes, el IA inicia procedimientos de control de flujo como los especificados en 11.3

10.5 Desconecta el servidor

El cliente y el servidor deben compartir el conocimiento de que la conexión está finalizando. El cliente señalará al servidor que ha concluido la transmisión del mensaje en curso enviando una notificación de cierre del cliente. El servidor debe concluir el procesamiento de lectura TLS pendiente y ejecutar a continuación los siguientes procedimientos:

- Ejecutar la notificación de cierre TLS:

  Esta función cierra la sesión TLS con el par. No han de esperarse comunicaciones posteriores. Normalmente esta función se efectuará después de que el cliente haya enviado el mensaje notificación de cierre del cliente y todos los datos entrantes hayan sido leídos en el flujo de datos entrante.

- Ejecutar cierre del zócalo:

  El servidor cierra la conexión por zócalo con la función cierre() *close()*, pasando el descriptor de zócalo como parámetros.
- Ejecutar la supresión de estructura de datos TLS:

  Normalmente esta función libera los recursos utilizados por la conexión TLS. Esto sólo se efectúa si la sesión ha de ser terminada y no se efectúa en las sesiones que puedan reanudarse más tarde. Según la implementación, la memoria utilizada por la estructura de datos puede tener que ser desasignada tras su utilización.

10.6 Análisis sintáctico (*parsing*) del mensaje recibido

Una vez que un IA ha recibido un mensaje de un IA par, se analiza el mensaje. Al analizar el mensaje se examina la etiqueta inicial para determinar cuál de la sintaxis normalizadas se utilizará para decodificar e interpretar el resto del mensaje.

10.7 Transferencia de datos al usuario inmediato (traductor/módulo de seguridad)

En el caso de que el mensaje sea del tipo "básico", los datos de usuario contenidos en el mensaje deben pasarse al usuario inmediato del IA (generalmente un traductor EDIFACT/ASC X12 EDI) para cualquier procesamiento ulterior que se requiera. El mecanismo por el que se consigue en el IA cae fuera del alcance de esta Recomendación UIT-T.

Si el mensaje es del tipo "mejorado", el contenido debe pasarse al módulo de seguridad pertinente para su posterior análisis y validaciones de seguridad requeridas.

Si fracasa la transferencia de datos al usuario inmediato del agente interactivo, el IA puede iniciar los procedimientos de control de flujo especificados en 11.3.

10.8 Registro cronológico de recibo

Las entradas de registro cronológico deben crearse como consecuencia del recibo de datos por el servidor. El conjunto mínimo de datos que han de registrarse deben ser los siguientes:

- Fecha/hora de recepción.
- Identificador de mensaje único (por ejemplo, segmento ASC X12 EDI ISA).
- Dirección IP distante y número de puerto local.
- Indicadores de éxito/fracaso de recepción y transferencia al usuario inmediato (normalmente un traductor EDIFACT/ASC X12 EDI o un módulo de seguridad).

11 Requisitos operacionales

11.1 Seguridad

Dos entidades pares comunicantes acordarán bilateralmente el nivel que ha de emplearse en el intercambio de IA, y el método para sustentar el nivel de seguridad convenido.

Se aplican las siguientes directrices al usuario de los servicios de seguridad de la capa de transporte (TLS):

- Se dotará a todas las asociaciones de autenticación rigurosa de entidades pares, basada en la encriptación de claves públicas.
- Se supone que los clientes IA y los servidores IA intercambiarán certificados digitales.
- La clave secreta de la sesión se encripta con la clave pública del receptor.
- El uso de encriptación de mensajes es opcional, pero recomendado.

- SHA-1 es el algoritmo digesto recomendado para la función integridad de bloques de transmisión TLS.
- Si se selecciona protección de privacidad TLS, se recomienda el modo norma de criptación de datos en el encadenamiento de bloques cifrados (DES-CBC, *data encryption standard in the cipher block chaining*) para la criptación de claves simétrica.
- Es necesario que toda entidad que utilice el IA obtenga un certificado de clave pública de un CA aceptable para a las entidades pares.
- Los certificados serán compatibles con la Recomendación X.509, versión 3.

NOTA 1 – Las sesiones reanudables no presentan ninguna amenaza adicional para la seguridad.

NOTA 2 – Véase en la Recomendación UIT-T Q.815 detalles de las especificaciones de seguridad a nivel de mensaje.

11.2 Certificados digitales

La arquitectura IA exige que ambas partes intercambien certificados digitales durante la entrada en contacto TLS. Al recibir un certificado de un par, la información de certificado debe pasarse de la TLS (capa de transporte) vía el IA al módulo de seguridad, donde será retenida y utilizada para operaciones de seguridad mejoradas tales como no repudio. El mismo par de claves y el certificado utilizados para la autenticación del sistema par se utilizará también para cualesquiera firmas digitales requeridas.

Las entidades pares deben acordar autoridades de certificados de confianza mutuamente aceptables. Sólo deben intercambiarse o referirse a ellos en las firmas digitales los certificados emitidos por estas autoridades de confianza.

En el caso de que los certificados digitales o las listas de certificados utilicen el tipo de datos ASN.1 **`UTCTime`**, ha de seguirse el siguiente procedimiento:

Antes de que se utilice un valor de **`Time`** en cualquier operación de comparación, y si se ha elegido la sintaxis de **`Time`** como el tipo de **`UTCTime`**, el valor del campo de año de dos cifras se racionalizará en un valor de año de cuatro cifras como sigue:

- Si el valor de dos cifras es 00 a 49 inclusive, se añadirá 2000 al valor.
- Si el valor de dos cifras es 50 a 99 inclusive, se añadirá 1900 al valor.

NOTA 1 – El uso de **GeneralizedTime** puede evitar el interfuncionamiento con implementaciones que no conocen la posibilidad de elegir **UTCTime** o **GeneralizedTime**. Corresponde a quienes especifican los dominios indicar en qué certificados definidos en esta especificación de directorio serán utilizados, por ejemplo, grupos de determinación de perfiles, así como cuándo puede utilizarse **GeneralizedTime**. En ningún caso se utilizará **UTCTime** para representar fechas superiores a 2049.

NOTA 2 – Existe el procedimiento de satisfacer un asunto Y2K creado por ASN.1 definiendo **UTCTime** que comprenda sólo un año de dos caracteres.

NOTA 3 – Se especifica más información relativa a **UTCTime** en certificados y firmas digitales X.509 para la Recomendación UIT-T X.509 (1995) en la Guía de implementadores de directorio del UIT-T (versión 11), que se incorpora en la Recomendación UIT-T X.509 (1997).

11.3 Control de flujo

El mecanismo de transporte especificado por el agente interactivo requiere que se establezca para cada mensaje una sesión TLS individual. La comunicación es esencialmente unidireccional, del cliente al servidor. El mensaje de situación de IA es un mecanismo que permite a entidades pares intercambiar errores y otros tipos de información de control de flujo. Si las entidades pares pueden definir códigos de mensaje específicos caen fuera del alcance de esta Recomendación UIT-T (véase el cuadro 2).

El servidor también tiene la opción de rehusar un establecimiento de sesión TLS ofrecido si las condiciones de su lado impiden el procesamiento inmediato de un mensaje entrante. En esta situación, se espera que el cliente reintente la conexión pasado un plazo convenido.

Cuando un servidor IA recibe un mensaje de situación con un valor que indica congestión de nivel superior o inferior, indica a su lado cliente, por un medio ajeno a esta Recomendación UIT-T, que cese la transmisión de mensajes IA subsiguientes destinados al originador del mensaje de situación hasta que una intervención exterior provoque la reanudación del procesamiento.

12 Asignaciones de puertos

Las asignaciones de puertos TCP/IP han de ser acordadas por las entidades pares. Estos puertos no necesitan ser los mismos en cada extremo de la conexión. Los puertos están asociados con la pila de protocolos TCP dentro de la configuración del sistema.

La autoridad de asignación de números Internet (IANA, *Internet assigned number authority*) ha registrado el protocolo descrito en esta Recomendación UIT-T y le ha asignado el número de protocolo 117. El protocolo se registra como "protocolo de transferencia de agente interactivo" y se abrevia ***iatp***.

La IANA ha asignado el número de puerto 6999 a ***iatp-normalpri***. Se recomienda el uso de este puerto para transacciones de prioridad normal que utilizan este protocolo.

La IANA ha asignado el número de puerto 6998 a ***iatp-highpri***. Se recomienda el uso de este puerto para transacciones de prioridad normal que utilizan este protocolo.

ANEXO A

Módulo de producción ASN.1

InteractiveAgent {itu-t(0) recommendation(0) q(17) q814(814) ia(0) messages(0)} DEFINITIONS IMPLICIT TAGS ::= BEGIN

-- EXPORTS everything

```
IaMessage ::= CHOICE {
        basicMessage        BasicMessage,       -- Basic Message
        iaStatusMessage     IaStatusMessage,    -- IA Status/Control Message
        enhancedMessage     EnhancedMessage     -- Security Module Enhanced Message
}

BasicMessage ::= CHOICE {

basicMessage1     GeneralString,

basicMessage2      IA5String

}
IaStatusMessage ::= BIT STRING ( SIZE (32) )
EnhancedMessage ::= OCTET STRING          -- Security Module Enhanced Message
END
```

ANEXO B

Consideraciones de diseño

Se identifican en este anexo las consideraciones/constricciones de diseño que harán al agente interactivo más eficiente y robusto.

B.1 Multiprocesamiento/multienlazamiento

La naturaleza y el uso de aplicación del IA exigirá que múltiples entidades IA concurrentes operen simultáneamente. Múltiples clientes pueden obtenerse simplemente invocando múltiples instancias de soporte lógico o procesos de cliente IA. Pueden utilizarse igualmente otros esquemas. Los procesos de servidor IA se estructuran normalmente para manejar múltiples peticiones de cliente simultáneas mediante procesos tales como *selección de proceso vástago, multienlazamiento, multiplexación,* u otras tecnologías comparables.

NOTA – Cada servidor IA debe considerar la provisión de hasta un máximo de 10 conexiones simultáneas por entidad par.

B.2 Conexiones no persistentes o persistentes

Cada conexión TLS puede sustentar la transferencia de un mensaje EDIFACT o ASC X12 EDI o de múltiples sesiones en una sola sesión. El primer caso exige que una sesión deba existir únicamente mientras dure la transmisión de un solo mensaje, en tanto que el segundo permite que una conexión se mantenga tanto tiempo como haya sido mutuamente convenido por las entidades pares.

La decisión de utilizar un solo mensaje por sesión o múltiples mensajes por sesión es una decisión de diseño basada en las necesidades de la aplicación. En una aplicación en la que el número de clientes puede ser grande, el paradigma de un solo mensaje por sesión puede hacer el mejor uso de los recursos. En cambio, en una aplicación que requiere intercambiar un gran número de mensajes entre un pequeño número de clientes, el caso de múltiples mensajes por sesión será probablemente el más eficiente.

B.3 Sesiones TLS reanudables

Por razones de calidad de funcionamiento se recomiendan las sesiones TLS reanudables. El establecimiento de TLS ordinario exige un trabajo intensivo del procesador, por lo que es necesario disponer de un mecanismo que permita reanudarse una sesión utilizando un subconjunto óptimo del procesamiento de establecimiento de TLS.

El tiempo que una sesión se mantendrá *reanudable* cae fuera del alcance de esta Recomendación UIT-T. Las implementaciones deben permitir que este parámetro sea configurable. Si las sesiones son para salir por edad (*age out*) de una sesión caché *reanudable*, este parámetro debe reconfigurarse para iniciar estas expiraciones después de que ha transcurrido el número indicado de minutos desde la última utilización del ID de sesión. Los valores recomendados son de la gama de 1 a 30 minutos.

ANEXO C

Tratamiento de errores/recuperación tras errores

Una sesión de agente interactivo puede terminar bruscamente debido a fallos técnicos. La recuperación en este caso consiste en establecer una nueva sesión y reenviar la transacción en curso. Sin embargo, existe un punto perfectamente definido en el que el cliente considera que una transacción se ha enviado con éxito. Este punto se produce cuando se recibe del servidor *notificación de cierre* (*close notify*). Después de que la recuperación de este punto está en el nivel de aplicación de extremo a extremo, que será desencadenada por la no recepción de una transacción de acuse de recibo (por ejemplo, ASC X12 997 – acuse de recibo funcional) dentro del periodo de temporización.

La recuperación del nivel de aplicación es indicada por la expiración de la temporización mientras se espera la transacción de acuse de recibo concordante (por ejemplo, ASC X12 997 – acuse de recibo funcional). Pueden especificarse temporizaciones separadas para tipos de transacción de prioridad normal y alta para cada entidad par.

APÉNDICE I

Referencias no normativas

– ISO 9735:1988, *Electronic Data Interchange For Administration, Commerce and Transport (EDIFACT) – Application level syntax rules*.

– ANSI ASC X12: American National Standards Institute (ANSI) Accredited Standards Committee X12. The Committee was chartered by ANSI in 1979 to develop uniform standards for the electronic interchange of business documents.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie B	Medios de expresión: definiciones, símbolos, clasificación
Serie C	Estadísticas generales de telecomunicaciones
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedios
Serie I	Red digital de servicios integrados
Serie J	Transmisiones de señales radiofónicas, de televisión y de otras señales multimedios
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	RGT y mantenimiento de redes: sistemas de transmisión, circuitos telefónicos, telegrafía, facsímil y circuitos arrendados internacionales
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Calidad de transmisión telefónica, instalaciones telefónicas y redes locales
Serie Q	**Conmutación y señalización**
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos y comunicación entre sistemas abiertos
Serie Y	Infraestructura mundial de la información y aspectos del protocolo Internet
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación

Ginebra, 2001