UNIÓN INTERNACIONAL DE TELECOMUNICACIONES

UIT-T X.830

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

(04/95)

REDES DE DATOS Y COMUNICACIÓN ENTRE SISTEMAS ABIERTOS

SEGURIDAD

TECNOLOGÍA DE LA INFORMACIÓN – INTERCONEXIÓN DE SISTEMAS ABIERTOS – SEGURIDAD GENÉRICA DE LAS CAPAS SUPERIORES: SINOPSIS, MODELOS Y NOTACIÓN

Recomendación UIT-T X.830

(Anteriormente «Recomendación del CCITT»)

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. En el UIT-T, que es la entidad que establece normas mundiales (Recomendaciones) sobre las telecomunicaciones, participan unos 179 países miembros, 84 empresas de explotación de telecomunicaciones, 145 organizaciones científicas e industriales y 38 organizaciones internacionales.

Las Recomendaciones las aprueban los Miembros del UIT-T de acuerdo con el procedimiento establecido en la Resolución N.° 1 de la CMNT (Helsinki, 1993). Adicionalmente, la Conferencia Mundial de Normalización de las Telecomunicaciones (CMNT), que se celebra cada cuatro años, aprueba las Recomendaciones que para ello se le sometan y establece el programa de estudios para el periodo siguiente.

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. El texto de la Recomendación UIT-T X.830 se aprobó el 10 de abril de 1995. Su texto se publica también, en forma idéntica, como Norma Internacional ISO/CEI 11586-1.

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.

RECOMENDACIONES UIT-T DE LA SERIE X

REDES DE DATOS Y COMUNICACIÓN ENTRE SISTEMAS ABIERTOS

(Febrero de 1994)

ORGANIZACIÓN DE LAS RECOMENDACIONES DE LA SERIE X

Dominio	Recomendaciones
REDES PÚBLICAS DE DATOS	
Servicios y facilidades	X.1-X.19
Interfaces	X.20-X.49
Transmisión, señalización y conmutación	X.50-X.89
Aspectos de redes	X.90-X.149
Mantenimiento	X.150-X.179
Disposiciones administrativas	X.180-X.199
INTERCONEXIÓN DE SISTEMAS ABIERTOS	
Modelo y notación	X.200-X.209
Definiciones de los servicios	X.210-X.219
Especificaciones de los protocolos en modo conexión	X.220-X.229
Especificaciones de los protocolos en modo sin conexión	X.230-X.239
Formularios para enunciados de conformidad de implementación de protocolo	X.240-X.259
Identificación de protocolos	X.260-X.269
Protocolos de seguridad	X.270-X.279
Objetos gestionados de capa	X.280-X.289
Pruebas de conformidad	X.290-X.299
INTERFUNCIONAMIENTO ENTRE REDES	
Generalidades	X.300-X.349
Sistemas móviles de transmisión de datos	X.350-X.369
Gestión	X.370-X.399
SISTEMAS DE TRATAMIENTO DE MENSAJES	X.400-X.499
DIRECTORIO	X.500-X.599
GESTIÓN DE REDES DE INTERCONEXIÓN DE SISTEMAS ABIERTOS Y ASPECTOS DE SISTEMAS	
Gestión de redes	X.600-X.649
Denominación, direccionamiento y registro	X.650-X.679
Notación de sintaxis abstracta uno	X.680-X.699
GESTIÓN DE INTERCONEXIÓN DE SISTEMAS ABIERTOS	X.700-X.799
SEGURIDAD	X.800-X.849
APLICACIONES DE INTERCONEXIÓN DE SISTEMAS ABIERTOS	
Cometimiento, concurrencia y recuperación	X.850-X.859
Tratamiento de transacciones	X.860-X.879
Operaciones a distancia	X.880-X.899
TRATAMIENTO ABIERTO DISTRIBUIDO	X.900-X.999

ÍNDICE

Sumario

Esta Recomendación forma parte de una serie de Recomendaciones que proporcionan diversas facilidades para la construcción de protocolos de capa superior de OSI que sustentan la prestación de servicios de seguridad. En esta Recomendación se definen:

a) modelos generales de funciones de protocolo de intercambio de seguridad y transformaciones de seguridad;

b) un conjunto de herramientas de notación para sustentar la especificación de requisitos de protección selectiva de los campos en una especificación de sintaxis abstracta y para sustentar la especificación de intercambios de seguridad y transformaciones de seguridad;

c) un conjunto de directrices informativas sobre la aplicación de las facilidades de seguridad genérica de las capas superiores abarcadas por esta serie de Recomendaciones.

Introducción

Esta Recomendación | Norma Internacional pertenece a una serie de Recomendaciones | Normas Internacionales multiparte que proporciona(n) un conjunto de facilidades para la construcción de protocolos de capa superior que sustentan la prestación de servicios de seguridad. Las partes son:

– Parte 1: Sinopsis, modelos y notación

– Parte 2: Definición de servicio del elemento de servicio de intercambio de seguridad

– Parte 3: Especificación del protocolo del elemento de servicio de intercambio de seguridad

– Parte 4: Especificación de la sintaxis de transferencia de protección

– Parte 5: Formulario PICS del elemento de servicio de intercambio de seguridad

– Parte 6: Formulario PICS de la sintaxis de transferencia de protección

Esta Recomendación | Norma Internacional constituye la Parte 1 de esta serie.

El Anexo G contiene directrices informativas sobre la aplicación de todas las facilidades descritas en esta serie.

Es importante resaltar que estas facilidades de seguridad genéricas no proporcionan por sí mismas servicios de seguridad; sencillamente son herramientas constructivas para protocolos relacionados con la seguridad. Además, estas facilidades no proporcionan necesariamente una solución autónoma a todos los requisitos de comunicaciones de seguridad de las aplicaciones. Puede aún ser necesario que las normas relativas a la aplicación incorporen aspectos de seguridad en sus propias especificaciones que actúen conjuntamente con los servicios de seguridad genéricos sustentados por las facilidades de seguridad genérica de las capas superiores.

NORMA INTERNACIONAL

RECOMENDACIÓN UIT-T

TECNOLOGÍA DE LA INFORMACIÓN – INTERCONEXIÓN DE SISTEMAS ABIERTOS – SEGURIDAD GENÉRICA DE LAS CAPAS SUPERIORES: SINOPSIS, MODELOS Y NOTACIÓN

1 Alcance

1.1 Esta serie de Recomendaciones | Normas Internacionales define un conjunto de facilidades genéricas destinadas a facilitar la prestación de servicios de seguridad en aplicaciones OSI, que comprenden:

a) un conjunto de herramientas de notación que permitan la especificación de requisitos de protección selectiva de los campos en una especificación de sintaxis abstracta, y la especificación de intercambios de seguridad y transformaciones de seguridad;

b) una definición de servicio, especificación de protocolo y formulario PICS para un elemento de servicio de aplicación (ASE) que permita la prestación de servicios de seguridad dentro de la capa de aplicación de OSI;

c) una especificación y un formulario PICS para una sintaxis de transferencia de seguridad, asociada con soporte de la capa de presentación para servicios de seguridad en la capa de aplicación.

1.2 Esta Recomendación | Norma Internacional define:

a) modelos generales de funciones de protocolo de intercambio de seguridad y transformaciones de seguridad basados en los conceptos descritos en el modelo de seguridad de capas superiores de OSI (Rec. UIT-T X.803 | ISO/CEI 10745);

b) un juego de herramientas de notación para sustentar la especificación de requisitos de protección selectiva de los campos en una especificación de sintaxis abstracta y para sustentar la especificación de intercambios de seguridad y transformaciones de seguridad;

c) un conjunto de directrices de información para la aplicación de facilidades genéricas de seguridad de capas superiores abarcadas por esta serie de Recomendaciones | Normas Internacionales.

1.3 Esta Recomendación | Norma Internacional no define:

a) un juego completo de facilidades de seguridad de capa superior que puede ser necesario en otras Recomendaciones | Normas Internacionales;

b) un conjunto completo de facilidades de seguridad para aplicaciones específicas;

c) los mecanismos utilizados para sustentar servicios de seguridad.

1.4 Se ha previsto que el modelo de intercambio de seguridad y la notación sustentante se utilicen como base para la definición del elemento de servicio de intercambio de seguridad en partes subsiguientes de esta serie de Recomendaciones | Normas Internacionales así como para su uso por otras ASE que necesitan importar intercambios de seguridad en su propia especificación.

2 Referencias normativas

Las siguientes Recomendaciones | Normas Internacionales contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación | Norma Internacional. Al efectuar esta publicación estaban en vigor las ediciones indicadas. Todas las Recomendaciones y Normas Internacionales son objeto de revisiones por lo que se invita a los participantes en acuerdos basados en la presente Recomendación | Norma Internacional a que

investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones y Normas enumeradas a continuación. Los miembros de la CEI y de la ISO mantienen registros de las Normas Internacionales actualmente vigentes. La Oficina de Normalización de las Telecomunicaciones de la UIT mantiene una lista de las Recomendaciones UIT-T actualmente vigentes.

2.1 Recomendaciones | Normas Internacionales idénticas

- Recomendación UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Modelo de referencia básico: El modelo básico.*
- Recomendación UIT-T X.207 (1993) | ISO/CEI 9545:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Estructura de la capa de aplicación.*
- Recomendación UIT-T X.214 (1993) | ISO/CEI 8072:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Definición del servicio de transporte.*
- Recomendación UIT-T X.216 (1994) | ISO/CEI 8822:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Definición del servicio de presentación.*
- Recomendación UIT-T X.217 (1995) | ISO/CEI 8649:...[1)], *Tecnología de la información – Interconexión de sistemas abiertos – Definición de servicio para el elemento de servicio de control de asociación.*
- Recomendación UIT-T X.226 (1994) | ISO/CEI 8823-1:1994, *Tecnología de la información – Interconexión de sistemas abiertos – Protocolo de presentación con conexión: Especificación del protocolo.*
- Recomendación UIT-T X.509 (1993) | ISO/CEI 9594-8:1995, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio – Marco de autenticación.*
- Recomendación UIT-T X.511 (1993) | ISO/CEI 9594-3:1994, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio – Definición de servicio abstracto.*
- Recomendación X.660 del CCITT (1992) | ISO/CEI 9834-1:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Procedimientos para la operación de autoridades de registro para interconexión de sistemas abiertos – Procedimientos generales.*
- Recomendación UIT-T X.680 (1994) | ISO/CEI 8824-1:1995, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de la notación básica.*
- Recomendación UIT-T X.681 (1994) | ISO/CEI 8824-2:1995, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de objetos de información.*
- Recomendación UIT-T X.682 (1994) | ISO/CEI 8824-3:1995, *Tecnología de la información – Notación de sintaxis abstracta uno: Especificación de constricciones.*
- Recomendación UIT-T X.683 (1994) | ISO/CEI 8824-4:1995, *Tecnología de la información – Notación de sintaxis abstracta uno: Parametrización de las especificaciones de la notación de sintaxis abstracta uno.*
- Recomendación UIT-T X.690 (1994) | ISO/CEI 8825-1:1995, *Tecnología de la información – Reglas de codificación de notación de sintaxis abstracta uno: Especificación de las reglas de codificación básica, de las reglas de codificación canónica y reglas de codificación distinguida.*
- Recomendación UIT-T X.803 (1994) | ISO/CEI 10745:1995, *Tecnología de la información – Interconexión de sistemas abiertos – Modelo de seguridad de las capas superiores.*
- Recomendación UIT-T X.811 (1995) | ISO/CEI 10181-2:1995, *Tecnología de la información – Interconexión de sistemas abiertos – Marcos de seguridad para sistemas abiertos – Marco de autenticación.*
- Recomendación UIT-T X.812[1)] | ISO/CEI 10181-3:...[1)], *Tecnología de la información – Interconexión de sistemas abiertos – Marcos de seguridad para sistemas abiertos: Marco de control de acceso.*

---

1) Actualmente en estado de proyecto.

2.2 Pares de Recomendaciones | Normas Internacionales de contenido técnico equivalente

– Recomendación X.800 del CCITT (1991), *Arquitectura de seguridad en la interconexión de sistemas abiertos para aplicaciones del CCITT.*

ISO 7498-2:1989, *Information processing systems – Open Systems Interconnection – Basic Reference Model – Part 2: Security Architecture*.

3 Definiciones

3.1 Se utiliza el siguiente término definido en la Rec. UIT-T X.200 | ISO/CEI 7498-1:

– sintaxis de transferencia.

3.2 Se utilizan los siguientes términos definidos en la Rec. X.800 del CCITT | ISO 7498-2:

– control de acceso;
– confidencialidad;
– autenticación del origen de los datos;
– descifrado;
– firma digital;
– cifrado;
– integridad;
– clave;
– gestión de claves;
– protección selectiva de los campos.

3.3 Se utilizan los siguientes términos definidos en la Rec. UIT.T X.216 | ISO/CEI 8822:

– sintaxis abstracta;
– contexto de presentación;
– valor de datos de presentación.

3.4 Se utilizan los siguientes términos definidos en la Rec. UIT-T X.207 | ISO/CEI 9545:

– asociación de aplicación;
– contexto de aplicación;
– elemento de servicio de aplicación (ASE);
– asociación de objeto de servicio de aplicación (asociación de ASO).

3.5 Se utilizan los siguientes términos definidos en la Rec. UIT-T X.811 | ISO/CEI 10181-2:

– intercambio de autenticación;
– declarante;
– autenticación de entidad;
– verificador.

3.6 Se utiliza el siguiente término definido en la Rec. UIT-T X.812 | ISO/CEI 10181-3:

– certificado de control de acceso.

3.7 Se utilizan los siguientes términos definidos en la Rec. UIT-T X.803 | ISO/CEI 10745:

– asociación de seguridad;
– función de comunicación de seguridad (SCF);
– intercambio de seguridad;
– elemento de intercambio de seguridad;

- función de intercambio de seguridad;
- transformación de seguridad;
- objeto de seguridad de sistema (SSO).

3.8 Para los fines de esta Recomendación | Norma Internacional se aplican las definiciones siguientes:

3.8.1 asociación de seguridad ligada al contexto de presentación: Asociación de seguridad cuyo establecimiento se produce conjuntamente con el de un contexto de presentación de protección, y que se aplica a todos los valores de datos de presentación transmitidos en un sentido en el contexto de presentación de protección. Los atributos de la asociación de seguridad se indican explícitamente junto con la codificación del primer valor de datos de presentación en el contexto de presentación de protección.

3.8.2 asociación de seguridad ligada a un solo elemento: Asociación de seguridad que se aplica a un solo valor de datos de presentación protegido independientemente, que no está asociado con un contexto de presentación. Los atributos de la asociación de seguridad se indican explícitamente junto con la codificación del valor de datos de presentación.

3.8.3 asociación de seguridad establecida externamente: Asociación de seguridad establecida de forma independiente de los casos de utilización y que tiene un identificador único global que le permite ser referenciado en el momento de utilización.

3.8.4 reglas de codificación inicial: Reglas de codificación en ASN.1 utilizadas para la generación de una cadena de bits no protegidos a partir de un valor de tipo ASN.1, cuando debe protegerse ese valor mediante una transformación de seguridad.

3.8.5 contexto de presentación de protección: Contexto de presentación que asocia una sintaxis de transferencia de protección con una sintaxis abstracta.

3.8.6 sintaxis de transferencia de protección: Sintaxis de transferencia que utiliza una transformación de seguridad.

3.8.7 correspondencia de protección: Especificación que relaciona un requisito de protección, identificado por un nombre en una especificación de sintaxis abstracta, con una transformación de seguridad específica utilizable para satisfacer ese requisito.

4 Abreviaturas

A los efectos de esta Recomendación | Norma Internacional se utilizan las siguientes abreviaturas:

ACSE Elemento de servicio de control de asociación (*association control service element*)

ASE Elemento de servicio de aplicación (*application-service-element*)

ASO Objeto de servicio de aplicación (*application-service-object*)

GULS Seguridad genérica de las capas superiores (*generic upper layers security*)

OSI Interconexión de sistemas abiertos (*open systems interconnection*)

PDU Unidad de datos de protocolo (*protocol-data-unit*)

PDV Valor de datos de presentación (*presentation data value*)

PICS Enunciado de conformidad de implementación de protocolo (*protocol implementation conformance statement*)

SCF Función de comunicación de seguridad (*security communication function*)

SEI Elemento de intercambio de seguridad (*security exchange item*)

SESE Elemento de servicio de intercambio de seguridad (*security exchange service element*)

SSO Objeto de seguridad de sistema (*system security object*)

5 Visión general

Las normas de seguridad genérica de las capas superiores (GULS) definen un conjunto de herramientas de construcción de protocolos y componentes de protocolo que sustentan la prestación de servicios de seguridad para las aplicaciones. Estas facilidades soportan la provisión de servicios de seguridad en las capas superiores de OSI (capa de aplicación, a veces con el soporte de la capa de presentación).

NOTA – Pueden prestarse servicios de seguridad para aplicaciones OSI utilizando mecanismos de seguridad en las capas superior o inferiores. En este último caso se logra la protección mediante la especificación de una calidad de servicio de protección apropiada (como la definida en la Rec. UIT-T X.214 | ISO/CEI 8072) al ACSE cuando se establece una asociación de aplicación. Esta calidad de servicio de protección se transfiere al servicio de transporte de forma transparente a través de las capas de presentación y de sesión. La prestación de servicios de seguridad en las capas inferiores queda fuera del ámbito de esta Recomendación | Norma Internacional.

Las facilidades proporcionadas en las normas de GULS comprenden:

- una forma general de elaborar componentes de protocolo de capa de aplicación para sustentar el intercambio de información relativa a la seguridad entre un par de invocaciones de entidad de aplicación que se comunican entre sí (concepto de intercambio de seguridad sustentado por el SESE); estas facilidades se describen en la cláusula 6;
- un método general para la utilización de facilidades de capa de presentación para realizar transformaciones relacionadas con la seguridad en los elementos de información con objeto de proteger tales elementos (sintaxis genérica de transferencia de protección); estas facilidades se describen en la cláusula 7;
- herramientas de notación de sintaxis abstracta para ayudar al diseñador de protocolos de aplicación en la especificación de la protección de seguridad que debe aplicar a campos seleccionados de este protocolo (un tipo parametrizado PROTECTED y la variante PROTECTED-Q de este tipo); estas facilidades se describen en la cláusula 8.

Los intercambios de seguridad se utilizan para aplicaciones tales como la autenticación de entidad y la gestión de la clave. Se utilizan las transformaciones de seguridad (así como la sintaxis genérica de transferencia de protección y/o el tipo parametrizado PROTECTED o sus variaciones) con fines de integridad, confidencialidad, autenticación del origen de los datos y/o no repudio.

El modelo de seguridad de las capas superiores (Rec. UIT-T X.803 | ISO/CEI 10745) proporciona el modelo de arquitectura para las especificaciones de la GULS. Describe los cometidos de las funciones de intercambio de seguridad y transformaciones de seguridad.

Las funciones de intercambio de seguridad proporcionan los medios para comunicar la información de seguridad entre invocaciones de entidad de aplicación como parte del funcionamiento de un mecanismo de seguridad, es decir generan y procesan información de control de protocolo de aplicación con una finalidad relacionada con la seguridad. Pueden especificarse los intercambios de seguridad utilizando la notación descrita en esta Recomendación | Norma Internacional e importarse después en una especificación de sintaxis abstracta. El elemento de servicio de intercambio de seguridad (SESE) es un elemento de servicio de aplicación (ASE) definido en la Rec. UIT-T X.831 | ISO/CEI 11586-2 y en la Rec. UIT-T X.832 | ISO/CEI 11586-3. El SESE proporciona un método para transportar intercambios de seguridad que sustenta el objetivo de conseguir ASE específicos de la aplicación independientes de los mecanismos de seguridad utilizados. Sin embargo, algunos aspectos de la especificación de una aplicación que incorporan directamente disposiciones de seguridad dependerán del mecanismo.

Pueden sustentarse las transformaciones de seguridad mediante la sintaxis genérica de transferencia de protección descrita en la Rec. UIT-T X.833 | ISO/CEI 11586-4.

6 Intercambios de seguridad

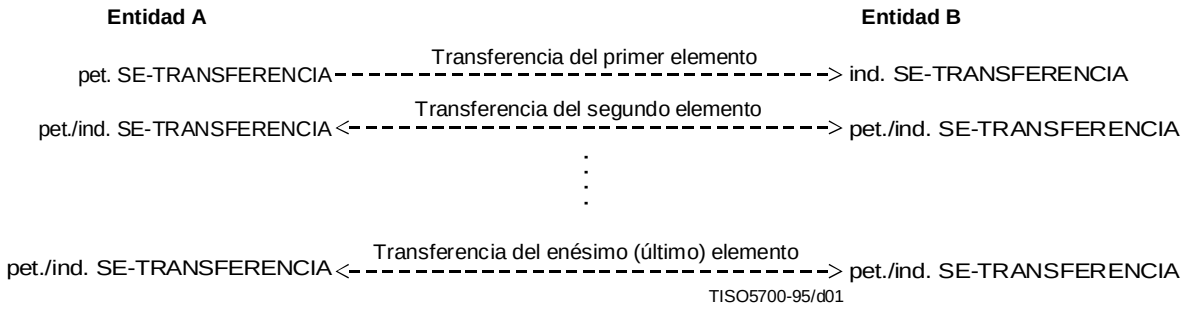
6.1 Modelo de intercambio de seguridad

En esta Recomendación | Norma Internacional se perfecciona el modelo de procedimiento de intercambio de seguridad introducido en la Rec. UIT-T X.803 | ISO/CEI 10745.

Un intercambio de seguridad se produce entre dos entidades A y B. Consiste en la transferencia de un elemento de intercambio de seguridad (SEI) de A a B seguida, posiblemente, de una secuencia de una o más transferencias de SEI en

cualquier sentido entre A y B. El número de transferencias depende de cada intercambio de seguridad concreto. Cada SEI puede comprender una estructura de datos arbitrariamente compleja representable por cualquier tipo ASN.1. Puede incluir componentes protegidas individualmente mediante la notación PROTECTED descrita en la cláusula 8.

El diagrama temporal de la Figura 1 representa la secuencia de transferencias de SEI para un intercambio de seguridad de n vías y las invocaciones de las primitivas del servicio SESE correspondiente definido en la Rec. UIT-T X.831 | ISO/CEI 11586-2.



NOTA – La doble flecha indica que la transferencia puede enviarse por A o por B.

Figura 1 – Modelo de intercambio de seguridad

Hay dos clases de intercambio:

- *Alternado* – Se producen transferencias de elementos sucesivas en sentidos alternos y sólo hay activa una transferencia en un momento dado.
- *Arbitrario* – No hay limitaciones de sentido en ninguna transferencia, y las transferencias en los dos sentidos pueden ser activas simultáneamente.

Cuando está en curso un intercambio de seguridad, pueden producirse otras transferencias de información y pueden también estar en curso otros intercambios de seguridad en la misma asociación de aplicación. Sin embargo, generalmente, las reglas del contexto de aplicación limitarán esas actividades superpuestas. Los valores de datos de presentación que transportan SEI pueden estar concatenados o intercalados con otros valores de datos de presentación o insertados en los mismos.

6.2 Notación para la especificación de los intercambios de seguridad

La especificación de un intercambio de seguridad comprende una especificación de los tipos de SEI que pueden intercambiarse, la indicación de cualquier clase de limitaciones de ordenación aplicables a las transferencias de esos SEI, la indicación de condiciones de error consecuencia de la transferencia de cada SEI y una indicación de la semántica asociada (o una referencia a la misma).

Toda definición de intercambio de seguridad comprende:

a) la asignación de un identificador de objeto global o un valor entero local al intercambio de seguridad, a fin de permitir que su utilización quede identificada inequívocamente en el protocolo;

b) una especificación de la sintaxis abstracta de los SEI y notificaciones de error transferidas en el intercambio de seguridad.

Para sustentar la especificación de esta información de una forma utilizable por el protocolo del SESE, se han previsto tres definiciones de clase de objeto de información ASN.1 (véase la Rec. UIT-T X.681 | ISO/CEI 8824-2), como sigue:

- se utiliza la clase de objeto de información SECURITY-EXCHANGE, para especificar un intercambio de seguridad determinado; un objeto de información de esta clase contiene uno o más objetos de información SEC-EXCHG-ITEM;

– se utiliza la clase de objeto de información SEC-EXCHG-ITEM, para definir un SEI; un objeto de información de esta clase puede contener uno o más objetos de información ERROR;

– se utiliza la clase de objeto de información SE-ERROR, para definir una condición de error que puede ser consecuencia de la transferencia de un SEI.

NOTA – El Anexo G contiene directrices que muestran la utilización de estas clases de objeto de información en la definición de un contexto de aplicación completo.

```
SECURITY-EXCHANGE ::= CLASS
-- This information object class definition is for use when
-- specifying a particular instance of a security exchange.
{
  &SE-Items        SEC-EXCHG-ITEM,
  -- This is an ASN.1 information object set, comprising a set
  -- of security exchange items
  &sE-Identifier   Identifier        UNIQUE
  -- A local or global identifier for the particular security exchange
}
WITH SYNTAX
-- The following syntax is used to specify a particular security exchange.
{
  SE-ITEMS         &SE-Items
  IDENTIFIER       &sE-Identifier
}
Identifier ::= CHOICE
{
  local            INTEGER,
  global           OBJECT IDENTIFIER
}

SEC-EXCHG-ITEM ::= CLASS
{
  &ItemType,
      -- ASN.1 type for this exchange item
  &itemId          INTEGER,
      -- Identifier for this item, e.g. 1, 2, 3, ..
  &Errors          SE-ERROR          OPTIONAL
      -- Optional list of errors which may result from transfer of this item
}
WITH SYNTAX
{
  ITEM-TYPE        &ItemType
  ITEM-ID          &itemId
  [ERRORS          &Errors]
}

SE-ERROR ::= CLASS
{
  &ParameterType   OPTIONAL,
  -- ASN.1 type of a parameter to accompany the signalling
  -- of the error condition back to the sender of the SEI
  &errorCode       Identifier        UNIQUE
  -- An identifier used in signalling the error condition
  -- back to the sender of the SEI
}
WITH SYNTAX
{
  [PARAMETER       &ParameterType]
  ERROR-CODE       &errorCode
}
```

En el Anexo C se facilitan ejemplos de utilización de esta notación.

7 Transformaciones de seguridad

7.1 Modelo de transformación de seguridad

Una transformación de seguridad es una función de seguridad (o una combinación de funciones de seguridad) aplicada a los datos de usuario para protegerlos en un proceso de comunicación o de almacenamiento así como un proceso de codificación que puede aplicarse (pero no siempre) tras la recepción o la recuperación de la información. Como ejemplos de transformaciones de seguridad pueden citarse:

a) la aplicación de un proceso de cifrado a la codificación de datos y el correspondiente proceso de descifrado a la decodificación;

b) la generación de un sello o una signatura y su adición a los datos en la codificación y la verificación y supresión del sello o signatura añadidos en la decodificación;

c) combinación de las funciones de a) y b) en una transformación de seguridad.

Las transformaciones de seguridad definidas mediante la notación de 7.2 son adecuadas para su utilización en aplicaciones OSI (junto con la sintaxis genérica de transferencia de protección definida en la Rec. UIT-T X.833 | ISO/CEI 11586-4) y para otras finalidades, incluida la protección fuera de línea en el almacenamiento local y en comunicaciones que no son del tipo OSI.

NOTA – En 7.1.5 se describe el empleo de transformaciones de seguridad en una conexión de presentación de OSI. En 7.1.6 se describe su utilización con independencia del protocolo de presentación de OSI.

Las transformaciones de seguridad pueden constituir el modo primario de proporcionar un servicio de seguridad (por ejemplo, confidencialidad, integridad, autenticación del origen de los datos) o pueden contribuir a la prestación de un servicio de seguridad (por ejemplo, autenticación de la entidad, control del acceso, no repudio).

En la Figura 2 se representan las etapas de la protección de un elemento de datos para su transferencia o almacenamiento.

En un sistema de codificación, el proceso de obtención de una representación transformada (protegida) de un elemento de datos no protegido se desarrolla como sigue:

a) si el elemento no protegido es un valor de un tipo ASN.1 como está especificado en la definición de sintaxis abstracta, se codificará a una representación de cadena de bits empleando las reglas de codificación inicial; seguidamente

b) se aplicará el proceso de codificación de la transformación de seguridad a la representación en cadena de bits del elemento no protegido utilizando posiblemente información de entrada local adicional, a fin de tener un elemento transformado que sea un valor del tipo ASN.1 XformedDataType (el tipo preciso se especifica como parte de la definición de la transformación de seguridad); a continuación

c) se codifica el valor ASN.1 resultante de b) (posiblemente como parte del proceso de codificación de un valor ASN.1 envolvente tal como la estructura de sintaxis de transferencia de protección definida en la Rec. UIT-T X.833 | ISO/CEI 11586-4).

En un sistema de decodificación, el proceso de recuperación del elemento de datos no protegido y/o verificación de un compromiso de seguridad, es el siguiente:

d) decodificar el elemento transformado recibido o recuperado que es un valor ASN.1 del tipo XformedDataType (este proceso de decodificación puede ser parte del proceso de decodificación de un valor ASN.1 envolvente, tal como la estructura de sintaxis de transferencia de protección definida en la Rec. UIT-T X.833 | ISO/CEI 11586-4); seguidamente

e) aplicar el proceso de decodificación de la transformación de seguridad al valor recibido o recuperado utilizando tal vez información de entrada local adicional y generar las salidas de conformidad con ese proceso de decodificación (en función de la transformación concreta las salidas pueden incluir una copia recuperada del elemento no protegido, una indicación de éxito/fallo de la verificación de la firma o del sello y/o una copia de la signatura para su almacenamiento local para una utilización posterior); seguidamente

f) si la salida de la etapa e) es una copia recuperada del elemento no protegido y si ese elemento es un valor de un tipo ASN.1 especificado en la definición de sintaxis abstracta, se decodificará ese elemento de datos utilizando las mismas reglas de codificación inicial que en la etapa a).

En 7.1.4 se describe la determinación de las reglas de codificación inicial para las etapas a) y f). Obsérvese que, en general, las transformaciones de seguridad pueden actuar sobre elemento de datos distintos de valores de tipo ASN.1 (por ejemplo, cadenas de bits arbitrarias), por lo que no siempre es necesario este proceso de codificación.

La determinación de las reglas de codificación para las etapas c) y d) depende del entorno de almacenamiento o comunicación y es independiente de la transformación de seguridad concreta utilizada.

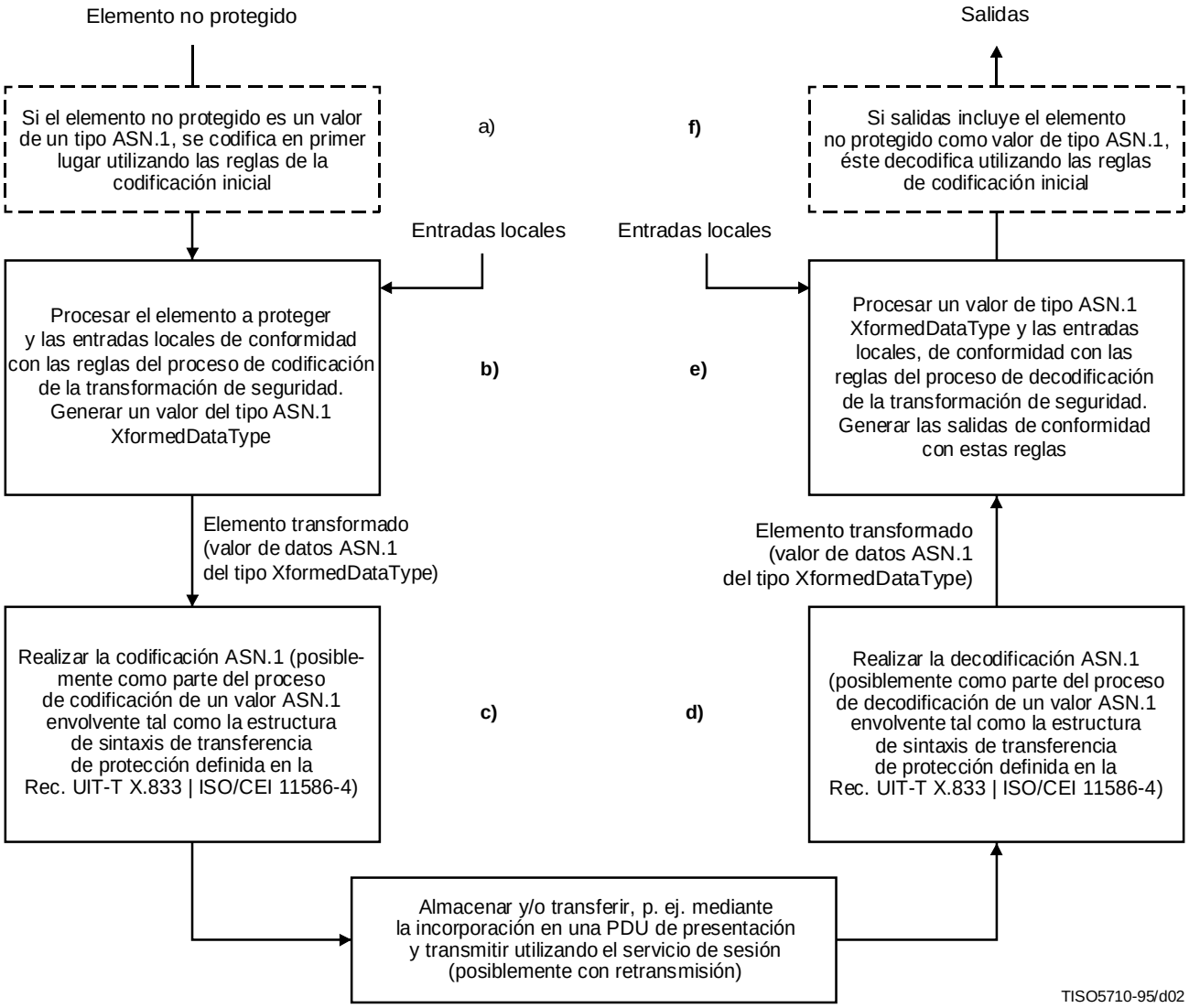


Figura 2 – Transferencia o almacenamiento protegidos de un elemento de datos

7.1.1 Situación arquitectural de las transformaciones de seguridad en las capas superiores de OSI

Una transformación de seguridad funciona en el contexto de una asociación de seguridad entre dos o más sistemas. En cada sistema existe un objeto de seguridad de sistema (SSO) que soporta esa asociación de seguridad. Estos SSO ejecutan los procesos de codificación/decodificación de la transformación de seguridad (por ejemplo, cifrado, generación/verificación de la signatura digital) y almacenan la información de estado de seguridad necesaria (por

ejemplo, claves, algoritmos, parámetros, estado de encadenamiento). El comportamiento interno de esos SSO está gobernado por especificaciones de transformación de seguridad específicas conjuntamente con las especificaciones de soporte, por ejemplo, para algoritmos (que quedan fuera del alcance de esta Recomendación | Norma Internacional). En términos de la Figura 2, las funciones indicadas en las casillas b) y e) se modelan en los SSO.

También existen funciones de comunicación de seguridad (SCF) en las entidades de presentación de los sistemas de codificación y decodificación. Estas SCF sustentan los requisitos de comunicación de los SSO. En términos de la Figura 2, las funciones indicadas en las casillas a), c), d) y f) se modelan en las SCF. En la cláusula 8 de esta Recomendación | Norma Internacional y en la Rec. UIT-T X.833 | ISO/CEI 11586-4 figuran las definiciones de comportamiento de las SCF.

7.1.2 Asociaciones de seguridad

Puede aplicarse repetidamente una transformación de seguridad a una sucesión de valores de datos ordenados lógicamente, por ejemplo, valores de datos de presentación transferidos de forma secuencial en un sentido entre dos sistemas. A cada valor de datos se le aplica la misma protección. La aplicación de una transformación de seguridad a tal secuencia está gobernada por una asociación de seguridad. Entre una pareja de sistemas puede existir al mismo tiempo más de una asociación de seguridad, proporcionando típicamente distintos tipos de protección.

Esta Recomendación | Norma Internacional se refiere a aspectos de una asociación de seguridad que son importantes para las comunicaciones de capas superiores o el almacenamiento de la información. Desde la perspectiva de las capas superiores de OSI, una asociación de seguridad es una forma de asociación ASO.

En esta Recomendación | Norma Internacional se distinguen tres clases de asociaciones de seguridad:

a) *Asociación de seguridad establecida externamente* – Asociación de seguridad establecida con independencia de los casos de utilización, que posee un identificador globalmente único que le permite ser referenciado en el momento de su utilización. En esta Recomendación | Norma Internacional no se especifican los medios para el establecimiento de esta asociación de seguridad, cuya duración temporal no queda restringida por las disposiciones de esta Recomendación | Norma Internacional. El identificador de una asociación de seguridad establecida externamente comprende un valor entero junto con la identidad del sistema que asigna ese valor entero. (Esta última identidad puede conocerse implícitamente, por ejemplo, el emisor o el receptor de los datos, por lo que no siempre es necesario transportar esta integridad en el protocolo.)

b) *Asociación de seguridad ligada a un solo elemento* – Asociación de seguridad aplicable a un único valor de datos de presentación de forma independiente, que no está asociada a un contexto de presentación. Los atributos de la asociación de seguridad se indican explícitamente junto con la codificación de datos de presentación. La duración temporal de una asociación de seguridad explícita (elemento único) está limitada por la duración temporal de un valor de datos de presentación.

c) *Asociación de seguridad explícita ligada al contexto de presentación* – Asociación de seguridad establecida en conjunción con el establecimiento de un contexto de presentación de protección y que se aplica a todos los valores de datos de presentación enviados en un sentido en ese contexto de presentación de protección. Los atributos de la asociación de seguridad se indican explícitamente junto con la codificación del primer valor de datos de presentación en el contexto de presentación de protección. Este tipo de asociación de seguridad únicamente puede aplicarse cuando se proporciona la protección en combinación con la utilización del servicio de presentación de OSI y el protocolo especificado en la Rec. UIT-T X.216 | ISO/CEI 8822 y Rec. UIT-T X.226 | ISO/CEI 8823-1, respectivamente. La duración temporal de la asociación de seguridad es la misma que la duración temporal del contexto de presentación de protección correspondiente.

El funcionamiento de una transformación de seguridad puede ser gobernado por la información del estado de seguridad local y/o por parámetros transferidos o almacenados con los valores de datos codificados. La información de estado de seguridad local puede mantenerse de una aplicación de una transformación de seguridad a la siguiente aplicación en una asociación de seguridad. Por ejemplo, en el caso de transformaciones que proporcionan la integridad de la sucesión de valores de datos de presentación dentro de una asociación de seguridad, la información de estado tal como un número de secuencia de integración o un valor de encadenamiento criptográfico puede mantenerse de una aplicación de la transformación a la siguiente. Asimismo se mantienen los valores de los parámetros estáticos (véase 7.1.3) a lo largo de una asociación de seguridad.

7.1.3 Parámetros de transformación de seguridad

Cuando se utiliza una transformación de seguridad puede ser necesario transferir valores de parámetro entre las funciones de codificación y decodificación junto con los valores de los datos transformados. Hay dos tipos de parámetros:

a) *Parámetros estáticos* – Estos parámetros mantienen valores constantes a lo largo de una asociación de seguridad y se especifican por el codificador de datos antes de que se aplique por primera vez la transformación de seguridad o en el curso de la misma en una asociación de seguridad.

b) *Parámetros dinámicos* – Los valores de estos parámetros pueden cambiar de forma dinámica en el curso de la transformación en una asociación de seguridad. El codificador de los datos indica esos cambios dentro del flujo de datos.

Como ejemplos de parámetros estáticos pueden citarse los siguientes:

– identificador o identificadores del algoritmo o algoritmos utilizados en una transformación de seguridad;

– modo de operación de un algoritmo si es necesario;

– clave o claves, identificador o identificadores de claves utilizados con el algoritmo o algoritmos mencionados anteriormente;

– valor o valores de los vectores de inicialización de ser necesario.

Como ejemplo de parámetro dinámico puede citarse la clave que se modifica tras un cierto periodo de uso.

Los valores de los parámetros pueden codificarse de forma no protegida o exigir por sí mismos protección. Los parámetros no protegidos se transportan en campos explícitos de la sintaxis de transferencia de protección que sustenta la transformación de seguridad. Los parámetros protegidos se consideran como entradas al proceso de codificación de la transformación de seguridad, junto con el valor que debe protegerse. Las reglas de la transformación de seguridad deben establecer la forma de representar estos parámetros, el modo de combinar su representación con el valor de sintaxis abstracta codificado y la manera en que debe procesarse el resultado para generar un valor de datos ASN.1 para su transferencia o almacenamiento.

NOTA – Como ejemplo de parámetros protegidos transportados véase la definición de la transformación de seguridad GULS SIGNED en D.4.

Los datos de parámetros (por ejemplo, claves) requeridos por las transformaciones de seguridad pueden también obtenerse por otros medios, como son:

– intercambios de protocolo de capa de aplicación anteriores (por ejemplo, intercambio de seguridad de obtención de clave transportado por el SESE);

– medios locales (por ejemplo, inserción manual de claves).

7.1.4 Determinación de las reglas de codificación inicial

Las reglas que gobiernan los procesos de codificación inicial (y codificación final), modelados en las casillas a) y f) de la Figura 2, se determinan de alguna de las siguientes formas:

a) una transformación de seguridad puede proporcionar el transporte de una indicación de reglas de codificación inicial en forma de parámetro estático (protegido o no protegido) de la transformación de seguridad;

b) cada especificación de transformación de seguridad identifica, por defecto, reglas de codificación inicial por defecto.

NOTA – Cuando se utilicen signaturas digitales para el no repudio del elemento transformado (esto es, los datos firmados) pueden tener que almacenarse en un sistema receptor y/o retransmitirse a otra entidad. En tales circunstancias debe preservarse el conocimiento de las reglas de codificación inicial utilizadas en la determinación de las necesidades de signatura. En el caso de signaturas digitales, se recomienda la utilización de las reglas de codificación por defecto contenidas en la especificación de la transformación de seguridad. De este modo, puede preservarse la información requerida almacenando/retransmitiendo el identificador de la transformación de seguridad junto con la signatura.

7.1.5 Utilización de transformaciones de seguridad en una conexión de presentación OSI

La capa de presentación OSI asocia una sintaxis de transferencia a cada sintaxis abstracta utilizada. Cuando se emplea una transformación de seguridad a la sintaxis de transferencia se le denomina sintaxis de transferencia de protección.

De conformidad con la Rec. X.208 del CCITT | ISO/CEI 8824 pueden transferirse los valores de datos de presentación:

a) dentro de un contexto de presentación negociado; o

b) fuera de un contexto de presentación (facultativamente cuando se emplee la notación ASN.1 EXTERNAL o EMBEDDED PDV).

En ambos casos, se representa el valor de datos de presentación que debe protegerse utilizando una sintaxis de transferencia de protección. La sintaxis de transferencia de protección definida de conformidad con la Rec. UIT-T X.833 | ISO/CEI 11586-4, sustenta la comunicación de parámetros de transformación de seguridad estáticos y dinámicos.

El caso a) anterior implica un contexto de presentación de protección. Todos los valores de datos de presentación transferidos en un sentido dentro de un contexto de presentación de protección se protegen utilizando la misma transformación de seguridad y se gobiernan mediante la asociación de seguridad propia. Cuando se establece un contexto de presentación de protección (utilizando los procedimientos de establecimiento de un contexto de presentación especificados en la Rec. UIT-T X.216 | ISO/CEI 8822 y en la Rec. UIT-T X.226 | ISO/CEI 8823-1), el primer valor de datos de presentación en cada sentido en ese contexto de presentación deberá ser:

a) referenciar una asociación de seguridad establecida externamente; o

b) definir una nueva asociación de seguridad ligada al contexto de presentación.

Cuando se codifique un valor de datos de presentación fuera de un contexto de presentación, el valor de datos de presentación será:

a) referenciar una asociación de seguridad establecida externamente; o

b) definir una nueva asociación de seguridad ligada a un solo elemento.

En una conexión de presentación OSI, se aplican asociaciones de seguridad distintas a cada sentido de flujo. Esas asociaciones de seguridad pueden utilizar la misma transformación de seguridad, pero no es necesario que lo hagan.

NOTA – La restricción anterior (es decir, que cuando se utiliza el protocolo de presentación OSI, se aplican asociaciones de seguridad distintas a cada dirección del flujo) asegura que no pueda haber variable de estado criptográficas comunes compartidas entre los dos sentidos de flujo. Si pudiera existir dicho estado compartido, habría necesidad de que los elementos de protocolo de mantenimiento de estado complejos de la capa de presentación tratasen dichos eventos como resincronización de la capa de sesión. En la práctica, es probable que las asociaciones de seguridad separadas para los dos sentidos tengan atributos comunes derivados de una asociación de seguridad que las abarque.

7.1.6 Utilización de transformaciones de seguridad con independencia del protocolo de presentación OSI

Pueden utilizarse transformaciones de seguridad independientemente del protocolo de presentación OSI, por ejemplo, para la protección del almacenamiento. Son aplicables los conceptos y procedimientos descritos en 7.1.2 a 7.1.5, con las siguientes restricciones.

Se representan todos los valores de datos de presentación protegidos fuera de los contextos de presentación.

Puede utilizarse una asociación de seguridad ligada a un solo elemento o una asociación de seguridad establecida externamente. No pueden aplicarse asociaciones de seguridad ligadas al contexto de presentación. Cuando la información protegida no se intercambia, sino que se protege únicamente para su uso por parte del originador de la misma, pueden también emplearse transformaciones de seguridad sin asociaciones de seguridad.

Si se utiliza una asociación de seguridad establecida externamente, la duración temporal de la asociación de seguridad establecida externamente deberá abarcar la duración temporal del almacenamiento de los datos protegidos.

7.2 Notación para la especificación de transformaciones de seguridad

Las especificaciones de la transformación de seguridad comprenden especificaciones de elemento de datos que deben ser reconocidas por la estructura de la sintaxis de transferencia de protección. Con este fin se proporciona la siguiente definición de clase de objeto de información ASN.1 (véase la Rec. UIT-T X.681 | ISO/CEI 8824-2):

```
SECURITY-TRANSFORMATION ::= CLASS
-- This information object class definition is for use when
-- specifying a particular instance of a security transformation.
{
  &sT-Identifier      OBJECT IDENTIFIER UNIQUE,
  -- Identifier to be used in signalling the application
  -- of the particular security transformation
```

```
    &initialEncodingRules OBJECT IDENTIFIER
        DEFAULT {joint-iso-ccitt asn1 (1) ber-derived (2)
        canonical-encoding (0)},
    -- Default initial encoding rules to generate a bit
    -- string prior to applying the encoding process of a
    -- security transformation.
    &StaticUnprotectedParm OPTIONAL,
    -- ASN.1 type for conveying static unprotected parameters
    &DynamicUnprotectedParm OPTIONAL,
    -- ASN.1 type for conveying dynamic unprotected parameters
    &XformedDataType,
    -- ASN.1 type of the ASN.1 value produced by the security
    -- transformations encoding process
    &QualifierType  OPTIONAL
    -- &QualifierType specifies the ASN.1 type of the qualifier
    -- parameter used with the PROTECTED-Q notation.
}
WITH SYNTAX
-- The following syntax is used to specify a particular security
-- transformation.
{
    IDENTIFIER                      &sT-Identifier
    [ INITIAL-ENCODING-RULES        &initialEncodingRules ]
    [ STATIC-UNPROT-PARM            &StaticUnprotectedParm ]
    [ DYNAMIC-UNPROT-PARM           &DynamicUnprotectedParm ]
    XFORMED-DATA-TYPE               &XformedDataType
    [ QUALIFIER-TYPE                &QualifierType ]
}
```

En el Anexo D se facilitan ejemplos de utilización de esta notación.

La especificación de la transformación de seguridad debe también estipular los siguientes detalles (aunque en esta Recomendación | Norma Internacional no se proporciona ninguna notación formal para sustentar esa especificación):

- *Proceso de codificación* – Descripción del proceso de transformación aplicado en el extremo de codificación al elemento no protegido y de los parámetros protegidos transferidos, a fin de generar el valor transformado resultante (que es un valor ASN.1 del tipo &XformedDataType).
- *Entradas locales al proceso de codificación* – Lista de entradas al proceso de codificación deducidas localmente.
- *Proceso de decodificación* – Descripción del proceso de transformación aplicado, en el extremo decodificador, al valor transformado recibido o recuperado (que es del tipo &XformedDataType) a fin de generar la cadena de bits de datos no protegidos resultante (si existe) y los valores de los parámetros protegidos transferidos.
- *Entradas locales al proceso de decodificación* – Lista de las entradas al proceso de decodificación, obtenidas localmente.
- *Salidas del proceso de decodificación* – Lista de salidas del proceso de decodificación (pueden o no incluir el valor recuperado del elemento no protegido).
- *Parámetros* – Descripción del significado semántico de todos los parámetros, valores por defecto de los parámetros y circunstancias en las que podrían producirse modificaciones en el parámetro dinámico.
- *Calificadores de la transformación* – Descripción de las reglas aplicables a los calificadores de la transformación especificados en la invocación (si existen) que se utilizan en esta transformación.
- *Errores* – Descripción de las condiciones de error que pudieran detectarse durante el proceso de decodificación.

8 Notación de sintaxis abstracta para la protección selectiva de los campos

La notación de sintaxis abstracta que se describe a continuación se utiliza para la especificación de los requisitos de protección abstracta para un tipo de datos ASN.1 seleccionado. La protección requerida se pone en correspondencia con una transformación de seguridad de un conjunto de ellas que proporciona (en un nivel abstracto) la forma de protección requerida. Algunas transformaciones de seguridad aceptan calificadores de entrada para comprobar el funcionamiento de la protección requerida, por ejemplo, el identificador de la asociación de seguridad para el que debe aplicarse la protección. Para estos casos, se define una ampliación de la notación básica que permita la especificación de calificadores por parte del usuario de la notación.

Esta cláusula especifica lo siguiente:

a) la notación de sintaxis abstracta protegida básica para la especificación de los requisitos de protección abstracta para un campo seleccionado, en una especificación de sintaxis abstracta;

b) la notación de sintaxis abstracta protegida calificada para especificar los requisitos de protección abstracta junto con el calificador asociado, para un campo seleccionado, en una especificación de sintaxis abstracta;

c) la notación de correspondencia de protección para especificar las posibles correspondencias con una o más transformaciones de seguridad que proporcionan la seguridad exigida.

8.1 Notación básica

A fin de ayudar al redactor de una sintaxis abstracta en la indicación de los requisitos de protección selectiva de los campos, se define el siguiente tipo parametrizado ASN.1 (véase la Rec. UIT-T X.683 | ISO/CEI 8824-4):

```
PROTECTED {BaseType, PROTECTION-MAPPING: protectionReqd}  ::=
CHOICE
{
  dirEncrypt BIT STRING (CONSTRAINED BY {BaseType
        -- dirEncrypt is for use only with the
        -- dirEncryptedTransformation,
        -- and generates the same encoding as the
        -- X.509/9594-8 ENCRYPTED type--}),
  dirSign   SEQUENCE
       {
          baseType BaseType  OPTIONAL,
           -- must be present for dirSignedTransformation
           -- and must be omitted for
           -- dirSignatureTransformation
          algorithmId AlgorithmIdentifier,
          encipheredHash BIT STRING (CONSTRAINED BY
               {BaseType -- contains enciphered hash
                         -- of a value of BaseType --})
       }
        -- dirSign is for use only with the
        -- dirSignedTransformation or
        -- dirSignatureTransformation, and generates
        -- the same encoding as the corresponding
        -- X.509/9594-8 SIGNED or SIGNATURE type--,
  noTransform [0] BaseType,
        -- noTransform invokes no security transformation.
        -- Subject to security policy, noTransform may be used
        -- if adequate protection is provided by lower layers
        -- and any application relays through which the data
        -- may pass are trusted to maintain the required
        -- protection. This alternative may only be used
        -- if protectionReqd.&bypassPermitted is TRUE.
```

```
    direct    [1] SyntaxStructure
                {{protectionReqd.&SecurityTransformation}},
              -- direct generates a protecting transfer syntax
              -- value, which is encoded using the same encoding
              -- rules as the surrounding ASN.1 (The type
              -- SyntaxStructure is imported from Rec. X.833 |
              -- ISO/IEC 11586-4)
    embedded  [2] EMBEDDED PDV (WITH COMPONENTS {
                  identification (WITH COMPONENTS {
                      presentation-context-id,
                      context-negotiation (WITH COMPONENTS {
                          transfer-syntax (CONSTRAINED BY
                            {OBJECT IDENTIFIER :
                            protectionReqd.&protTransferSyntax})}),
                      transfer-syntax (CONSTRAINED BY
                            {OBJECT IDENTIFIER :
                            protectionReqd.&protTransferSyntax})}),
              data-value (WITH COMPONENTS {notation (BaseType)})
               -- The data value encoded is a value of type BaseType
              })
}
-- BaseType is the type to be protected, and protectionReqd is an ASN.1
-- object of class PROTECTION-MAPPING. The use of PROTECTED requires
-- the importation into the user's module of the PROTECTED parameterized
-- type, together with the necessary PROTECTION-MAPPING object
-- definition.
```

En 8.3 se examina la clase de objeto PROTECTION-MAPPING y su significación. El conjunto de objetos posibles para «protectionReqd» variará según las distintas especificaciones de sintaxis abstracta, dependiendo de la gama de transformaciones diferentes requerida. Las correspondencias entre objetos PROTECTION-MAPPING y transformaciones figuran en un conjunto de definiciones de objetos PROTECTION-MAPPING. Este conjunto de definiciones puede estar especificado en un módulo ASN.1 separado a partir de la especificación de sintaxis abstracta (independiente del mecanismo) y de la definición de la transformación (independiente de la aplicación).

Se dispone de varias alternativas en el CHOICE para su empleo en distintas circunstancias, como sigue:

- *dirEncrypt y dirSign* – Estas alternativas generan simplemente el &XformedDataType de la transformación de seguridad utilizada. Se dispone de estas alternativas para proporcionar un medio mediante el cual la notación PROTECTED puede generar codificaciones de bits idénticas a los tipos parametrizados ENCRYPTED, SIGNED y SIGNATURE definidos en la Rec. UIT-T X.509 | ISO/CEI 9594-8.

- *noTransform* – Esta alternativa no utiliza la transformación de seguridad. Se permite si la correspodencia de protección utilizada (véanse 8.3 y 8.4) indica &bypassPermitted = TRUE. El elemento se codifica en su forma sin protección. De acuerdo con la política de seguridad, puede utilizarse noTransform si se proporciona una protección adecuada en las capas inferiores y se confía en que todos los retransmisores de aplicación a través de los cuales deben pasar los datos mantendrán la protección requerida.

- *direct* – Esta alternativa importa directamente un valor de sintaxis de transferencia de protección como el definido en la Rec. UIT-T X.833 | ISO/CEI 11586-4 en la especificación ASN.1 envolvente. Sustenta la utilización de una asociación de seguridad establecida externamente o de una asociación de seguridad de un solo elemento. No permite el empleo de un contexto de presentación negociado. Se obliga a que las reglas de codificación utilizadas para la codificación de la estructura de sintaxis de transferencia de protección [modeladas en las casillas c) y d) de la Figura 2 de 7.1] sean las mismas que las utilizadas para el tipo ASN.1 que abarca la notación PROTECTED.

- *embedded* – Esta alternativa proporciona la máxima flexibilidad, incluidas la aptitud para asociar la protección con un contexto de presentación negociado y la posibilidad de utilizar una sintaxis de transferencia de protección distinta de la definida en la Rec. UIT-T X.833 | ISO/CEI 11586-4.

NOTA – Se recomienda que el empleo de estas opciones se seleccione como sigue:

a) Empleo de la opción directa si no se aplica ninguna de las b), c) o d).

b) Cuando se requiera compatibilidad de bits con los tipos parametrizados ENCRYPTED, SIGNED o SIGNATURE por razones de compatibilidad hacia atrás, se utiliza la opción dirEncrypt o dirSign, según sea aplicable.

c) Cuando la correspondencia de protección utilizada indica &bypassPermitted = TRUE y cuando lo permita la política de seguridad, emplear la opción noTransform.

d) Cuando sea necesario asociar la protección con un contexto de presentación negociado, utilizar la opción insertada.

Al procesar un valor en un campo protegido en el sistema de decodificación pueden detectarse condiciones de error. Puede emplearse la notación de tratamiento de las excepciones ASN.1 definida en la Rec. UIT-T X.682 | ISO/CEI 8824-3 para afrontar esas condiciones de error.

En I.1 figuran ejemplos de utilización de esa notación.

8.2 Notación con el calificador de transformación

Como alternativa a la notación PROTECTED descrita en 8.1, la notación PROTECTED-Q permite a su usuario proporcionar adicionalmente un parámetro calificador. Este parámetro calificador se utiliza para una de las siguientes finalidades o ambas:

a) identificar una asociación de seguridad específica establecida externamente;

b) proporcionar uno o más parámetros para su uso por la transformación de seguridad, por ejemplo, un algoritmo, un modo de funcionamiento y/o identificadores de clave.

NOTA – Algunos identificadores de algoritmo pueden implicar un modo de funcionamiento específico. En otros casos, el modo de funcionamiento puede especificarse como un parámetro adicional.

Pueden especificarse los calificadores múltiples utilizando un tipo ASN.1 SEQUENCE o SET apropiado. Dentro del sistema de codificación, las funciones de sistemas locales utilizan el calificador para determinar el SSO apropiado y/o transportar un parámetro al SSO. Todo calificador transportado a un SSO debe ser compatible con la transformación de seguridad en uso, como se indica en la especificación de transformación de seguridad. Cuando la correspondencia de protección especificada permita la elección de transformaciones de seguridad, la transformación seleccionada por cualquier ejemplo de utilización debe ser aquella cuyo &QualifierType sea coherente con el valor especificado por el usuario de la notación PROTECTED-Q. El valor del calificador debe ser (aunque no necesariamente) transportado al sistema de decodificación dentro de la sintaxis de transferencia de protección (por ejemplo, como el identificador de la asociación de seguridad externa establecida o como un parámetro de transformación de seguridad).

Se define el siguiente tipo ASN.1 parametrizado (véase la Rec. UIT-T X.683 | ISO/CEI 8824-4).

```
PROTECTED-Q {BaseType, PROTECTION-MAPPING: protectionReqd,
  PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier} ::=
    PROTECTED {BaseType, protectionReqd} (CONSTRAINED BY
    {PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier
        -- The value of qualifier must be made available to
        -- the security transformation used
       })
  -- BaseType is the type to be protected, and protectionReqd is an object of class
  -- PROTECTION-MAPPING. The use of PROTECTED requires the importation into the user's
  -- module of the PROTECTED parameterized type, together with the necessary
  -- PROTECTION-MAPPING object definition.
```

En I.2 e I.3 figuran ejemplos de utilización de esta notación.

8.3 Correspondencia entre requisitos de protección y transformaciones de seguridad

Una correspondencia de protección relaciona un requisito de protección identificado por un nombre en una especificación de sintaxis abstracta, con una transformación específica que debe utilizarse para cumplir ese requisito. Se introduce este concepto para permitir la especificación de esas correspondencias de forma separada de la especificación de la sintaxis abstracta principal que así puede ser independiente del mecanismo. Para la protección denominada en una sintaxis abstracta, la transformación real utilizada puede ser distinta en contexto de aplicación diferente.

Una correspondencia de protección puede limitar la selección de una transformación de seguridad de las siguientes formas:

- proporcionando una lista de transformaciones de seguridad; en el momento de utilización se seleccionará de esta lista una transformación de seguridad determinada sobre la base de la política de seguridad local u otras consideraciones del sistema local;
- estableciendo reglas de selección especializadas.

Como ejemplos de correspondencia de protección definidas completamente en el Anexo E, pueden citarse los siguientes:

- *Confidencialidad* – Datos protegidos con confidencialidad mediante cifrado/descifrado, pero permitiendo saltarse el cifrado/descifrado si así lo dicta la política de seguridad.
- *Cifrado* – Realización del cifrado/descifrado mediante un tipo de algoritmo no especificado.
- *Firmada* – Generación/verificación de una signatura digital para su transferencia junto con los datos firmados.
- *Signatura* – Generación/verificación de una signatura digital para su transferencia separada respecto de los datos firmados.

Son posibles otras correspondencias de protección, por ejemplo, para la puesta en correspondencia específica con transformaciones para cifrado de claves públicas, cifrado simétrico, sellado, funciones hash o cifrado unilateral.

8.4 Notación para la especificación de correspondencias de protección

Para definir correspondencias de protección específicas se ha previsto la siguiente definición de clase de objeto de información ASN.1 (véase la Rec. UIT-T X.681 | ISO/CEI 8824-2):

```
PROTECTION-MAPPING ::= CLASS
{
    &SecurityTransformation SECURITY-TRANSFORMATION,
    -- &SecurityTransformation specifies an ASN.1 object set of the SECURITY-TRANSFORMATION class.
    -- Use of the particular protection mapping implies use of one of the specified transformations,
    -- with the choice being left to the encoding system. Rules for selecting between these security
    -- transformations may be specified in comments.
    &protTransferSyntax OBJECT IDENTIFIER
            DEFAULT {joint-iso-ccitt genericULS (20)
            generalTransferSyntax (2)},
    -- Identifies the particular protecting transfer syntax to be used in an EMBEDDED PDV
    -- encoding for the embedded option.
    &bypassPermitted BOOLEAN DEFAULT FALSE
    -- Indicates if bypassing of protection is permitted
}
WITH SYNTAX
{
    SECURITY-TRANSFORMATION          &SecurityTransformation
    [ PROTECTING-TRANSFER-SYNTAX     &protTransferSyntax ]
    [ BYPASS-PERMITTED               &bypassPermitted ]
}
```

9 Conformidad

Un sistema que pretenda ser conforme con esta Recomendación | Norma Internacional, debe serlo también respecto de la utilización de los intercambios de seguridad de GULS o transformaciones de seguridad especificadas en los Anexos C y D:

a) cuando haga uso de cualquiera de los intercambios de seguridad especificados en el Anexo C e identificados por el identificador de objetos de ASN.1 para el módulo «GulsSecurityExchanges» del Anexo C, el sistema deberá sustentar las estipulaciones ASN.1 aplicables y cualesquiera otras asociadas del Anexo C;

b) cuando haga uso de cualquiera de las transformaciones de seguridad especificadas en el Anexo D e identificadas por el identificador de objetos de ASN.1 para el módulo «GulsSecurityTransformations» del Anexo D, el sistema deberá sustentar las estipulaciones ASN.1 aplicables y las estipulaciones asociadas del Anexo D.

En las subcláusulas pertinentes de los Anexos C y D se establecen requisitos específicos de conformidad estática y dinámica.

La implementación de las construcciones definidas en los Anexos C, D y E es facultativa para el usuario de esta Recomendación | Norma Internacional y no constituye un requisito de conformidad obligatorio.

Anexo A

Definiciones de ASN.1

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

El módulo de ASN.1 que sigue proporciona las especificaciones ASN.1 definitivas para el texto principal de esta Recomendación | Norma Internacional.

```
Notation {joint-iso-ccitt genericULS (20)
        modules (1) notation (1)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

-- EXPORTS All --

IMPORTS
  -- From Directory Standards: --
  informationFramework, selectedAttributeTypes,
  authenticationFramework
      FROM UsefulDefinitions {joint-iso-ccitt ds (5) module (1)
            usefulDefinitions (0) 2}
  Name
      FROM InformationFramework          informationFramework
  UniqueIdentifier
      FROM SelectedAttributeTypes        selectedAttributeTypes
  AlgorithmIdentifier
      FROM AuthenticationFramework       authenticationFramework

  -- From Other GULS Modules: --
  genericProtectingTransferSyntax
      FROM ObjectIdentifiers {joint-iso-ccitt genericULS (20)
      modules (1) objectIdentifiers (0)}
  SyntaxStructure { }
      FROM GenericProtectingTransferSyntax
            genericProtectingTransferSyntax;

-- ************************************************ --
-- Notation for security identity and SA-identifiers --
-- ************************************************ --

-- Values of the SecurityIdentity type are used to identify entities
-- which assign externally-established security association identifiers,
-- and for other security-related purposes requiring globally-unique
-- identifiers.

SecurityIdentity ::=  CHOICE
{
  directoryName          Name,
  objectIdentifier       OBJECT IDENTIFIER
}
ExternalSAID ::=  SEQUENCE
{
  localSAID              INTEGER,
  assignerIdentity       SecurityIdentity    OPTIONAL
  -- Identity of the system which assigned the integer value
}
```

```
-- ****************************************** --
-- Notation for specifying security exchanges --
-- ****************************************** --

SECURITY-EXCHANGE ::= CLASS
-- This information object class definition is for use when
-- specifying a particular instance of a security exchange.
{
  &SE-Items         SEC-EXCHG-ITEM,
  -- This is an ASN.1 information object set, comprising a set
  -- of security exchange items
  &sE-Identifier    Identifier          UNIQUE
  -- A local or global identifier for the particular security
  -- exchange
}
WITH SYNTAX
-- The following syntax is used to specify a particular security
-- exchange.
{
  SE-ITEMS          &SE-Items
  IDENTIFIER        &sE-Identifier
}
Identifier ::= CHOICE
{
  local             INTEGER,
  global            OBJECT IDENTIFIER
}

SEC-EXCHG-ITEM ::= CLASS
{
  &ItemType,
      -- ASN.1 type for this exchange item
  &itemId           INTEGER,
      -- Identifier for this item, e.g. 1, 2, 3, ..
  &Errors           SE-ERROR            OPTIONAL
      -- Optional list of errors which may result from
      -- transfer of this item
}
WITH SYNTAX
{
  ITEM-TYPE         &ItemType
  ITEM-ID           &itemId
  [ERRORS           &Errors]
}

SE-ERROR ::= CLASS
{
  &ParameterType    OPTIONAL,
  -- ASN.1 type of a parameter to accompany the signalling
  -- of the error condition back to the sender of the SEI
  &errorCode        Identifier          UNIQUE
  -- An identifier used in signalling the error condition
  -- back to the sender of the SEI
}
WITH SYNTAX
{
  [PARAMETER        &ParameterType]
  ERROR-CODE        &errorCode
}
```

```
-- ********************************************** --
-- Notation for specifying security transformations --
-- ********************************************** --

SECURITY-TRANSFORMATION ::= CLASS
-- This information object class definition is for use when
-- specifying a particular instance of a security transformation.
{
   &sT-Identifier      OBJECT IDENTIFIER UNIQUE,
   -- Identifier to be used in signalling the application
   -- of the particular security transformation
   &initialEncodingRules OBJECT IDENTIFIER
        DEFAULT {joint-iso-ccitt asn1 (1) ber-derived (2)
        canonical-encoding (0)},
   -- Default initial encoding rules to generate a bit
   -- string prior to applying the encoding process of a
   -- security transformation.
   &StaticUnprotectedParm OPTIONAL,
   -- ASN.1 type for conveying static unprotected parameters
   &DynamicUnprotectedParm OPTIONAL,
   -- ASN.1 type for conveying dynamic unprotected parameters
   &XformedDataType,
   -- ASN.1 type of the ASN.1 value produced by the security
   -- transformations encoding process
   &QualifierType  OPTIONAL
   -- &QualifierType specifies the ASN.1 type of the qualifier
   -- parameter used with the PROTECTED-Q notation.

}
WITH SYNTAX
-- The following syntax is used to specify a particular security
-- transformation.
{
   IDENTIFIER                          &sT-Identifier
   [ INITIAL-ENCODING-RULES            &initialEncodingRules ]
   [ STATIC-UNPROT-PARM                &StaticUnprotectedParm ]
   [ DYNAMIC-UNPROT-PARM               &DynamicUnprotectedParm ]
   XFORMED-DATA-TYPE                   &XformedDataType
   [ QUALIFIER-TYPE                    &QualifierType ]
}


-- ************************************************ --
-- Notation for specifying selective field protection --
-- ************************************************ --

PROTECTED {BaseType, PROTECTION-MAPPING: protectionReqd} ::=
CHOICE
{
   dirEncrypt BIT STRING (CONSTRAINED BY {BaseType
            -- dirEncrypt is for use only with the
            -- dirEncryptedTransformation,
            -- and generates the same encoding as the
            -- X.509/9594-8 ENCRYPTED type--}),
   dirSign    SEQUENCE
          {
             baseType BaseType  OPTIONAL,
               -- must be present for dirSignedTransformation
               -- and must be omitted for
               -- dirSignatureTransformation
```

```
            algorithmId AlgorithmIdentifier,
            encipheredHash BIT STRING (CONSTRAINED BY
                {BaseType -- contains enciphered hash
                          -- of a value of BaseType --})
        }
        -- dirSign is for use only with the
        -- dirSignedTransformation or
        -- dirSignatureTransformation, and generates
        -- the same encoding as the corresponding
        -- X.509/9594-8 SIGNED or SIGNATURE type--,
  noTransform [0] BaseType,
        -- noTransform invokes no security transformation.
        -- Subject to security policy, noTransform may be used
        -- if adequate protection is provided by lower layers
        -- and any application relays through which the data
        -- may pass are trusted to maintain the required
        -- protection. This alternative may only be used
        -- if protectionReqd.&bypassPermitted is TRUE,
  direct    [1] SyntaxStructure
            {{protectionReqd.&SecurityTransformation}},
        -- direct generates a protecting transfer syntax
        -- value, which is encoded using the same encoding
        -- rules as the surrounding ASN.1  (The type
        -- SyntaxStructure is imported from Rec. X.833 |
        -- ISO/IEC 11586-3)
  embedded  [2] EMBEDDED PDV (WITH COMPONENTS {
              identification (WITH COMPONENTS {
                  presentation-context-id,
                  context-negotiation (WITH COMPONENTS {
                      transfer-syntax (CONSTRAINED BY
                        {OBJECT IDENTIFIER :
                        protectionReqd.&protTransferSyntax})}),
                  transfer-syntax (CONSTRAINED BY
                      {OBJECT IDENTIFIER :
                      protectionReqd.&protTransferSyntax})}),
        data-value (WITH COMPONENTS {notation (BaseType)})
        -- The data value encoded is a value of type BaseType
        })
}
-- BaseType is the type to be protected, and protectionReqd is an ASN.1
-- object of class PROTECTION-MAPPING. The use of PROTECTED requires
-- the importation into the user's module of the PROTECTED parameterized
-- type, together with the necessary PROTECTION-MAPPING object
-- definition.

PROTECTED-Q {BaseType, PROTECTION-MAPPING: protectionReqd,
  PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier}  ::=
    PROTECTED {BaseType, protectionReqd} (CONSTRAINED BY
    {PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier
        -- The value of qualifier must be made available to
        -- the security transformation used
        })
  -- BaseType is the type to be protected, and protectionReqd is an
  -- object of class PROTECTION-MAPPING. The use of PROTECTED requires
  -- the importation into the user's module of the PROTECTED parameterized
  -- type, together with the necessary PROTECTION-MAPPING object
  -- definition.
```

```
-- ****************************************** --
-- Notation for specifying protection mappings --
-- ****************************************** --

PROTECTION-MAPPING ::=  CLASS
{
    &SecurityTransformation SECURITY-TRANSFORMATION,
     -- &SecurityTransformation specifies an ASN.1 object set of the
     -- SECURITY-TRANSFORMATION class. Use of the particular
     -- protection mapping implies use of one of the specified
     -- transformations, with the choice being left to the
     -- encoding system. Rules for selecting between these security
     -- transformations may be specified in comments.
    &protTransferSyntax OBJECT IDENTIFIER
        DEFAULT {joint-iso-ccitt genericULS (20)
        generalTransferSyntax (2)},
     -- Identifies the particular protecting transfer syntax to
     -- be used in an EMBEDDED PDV encoding for the embedded
     -- option.
    &bypassPermitted BOOLEAN DEFAULT FALSE
     -- Indicates if bypassing of protection is permitted
}
WITH SYNTAX
{
    SECURITY-TRANSFORMATION             &SecurityTransformation
    [ PROTECTING-TRANSFER-SYNTAX        &protTransferSyntax ]
    [ BYPASS-PERMITTED                  &bypassPermitted ]
}

END
```

Anexo B

Registro de intercambios de seguridad y transformaciones de seguridad

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

B.1 Introducción

La identificación de intercambios de seguridad y transformaciones de seguridad para su utilización de conformidad con las diversas partes de las especificaciones de seguridad genéricas de capas superiores, requiere una denominación inequívoca de esos objetos de seguridad. En este anexo se especifican los procedimientos para la asignación de esos nombres.

B.2 Procedimientos de registro

En esta subcláusula se especifican procedimientos de registro para intercambios de seguridad y transformaciones de seguridad especificadas:

a) en Recomendaciones UIT-T | Normas Internacionales, o

b) por organizaciones que tengan necesidad de los mismos.

B.2.1 Registro en Recomendaciones UIT-T | Normas Internacionales

En algunos casos se especifican los nombres de los intercambios de seguridad o transformaciones de seguridad en Recomendaciones UIT-T | Normas Internacionales que hacen referencia a esta Recomendación UIT-T | Norma Internacional. El nombre deberá definirse de conformidad con la Rec. X.660 del CCITT | ISO/CEI 9834-1. No está prevista la participación de autoridades de registro internacional que se ocupen de estos tipos de objetos de información.

La Recomendación UIT-T | Norma Internacional referenciante asignará un nombre conforme con la Rec. X.660 del CCITT | ISO/CEI 9834-1, aunque no es necesario que referencie la Rec. X.660 del CCITT | ISO/CEI 9834-1.

B.2.2 Registro por alguna organización que lo necesite

La asignación de nombres para el intercambio de seguridad, transformación de seguridad o especificaciones de etiqueta de seguridad será conforme con los procedimientos generales y revestirá la forma especificada en la Rec. X.660 del CCITT | ISO/CEI 9834-1.

Las organizaciones que deseen asignar estos nombres deberán buscar un superior apropiado en el árbol de denominación de la Rec. X.660 del CCITT | ISO/CEI 9834-1 y solicitar que le sea asignado un arco al mismo.

NOTA – Esto afecta a organismos nacionales ISO/CEI, organizaciones con designadores de código internacional asignados según ISO 6523, administraciones de telecomunicaciones y empresas de explotación reconocidas (EER).

B.3 Otros registros pertinentes

Las definiciones de transformaciones de seguridad pueden, aunque no están obligadas a ello, utilizar asientos de registro en el registro de algoritmos de criptografía establecidos de conformidad con ISO/CEI 9979 para su posible utilización como parámetros de transformación de seguridad.

Anexo C

Especificaciones de intercambios de seguridad

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

Los intercambios de seguridad pueden definirse en Recomendaciones UIT-T | Normas Internacionales o definirse como normas externas registradas por cualquier organización apta para la asignación de identificadores de objetos. Las definiciones de intercambios de seguridad deberán tener la mayor gama de aplicación posible, de forma que puedan reutilizarse en múltiples aplicaciones. En este anexo se definen algunos intercambios de seguridad que se consideran generalmente útiles. No hay ningún requisito implícito de aplicaciones o implementaciones de los mismos para emplear los intercambios de seguridad específicos aquí definidos con preferencia a otros intercambios de seguridad.

C.1 Intercambio de seguridad de directorio (unilateral)

El intercambio de seguridad de directorio (unilateral) se basa en el intercambio de autenticación utilizado en el protocolo de directorio (Rec. UIT-T X.511 | ISO/CEI 9594-3) para una autenticación de entidad unilateral simple o rigurosa. Para más detalles sobre el elemento de datos de credenciales, véase la Rec. UIT-T X.511 | ISO/CEI 9594-3 y para una descripción de la semántica asociada, consúltese la Rec. UIT-T X.509 | ISO/CEI 9594-8.

```
dirAuthenticationOneWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS    {credentials}
  IDENTIFIER   global : {securityExchanges dir-authent-one-way (1)}
}
credentials    SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID     1
}
```

Este intercambio de seguridad implica la transferencia de un único SEI del reclamante al verificador. No se definen errores. La indicación de las condiciones de error se reserva a otro protocolo de aplicación.

C.1.1 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá cumplir los siguientes requisitos de conformidad:

- *Requisitos de declaración* – Un implementador deberá indicar si la implementación actúa como iniciador (inicia el intercambio de seguridad), respondedor (responde a una iniciación procedente de otro sistema) o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como iniciador deberá ser capaz de generar el siguiente elemento de intercambio de seguridad: credenciales. Una implementación que actúe como respondedor deberá ser capaz de procesar el siguiente elemento de intercambio de seguridad: credenciales.
- *Requisitos dinámicos* – Una implementación deberá establecer los procedimientos aplicables descritos en este anexo, en la Rec. UIT-T X.511 | ISO/CEI 9594-3 y en la Rec. UIT-T X.509 | ISO/CEI 9594-8.

C.2 Intercambio de autenticación de directorio (bilateral)

El intercambio de autenticación de directorio (bilateral) se basa en el intercambio de autenticación utilizado en el protocolo de directorio (Rec. UIT-T X.511 | ISO/CEI 9594-3) para la autenticación de entidad mutua intensa o simple. Para más detalles sobre el elemento de datos de credenciales, véase la Rec. UIT-T X.511 | ISO/CEI 9594-3 y para la descripción de la semántica asociada, consúltese la Rec. UIT-T X.509 | ISO/CEI 9594-8.

```
dirAuthenticationTwoWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS    {initiatorCredentials | responderCredentials}
  IDENTIFIER   global : {securityExchanges dir-authent-two-way (2)}
}
initiatorCredentials     SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        1
  ERRORS  {authenticationFailure}
}
responderCredentials     SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE      DirectoryAbstractService.Credentials
  ITEM-ID          2
}
authenticationFailure    SE-ERROR ::=
{
  PARAMETER    DirectoryAbstractService.SecurityProblem
  ERROR-CODE    local : 1
}
```

Este intercambio de seguridad implica dos elementos de intercambio de seguridad, el primero de los cuales se transfiere del iniciador al respondedor. Si tras la primera transferencia se detecta un error, el respondedor deberá abortar el intercambio de seguridad. Facultativamente, puede utilizar el código de error authenticationFailure o puede abortar sin especificar el motivo del error. Si tras la primera transferencia no se detecta ningún error, se transfiere el SEI responderCredentials del respondedor al iniciador. La indicación de condiciones de error se reserva para otro protocolo de aplicación.

C.2.1 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá satisfacer los siguientes requisitos de conformidad:

- *Requisitos de declaración* – Un implementador deberá indicar si la implementación actúa como iniciador (inicia el intercambio de seguridad), respondedor (responde a una iniciación procedente de otro sistema) o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como un iniciador deberá ser capaz de generar el siguiente elemento de intercambio de seguridad initiatorCredentials y deberá poder procesar el siguiente elemento de intercambio de seguridad: responderCredentials. Una implementación que actúe como respondedor deberá poder generar el siguiente elemento de intercambio de seguridad: responderCredentials y deberá poder procesar el siguiente elemento de intercambio de seguridad: initiatorCredentials.
- *Requisitos dinámicos* – Una implementación deberá establecer los procedimientos aplicables descritos en este anexo, en la Rec. UIT-T X.511 | ISO/CEI 9594-3 y en la Rec. UIT-T X.509 | ISO/CEI 9594-8.

C.3 Negociación simple del intercambio de seguridad

Un contexto de aplicación puede comprender la sustentación de más de un intercambio de seguridad para proporcionar los mismos servicios de seguridad a través de protocolos o mecanismos de seguridad diferentes. La sustentación de intercambios de seguridad o mecanismos de seguridad alternados, permite la operación mutua con entidades pares que implementan cualquiera de las alternativas.

Se ha previsto el SE-Negociación para determinar los intercambios de seguridad que han de emplearse en el momento de la utilización. El SE-Negociación, objeto de la clase SECURITY-EXCHANGE, se utiliza para negociar intercambios de seguridad particulares. Este objeto de información está compuesto de uno o más identificadores de intercambio de seguridad. La aplicación iniciadora utiliza el SE-Negociación para proponer uno o más intercambios de seguridad. La aplicación respondedora utiliza el SE-Negociación para indicar cuál de las elecciones propuestas se utilizará en las operaciones subsiguientes. Puede utilizarse el SE-Negociación en cualquier momento para modificar los intercambios de seguridad en uso.

Los contextos de aplicación que requieren negociaciones deben especificar el empleo del SE-Negociación.

El SE-Negociación consta de dos SEI: «offeredIds» y «acceptedIds», como se muestra seguidamente.

```
simpleNegotiationSE  SECURITY-EXCHANGE   ::=
{
  SE-ITEMS     {offeredIds | acceptedIds}
  IDENTIFIER   global : {securityExchanges simple-negotiation-se (3)}
}
offeredIds       SEC-EXCHG-ITEM      ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID      1
}
acceptedIds    SEC-EXCHG-ITEM      ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID      2
}

Negotiation-SEI ::=  SEQUENCE OF OBJECT IDENTIFIER
```

C.3.1 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá cumplir los siguientes requisitos de conformidad:

- *Requisitos de declaración* – Un implementador deberá indicar si la implementación actúa como iniciador (inicia el intercambio de seguridad), respondedor (responde a una iniciación procedente de otro sistema) o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como iniciador deberá poder generar el siguiente elemento de intercambio de seguridad: offerredIds y deberá poder procesar el siguiente elemento de seguridad: acceptedIds. Una implementación que actúe como respondedor deberá poder generar el siguiente elemento de intercambio de seguridad: acceptedIds y deberá poder procesar el siguiente elemento de intercambio de seguridad: offeredIds.
- *Requisitos dinámicos* – Toda implementación deberá establecer los procedimientos aplicables descritos en este anexo.

C.4 Especificación ASN.1 definitiva

```
GulsSecurityExchanges {joint-iso-ccitt genericULS (20)
      modules (1) gulsSecurityExchanges (2)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

-- EXPORTS All --

IMPORTS
  securityExchanges, notation
    FROM ObjectIdentifiers {joint-iso-ccitt genericULS (20)
    modules (1) objectIdentifiers (0)}
  SECURITY-EXCHANGE, SEC-EXCHG-ITEM, SE-ERROR
    FROM Notation notation
  Credentials, SecurityProblem
    FROM DirectoryAbstractService {joint-iso-ccitt ds (5)
    module (1) directoryAbstractService (2) 2};
```

```
-- ****************************************** --
-- Directory Authentication Exchange (One-way) --
-- ****************************************** --

dirAuthenticationOneWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS      {credentials}
  IDENTIFIER   global : {securityExchanges dir-authent-one-way (1)}
}
credentials       SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        1
}


-- ****************************************** --
-- Directory Authentication Exchange (Two-way) --
-- ****************************************** --

dirAuthenticationTwoWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS      {initiatorCredentials | responderCredentials}
  IDENTIFIER   global : {securityExchanges dir-authent-two-way (2)}
}
initiatorCredentials            SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        1
  ERRORS  {authenticationFailure}
}
responderCredentials          SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        2
}
authenticationFailure    SE-ERROR ::=
{
  PARAMETER    DirectoryAbstractService.SecurityProblem
  ERROR-CODE   local : 1
}

-- ***************************** --
-- Simple Negotiation Exchange --
-- ***************************** --

simpleNegotiationSE  SECURITY-EXCHANGE   ::=
{
  SE-ITEMS     {offeredIds | acceptedIds }
  IDENTIFIER   global : {securityExchanges simple-negotiation-se (3)}
}
offeredIds        SEC-EXCHG-ITEM     ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID        1
}
acceptedIds      SEC-EXCHG-ITEM     ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID        2
}

Negotiation-SEI ::=  SEQUENCE OF OBJECT IDENTIFIER

END
```

Anexo D

Especificaciones de la transformación de seguridad

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

Las transformaciones de seguridad pueden definirse en Recomendaciones UIT-T | Normas Internacionales o como normas aparte y registrarse por cualquier organización, de conformidad con el Anexo B. Las definiciones de la transformación de seguridad deben tener la más amplia gama de aplicación posible de forma que pueden reutilizarse en múltiples aplicaciones. En este anexo, se definen algunas transformaciones de seguridad consideradas de utilidad general. No hay ninguna exigencia implícita de que esas aplicaciones o implementaciones tengan que usar las transformaciones de seguridad específicas aquí definidas con preferencia sobre otras transformaciones de seguridad.

D.1 Transformación de seguridad directorio cifrado (directory ENCRYPTED)

La transformación de seguridad Directory Encrypted, cuando se utiliza junto a un proceso de codificación inicial que emplea las reglas de codificación básica en ASN.1, es equivalente funcionalmente al tipo parametrizado ENCRYPTED, definido en la Rec. UIT-T X.509 | ISO/CEI 9594-8 y proporciona el cifrado y el descifrado.

```
dirEncryptedTransformation SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-encrypted (1) }
  -- This transformation transforms a string of octets to a
  -- new bit string using an encipherment process.
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber (1)}
  XFORMED-DATA-TYPE BIT STRING
}
```

D.1.1 Otros detalles

Proceso de codificación:	Proceso de cifrado basado en cualquier algoritmo elegido.
Entradas locales al proceso de codificación:	Algoritmo, parámetros de algoritmos, información de clave de cifrado.
Proceso de decodificación:	Proceso de descrifrado, basado en el mismo algoritmo.
Entradas locales al proceso de decodificación:	Algoritmo, parámetros de algoritmo, información de la clave de descifrado.
Salidas del proceso de decodificación:	Elemento recuperado que debe protegerse en forma de valor de tipo ASN.1.
Parámetros:	Ninguno.
Calificadores de transformación:	Ninguno.
Errores:	No se ha especificado comportamiento de errores.
Servicios de seguridad:	Confidencialidad.

D.1.2 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá cumplir los siguientes requisitos de conformidad:

- *requisitos de declaración* – Un implementador deberá indicar si la implementación actúa como codificador, decodificador o ambas cosas.
- *requisitos estáticos* – Una implementación que actúe como codificador deberá poder generar el elemento transformado. Una implementación que actúe como respondedor deberá poder procesar el elemento transformado.
- *requisitos dinámicos* – Toda implementación deberá establecer los procedimientos aplicables descritos en este anexo.

D.2 Transformación de seguridad directorio firmado (directory SIGNED)

La transformación de seguridad Directory SIGNED es equivalente funcionalmente al tipo parametrizado SIGNED, definido en la Rec. UIT-T X.509 | ISO/CEI 9594-8. Proporciona la signatura digital con un apéndice, con el elemento transformado que incluye los datos no protegidos que han de firmarse y el apéndice de signatura.

```
dirSignedTransformation   SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-signed (2) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            distinguished-encoding (1)}
  XFORMED-DATA-TYPE SEQUENCE
  {
    toBeSigned          ABSTRACT-SYNTAX.&Type (CONSTRAINED BY {
     -- this type is constrained to being the to-be-signed
     -- type -- } ),
    algorithmId         AlgorithmIdentifier,
     -- of the algorithms used to compute the signature --
    encipheredHash      BIT STRING
  }
}
```

D.2.1 Otros detalles

Proceso de codificación: El proceso de codificación opera sobre la codificación ASN.1 DER completa (valor, longitud, etiqueta) de un valor de un tipo ASN.1 único ("elemento no protegido") y produce un "elemento transformado" (valor de tipo SEQUENCE como el definido anteriormente). El elemento no protegido codificado está sometido a una función (por ejemplo, hashing) que genera una cadena de octetos intermedia. La cadena de octetos intermedia se codifica empleando las reglas de codificación básicas en ASN.1 cifrándose el resultado para obtener una cadena de bits "encipheredHash". Seguidamente se construye el elemento transformado.

Entradas locales al proceso de codificación: Identificador de la función hashing y algoritmo de cifrado, parámetros de algoritmo, información de la clave de cifrado.

Proceso de decodificación: El valor del elemento no protegido se extrae del elemento transformado y se le da salida. Si hay que verificar la signatura se aplica asimismo el siguiente proceso. La verificación de signatura requiere la codificación DER del elemento no protegido. Este puede obtenerse a partir del elemento transformado pero puede requerir la decodificación y recodificación con DER. Se somete a los octetos a una función (por ejemplo, hashing) que genera una cadena de octetos intermedia. Utilizando las reglas de codificación básica en ASN.1, se descifra y se decodifica el valor encipheredHash comparándose el resultado con la cadena de octetos intermedia. Si hay coincidencia se ha verificado la signatura correctamente. En cualquier otro caso se indica error.

Entradas locales al proceso de decodificación: Identificador de la función hashing y del algoritmo de cifrado, parámetros del algoritmo, información de la clave de descifrado. Obsérvese que pueden obtenerse el algoritmo y los parámetros del algoritmo a partir del elemento transformado aunque se han almacenado/transferido sin protección. En consecuencia, se recomienda que se obtengan esos valores como entradas locales derivadas tal vez de campos transportados con el elemento no protegido.

Salidas del proceso de decodificación: Elemento no protegido recuperado en forma de valor de tipo ASN.1. Además, pueden producirse facultativamente una o ambas de las siguientes salidas:

a) indicador de si ha sido o no verificada correctamente la signatura;

b) copia del elemento transformado o del valor encipheredHash, para almacenamiento local para una posible verificación de signatura subsiguiente.

Parámetros:	Ninguno.
Calificadores de transformación:	Ninguno.
Errores:	Si la verificación de signatura falla, se produce una condición de error.
Servicios de seguridad:	Autenticación del origen de datos, integridad de datos y (en ciertas situaciones) no repudio.

D.2.2 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá cumplir los siguientes requisitos de seguridad:

- *Requisitos de declaración* – Un implementador deberá indicar si la implementación actúa como codificador, decodificador o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como codificador deberá poder generar el elemento transformado. Una implementación que actúe como respondedor deberá poder procesar el elemento transformado.
- *Requisitos dinámicos* – Toda implementación deberá establecer los procedimientos aplicables descritos en este anexo.

D.3 Transformación de seguridad firma de directorio (directory SIGNATURE)

La transformación de seguridad Directory SIGNATURE es equivalente funcionalmente al tipo parametrizado SIGNATURE definido en la Rec. UIT-T X.509 | ISO/CEI 9594-8. Proporciona la signatura digital con un apéndice, con el elemento transformado que incluye el apéndice de signatura, pero no los datos no protegidos que han de firmarse.

```
dirSignatureTransformation SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-signature (3) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            distinguished-encoding (1)}
  XFORMED-DATA-TYPE SEQUENCE
  {
    algorithmId    AlgorithmIdentifier,
     -- of the algorithms used to compute the signature --
    encipheredHash BIT STRING
  }
}
```

D.3.1 Otros detalles

Proceso de codificación:	El proceso de codificación actúa sobre la codificación ASN.1 DER completa (valor, longitud, etiqueta) de un valor de un tipo ASN.1 único ("elemento no protegido") y proporciona un "elemento transformado" (un valor de tipo SEQUENCE como el definido anteriormente). El elemento no protegido codificado está sujeto a una función (por ejemplo, hashing) que genera una cadena de octetos intermedia. La cadena de octetos intermedia se codifica empleando las reglas de codificación básicas ASN.1 cifrándose el resultado para obtener una cadena de bits "encipheredHash". A continuación se construye el elemento transformado.
Entradas locales al proceso de codificación:	Identificador de la función hashing y algoritmo de cifrado, parámetros de algoritmo, información de la clave de cifrado.
Proceso de decodificación:	Si debe verificarse la signatura se aplicará el proceso siguiente. La verificación de signatura exige la codificación DER del elemento no protegido, que se obtiene como entrada local. Los octetos se someten a una función (por ejemplo, hashing) la cual genera una cadena de octetos intermediaria. El valor cifrado hash se descifra y decodifica empleando las reglas de codificación básica ASN.1 y se compara el resultado con la cadena de octetos intermediaria. Si ambos son iguales se ha verificado la signatura correctamente. En cualquier otro caso, se indica un error.

Entradas locales al proceso de decodificación:	Elemento no protegido, identificador de la función hashing y algoritmo de cifrado, parámetros del algoritmo, información de la clave de descifrado. Obsérvese que el algoritmo y los parámetros del algoritmo pueden obtenerse a partir del elemento transformado, aunque se han almacenado/transferido sin protección. Se recomienda, en consecuencia, que se obtengan esos valores como entradas locales derivadas tal vez de campos transportados con el elemento no protegido.
Salidas del proceso de decodificación:	Pueden producirse facultativamente una o ambas de las siguientes salidas: a) un indicador cuya signatura se haya o no verificado correctamente; b) una copia del elemento transformado o del valor encipheredHash para almacenamiento local y posible verificación de signatura subsiguiente.
Parámetros:	Ninguno.
Calificadores de transformación:	Ninguno.
Errores:	Si la verificación de signatura falla, se produce una condición de error.
Servicios de seguridad:	Autenticación del origen de datos, integridad de datos y (en ciertas situaciones) no repudio.

D.3.2 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá satisfacer los siguientes requisitos de conformidad:

- *Requisitos de declaración* – Un implementador deberá indicar si la implementación actúa como codificador, decodificador o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como codificador deberá poder generar el elemento transformado. Una implementación que actúe como decodificador deberá poder procesar el elemento transformado.
- *Requisitos dinámicos* – Una implementación debe establecer los procedimientos aplicables descritos en este anexo.

D.4 Transformación de seguridad guls firmada (GULS SIGNED)

La transformación de seguridad GULS SIGNED, proporciona la signatura digital o el sello con apéndice con el elemento transformado, incluyendo tanto los datos no protegidos que deben firmarse como el apéndice signatura/sello. Realiza una función similar a la de la transformación de seguridad Directory SIGNED, pero tiene las siguientes características:

- puede sustentar cualquier signatura basada en un apéndice o técnica de sellado, es decir no está restringida a una técnica hash cifrada como Directory SIGNED;
- suprime la restricción de utilización de reglas de codificación distinguida únicamente, pudiendo utilizarse cualquier regla de codificación univaluada (incluidas las reglas de codificación canónica);
- sustenta parámetros protegidos para indicar reglas de codificación inicial, identificadores de algoritmo, parámetros de algoritmo e información de clave;
- proporciona la identificación del algoritmo de signatura digital y de la función hash por medio de identificadores de algoritmos diferentes;
- asegura que la signatura digital se compute en la misma codificación digital del elemento de datos firmado que se transfiere, avisando así de la posible necesidad de que el decodificador decodifique y recodifique los datos cuando verifique la signatura.

En el Anexo E, se definen correspondencias de protección alternativas que permiten poner en correspondencia la notación PROTECTED (como BaseType firmada) con la transformación de seguridad Directory SIGNED o GULS SIGNED.

```
gulsSignedTransformation {KEY-INFORMATION: SupportedKIClasses}
          SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signed (4) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
              canonical-encoding (0) }
   -- This default for initial encoding rules may be overridden
   -- using a static protected parameter (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
    intermediateValue   EMBEDDED PDV (WITH COMPONENTS {
        identification (WITH COMPONENTS
            {transfer-syntax (CONSTRAINED BY {
              -- The transfer syntax to be used is that
              -- indicated by the initEncRules value within
              -- the intermediate value -- }) PRESENT}),
        data-value (WITH COMPONENTS {notation (IntermediateType
          { {SupportedKIClasses} })})
            -- The data value encoded is a value of type IntermediateType
        }),
    appendix      BIT STRING (CONSTRAINED BY {
         -- the appendix value must be generated following
         -- the procedure specified in D.4 of DIS 11586-1 -- })
  }
}
IntermediateType {KEY-INFORMATION: SupportedKIClasses }::= SEQUENCE
{
  unprotectedItem   ABSTRACT-SYNTAX.&Type
                -- this type is constrained to being
                -- the type of the unprotected item, or
                -- BIT STRING if the unprotected item is
                -- not derived from an ASN.1 abstract
                -- syntax --,
  initEncRules      OBJECT IDENTIFIER DEFAULT
              {joint-iso-itu-t asn1 (1) ber-derived (2)
               canonical-encoding (0)},
  signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
                -- Identifies the signing or
                -- sealing algorithm, and can convey
                -- algorithm parameters --
  hashAlgorithm     AlgorithmIdentifier OPTIONAL,
                -- Identifies a hash function,
                -- for use if a hash function is required
                -- and the signOrSealAlgorithm identifier
                -- does not imply a particular hash
                -- function. Can also convey algorithm
                -- parameters.--
  keyInformation    SEQUENCE
              {
                 kiClass              KEY-INFORMATION.&kiClass
                                      ({SupportedKIClasses}),
                 keyInfo              KEY-INFORMATION.&KiType
                                      ({SupportedKIClasses}
                                      {@.kiClass})
              } OPTIONAL
                -- Key information may assume various
                -- formats, governed by supported members
                -- of the KEY-INFORMATION information
                -- object class (defined at start of the
                -- definitive ASN.1 module)
}
```

D.4.1 Otros detalles

Proceso de codificación: El proceso de codificación actúa sobre la codificación de un valor de un tipo ASN.1 único ("elemento no protegido") y proporciona un "elemento transformado" (un valor de tipo SEQUENCE como el definido anteriormente). (Si el elemento no protegido no se ha derivado de una especificación de sintaxis abstracta ASN.1 puede considerarse que es un valor del tipo ASN.1 BIT STRING.) En primer lugar, se genera un "valor intermedio" del tipo IntermediateType ASN.1. Este valor se codifica empleando las reglas de codificación inicial, como se ha especificado en 7.1.4. Los octetos resultantes (codificación completa valor-longitud-etiqueta), se someten a un proceso de firma o de sellado que puede o no emplear la función hash. Este proceso genera un valor apéndice en forma cadena de bits. Seguidamente se construye el elemento transformado.

NOTA – Ejemplos del "proceso de firma o de sello" para diferentes algoritmos:

a) Calcular un código de autenticación de mensaje de acuerdo con ISO 8730 (este es un tipo de sello).

b) Concatenar una codificación de BIT STRING que contenga un valor clave secreto de la codificación del IntermediateType, y luego aplicar una función hash al resultado (este es un tipo de sello).

c) Aplicar una función hash a la codificación de IntermediateType y luego firmar el valor hash resultante utilizando una signatura digital o un algoritmo de cifrado de claves públicas.

Entradas locales al proceso de codificación: Identificador del algoritmo de firma o de sello, identificador del algoritmo hashing (facultativo), parámetros del algoritmo, información de la clave de firma o de sello.

Proceso de decodificación: El valor del elemento no protegido se extrae del elemento transformado, dándosele salida. Cuando deba verificarse la signatura, se aplicará el siguiente proceso. La verificación de la signatura requiere una codificación del valor intermedio empleando las reglas de codificación inicial. Esto puede obtenerse a partir del elemento transformado. Se ejecuta el proceso de verificación de la signatura o sello.

Entradas locales al proceso de decodificación: Información de la clave de verificación signatura/sello. Pueden ser también necesarios la identificación del algoritmo de firma o sello, la identificación del algoritmo hashing y/o parámetros del algoritmo, si no se transportaron como parámetros protegidos.

Salidas del proceso de decodificación: Elemento no protegido recuperado en forma de valor del tipo ASN.1. Además, pueden producirse facultativamente una de las siguientes salidas:

a) indicador de si la signatura ha sido o no verificada correctamente;

b) copia del elemento transformado o del valor apéndice para su almacenamiento local y posible verificación subsiguiente de la signatura.

Parámetros: Los parámetros protegidos estáticos facultativos son: reglas de codificación inicial, identificador del algoritmo de signatura/sello, parámetros del algoritmo signatura/sello, identificador del algoritmo hashing, parámetros del algoritmo hashing, información de clave.

Calificadores de transformación: Ninguno.

Errores: Se produce una situación de error si falla la verificación de la signatura/sello.

Servicios de seguridad: Autenticación del origen de datos, integridad de los datos y (en algunos casos) no repudio.

D.4.2 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad deberá cumplir los siguientes requisitos de conformidad:

- *Requisitos de declaración* – Un implementador deberá declarar si la implementación actúa como codificador, decodificador o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como codificador deberá poder generar el elemento transformado. Una implementación que actúe como respondedor deberá poder procesar el elemento transformado.
- *Requisitos dinámicos* – Una implementación deberá establecer los procedimientos aplicables descritos en este anexo.

D.5 Transformación de seguridad signatura guls (GULS SIGNATURE)

La transformación de seguridad GULS SIGNATURE proporciona la signatura digital o el sellado con apéndice con el elemento transformado incluido el apéndice signatura/sello pero no el dato no protegido que debe firmarse. Realiza una función similar a la de la transformación de seguridad Directory SIGNATURE, pero tiene las siguientes características:

- puede sustentar cualquier signatura basada en apéndice o técnica de sellado, es decir no está limitada a una técnica hash-cifrada como Directory SIGNATURE;
- suprime la limitación de utilización de reglas de codificación distinguida únicamente, pudiendo utilizarse cualquier regla de codificación univaluada (incluidas las reglas de codificación canónica);
- sustenta parámetros protegidos para indicar reglas de codificación inicial, identificadores de algoritmo, parámetros de algoritmo e información de clave;
- proporciona la identificación del algoritmo de signatura digital y de la función hash por medio de identificadores de algoritmos diferentes;
- se han simplificado los procesos de codificación y de decodificación.

En el Anexo E, se definen correspondencias de protección alternativas que permiten poner en correspondencia la notación PROTECTED {BaseType, signed}, con la transformación de seguridad Directory SIGNATURE o GULS SIGNATURE.

```
gulsSignatureTransformation {KEY-INFORMATION: SupportedKIClasses }
        SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signature (5) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            canonical-encoding (0)}
   -- This default for initial encoding rules may be overridden
   -- using a static protected parameter (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
  initEncRules     OBJECT IDENTIFIER DEFAULT
            {joint-iso-itu-t asn1 (1) ber-derived (2)
             canonical-encoding (0)},
  signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
              -- Identifies the signing or
              -- sealing algorithm, and can convey
              -- algorithm parameters --
  hashAlgorithm     AlgorithmIdentifier OPTIONAL,
              -- Identifies a hash function,
              -- for use if a hash function is required
              -- and the signOrSealAlgorithm identifier
              -- does not imply a particular hash
              -- function. Can also convey algorithm
              -- parameters.--
```

```
    keyInformation    SEQUENCE
                {
                   kiClass              KEY-INFORMATION.&kiClass
                                        ({SupportedKIClasses}),
                   keyInfo              KEY-INFORMATION.&KiType
                                        ({SupportedKIClasses}
                                        {@.kiClass})
                } OPTIONAL,
                  -- Key information may assume various
                  -- formats, governed by supported members
                  -- of the KEY-INFORMATION information
                  -- object class (defined at start of the
                  -- definitive ASN.1 module)
    appendix        BIT STRING (CONSTRAINED BY {
          -- the appendix value must be generated following
          -- the procedure specified in D.5 of DIS 11586-1 -- })
    }
}
```

D.5.1 Otros detalles

Proceso de codificación:

El proceso de codificación actúa sobre un valor de un tipo ASN.1 único ("elemento no protegido") y proporciona un "elemento transformado" (valor de tipo SEQUENCE como el definido anteriormente). (Si el elemento no protegido no se ha derivado de una especificación de sintaxis abstracta ASN.1, puede considerarse que es un valor del tipo ASN.1 BIT STRING.) En primer lugar, se genera un "valor intermedio" del tipo IntermediateType ASN.1 definido en D.4. Este valor se codifica empleando las reglas de codificación inicial, como se ha especificado en 7.1.4. Los octetos resultantes (codificación completa valor-longitud-etiqueta), se someten a un proceso de firma o de sellado que puede o no emplear la función hash. Este proceso genera un valor apéndice en forma cadena de bits. Seguidamente se construye el elemento transformado.

NOTA – Ejemplos del "proceso de firma o de sellado" para diferentes algoritmos son:

a) Calcular un código de autenticación de mensaje de acuerdo con ISO 8730 (este es un tipo de sello).

b) Concatenar una codificación de BIT STRING que contenga un valor clave secreto de la codificación del IntermediateType, y luego aplicar una función hash al resultado (este es un tipo de sello).

c) Aplicar una función hash a la codificación de IntermediateType y luego firmar el valor hash resultante utilizando una signatura digital o un algoritmo de cifrado de claves públicas.

Entradas locales al proceso de codificación:

Identificador de firma o algoritmo de sellado, identificador (facultativo) del algoritmo hashing, parámetros de algoritmo, información de la clave de la firma/sellado.

Proceso de decodificación:

Cuando deba verificarse la signatura se aplicará el siguiente proceso. La verificación de la signatura requiere la codificación del valor intermedio empleando las reglas de codificación inicial. Esto exige el valor del elemento no protegido que se obtiene como entrada local. Los valores del parámetro protegido pueden obtenerse del elemento transformado, si bien pueden necesitar la decodificación y recodificación con las reglas de codificación requeridas. Se ejecuta el proceso de verificación de la signatura o sello.

Entradas locales al proceso de decodificación:	Elemento no protegido, información de la clave de verificación signatura/sello. Pueden ser también necesarios la identificación del algoritmo de firma o sello, la identificación del algoritmo hashing y/o parámetros del algoritmo, si no se transportaron como parámetros protegidos.
Salidas del proceso de decodificación:	Pueden producirse facultativamente una de las siguientes salidas o ambas: a) indicador de si la signatura ha sido o no verificada correctamente; b) copia del elemento transformado o del valor apéndice para su almacenamiento local y posible verificación subsiguiente de la signatura.
Parámetros:	Son parámetros protegidos estáticos facultativos los siguientes: Reglas de codificación inicial, identificador del algoritmo signatura/sello, parámetros del algoritmo signatura/sello, identificador del algoritmo hashing, parámetros del algoritmo hashing, información de clave.
Calificadores de transformación:	Ninguno.
Errores:	Se produce una situación de error si falla la verificación de la signatura/sello.
Servicios de seguridad:	Autenticación del origen de datos, integridad de los datos y (en algunos casos) no repudio.

D.5.2 Conformidad

Una implementación que pretenda ser conforme con esta definición de intercambio de seguridad, deberá cumplir los siguientes requisitos de conformidad:

- *Requisitos de declaración* – Un implementador deberá declarar si la implementación actúa como codificador, decodificador o ambas cosas.
- *Requisitos estáticos* – Una implementación que actúe como codificador deberá poder generar el elemento transformado. Una implementación que actúe como respondedor deberá poder procesar el elemento transformado.
- *Requisitos dinámicos* – Una implementación deberá establecer los procedimientos aplicables descritos en este anexo.

D.6 Especificación ASN.1 definitiva

```
GulsSecurityTransformations {joint-iso-itu-t genericULS (20)
      modules (1) gulsSecurityTransformations (3) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

-- EXPORTS All --

IMPORTS
  securityTransformations, notation
    FROM ObjectIdentifiers {joint-iso-itu-t genericULS (20)
    modules (1) objectIdentifiers (0) }
  SECURITY-TRANSFORMATION, SecurityIdentity
    FROM Notation notation
  AlgorithmIdentifier
    FROM AuthenticationFramework {joint-iso-itu-t ds (5)
    module (1) authenticationFramework(7) 2 };

-- ************************************** --
-- Notation for specifying key information --
-- ************************************** --

KEY-INFORMATION ::= CLASS
-- This information object class definition is for use when
-- specifying key information relating to particular classes
```

```
  -- of protection mechanisms (e.g. symmetric, asymmetric).
  -- It may be useful in defining various security transformations.
  {
    &kiClass
      CHOICE
      {   local       INTEGER,
          -- local objects can only be defined within this
          -- ASN.1 module.
          global   OBJECT IDENTIFIER
          -- global objects are defined elsewhere
      }UNIQUE,
    &KiType
  }
  WITH SYNTAX
  {
    KEY-INFO-CLASS     &kiClass
    KEY-INFO-TYPE      &KiType
  }

  symmetricKeyInformation KEY-INFORMATION ::= {
    KEY-INFO-CLASS       local: 0
    KEY-INFO-TYPE SEQUENCE
    {
      entityId              SecurityIdentity,
      keyIdentifier         INTEGER
    }
  }

  asymmetricKeyInformation KEY-INFORMATION ::= {
    KEY-INFO-CLASS       local: 1
    KEY-INFO-TYPE SEQUENCE
    {
      issuerCAName      SecurityIdentity OPTIONAL,
      certSerialNumber     INTEGER OPTIONAL,
      signerName        SecurityIdentity OPTIONAL,
      keyIdentifier     BIT STRING OPTIONAL
    }
  }

  -- ****************************************** --
  -- Directory ENCRYPTED Security Transformation --
  -- ****************************************** --

  dirEncryptedTransformation SECURITY-TRANSFORMATION ::=
  {
    IDENTIFIER {securityTransformations dir-encrypted (1) }
    -- This transformation transforms a string of octets to a
    -- new bit string using an encipherment process.
    INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber (1) }
    XFORMED-DATA-TYPE BIT STRING
  }

  -- ****************************************** --
  -- Directory SIGNED Security Transformation --
  -- ****************************************** --

  dirSignedTransformation   SECURITY-TRANSFORMATION ::=
  {
    IDENTIFIER {securityTransformations dir-signed (2) }
    INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
                distinguished-encoding (1)}
    XFORMED-DATA-TYPE SEQUENCE
    {
      toBeSigned          ABSTRACT-SYNTAX.&Type (CONSTRAINED BY {
       -- this type is constrained to being the to-be-signed type -- } ),
```

```
    algorithmId          AlgorithmIdentifier,
     -- of the algorithms used to compute the signature --
    encipheredHash BIT STRING
  }
}
-- ****************************************** --
-- Directory SIGNATURE Security Transformation --
-- ****************************************** --

dirSignatureTransformation SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-signature (3) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            distinguished-encoding (1)}
  XFORMED-DATA-TYPE SEQUENCE
  {
    algorithmId       AlgorithmIdentifier,
     -- of the algorithms used to compute the signature --
    encipheredHash BIT STRING
  }
}

-- ********************************** --
-- GULS SIGNED Security Transformation --
-- ********************************** --

gulsSignedTransformation {KEY-INFORMATION: SupportedKIClasses }
          SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signed (4) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
              canonical-encoding (0)}
   -- This default for initial encoding rules may be overridden
   -- using a static protected parameter (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
    intermediateValue   EMBEDDED PDV (WITH COMPONENTS {
        identification (WITH COMPONENTS
            {transfer-syntax (CONSTRAINED BY {
              -- The transfer syntax to be used is that
              -- indicated by the initEncRules value within
              -- the intermediate value -- }) PRESENT}),
        data-value (WITH COMPONENTS {notation (IntermediateType
          { {SupportedKIClasses} })})
              -- The data value encoded is a value of type
              -- IntermediateType
        }),
    appendix      BIT STRING (CONSTRAINED BY {
         -- the appendix value must be generated following
         -- the procedure specified in D.4 of DIS 11586-1 -- })
  }
}
IntermediateType {KEY-INFORMATION: SupportedKIClasses }::= SEQUENCE
{
  unprotectedItem   ABSTRACT-SYNTAX.&Type
                -- this type is constrained to being
                -- the type of the unprotected item, or
                -- BIT STRING if the unprotected item is
                -- not derived from an ASN.1 abstract
                -- syntax --,
  initEncRules      OBJECT IDENTIFIER DEFAULT
              {joint-iso-itu-t asn1 (1) ber-derived (2)
               canonical-encoding (0)},
```

```
    signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
                -- Identifies the signing or
                -- sealing algorithm, and can convey
                -- algorithm parameters --
    hashAlgorithm     AlgorithmIdentifier OPTIONAL,
                -- Identifies a hash function,
                -- for use if a hash function is required
                -- and the signOrSealAlgorithm identifier
                -- does not imply a particular hash
                -- function. Can also convey algorithm
                -- parameters.--
    keyInformation    SEQUENCE
                {
                   kiClass    KEY-INFORMATION.&kiClass
                              ({SupportedKIClasses}),
                   keyInfo    KEY-INFORMATION.&KiType
                              ({SupportedKIClasses}
                              {@.kiClass})
                } OPTIONAL
                  -- Key information may assume various
                  -- formats, governed by supported members
                  -- of the KEY-INFORMATION information
                  -- object class (defined at start of the
                  -- definitive ASN.1 module)
}


-- ************************************** --
-- GULS SIGNATURE Security Transformation --
-- ************************************** --

gulsSignatureTransformation {KEY-INFORMATION: SupportedKIClasses }
          SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signature (5) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
                canonical-encoding (0)}
   -- This default for initial encoding rules may be overridden
   -- using a static protected parameter (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
  initEncRules      OBJECT IDENTIFIER DEFAULT
                {joint-iso-itu-t asn1 (1) ber-derived (2)
                 canonical-encoding (0)},
  signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
                -- Identifies the signing or
                -- sealing algorithm, and can convey
                -- algorithm parameters --
  hashAlgorithm     AlgorithmIdentifier OPTIONAL,
                -- Identifies a hash function,
                -- for use if a hash function is required
                -- and the signOrSealAlgorithm identifier
                -- does not imply a particular hash
                -- function. Can also convey algorithm parameters.--
  keyInformation    SEQUENCE
                {
                   kiClass    KEY-INFORMATION.&kiClass
                              ({SupportedKIClasses}),
                   keyInfo    KEY-INFORMATION.&KiType
                              ({SupportedKIClasses}
                              {@.kiClass})
                } OPTIONAL,
```

```
                -- Key information may assume various
                -- formats, governed by supported members
                -- of the KEY-INFORMATION information
                -- object class (defined at start of the
                -- definitive ASN.1 module)
  appendix      BIT STRING (CONSTRAINED BY {
        -- the appendix value must be generated following
        -- the procedure specified in D.5 of DIS 11586-1 -- })
  }
}

END
```

Anexo E

Especificaciones de la correspondencia de protección

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

Las correspondencias de protección se definen en módulos ASN.1. Puede definirse cualquiera de estos módulos en Recomendaciones UIT-T | Normas Internacionales o mediante otras normas y registrarse por cualquier organización habilitada para la asignación de identificadores de objeto. Las definiciones de la correspondencia de protección deberán redactarse de forma que sean lo más ampliamente aplicables posible de manera que puedan reutilizarse en múltiples aplicaciones. En este anexo, se definen algunas correspondencias de protección que se consideran de utilidad general. No hay ninguna exigencia implícita de que estas aplicaciones o implementaciones tengan que utilizar las correspondencias de protección específicas aquí definidas con preferencia sobre otras correspondencias de protección.

```
DirectoryProtectionMappings {joint-iso-itu-t genericULS (20)
        modules (1) dirProtectionMappings (4) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
-- These protection mappings generate bit-compatible encodings
-- to the parameterized types in the Directory Authentication
-- Framework

-- EXPORTS All --

IMPORTS
  notation, gulsSecurityTransformations
    FROM ObjectIdentifiers {joint-iso-itu-t genericULS (20)
    modules (1) objectIdentifiers (0) }
  PROTECTION-MAPPING
    FROM Notation notation
  dirEncryptedTransformation, dirSignedTransformation,
  dirSignatureTransformation
    FROM GulsSecurityTransformations
          gulsSecurityTransformations;

-- ************************************* --
-- Directory encrypted Protection Mapping --
-- ************************************* --

-- This protection mapping enables the notation
-- PROTECTED {BaseType, encrypted}
-- to replace the notation
-- ENCRYPTED {BaseType}
-- as provided by ITU-T Rec. X.509 | ISO/IEC 9594-8:1994, and to
-- generate an identical bit-encoding.
-- Security Service:  confidentiality

encrypted PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {dirEncryptedTransformation }
}


-- ********************************** --
-- Directory signed Protection Mapping --
-- ********************************** --

-- This protection mapping enables the notation
-- PROTECTED {BaseType, signed}
-- to replace the notation
-- SIGNED {BaseType}
```

```
-- as provided by ITU-T Rec. X.509 | ISO/IEC 9594-8:1994, and to
-- generate an identical bit-encoding.
-- Security Service:  data origin authentication, data integrity and
-- (in certain situations) non-repudiation.
signed PROTECTION-MAPPING  ::=
{
  SECURITY-TRANSFORMATION {dirSignedTransformation }
}

-- ************************************ --
-- Directory signature Protection Mapping --
-- ************************************ --

-- This protection mapping enables the notation
-- PROTECTED {BaseType, signature}
-- to provide a functionally-equivalent replacement of the notation
-- SIGNATURE BaseType
-- as provided by ITU-T Rec. X.509 | ISO/IEC 9594-8.
-- Security Service:  data origin authentication, data integrity and
-- (in certain situations) non-repudiation.

signature PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION  {dirSignatureTransformation }
}
END


GULSProtectionMappings {joint-iso-itu-t genericULS (20)
        modules (1) gulsProtectionMappings (5) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
-- These protection mappings are more versatile that the
-- preceding protection mappings which were specifically designed
-- to generate identical bit-encodings as the Directory
-- Authentication Framework parameterized types.

-- EXPORTS All --

IMPORTS
  notation, gulsSecurityTransformations
    FROM ObjectIdentifiers {joint-iso-itu-t genericULS (20)
    modules (1) objectIdentifiers (0) }
  PROTECTION-MAPPING
    FROM Notation notation
  dirEncryptedTransformation, gulsSignedTransformation,
  gulsSignatureTransformation, symmetricKeyInformation,
  asymmetricKeyInformation
    FROM GulsSecurityTransformations
            gulsSecurityTransformations;


-- ********************************** --
-- confidentiality Protection Mapping --
-- ********************************** --

-- This protection mapping enables the notation
-- PROTECTED {BaseType, confidentiality}
-- to map to either dirEncryptedTransformation or to no transformation
-- at the choice of the encoding system, dependent upon local security
-- policy and other local environment considerations.
-- Security Service:  confidentiality
```

```
confidentiality PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {dirEncryptedTransformation }
  BYPASS-PERMITTED TRUE
}

-- ****************************** --
-- GULS signed Protection Mapping --
-- ****************************** --

-- This protection mapping causes the notation
-- PROTECTED {BaseType, signed}
-- to map to the gulsSignedTransformation.
-- Security Service:  data origin authentication, data integrity and
-- (in certain situations) non-repudiation.

signed PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {gulsSignedTransformation
    {{symmetricKeyInformation | asymmetricKeyInformation }}}
}

-- ********************************* --
-- GULS signature Protection Mapping --
-- ********************************* --

-- This protection mapping causes the notation
-- PROTECTED {BaseType, signature}
-- to map to the gulsSignatureTransformation.
-- Security Service:  data origin authentication, data integrity and
-- (in certain situations) non-repudiation.

signature PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {gulsSignatureTransformation
    {{symmetricKeyInformation | asymmetricKeyInformation }}}
}

END
```

Anexo F

Utilización del identificador de objetos

(Este anexo es parte integrante de esta Recomendación | Norma Internacional)

En este anexo se documentan los límites superiores del subárbol identificador de objetos donde residen todos los identificadores de objetos asignados en esta serie de especificaciones. Ello se realiza proporcionando un módulo ASN.1 denominado ObjectIdentifiers en el cual todos los nodos del subárbol carentes de hojas son nombres asignados. Se identifica también el conjunto completo de módulos ASN.1 definidos en esta serie de Recomendaciones.

```
ObjectIdentifiers {joint-iso-itu-t genericULS (20)
        modules (1) objectIdentifiers (0) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

-- EXPORTS All --

genericULS                  OBJECT IDENTIFIER ::=
                            {joint-iso-itu-t genericULS (20) }

-- Categories of information object --

modules                     OBJECT IDENTIFIER ::= {genericULS 1}
generalTransferSyntax       OBJECT IDENTIFIER ::= {genericULS 2}
specificTransferSyntax      OBJECT IDENTIFIER ::= {genericULS 3}
securityExchanges           OBJECT IDENTIFIER ::= {genericULS 4}
securityTransformations     OBJECT IDENTIFIER ::= {genericULS 5}

-- ASN.1 modules --

objectIdentifiers           OBJECT IDENTIFIER ::= {modules 0}
notation                    OBJECT IDENTIFIER ::= {modules 1}
gulsSecurityExchanges       OBJECT IDENTIFIER ::= {modules 2}
gulsSecurityTransformations
                            OBJECT IDENTIFIER ::= {modules 3}
dirProtectionMappings
                            OBJECT IDENTIFIER ::= {modules 4}
gulsProtectionMappings
                            OBJECT IDENTIFIER ::= {modules 5}
seseAPDUs                   OBJECT IDENTIFIER ::= {modules 6}
genericProtectingTransferSyntax
                            OBJECT IDENTIFIER ::= {modules 7}
END
```

Anexo G

Directrices para la utilización de facilidades de seguridad genérica de las capas superiores

(Este anexo no es parte integrante de esta Recomendación | Norma Internacional)

G.1 Introducción

En este anexo se explica como pueden utilizarse las normas de la GULS para proporcionar seguridad en una aplicación específica supuesto que esas herramientas GULS sean idóneas para proporcionar seguridad para esa aplicación.

Es conveniente que el diseñador de cualquier protocolo de aplicación de OSI utilice las mismas soluciones de seguridad que las empleadas en otros protocolos de aplicación de OSI. En general esto no siempre puede lograrse totalmente, debido a que aplicaciones diferentes comportan distintos requisitos de seguridad, por lo que las soluciones de seguridad requerirán algún tipo de adaptación para ajustarse a las necesidades de aplicaciones diferentes. Sin embargo, generalmente es posible que aplicaciones diferentes adopten soluciones comunes basadas en la identificación de requisitos de seguridad comunes.

Esta serie de Recomendaciones | Normas Internacionales tiene por finalidad proporcionar un conjunto de facilidades de protocolos de seguridad que pueden contribuir a la incorporación de soluciones de seguridad en cualquier protocolo de aplicación y que alientan la adopción de soluciones de seguridad comunes en aplicaciones diferentes. Sin embargo, estas especificaciones no proporcionan por sí mismas la totalidad de la especificación para soluciones de seguridad comunes.

G.2 Facilidades genéricas proporcionadas

Las facilidades proporcionadas en las normas de GULS comprenden:

- un medio general para construir componentes de protocolo de capa de aplicación que sustenten el intercambio de información relacionada con la seguridad entre una pareja de invocaciones de entidad de aplicación que se comunican entre sí (se trata del *concepto intercambio de seguridad* sustentado por el *SESE*);
- un enfoque general para la utilización de las facilidades de capa de presentación a fin de que realicen transformaciones relacionadas con la seguridad sobre elementos de información, con objeto de proteger esos elementos (es la *sintaxis genérica de transferencia de protección*);
- herramientas de notación de sintaxis abstracta que ayuden a un diseñador de protocolos de aplicación en la especificación de la protección de seguridad que debe aplicarse a campos seleccionados de su protocolo (tipo parametrizado *PROTECTED* y la variante *PROTECTED-Q* de este tipo).

Otra facilidad de seguridad genérica de esta naturaleza es la unidad funcional de autenticación del ACSE. Si bien esta facilidad se ha definido en la Rec. UIT-T X.217 | ISO/CEI 8649 y en la Rec. X.227 del CCITT | ISO/CEI 8650-1, en vez de en esta Recomendación | Norma Internacional, en el presente anexo se tratará del empleo de esa facilidad para el desarrollo de soluciones de seguridad para las aplicaciones.

Se ha previsto que las facilidades descritas sean empleadas por los diseñadores de nuevas aplicaciones cuando se ocupen de sus necesidades de seguridad. Sin embargo, pueden también emplearse estas facilidades para añadir características de seguridad a los protocolos de aplicación de OSI existentes. En cierta medida, esto puede lograrse construyendo un nuevo contexto de aplicación que incorpore esas facilidades sin tener que modificar necesariamente las especificaciones de ASE existentes. Sin embargo, para proporcionar algunos servicios de seguridad (por ejemplo, confidencialidad o integridad selectiva de los campos) será necesario efectuar modificaciones en otras especificaciones de ASE.

G.3 Aspectos de soluciones de seguridad no recogidos en esta Recomendación | Norma Internacional

El alcance de esta Recomendación | Norma Internacional está limitado a la comunicación de información asociada con la provisión de servicios de seguridad, es decir no se extiende a la totalidad de los detalles relativos a la prestación de cualquier servicio de seguridad o a la implementación de cualquier mecanismo de seguridad. En las Recomendaciones | Normas Internacionales relativas al marco de seguridad (Recs. UIT-T X.811, X.812, X.813, X.814, X.815 | ISO/CEI 10181) se describen aspectos generales de los mecanismos de seguridad. El JTC1/SC27 de ISO/CEI desarrolla Recomendaciones | Normas Internacionales para ciertos mecanismos de seguridad específicos y técnicas de seguridad sustentantes de los mismos.

En particular, la implementación de las facilidades genéricas descritas en esta Recomendación | Norma Internacional depende de una o ambas especificaciones siguientes:

- especificaciones de *intercambios de seguridad* particulares, diseñados para la sustentación de mecanismos de seguridad específicos (por ejemplo, un intercambio de autenticación específico);
- especificaciones de *transformaciones de seguridad* particulares, que transforman datos de usuarios con fines de protección de alguna forma particular (por ejemplo, un proceso de cifrado).

En esta Recomendación | Norma Internacional no se proporcionan esas especificaciones (salvo para algunos ejemplos generalmente útiles descritos en los Anexos C y D). Sin embargo, esta Recomendación | Norma Internacional incluye herramientas y directrices que facilitan la elaboración de tales especificaciones. Debe observarse que cuando se elaboran tales especificaciones deberían utilizarse para sustentar numerosas aplicaciones diferentes cuando se empleen conjuntamente con las facilidades descritas en esta Recomendación | Norma Internacional.

Además esta Recomendación | Norma Internacional no especifica procedimientos para el establecimiento de asociaciones de seguridad establecidas externamente.

Esta Recomendación | Norma Internacional no incluye la definición de interfaz de servicio con los intercambios de seguridad que es independiente de los mecanismos utilizados.

G.4 Utilización de las facilidades de GULS para proporcionar servicios de seguridad

A continuación se indica como pueden emplearse las facilidades genéricas descritas en estas Recomendaciones | Normas Internacionales para la sustentación de la provisión de los servicios de seguridad indicados en la Rec. X.800 del CCITT | ISO 7498-2 para la capa de aplicación. Estos servicios contarán vulnerabilidades identificadas por grupos de aplicación específicos.

En otras Recomendaciones UIT-T | Normas Internacionales, pueden definirse separadamente detalles sobre la provisión de los servicios de seguridad indicados más adelante que utilizan herramientas de GULS.

G.4.1 Autenticación de entidad

La autenticación de entidad (como se describe en la Rec. UIT-T X.811 | ISO/CEI 10181-2) implica, en general, un *intercambio de autenticación* que es un intercambio n-lateral de información de autenticación entre dos partes (típicamente n toma los valores 1, 2 ó 3 pero puede ser mayor). En consecuencia, un intercambio de autenticación puede considerarse como un caso especial de intercambio de seguridad.

Hay dos formas posibles de sustentar un intercambio de autenticación utilizando las facilidades de seguridad genérica de capas superiores:

- en el caso particular en que el intercambio de autenticación esté limitado a ser unilateral o bilateral y cuando solamente pueda producirse una vez en conjunción con el establecimiento de la asociación, puede transportarse el intercambio de autenticación empleando la unidad funcional de autenticación ACSE;
- en todos los casos (no se aplican las limitaciones anteriores), el intercambio de autenticación puede transportarse mediante el SESE.

Obsérvese que el tipo de identidad que se está autenticando es inmaterial y no está limitado a ser entidad OSI. Además la autenticación de identidad no tiene porqué producirse al comienzo de una asociación (por ejemplo, podría ocurrir al comienzo de un diálogo TP o en cualquier momento dentro de una asociación).

La autenticación de entidad puede también implicar la comunicación con un tercero. Para esta finalidad el protocolo podría ser un protocolo de aplicación, en cuyo caso podrían también emplearse en este protocolo los intercambios de seguridad.

G.4.2 Autenticación del origen de datos

Un método habitual de autenticación del origen de datos es añadir una signatura o sello al elemento cuya fuente se está autenticando. Esto puede realizarse transportando el elemento en una asociación de seguridad que utiliza una transformación de seguridad de tipo firma o sello.

A fin de proteger de esta forma una clase de PDU completa, la especificación del contexto de aplicación deberá contener reglas que indiquen que esa PDU debe transportarse en un contexto de presentación de protección. Para proteger un elemento de información individual dentro de una sintaxis abstracta, puede utilizarse el tipo parametrizado PROTECTED.

G.4.3 Control de acceso

Hay muchos aspectos del control de acceso que son específicos de la aplicación por lo que no pueden contemplarse de una forma genérica. Sin embargo, utilizando un intercambio de seguridad puede lograrse la comunicación de la información de control de acceso (relativa a la concesión, reforzamiento y revocación de los derechos del control de acceso). Por ejemplo, puede contemplarse la transferencia de un certificado de control de acceso como un intercambio de seguridad simple (unidireccional). En ese caso, puede agregarse tal certificado a cualquier otra PDU transportándola mediante servicios de intercambio de seguridad del SESE. En I.4 se facilita un ejemplo de utilización del SESE para esta finalidad.

Generalmente son también muy importantes la integridad y/o autenticación del origen de los datos de la información de control de acceso intercambiada. Puede proveerse la protección necesaria transportando la información de control de acceso en una asociación de seguridad que emplea una transformación de seguridad de tipo firma o sello.

G.4.4 Confidencialidad con conexión y sin conexión

Puede lograrse la confidencialidad de una PDU completa transportándola en una asociación de seguridad que emplea una transformación de seguridad de tipo cifrado. Para proteger una categoría de PDU completa de esta manera, la especificación del contexto de aplicación deberá contener reglas que indiquen que es necesario transportar esas PDU en un contexto de presentación de protección.

G.4.5 Confidencialidad selectiva de los campos

Puede conseguirse la confidencialidad de cualquier campo de protocolo transportándolo en una asociación de seguridad que utilice una transformación de seguridad de tipo cifrado. Para identificar los elementos de información individuales dentro de una sintaxis abstracta que requieren tal protección, puede emplearse el tipo parametrizado PROTECTED.

G.4.6 Confidencialidad del flujo de tráfico

Un proceso de codificación de transformación podría proporcionar la adición de datos de relleno al elemento protegido pero no permitiría la generación de PDU que contuvieran datos de relleno únicamente.

G.4.7 Integridad con conexión y sin conexión

Puede conseguirse la integridad de una PDU completa transportándola en una asociación de seguridad que utilice una transformación de seguridad de tipo firma o sello. Para proteger de esta forma una categoría de PDU completa, la especificación del contexto de aplicación debería contener reglas indicando que esas PDU deben transportarse en un contexto de presentación de aplicación.

G.4.8 Integridad selectiva de los campos

Puede conseguirse la integridad de cualquier campo de protocolo transportándolo en una asociación de seguridad que utilice una transformación de seguridad de tipo firma o sello. Para identificar los elementos de información individuales, en una sintaxis abstracta, que requieren tal protección, podría utilizarse el tipo parametrizado PROTECTED.

G.4.9 No repudio

La provisión de un servicio de no repudio (con prueba de origen o prueba de entrega) requiere típicamente la integridad y/o autenticación del origen de los datos que han de aplicarse a los datos comunicados. Puede lograrse la provisión de la protección necesaria transportando los datos en una asociación de seguridad que utilice una transformación de seguridad de tipo firma o sello.

Algunos mecanismos de no repudio se basan en el empleo de signaturas no repudiables aplicadas a los datos comunicados. Esto puede lograrse transportando los datos en una asociación de seguridad que utilice una transformación de seguridad de tipo firma mediante una técnica de cifrado asimétrica.

G.4.10 Auditoría

La provisión de un servicio de auditoría suele exigir otros servicios de seguridad, aparte de los descritos en G.4.1 a G.4.9. El SESE puede utilizarse para intercambiar información tal como mensajes de alarma de seguridad y de auditoría entre entidades. (Sin embargo, existen también otras Recomendaciones | Normas Internacionales que tratan dicho intercambio de información, por ejemplo, Rec. X.736 del CCITT | ISO/CEI 10164-7 y Rec. X.740 del CCITT | ISO/CEI 10164-8.)

G.5 Gestión de la clave

La gestión de la clave es una actividad compleja, muchos de cuyos aspectos quedan fuera del ámbito de OSI. Sin embargo, la utilización de numerosos tipos de funciones de transformación de protección en la capa de presentación dependerá de las claves que se hayan establecido.

Hay varias formas de establecimiento de las claves, tales como:

a) distribución manual o de otra forma completamente fuera del ámbito de OSI;

b) establecimiento de las claves en una asociación separada (más temprana o superpuesta), por ejemplo, utilizando servicios de gestión de sistemas OSI;

c) establecimiento de claves dentro de la misma asociación, pero antes de que la transformación requiera la clave. Esto podría implicar por ejemplo un intercambio de obtención de clave de tipo Diffie-Hellman o el envío de una clave protegida con fines de confidencialidad en alguna otra transformación y/o alguna otra clave.

   En el caso c), la derivación de la clave o intercambio de distribución podría realizarse como un intercambio de seguridad y podría utilizar los servicios de intercambio de seguridad del SESE. Esto se realizaría como parte del protocolo que sustenta el establecimiento de una asociación de seguridad establecida externamente.

Puede proporcionarse una derivación o distribución de la clave como parte integrante de un intercambio de seguridad que sustenta otro servicio, por ejemplo, autenticación de entidad.

Pueden también pertenecer a la gestión de la clave, los parámetros dinámicos de transformación de seguridad transportados en la sintaxis de transferencia de seguridad, por ejemplo, indicando la clave particular que debe utilizarse desde un cierto punto en adelante.

G.6 Directrices para la especificación de contextos de aplicación

En general, la utilización del SESE requerirá la redacción de reglas especiales, que no son parte de una especificación de ASE, en una especificación de contexto de aplicación. Tales reglas deberán especificar lo siguiente:

a) *ASE* – Inclusión del SESE como uno de los ASE del contexto de aplicación.

b) *Intercambios de seguridad* – Conjunto particular de intercambios de seguridad que deben sustentarse, lo que implica una sintaxis abstracta SESE específica.

c) *Correspondencias SESE PDU* – Correspondencias de los SESE PDU con otros servicios, por ejemplo, el servicio P-DATA o como un valor de datos de presentación insertado en la PDU de otro ASE.

d) *Requisitos de concatenación de los PDU* – Requisitos para la concatenación de SESE PDU particulares con los valores de datos de presentación de otros ASE.

e) *Limitaciones insertadas de los PDU* – Requisitos para la inserción de otros valores de datos de presentación en las SESE PDU.

f) *Limitaciones de procedimiento* – Reglas relativas a las interacciones de la máquina de estados del SESE con las máquinas de estados de otros ASE, por ejemplo, para asegurar que el estado de otras máquinas de protocolo del ASE, cuando cada intercambio de seguridad termina con éxito o aborta, está bien definido y no queda estancado.

g) *Limitaciones del contexto de presentación* – Requisitos para el establecimiento de sintaxis de transferencia particulares para sintaxis abstractas particulares.

G.7 Ejemplo

Supóngase que se desea elaborar un nuevo contexto de aplicación para transferencia, acceso y gestión de ficheros (*FTAM, file transfer, access and management*) de OSI definida en ISO 8571 que añada tres características de seguridad al protocolo FTAM básico:

a) utilización de un intercambio de autenticación manual fuerte en conjunción con el establecimiento de la asociación;

b) un certificado de control de acceso cuyo formato se defina en alguna otra norma y que esté limitado por cada petición F-SELECT o F-CREATE;

c) aplicación de confidencialidad y protección de integridad a todos los datos transmitidos de contenido de los ficheros.

Se desea alcanzar este objetivo sin modificar la sintaxis abstracta FTAM ASE.

El primer paso es identificar o especificar, si es necesario, los intercambios de seguridad requeridos. Para la característica a) se requiere un intercambio de seguridad bidireccional, mientras que para la característica b) es necesario un intercambio de seguridad unilateral. Si sólo se requiriera la autenticación podría emplearse para a) el intercambio de seguridad dirAuthenticationTwoWay definido en el Anexo C. Alternativamente, podría utilizarse un intercambio de seguridad que combine la autenticación y el establecimiento de la clave, en cuyo caso podrían utilizarse la clave o claves derivadas del intercambio para proporcionar la característica c). Para los fines de este ejemplo, se supondrá que se emplea el intercambio de seguridad dirAuthenticationTwoWay. El intercambio de seguridad para la característica b) podría ser el intercambio de seguridad boundAccessControlCert definido en el ejemplo I.4.

El paso siguiente es especificar una sintaxis abstracta del SESE para sustentar esos intercambios de seguridad. El ejemplo indicado en la parte 3 de estas especificaciones mostrará cómo se realiza esto.

El paso final es la especificación del contexto de aplicación necesario. Siguiendo las directrices de G.6, esta especificación incluirá las siguientes reglas:

a) *ASE* – El conjunto de los ASE incluye los SESE así como la FTAM (no modificada) y los ACSE ASE.

b) *Intercambios de seguridad* – Se sustentan los intercambios de seguridad dirAuthenticationTwoWay y boundAccessControlCert.

c) *Correspondencias SESE PDU* – Las SE-TRANSFER PDU que transportan el intercambio de seguridad dirAuthenticationTwoWay, ponen en correspondencia una petición A-ASSOCIATE y las PDU de respuesta, respectivamente (por ejemplo, como componentes adicionales al campo de información de usuario). La PDU SE-TRANSFER que transporta el intercambio de seguridad boundAccessControlCert se corresponde con P-DATA.

d) *Limitaciones de concatenación del PDV* – Ninguna.

e) *Limitaciones de inserción de la PDV* – Cada FTAM PDU (o grupo PDU) que contenga una petición F-SELECT o F-CREATED está insertada en una SE-TRANSFER PDU que transporta un intercambio de seguridad boundAccessControlCert.

f) *Limitaciones de procedimiento* – Cualquier condición de error encontrada en el intercambio de seguridad dirAuthenticationTwoWay produce un aborto de la asociación de aplicación.

g) *Limitaciones del contexto de presentación* – El contexto de presentación utilizado para la transferencia de datos de contenido de ficheros debe emplear una sintaxis de transferencia de protección con una correspondencia de protección que implique la confidencialidad y la protección de integridad.

A este contexto de aplicación se le asigna un nuevo identificador de objetos.

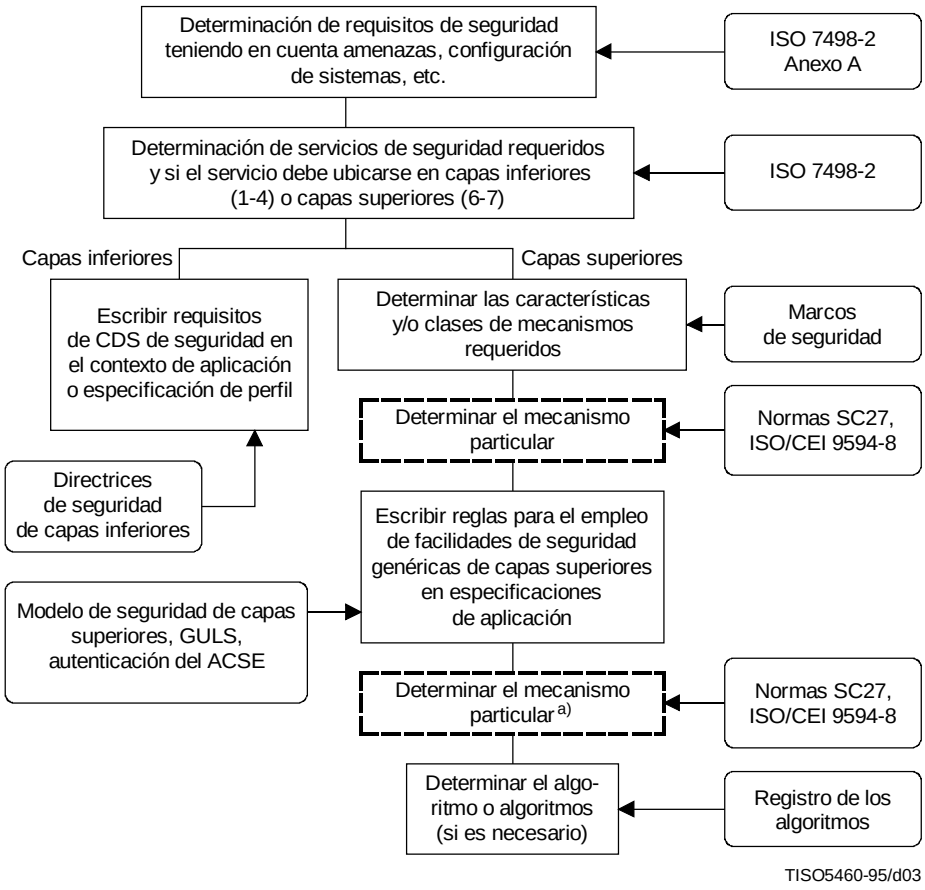
Anexo H

Relación con otras normas

(Este anexo no es parte integrante de esta Recomendación | Norma Internacional)

En este anexo se explica la relación entre las especificaciones de la GULS y otras normas suponiendo que las herramientas de la GULS sean idóneas para proporcionar seguridad a una aplicación particular.

En la Figura H.1 se representa el proceso global de incorporación de seguridad en una norma de protocolo de aplicación.



a) Algunos aspectos de la determinación del mecanismo pueden posponerse a una etapa de elaboración de perfiles, después de la construcción del protocolo.

Figura H.1 – Directrices para la incorporación de seguridad en un protocolo de capa de aplicación

En la Figura H.2 se indica donde encajan las facilidades de la GULS en este proceso global. Seguidamente, se efectúan algunos comentarios a casillas específicas de la Figura H.2.

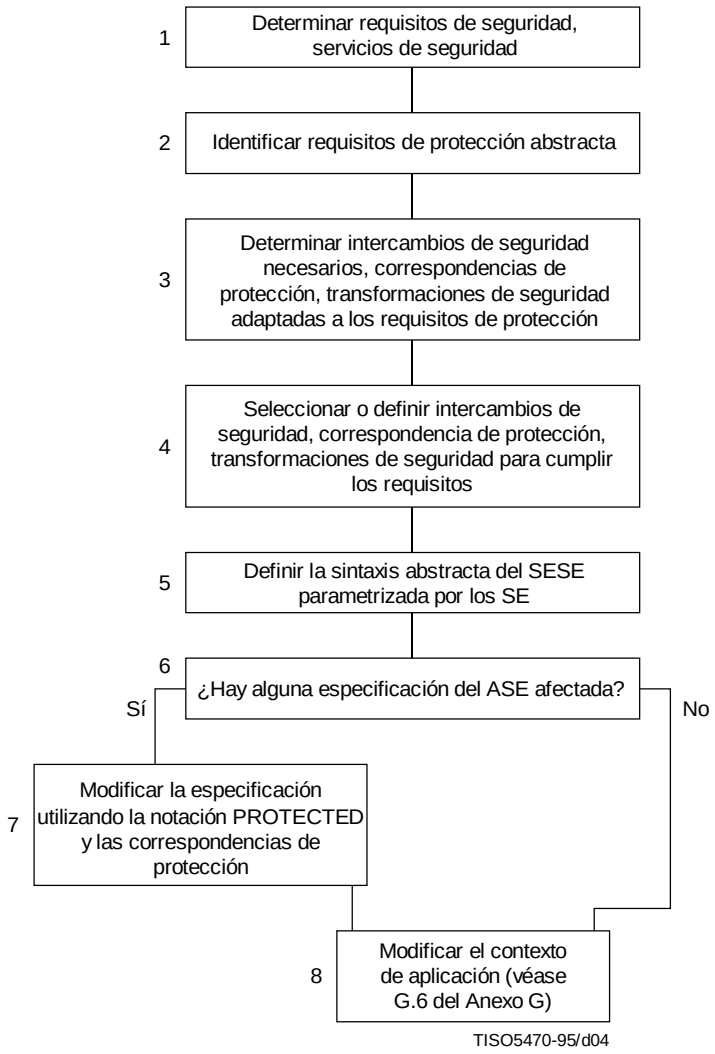


Figura H.2 – Incorporación de la GULS en un protocolo de aplicación de OSI

Casilla 4:

Los intercambios de seguridad, las transformaciones de seguridad y las correspondencias de protección pueden especificarse mediante múltiples tipos distintos de organizaciones. El propósito general es que tales especificaciones puedan reutilizarse en aplicaciones diferentes, en vez de tener que elaborar nuevas especificaciones que ejecuten la misma función básica. El encargado del desarrollo de un protocolo de aplicación deberá investigar las especificaciones existentes en las siguientes fuentes:

- anexos a esta parte de la especificación de seguridad genérica de las capas superiores;
- especificaciones contenidas en otras Recomendaciones UIT-T o Normas Internacionales ya sea bajo la forma de una especificación útil para diversas aplicaciones o para una aplicación OSI determinada;
- especificaciones registradas existentes, por ejemplo, especificaciones desarrolladas y registradas por foros que desarrollan perfiles.

Si no puede encontrarse ninguna especificación adecuada, la organización que lo necesite deberá desarrollar una especificación y normalizarla o registrarla con miras a su utilización futura en otras aplicaciones.

Casilla 6:

En general, una especificación de ASE sólo tendrá que modificarse si es necesario realizar un cambio en la especificación de una sintaxis abstracta. Esto solamente se requiere cuando se introducen funciones de seguridad de los campos (confidencialidad, integridad o autenticación del origen de los datos) que tengan una granularidad inferior a la del valor de datos de presentación. En los demás casos, pueden incorporarse servicios de seguridad mediante cambios de las especificaciones del contexto de aplicación sin afectar a las especificaciones del ASE.

Para sustentar específicamente la seguridad de una aplicación de OSI, será necesario elaborar:

a) Especificaciones para los protocolos que sustenten la seguridad mediante el empleo de una clase particular de mecanismo. Esas especificaciones pueden incluir:

   – intercambios de seguridad que pueden especificarse empleando la notación SECURITY-EXCHANGE definida en 6.2;

   – transformaciones de seguridad que pueden especificarse utilizando la notación SECURITY-TRANSFORMATION definida en 7.2.

   Deberán también elaborarse especificaciones adicionales para definir:

   – la utilización de servicios proporcionados por otros procesos de aplicación de OSI, por ejemplo procesos de directorio, procesos de gestión de la clave;

   – interacciones e interdependencias entre transformaciones de seguridad, intercambios de seguridad y utilización de otros procesos de aplicación de OSI.

   En la medida de lo posible, tales especificaciones deberán ser aplicables a una gama de aplicaciones de OSI.

b) Especificaciones incorporadas en especificaciones de protocolo de aplicación de OSI para relacionar las disposiciones de seguridad con objetos específicos del protocolo de aplicación. Tales especificaciones pueden incluir:

   – *Protecciones selectivas de los campos necesarias en los objetos de datos de aplicación* – Pueden especificarse utilizando la notación PROTECTED o PROTECTED-Q definida en 8.1 y 8.2.

   – *Requisitos para el establecimiento de asociaciones de seguridad* – Las herramientas para la especificación de tales requisitos pueden ser objeto de una normalización separada.

   En la medida de lo posible esto deberá realizarse de forma independiente de una clase determinada de mecanismo.

c) Especificaciones para la aplicación de clases específicas de mecanismos para asegurar aplicaciones de OSI específicas. Tales especificaciones pueden incluir:

   – contextos ASO que especifiquen la utilización de ASE/ASO (por ejemplo, SESE) con fines de seguridad junto con otros ASE/ASO;

   – correspondencias entre los tipos protección requeridos y las transformaciones de seguridad que pueden especificarse empleando la notación PROTECTION-MAPPING definida en 8.4;

   – requisitos para aplicar transformaciones de seguridad específicas a todos los PDV de sintaxis abstractas particulares.

Anexo I

Ejemplos de utilización de las facilidades de seguridad genérica de las capas superiores

(Este anexo no es parte integrante de esta Recomendación | Norma Internacional)

I.1 Ejemplo de utilización de la notación PROTECTED anidada

Como ejemplo ilustrativo de la utilización del tipo parametrizado PROTECTED, supóngase que un diseñador de protocolo de aplicación necesita especificar una PDU con las siguientes características:

a) debe sellarse la PDU completa mediante un mecanismo de integralidad;

b) las PDU contienen campos separados con las siguientes características:

1) un campo (de tipo TypeOne) que no requiere protección de seguridad ulterior;

2) otro campo (de tipo TypeTwo) que debe protegerse confidencialmente empleando un algoritmo simétrico; en la PDU debe también transportarse la clave de cifrado, se realiza el cifrado utilizando un algoritmo asimétrico bajo control de la clave pública del destinatario;

3) otro campo (de tipo TypeThree) que debe ser firmado utilizando la clave privada del enviador.

Esta PDU puede especificarse como un tipo ASN.1 como sigue:

```
SecurePDU ::= PROTECTED
{
    SEQUENCE
    {
        encipheredConfKey       EncipheredConfKey,
        confidentialInfo        ConfidentialInfo,
        signedInfo              SignedInfo,
        clearInfo               TypeOne
    },
    sealed
}
EncipheredConfKey ::=   PROTECTED { ConfKey, encipheredKey}
ConfidentialInfo ::=    PROTECTED { TypeTwo, enciphered }
SignedInfo ::=          PROTECTED { TypeThree, signed }

ConfKey ::= BIT STRING
    -- Value sent is the randomly-generated value supplied
    -- and used by the security transformation used for
    -- the sym-enciphered protection mapping.
```

Esta ASN.1 generará, para cada caso el tipo PROTECTED, una codificación como se especifica en la cláusula 8. La PDU completa es una codificación de este tipo. Las restantes codificaciones similares están anidadas dentro de la primera. La protección proporcionada por la codificación exterior se aplica a la totalidad del contenido interior.

Esta especificación depende de las correspondencias de protección "encipheredKey", "enciphered", "signed" y "sealed", que las ponen en correspondencia con transformaciones. Las últimas definiciones podrían estar en el mismo módulo ASN.1 que la SecurePDU o podrían ser parámetros de ese módulo proporcionados en una fase posterior del desarrollo del contexto de aplicación total.

A continuación se facilita un ejemplo de un conjunto de definiciones de PROTECTION-MAPPING.

```
encipheredKey PROTECTION-MAPPING ::=
{   -- enciphered using an asymmetric algorithm using the public key of the recipient
    SECURITY-TRANSFORMATION { dirEncryptedTransformation }
}
enciphered PROTECTION-MAPPING ::=
{   -- enciphered using a symmetric algorithm; the key used is
    -- the last value delivered under the protection-mapping
    -- "pk-enciphered"
    SECURITY-TRANSFORMATION { dirEncryptedTransformation }
}
signed PROTECTION-MAPPING ::=
{   -- signed using the private key of the sender
    SECURITY-TRANSFORMATION { dirSignedTransformation }
}
```

```
sealed PROTECTION-MAPPING ::=
{   -- sealed under an integrity mechanism
    SECURITY-TRANSFORMATION { sealedTransformation }
    -- sealedTransformation is not currectly defined in this
    -- Specification.
}
```

I.2 Utilización de la notación PROTECTED con calificador de transformación – Ejemplo 1

A continuación se facilita una ilustración sobre el empleo del tipo parametrizado PROTECTED-Q basada en el ejemplo de I.1, pero con calificadores especificados para su utilización por la transformación de seguridad. Los calificadores indican un algoritmo concreto o una fuente de algoritmos para cada transformación de seguridad, así como el tipo de clave que debe utilizarse para cada transformación de seguridad.

Puede especificarse la PDU como un tipo ASN.1 del modo siguiente:

```
SecurePDU ::= PROTECTED-Q
{
    SEQUENCE
    {
        encipheredConfKey       EncipheredConfKey,
        confidentialInfo        ConfidentialInfo,
        signedInfo              SignedInfo,
        clearInfo               TypeOne
    },
    sealed, { sealAlgorithm, preEstablishedKey }
}
EncipheredConfKey ::=   PROTECTED-Q { ConfKey, encipheredKey,
                        { rsaAlgorithm, receiverAsymKeyPair }}
ConfidentialInfo ::=    PROTECTED-Q { TypeTwo, enciphered,
                        { deaAlgorithm, accompanyingEncipheredKey }}
SignedInfo ::=          PROTECTED-Q { TypeThree, signed,
                        { signAlgorithm, senderAsymKeyPair }}

ConfKey ::= BIT STRING

rsaAlgorithm    AlgorithmSelector       ::= specificAlgorithm: { iso ... }
deaAlgorithm    AlgorithmSelector       ::= specificAlgorithm: { iso ... }
signAlgorithm   AlgorithmSelector       ::= algorithmSource: userDependent
sealAlgorithm   AlgorithmSelector       ::= algorithmSource: systemDefault
```

En este ejemplo, el tipo ASN.1 para todos los calificadores es el siguiente:

```
QualifierType ::= SEQUENCE
{
    algorithmSelector   AlgorithmSelector,
    keySelector         KeySelector
}
AlgorithmSelector ::= CHOICE
{
    specificAlgorithm   OBJECT IDENTIFIER,
    algorithmSource     BIT STRING
    {
        systemDefault (0),
        -- Standard system default algorithm to be used.
        userDependent (1)
        -- Algorithm selection based on local user information.
    }
}
KeySelector ::= BIT STRING
{
    preEstablishedKey               (0),
    -- Key has been previously established between the parties.
    userSuppliedKey                 (1),
    -- Key is supplied by the sending user.
```

```
        accompanyingEncipheredKey       (2),
        -- Key accompanies the protected field, conveyed in another
        -- PROTECTED field using the encipheredKey protection mapping, as
        -- another component of the same enclosing ASN.1 construct.
        senderAsymKeyPair               (3),
        -- Encoding key is the private key of the sender; decoding key is
        -- corresponding public key
        receiverAsymKeyPair             (4)
        -- Encoding key is the public key of the receiver; decoding key is
        -- corresponding private key
}
```

Las definiciones de correspondencia de protección necesitan reflejar la utilización posible de los calificadores de transformación, por ejemplo:

```
encipheredKey PROTECTION-MAPPING ::=
{       -- enciphers a key for use in protecting another field
        SECURITY-TRANSFORMATION { qualEncryptedTransformation }
        -- a variant of dirEncryptedTransformation which accepts
        -- algorithm and/or key source qualifier(s) of type
        -- QualifierType
}
enciphered PROTECTION-MAPPING ::=
{       -- general encipherment
        SECURITY-TRANSFORMATION { qualEncryptedTransformation }
        -- a variant of dirEncryptedTransformation which accepts
        -- algorithm and/or key source qualifier(s) of type
        -- QualifierType
}
signed PROTECTION-MAPPING ::=
{       -- general digital signature
        SECURITY-TRANSFORMATION { qualSignedTransformation }
        -- a variant of gulsSignedTransformation which accepts
        -- algorithm and/or key source qualifier(s) of type
        -- QualifierType
}
sealed PROTECTION-MAPPING ::=
{       -- sealed under an integrity mechanism
        SECURITY-TRANSFORMATION { qualSealedTransformation }
        -- a variant of gulsSignedTransformation which accepts
        -- algorithm and/or key source qualifier(s) of type
        -- QualifierType
}
```

I.3 Empleo de la notación PROTECTED con calificador de transformación – Ejemplo 2

A continuación se facilita una ilustración sobre el empleo del tipo parametrizado PROTECTED-Q utilizando un identificador de asociación de seguridad como calificador de un requisito de protección de "confidencialidad" (véase E.4)

Con antelación al empleo de la protección de "confidencialidad" a los datos, se establece una asociación de seguridad establecida externamente entre los dos sistemas en comunicación. Esto establece la transformación de seguridad y los parámetros estáticos necesarios para controlar su funcionamiento y proporcionar la protección de confidencialidad requerida (es decir el algoritmo, modo de operación y claves necesarios). Esto puede conseguirse, por ejemplo, utilizando un protocolo de capa de aplicación de OSI que hace uso del SESE para sustentar el intercambio de seguridad necesario. Además de establecer los parámetros estáticos, este protocolo de establecimiento de la asociación de seguridad constituye un identificador de asociación de seguridad sa-id, que puede utilizarse por los sistemas de codificación y de decodificación para hacer referencia al conjunto de parámetros estáticos.

La especificación indicativa de que un elemento de datos de tipo ClearInfo está protegido con "confidencialidad" mediante los parámetros estáticos identificados por pc-id deberá ser de la forma:

```
PROTECTED-Q { ClearInfo, confidentiality, sa-id }
```

En este ejemplo el calificador es de tipo:

SecurityAssociationId ::= ExternalSAID

como el definido en el módulo de notación ASN.1, Anexo A.

I.4 Ejemplo de utilización del intercambio de seguridad y de la notación PROTECTED en combinación

En este ejemplo el sistema A envía una petición de acceso al sistema B mediante un certificado de control de acceso. El certificado de control de acceso está protegido contra su utilización no autorizada por el método descrito en el Anexo B de la Rec. UIT-T X.812 | ISO/CEI 10181-3 (marco de control de acceso). El certificado de control de acceso contiene un valor de protección (PV) vinculado a un valor de control (CV) mediante la siguiente relación:

$$PV = OWF (CV),$$

donde OWF representa una función unidireccional. El conocimiento del CV constituye la prueba de la propiedad del certificado del control de acceso. Esto implica que debe enviarse CV en forma cifrada a B. Supongamos que A y B tienen pares de claves públicas. Entonces es necesario enviar CV cifrado con la clave pública de B. Además, también se envían el certificado de control de acceso y la respuesta, sellados con la clave privada A.

Puede cumplirse este requisito definiendo un intercambio de seguridad que transporte la información de seguridad necesaria con la petición de acceso (típicamente un valor de datos de presentación procedente de un ASE específico de la aplicación) insertado en el intercambio de seguridad. La definición del intercambio de seguridad podría ser la siguiente:

```
boundAccessControlCert  SECURITY-EXCHANGE  ::=
{
    SE-ITEMS            { boundACC }
    IDENTIFIER                  { ... object identifier ... }
}
boundACC    SEC-EXCHG-ITEM  ::=
{
    ITEM-TYPE                   PROTECTED { SealedSequence, sealed }
    ITEM-ID  1
}
SealedSequence ::= SEQUENCE
{
    accessControlCert   AccessControlCert,
    encipheredCV        EncipheredCV,
    accessRequest       EMBEDDED PDV
                -- The access request PDU is embedded here
}
AccessControlCert       ::= PROTECTED {...certificate contents..., signed }

EncipheredCV            ::= PROTECTED { BIT STRING, encrypted }
```

Anexo J

Bibliografía

(Este anexo no es parte integrante de esta Recomendación | Norma Internacional)

– ISO 8730:1990, Banking – *Requirements for message authentication (wholesale)*.

– Recomendación X.227 del CCITT (1992), *Especificación de protocolo con conexión para el elemento de servicio de control de asociación.*

ISO 8650-1:1988, *Information processing systems – Open Systems Interconnection – Protocol specification for the Association Control Service Element*.

– Recomendación X.736 del CCITT (1992) | ISO/CEI 10164-7:1992, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función señaladora de alarmas de seguridad.*

– Recomendación X.740 del CCITT (1992) | ISO/CEI 10164-8:1993, *Tecnología de la información – Interconexión de sistemas abiertos – Gestión de sistemas: Función de pista de auditoría de seguridad.*

– Recomendación UIT-T X.813[2)] | ISO/CEI 10181-4:...[2)], *Tecnología de la información – Interconexión de sistemas abiertos – Marcos de seguridad en sistemas abiertos: No repudio.*

– Recomendación UIT-T X.814[2)] | ISO/CEI 10181-5:...[2)], *Tecnología de la información – Interconexión de sistemas abiertos – Marcos de seguridad para sistemas abiertos: Marco de confidencialidad.*

– Recomendación UIT-T X.815[2)] | ISO/CEI 10181-6:...[2)], *Tecnología de la información – Interconexión de sistemas abiertos – Marcos de seguridad para sistemas abiertos: Marco de integridad.*

---

2) Actualmente en estado de proyecto.