Unión Internacional de Telecomunicaciones

UIT-T

SECTOR DE NORMALIZACIÓN
DE LAS TELECOMUNICACIONES
DE LA UIT

Y.2722

(01/2011)

SERIE Y: INFRAESTRUCTURA MUNDIAL DE LA INFORMACIÓN, ASPECTOS DEL PROTOCOLO INTERNET Y REDES DE LA PRÓXIMA GENERACIÓN

Redes de la próxima generación – Seguridad

Mecanismos de gestión de identidad en las NGN

Recomendación UIT-T Y.2722

RECOMENDACIONES UIT-T DE LA SERIE Y

INFRAESTRUCTURA MUNDIAL DE LA INFORMACIÓN, ASPECTOS DEL PROTOCOLO INTERNET Y REDES DE LA PRÓXIMA GENERACIÓN

INFRAESTRUCTURA MUNDIAL DE LA INFORMACIÓN	
Generalidades	Y.100–Y.199
Servicios, aplicaciones y programas intermedios	Y.200–Y.299
Aspectos de red	Y.300–Y.399
Interfaces y protocolos	Y.400–Y.499
Numeración, direccionamiento y denominación	Y.500–Y.599
Operaciones, administración y mantenimiento	Y.600–Y.699
Seguridad	Y.700–Y.799
Características	Y.800–Y.899
ASPECTOS DEL PROTOCOLO INTERNET	
Generalidades	Y.1000–Y.1099
Servicios y aplicaciones	Y.1100–Y.1199
Arquitectura, acceso, capacidades de red y gestión de recursos	Y.1200–Y.1299
Transporte	Y.1300–Y.1399
Interfuncionamiento	Y.1400–Y.1499
Calidad de servicio y características de red	Y.1500–Y.1599
Señalización	Y.1600–Y.1699
Operaciones, administración y mantenimiento	Y.1700–Y.1799
Tasación	Y.1800–Y.1899
Televisión IP sobre redes de próxima generación	Y.1900–Y.1999
REDES DE LA PRÓXIMA GENERACIÓN	
Marcos y modelos arquitecturales funcionales	Y.2000–Y.2099
Calidad de servicio y calidad de funcionamiento	Y.2100–Y.2199
Aspectos relativos a los servicios: capacidades y arquitectura de servicios	Y.2200–Y.2249
Aspectos relativos a los servicios: interoperabilidad de servicios y redes en las redes de la próxima generación	Y.2250–Y.2299
Numeración, denominación y direccionamiento	Y.2300–Y.2399
Gestión de red	Y.2400–Y.2499
Arquitecturas y protocolos de control de red	Y.2500–Y.2599
Smart ubiquitous networks	Y.2600–Y.2699
Seguridad	**Y.2700–Y.2799**
Movilidad generalizada	Y.2800–Y.2899
Entorno abierto con calidad de operador	Y.2900–Y.2999
Redes futuras	Y.3000–Y.3099

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

Recomendación UIT-T Y.2722

Mecanismos de gestión de identidad en las NGN

Resumen

En la Recomendación UIT-T Y.2722 se presentan los mecanismos de gestión de identidad que pueden utilizarse para satisfacer los requisitos de la gestión de identidad (IdM) y las necesidades del despliegue de las redes de la próxima generación (NGN).

Historia

Edición	Recomendación	Aprobación	Comisión de Estudio
1.0	ITU-T Y.2722	2011-01-28	13

Palabras clave

Gestión de identidad, identidad federada, mecanismos de gestión de identidad, red de la próxima generación, seguridad.

PREFACIO

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

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

La aprobación de Recomendaciones por los Miembros del UIT-T es el objeto del procedimiento establecido en la Resolución 1 de la AMNT.

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

NOTA

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

La observancia de esta Recomendación es voluntaria. Ahora bien, la Recomendación puede contener ciertas disposiciones obligatorias (para asegurar, por ejemplo, la aplicabilidad o la interoperabilidad), por lo que la observancia se consigue con el cumplimiento exacto y puntual de todas las disposiciones obligatorias. La obligatoriedad de un elemento preceptivo o requisito se expresa mediante las frases "tener que, haber de, hay que + infinitivo" o el verbo principal en tiempo futuro simple de mandato, en modo afirmativo o negativo. El hecho de que se utilice esta formulación no entraña que la observancia se imponga a ninguna de las partes.

PROPIEDAD INTELECTUAL

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

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

© UIT 2011

Reservados todos los derechos. Ninguna parte de esta publicación puede reproducirse por ningún procedimiento sin previa autorización escrita por parte de la UIT.

ÍNDICE

Página

1 Alcance 1
2 Referencias 1
3 Definiciones 2
4 Abreviaturas 2
5 Convenios 3
6 Mecanismos y procedimientos que soportan las funciones de la IdM 4
6.1 Gestión del ciclo de vida 4
6.2 Autentificación y garantía de la autentificación 4
6.3 Correlación y vinculación 24
6.4 Descubrimiento 25
6.5 Comunicaciones e intercambio de información del IdM 26
6.6 Protección de la información personal identificable (PII) 30
6.7 Funciones de identidad federada 30
6.8 Control de acceso a la información de identidad 31
6.9 Inicio de sesión único 31
6.10 La desconexión única 32
7 Seguridad 36
Apéndice I – Autentificación del mensaje WSS X.509 v3 37
Apéndice II – Mecanismo de control de acceso basado en "OpenID + OAuth" 39
II.1 OAuth [b-IETF RFC 5849] 39
II.2 Uso conjunto de OpenID y OAuth 39
III.3 Flujo de la autorización OpenID + OAuth 39
Bibliografía 42

Recomendación UIT-T Y.2722

Mecanismos de gestión de identidad en las NGN

1 Alcance

En [UIT-T Y.2721], *Requisitos de gestión de identidad en las NGN y ejemplos de utilización*, se especifican los requisitos de gestión de identidad (IdM) para las redes de la próxima generación (NGN). En esta Recomendación se describen los mecanismos específicos de la IdM y las series de opciones que deben utilizarse para satisfacer los requisitos estipulados en [UIT-T Y.2721]. Además, sugiere prácticas óptimas y directrices para soportar la interoperabilidad y otras necesidades.

Está Recomendación se ha concebido para utilizarse junto con [ITU-T Y.2720] e [ITU-T Y.2721], dado que los conceptos arquitectónicos fundamentales, los requisitos y los ejemplos de utilización no se reproducen en esta Recomendación.

NOTA – Los implementadores y usuarios de los mecanismos descritos deberán cumplir toda la legislación, la normativa y las políticas nacionales y regionales aplicables. En ciertas normativas y legislaciones se exige la aplicación de mecanismos para proteger la información de identificación personal.

2 Referencias

Las siguientes Recomendaciones del UIT-T y otras referencias contienen disposiciones que, mediante su referencia en este texto, constituyen disposiciones de la presente Recomendación. Al efectuar esta publicación, estaban en vigor las ediciones indicadas. Todas las Recomendaciones y otras referencias son objeto de revisiones por lo que se alienta a los usuarios de esta Recomendación a que investiguen la posibilidad de aplicar las ediciones más recientes de las Recomendaciones y otras referencias citadas a continuación. Se publica periódicamente una lista de las Recomendaciones UIT-T actualmente vigentes. En la presente Recomendación, la referencia a un documento no confiere a este último, como documento autónomo, la categoría de una Recomendación.

[UIT-T X.509] Recomendación UIT-T X.509 (2005) | ISO/IEC 9594-8:2005, *Tecnología de la información – Interconexión de sistemas abiertos – El directorio: Marcos para certificados de claves públicas y atributos*.

[UIT-T X.1035] Recomendación UIT-T X.1035 (2007), *Password-authenticated key exchange (PAK) protocol*.

[UIT-T X.1141] Recomendación UIT-T X.1141 (2006), *Security Assertion Markup Language (SAML 2.0)*.

[UIT-T X.1252] Recomendación UIT-T X.1252 (2010), *Marco general para la gestión de identidades en las redes de la próxima generación*.

[UIT-T Y.2012] Recomendación UIT-T Y.2012 (2006), *Functional requirements and architecture of NGN release 1*.

[UIT-T Y.2701] Recomendación UIT-T Y.2701 (2007), *Mecanismos y procedimientos de seguridad para las NGN*.

[UIT-T Y.2702] Recomendación UIT-T Y.2702 (2008), *Authentication and authorization requirements for NGN release 1*.

[UIT-T Y.2704] Recomendación UIT-T Y.2704 (2010), *Security mechanisms and procedures for NGN*.

[UIT-T Y.2720] Recomendación UIT-T Y.2720 (2009), *NGN identity management framework*.

[UIT-T Y.2721] Recomendación UIT-T Y.2721 (2010), *NGN identity management requirements and use cases*.

[3GPP TS 23.228] 3GPP TS 23.228 (en vigor), *IP Multimedia Subsystem (IMS); Stage 2*.

[ATIS 33102] ATIS.3GPP.33.102V710-2007, *Security Architecture*.

[IETF RFC 2289] IETF RFC 2289 (1998), *A One-Time Password System*.

3 Definiciones

La presente Recomendación se basa en los siguientes términos definidos en [UIT-T Y.2720] y [UIT-T X.1252].

En particular, se han adoptado de [UIT-T X.1252] las siguientes definiciones:

3.1 proveedor de identidad (IdP): Véase Proveedor de servicios de identidad (IdSP)

3.2 proveedor de servicios de identidad (IdSP): Entidad que verifica, mantiene, gestiona y puede crear y asignar información de identidad de otras entidades.

4 Abreviaturas

En la presente Recomendación se utilizan las siguientes abreviaturas y acrónimos:

AKA Acuerdo de autentificación y clave (*authentication and key agreement*)

ASP Proveedor de servicios de aplicación (*application service provider*)

AuC Centro de Autentificación (*authentication centre*)

AV Vector de autentificación (*authentication vector)*

BSF Función de servidor de inicialización (*bootstrapping server function*)

CK Clave de cifrado (*ciphering key*)

GBA Arquitectura de inicialización genérica (*generic bootstrapping architecture*)

HSS Sistema de abonado residencial (*home subscriber system*)

IdM Gestión de identidad

IdP Proveedor de identidad

IdSP Proveedor de servicios de identidad

IK Clave de integridad (*integrity key*)

IMPI Identidad de usuario privado de multimedios IP (*IP multimedia private user identity*)

IMPU Identidad de usuario público de multimedios IP (*IP multimedia public user identity*)

IMS Subsistema de multimedios IP (*IP multimedia subsystem*)

IMSI Identidad de abonado móvil internacional (*international mobile subscriber identity*)

IPTV Televisión por Protocolo de Internet (*Internet protocol television*)

ISIM Módulo de identidad del abonado IMS (*IMS subscriber identity module*)

LDAP Protocolo ligero de acceso al directorio (*lightweight directory access protocol*)

MS Estación móvil (*mobile station*)

NAF Función de aplicación de red (*network application function*)

NGN Red de la próxima generación (*next generation network*)

OASIS	Organización para el progreso de la normalización de la información estructurada (*Organization for the advancement of structured information standards*)
OTP	Contraseña de uso único (*one time password*)
PII	Información de identificación personal (*personally identifiable information*)
PKI	Infraestructura de clave pública (*public key infrastructure*)
RP	Parte confiante (*relying party*)
SAML	Lenguaje de marcación de aserción de seguridad (*security assertion markup language*)
SIP	Protocolo de inicio de sesión (*session initiation protocol*)
SLF	Función de localizador de abonado (*subscriber locator function*)
SOAP	Protocolo simple de acceso a objetos (*simple object access protocol*)
SQL	Lenguaje de consulta estructurada (*structured query language*)
SSO	Inicio de sesión único (*single sign-on*)
UE	Equipo de usuario (*user equipment*)
UICC	Tarjeta universal de circuito integrado (*universal integrated circuit card*)
USIM	Módulo de identificador de abonado universal (*universal subscriber identifier module*)
WAP	Protocolo de aplicaciones inalámbricas (*wireless application protocol*)
WSS	Seguridad de los servicios web (*web services security*)
XML	Lenguaje de marcación extensible (*eXtensible markup language*)
XRDS	Secuencia eXtensible de descriptores de recursos (*eXtensible resource descriptor sequence*)

5 Convenios

En esta Recomendación:

La expresión "se le exige que" indica un requisito que debe cumplirse estrictamente, no permitiéndose desviación alguna si la Recomendación pretende reclamar su conformidad.

La expresión "se recomienda" indica un requisito recomendado pero que no se exige con carácter taxativo. Por ello no es necesario cumplir este requisito para reclamar su conformidad.

La expresión "se le prohíbe" indica un requisito que debe cumplirse estrictamente, sin permitirse desviación alguna si la Recomendación pretende ser conforme.

La expresión "puede opcionalmente" indica un requisito opcional admisible que no reviste en absoluto el carácter de recomendación. Esta expresión no pretende dar a entender que la implementación del fabricante deba suministrar una opción o característica que puedan ser activadas opcionalmente por el operador de red o proveedor del servicio. Más bien significa que el fabricante puede proporcionar opcionalmente esta característica sin menoscabo de su derecho de reclamar la conformidad con la Recomendación.

En la parte principal de esta Recomendación y sus apéndices, los anteriores requisitos podrán expresarse por medio de tiempos verbales o la utilización del verbo deber (formulación afirmativa o negativa). La utilización de tales tiempos o expresiones en los apéndices o partes explícitamente *informativas* no tendrá propósito normativo.

6 Mecanismos y procedimientos que soportan las funciones de la IdM

6.1 Gestión del ciclo de vida

Para mayor información la gestión del ciclo de vida de la identidad, véase [UIT-T Y.2720], *Marco general para la gestión de identidades en las redes de la próxima generación*.

6.2 Autentificación y garantía de la autentificación

En esta cláusula se describen mecanismos de autentificación y garantía de la identidad y de la información de la identidad. Se mencionan mecanismos de autentificación definidos en otro contexto.

El IdSP puede utilizar mecanismos de autentificación específicos, tales como la autentificación basada en el perfil del lenguaje de marcación de aserción de seguridad (SAML) de los servicios web (WS), la autentificación por certificado y la autentificación por contraseña (incluida la contraseña de uso único (OTP)), para aplicaciones o servicios específicos en función del contexto y el nivel de garantía requerido. El método (o los métodos) de autentificación se selecciona(n) con arreglo a los requisitos del nivel de garantía. El IdSP puede solicitar información del nivel de garantía para determinar los métodos de autentificación que satisfagan los requisitos del nivel de garantía del proveedor de servicios.

6.2.1 Autentificación basada en el perfil SAML de seguridad del WS

6.2.1.1 Aserciones SAML

El lenguaje de marcación de aserción de seguridad (SAML) [UIT-T X.1141] especifica el formato de las aserciones que pueden utilizarse en la para intercambiar información de seguridad a efectos de IdB. Entre las funciones IdM que pueden implementarse utilizando SAML se encuentran la autentificación, el intercambio de atributos y la autorización, que se corresponden con los tres tipos de declaración sobre el sujeto de una aserción SAML:

- Declaración de autentificación – Transporta la información de que el sujeto de la aserción ha sido autenticado por un medio específico en un instante determinado.
- Declaración de atributo – Transporta la información de que el sujeto de la aserción está asociado con los atributos relacionados.
- Declaración de decisión de autorización – Transporta la información de que se ha permitido al sujeto de la aserción acceder a un recurso especificado, o de que se le ha denegado dicho acceso.

El contenido de una aserción SAML puede describirse a alto nivel del siguiente modo: el expedidor **R** expidió la aserción **A** en el instante **t** para el sujeto **S** siempre que las condiciones **C** sean válidas.

Las aserciones SAML empleadas para comunicar la información de autentificación, atributo y autorización se transportan en mensajes del protocolo simple de acceso a objetos (SOAP). Cuando se intercambian mensajes SOAP por un sistema de transporte no protegido, se recomienda encarecidamente que se utilice la firma [b-W3C XML signature] para verificar la relación entre el mensaje SOAP y las declaraciones de aserción transportadas en el mensaje. En la norma *Seguridad de los servicios web (WSS): Perfil del testigo SAML* [b-OASIS SAML token] se describe el modo en que:

- Se transportan las aserciones SAML (denominadas también testigos SAML) en un mensaje SOAP y se referencian a partir de éste.
- Se utiliza la firma XML para vincular un sujeto y las declaraciones de una aserción SAML con un mensaje SOAP.

En la figura 1 se representa una utilización característica del testigo SAML con un mensaje SOAP construido con arreglo a esta Recomendación.

En este ejemplo, un mensaje SOAP firmado contiene una aserción SAML con una declaración de atributo. A partir de la información de esta declaración, el destinatario podría tomar decisiones de control de acceso.

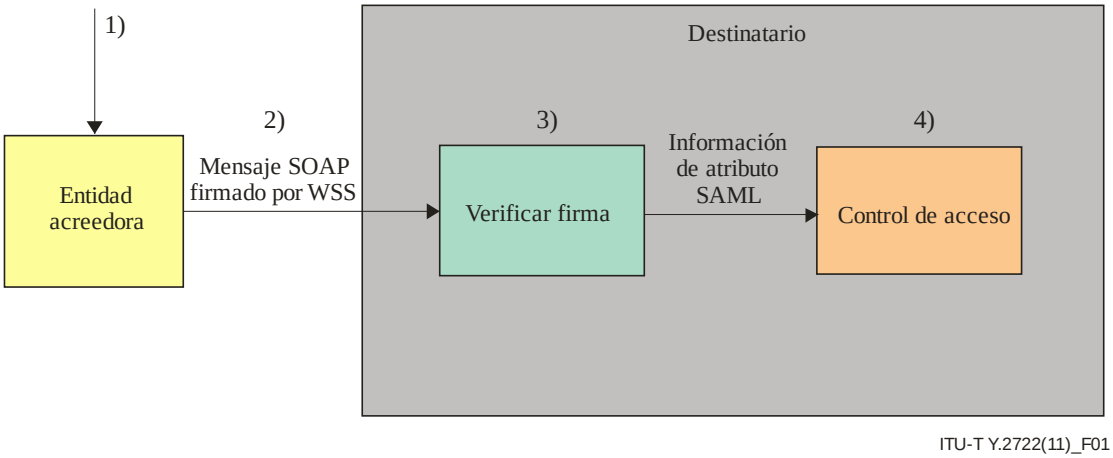


Figura 1 – Pasos típicos de la construcción y procesamiento de un mensaje SOAP con un testigo SAML

1) La entidad acreditadora obtiene una aserción SAML con una declaración de atributo y la construye e incluye en el mensaje SOAP con arreglo a [b-OASIS SAML token].

2) La entidad acreditadora envía un mensaje SOAP firmado por WSS al destinatario.

3) El destinatario verifica la firma digital.

4) La información de la declaración SAML puede utilizarse para tomar decisiones de control de acceso.

6.2.1.2 Métodos de confirmación del sujeto de los testigos SAML

En la norma OASIS *Seguridad de los servicios web (WSS): Perfil del testigo SAML* [b-OASIS SAML token] se especifica cómo adjuntar una aserción SAML a un mensaje SOAP y se definen dos métodos obligatorios de confirmación del sujeto:

- el del poseedor de la clave, y
- el del remitente lo garantiza.

En la figura 2 se muestran los principales elementos XML del mensaje SOAP construido con arreglo a [b-OASIS WSS SOAP].

La aserción SAML se coloca en el encabezamiento <wsse:Security>, que también contiene la firma digital <ds:Signature>. El destinatario del mensaje SOAP utiliza la firma digital para verificar que el remitente del mensaje conoce la clave utilizada para calcular la firma sobre el compendio del cuerpo SOAP y para comprobar su integridad. El algoritmo del compendio es el SHA 1 y el de la firma el RSA_SHA 1, y ambos están especificados en [b-OASIS WSS SOAP]. El valor de la firma se puede obtener del elemento de la firma digital <ds:Signature Value> de <ds:Signature>.

Hay dos métodos diferentes de confirmación del sujeto que, a su vez, definen modos distintos de transmitir al destinatario la información relativa a la clave.

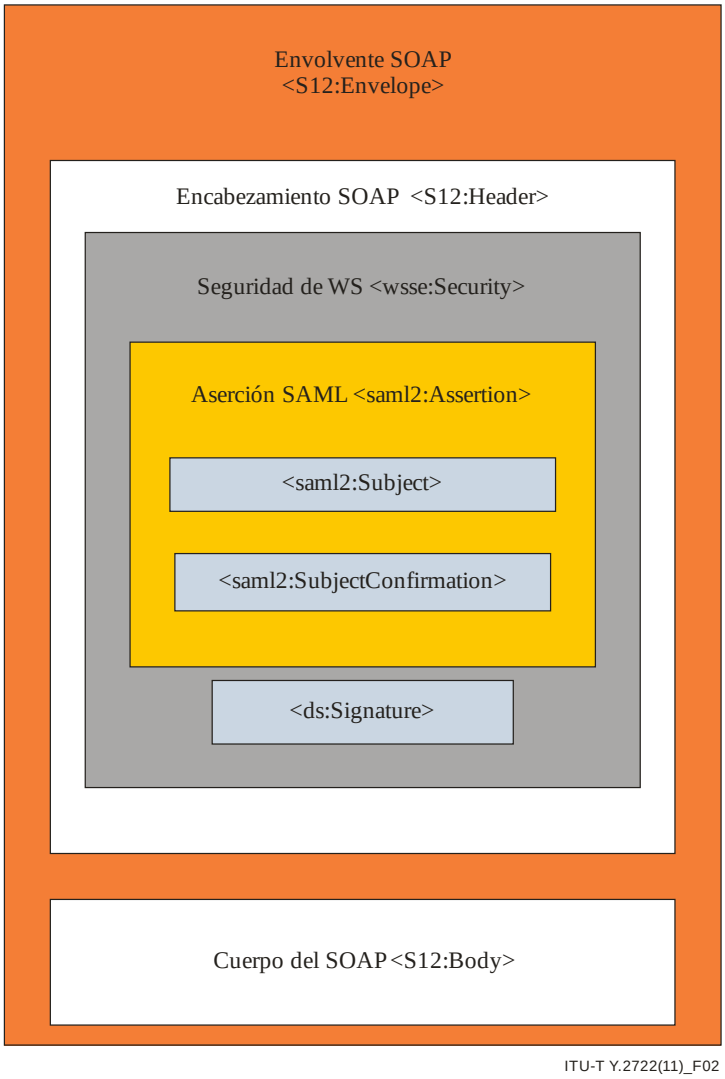


Figura 2 – Estructura del mensaje SOAP con aserción SAML

En las siguientes cláusulas se describen los dos métodos de confirmación del sujeto.

6.2.1.2.1 Método de confirmación del sujeto: poseedor de la clave

En la figura 3 se representa la estructura de la aserción SAML que se utiliza en el método de confirmación del sujeto "poseedor de la clave". El atributo método del elemento <saml2:SubjectConfirmation> indica el método de confirmación del sujeto (poseedor de la clave).

Según este método, el remitente (también denominado Entidad acreditadora) debe probar que tiene derecho a formular las declaraciones acerca del sujeto demostrando que conoce la clave, definida en el elemento <ds:KeyValue> contenido en el elemento <ds:KeyInfo> de la aserción SAML. El elemento <ds:KeyInfo> define una clave secreta o pública que se utiliza para confirmar la identidad del sujeto. Este método especifica además que el remitente puede hacerlo, firmando un compendio del cuerpo SOAP con dicha clave. La firma está contenida en el elemento <ds:Signature> del encabezamiento de Seguridad WS como se muestra en la figura 2.

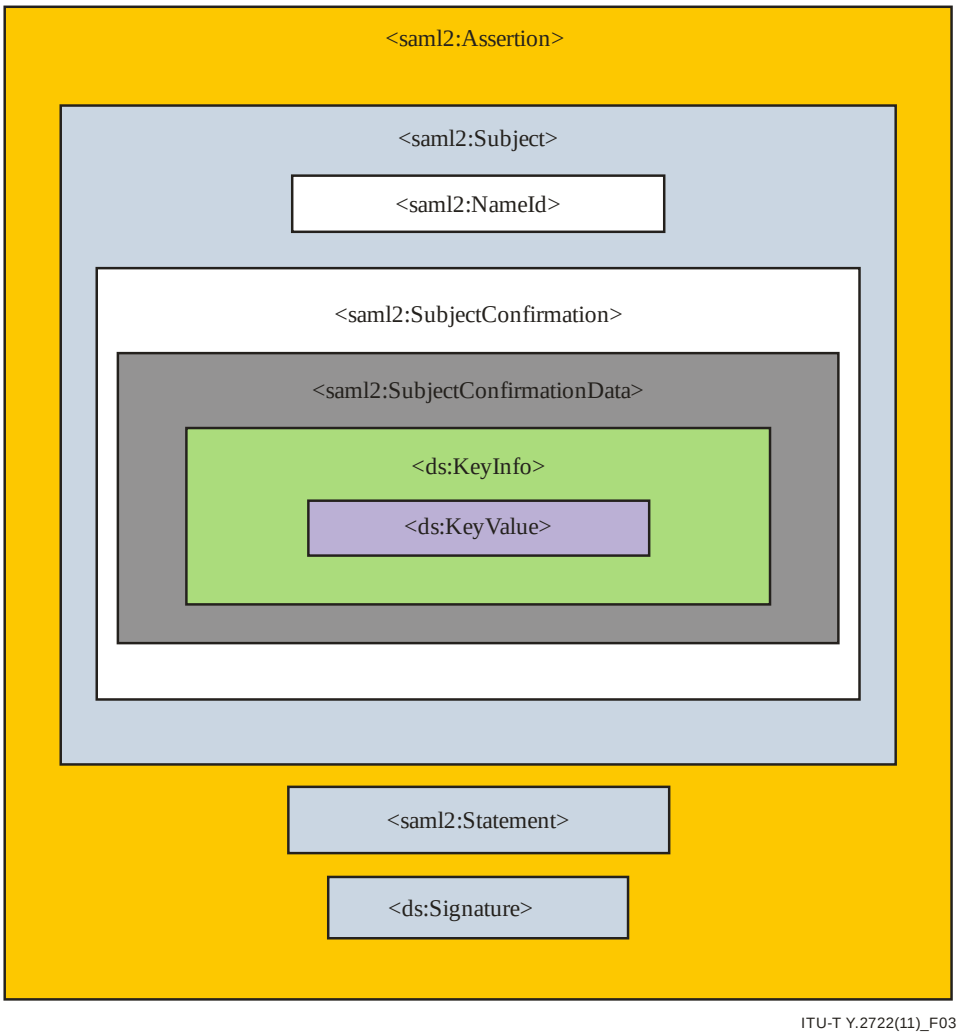


Figura 3 – Estructura de la aserción SAML utilizada en el método de confirmación del sujeto "poseedor de la clave"

El destinatario del mensaje SOAP obtiene la clave gracias a la información ofrecida por la entidad acreditadora en el elemento <ds:keyInfo>. Acto seguido, el destinatario calcula la firma digital del cuerpo SOAP y comprueba su concordancia con la firma presentada por la entidad acreditadora. Si concuerdan, el sujeto y las declaraciones de la aserción SAML pueden atribuirse a la entidad acreditadora y puede considerarse que el contenido del cuerpo SOAP, cuya integridad está protegida por la clave, ha sido entregado por la entidad acreditadora.

6.2.1.2.2 Método de confirmación del sujeto: el remitente lo garantiza

En la figura 4 se representa la estructura de la aserción SAML utilizada en el método de confirmación del sujeto "el remitente lo garantiza". El atributo método del elemento <saml2:SubjectConfirmation> indica el método de confirmación del sujeto (el remitente lo garantiza, *sender-vouches*).

El destinatario confía en las aserciones SAML de la entidad acreditadora relativas al sujeto, a condición de que el valor del atributo del elemento <SubjectConfirmation> indique que se utiliza el método "el remitente lo garantiza (*sender-vouches*)".

La entidad acreditadora obtiene una o más aserciones, o referencias a aserciones, de una o más autoridades y las introduce en el mensaje SOAP. Acto seguido, calcula la firma del compendio de aserciones SAML y el cuerpo del mensaje SOAP. La firma está contenida en el elemento <ds:Signature> del encabezamiento de Seguridad WS (como se muestra en la figura 2).

Opcionalmente, la entidad acreditadora ofrece al destinatario información sobre la clave utilizada para calcular la firma. Se espera que, de no existir dicha información, el destinatario tenga otra forma de identificar la clave.

El destinatario comprueba la validez de la firma. Si ésta es válida, el destinatario establece que la entidad acreditadora es la autora de las declaraciones.

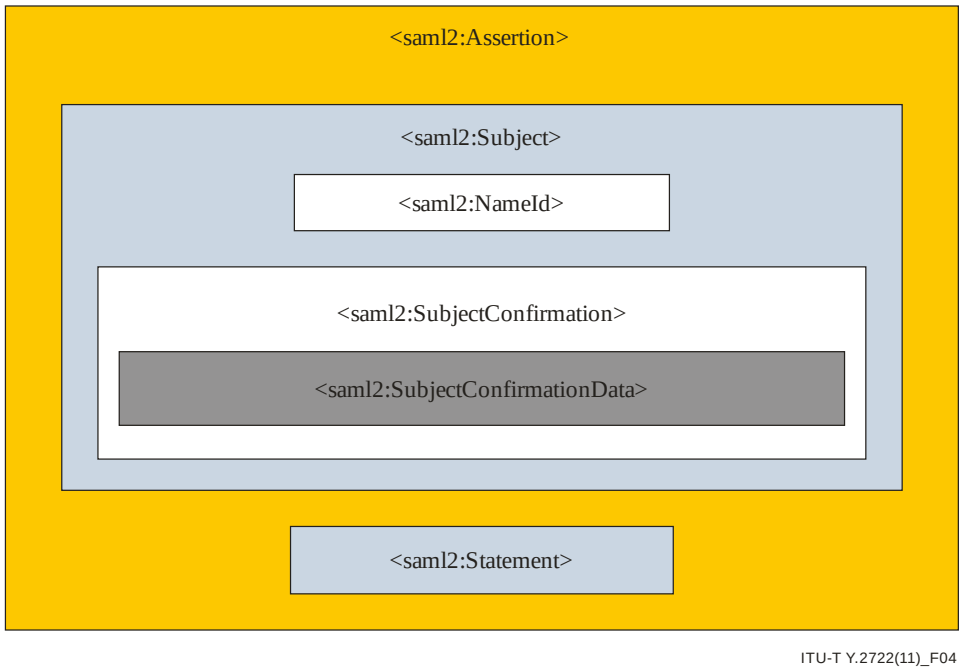


Figura 4 – Estructura de la aserción SAML utilizada en el método de confirmación del sujeto "el remitente lo garantiza"

6.2.2 Autentificación basada en certificado

Pueden emplearse certificados [ITU-T X.509] para aplicaciones o servicios específicos en función del contexto y el nivel de garantía requerido. La utilización de los certificados [UIT-T X.509] para la autentificación se describe en [UIT-T Y.2704], *Mecanismos y procedimientos de seguridad para las NGN*.

6.2.3 Autentificación basada en contraseña

La autentificación basada en contraseña puede utilizarse para aplicaciones o servicios específicos dependiendo del contexto y del nivel de garantía requerido. Remítase el lector a [UIT-T X.1035] donde se describe el mecanismo de autentificación basado en contraseña.

6.2.4 Contraseña de uso único (OTP)

La OTP puede utilizarse para aplicaciones o servicios específicos dependiendo del contexto y del nivel de garantía requerido. En [b-IETF RFC 2289] se describe un método de implementar la OTP.

6.2.5 Utilización del Acuerdo de autentificación y clave (AKA) para autentificación mutua

El protocolo de acuerdo de autentificación y clave del sistema de telecomunicaciones móviles universales (UMTS) puede utilizarse para ofrecer la autentificación mutua de la estación móvil (MS) y la red. El AKA UMTS es un protocolo de pregunta-respuesta que utiliza una clave a largo plazo compartida entre el módulo de identidad del abonado UMTS (USIM) y el centro de autentificación (AuC), que puede ser un componente del IdSP. Estas entidades residen en la tarjeta universal de circuito integrado (UICC) de la MS y en la red originaria de la estación móvil, respectivamente. En determinados acuerdos comerciales, el IdSP podría ofrecer las funciones del AuC. Las especificaciones del protocolo AKA figuran en [ATIS 33102].

6.2.6 Integración de la autentificación basada en PKI con el IMS

La seguridad de los subsistemas multimedios IP (IMS) se basa en el mecanismo AKA, que utiliza un protocolo de pregunta-respuesta y secreto compartido para la autentificación. Sin embargo, la seguridad de algunos servicios NGN (por ejemplo, la IPTV) se basa en certificados de la infraestructura de clave pública (PKI). Para permitir la combinación de los servicios NGN utilizando certificados PKI y seguridad IMS sería conveniente integrar la autentificación basada en PKI con la autentificación IMS de modo que se potencie la seguridad del IMS.

La integración de la autentificación IMS con la autentificación basada en PKI permite que el equipo del usuario y la red se autentiquen recíprocamente de acuerdo con sus respectivos certificados para acordar un conjunto de claves criptográficas generadas con los mismos algoritmos que el AKA. Para ello, hay que suministrar al equipo del usuario y a la red los correspondientes certificados y claves privadas, para que puedan realizar las operaciones PKI.

En cuanto al acuerdo sobre la clave de cifrado (*CK*) y la clave de integridad (*IK*), en el mecanismo de integración descrito se especifican dos opciones:

1) Establecer un acuerdo entre las claves *CK* e *IK* utilizando el secreto compartido entre la función de usuario final y la entidad funcional Perfil de usuario del servicio S-5 (SUP-FE), definido en [UIT-T Y.2012].

2) Establecer un acuerdo sobre las claves *CK* e *IK* sin utilizar el secreto compartido.

El flujo de la llamada genérica correspondiente a la primera y la segunda opciones se representan en las figura 5 y 6, respectivamente.

6.2.6.1 Convenios

En esta cláusula se utilizan los siguientes conectores:

– "|" indica concatenación de cadenas de caracteres.

– *CK* indica Clave de cifrado.

– *IK* indica Clave de integridad.

– *K()* indica una encriptación de clave simétrica.

– N_{pr} *[]* indica encriptación con clave de red privada N_{pr}.

– N_{pu} *[]* indica encriptación con clave de red pública N_{pu} obtenida del certificado de red.

– U_{pr} *[]* indica encriptación con la clave de usuario privada U_{pr}.

6.2.6.2 Entidades implicadas en la autentificación

- S-5 – Entidad funcional Perfil de usuario del servicio (SUP-FE).
- Función de usuario final. Entidad capaz de ejecutar un cliente SIP.
- Entidad funcional Control de sesión de la llamada S-n (CSE-FE), siendo S-n una de las entidades siguientes:
    - S-1 entidad funcional control de sesión de la llamada servidora (S-CSC-FE).
    - S-2 entidad funcional control de sesión de la llamada intermediaria (P-CSC-FE).
    - S-3 entidad funcional control de sesión de la llamada interrogante (I-CSC-FE).

Para designar a una de estas entidades se utiliza S-n, sin que existan diferencias entre ellas en lo que al procedimiento de autentificación descrito se refiere. En [ITU-T Y.2012] figuran las descripciones de las entidades funcionales de las NGN (S-1, S-2, S-3, S-5 y función de usuario final).

6.2.6.3 Establecimiento del acuerdo entre las claves CK e IK utilizando un secreto compartido entre la función de usuario final y S-5 (opción 1)

En la figura 5 se representa el flujo de la llamada. Los pasos básicos son los siguientes:

1) La función de usuario final envía a S-n la petición de Registro SIP con la *IMPU* y la *IMPI* del usuario.

2) S-1 solicita a S-5 una pregunta aleatoria *RAND, CK e IK*. Los valores *RAND, CK e IK* se especifican en [ATIS 33102].

3) S-1 recibe de S-5 *RAND, CK e IK* correspondiente al usuario.

4) S-n envía a la función de usuario final el mensaje SIP 401 no autorizado con una pregunta *RAND* y su valor encriptado $N_{pr}[RAND]$.

   La función de usuario final:

   - recibe el valor *A*, supuestamente igual a *RAND*, y el *B*, supuestamente igual a $N_{pr}[RAND]$;
   - recupera la clave pública de red N_{pu};
   - desencripta *B* con N_{pu} y compara el resultado con *A*. Si su valor coincide, se autentica la red, en caso contrario, se cancela el procedimiento de autentificación;
   - genera *IK* y *CK* utilizando el secreto compartido K_s;
   - genera el valor $U_{pr}[N_{pu}[K]|K(RAND)]$.

5) La función de usuario final envía a S-n el mensaje de Registro SIP con los identificadores *IMPI* e *IMPU* y el valor $U_{pr}[N_{pu}[K]|K(RAND)]$.

6) S-1 envía a S-5 los datos recibidos en el paso 5 y solicita una verificación y el registro del usuario.

   S-5:

   - consulta el certificado del usuario para obtener la clave pública de éste U_{pu};
   - desencripta con U_{pu} el valor recibido *C*, supuestamente igual a $U_{pr}[N_{pu}[K]|K(RAND)]$ para recuperar el valor *D*|*E*, siendo *D* supuestamente igual a $N_{pu}[K]$ y *E* supuestamente igual a *K(RAND)*;
   - desencripta con el valor de la clave privada de la red N_{pr} el valor *D* para obtener *K'*;
   - desencripta con *K'* el valor *E* para obtener *RAND'*;
   - compara *RAND'* con *RAND*. Si concuerdan, el usuario queda autenticado.

7) S-5 comunica a S-1 el resultado de la autentificación y el registro del usuario.

8) S-1 utiliza el registro para comprobar si el usuario autenticado está autorizado a registrar y recibir el servicio solicitado. De ser éste el caso, S-n notifica a la función de usuario final la concesión del acceso, utilizando el mensaje SIP 200 OK.

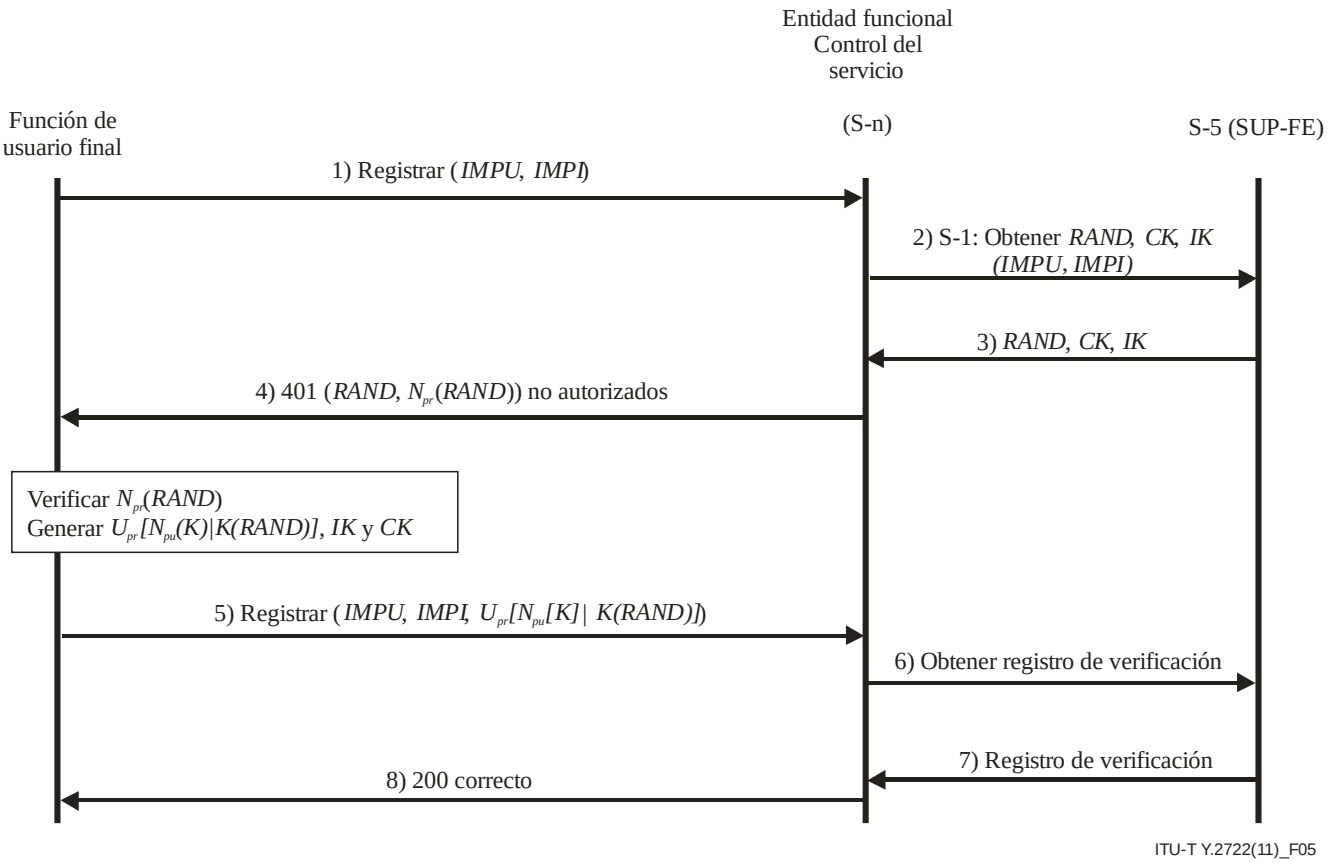


Figura 5 – Integración del mecanismo de autentificación IMS con el basado en PKI (opción 1)

6.2.6.4 Establecimiento de un acuerdo sobre las claves CK y CI sin utilizar un secreto compartido entre la función de usuario final y S-5 (opción 2)

En la figura 6 se representa el flujo de la llamada. Los pasos básicos son los siguientes:

1) La función de usuario final envía a S-n una petición de Registro SIP con la *IMPI* y la *IMPU* del usuario.

2) S-1 solicita una pregunta aleatoria *RAND* a S-5. El valor de *RAND* se especifica en [ATIS 33102].

3) S-1 recibe de S-5 la *RAND* correspondiente al usuario especificado.

4) S-n envía a la función de usuario final el mensaje SIP 401 no autorizado con una pregunta *RAND* y su valor encriptado $N_{pr}[RAND]$.

   La función de usuario final:

   - recibe el valor *A*, supuestamente igual a *RAND*, y el *B*, supuestamente igual a $N_{pr}[RAND]$;
   - recupera la clave pública de red N_{pu};
   - desencripta *B* con N_{pu} y compara el resultado con *A*. Si los valores son iguales, la red queda autenticada, de lo contrario se cancela el procedimiento;
   - genera *IK* y *CK* utilizando la clave generada aleatoriamente *K*;
   - genera el valor $U_{pr}[N_{pu}[K]|K(RAND)]$.

5) La función de usuario final envía a S-n el mensaje de Registro SIP con los identificadores *IMPU* e *IMPI* y el valor $U_{pr}[N_{pu}[K]|K(RAND)]$.

6) S-1 envía a S-5 los datos recibidos en el paso 5 y solicita la verificación, el registro del usuario y las claves *CK* e *IK*.

S-5:

- consulta el certificado del usuario para obtener la clave pública de usuario U_{pu};
- desencripta con U_{pu} el valor recibido *C*, supuestamente igual a $U_{pr}[N_{pu}[K]|K(RAND)]$ para recuperar el valor *D*|*E*, siendo *D* supuestamente igual a $N_{pu}[K]$ y *E* supuestamente igual a *K(RAND)*;
- desencripta con el valor de la clave privada de la red N_{pr} el valor *D* para obtener *K'*;
- desencripta con *K'*el valor *E* para obtener *RAND'*;
- compara *RAND'* con *RAND*. Si concuerdan, el usuario queda autenticado y *K' = K*. O sea, la función de usuario final y S-5 comparten ahora la clave *K*;
- genera las claves *CK* e *IK* utilizando la clave compartida *K*. Por ejemplo, las mismas funciones para generar *CK* e *IK* especificadas en [ATIS 33102] pueden emplearse con *K* como parámetro de entrada.

7) S-5 comunica a S-1 el resultado de la autentificación, o sea el registro del usuario y las claves *CK* e *IK*.

8) S-1 utiliza el registro para comprobar si el usuario autenticado está autorizado a registrar y recibir el servicio solicitado. De estarlo, S-n notifica a la función de usuario final la concesión del acceso, utilizando el mensaje SIP 200 OK.

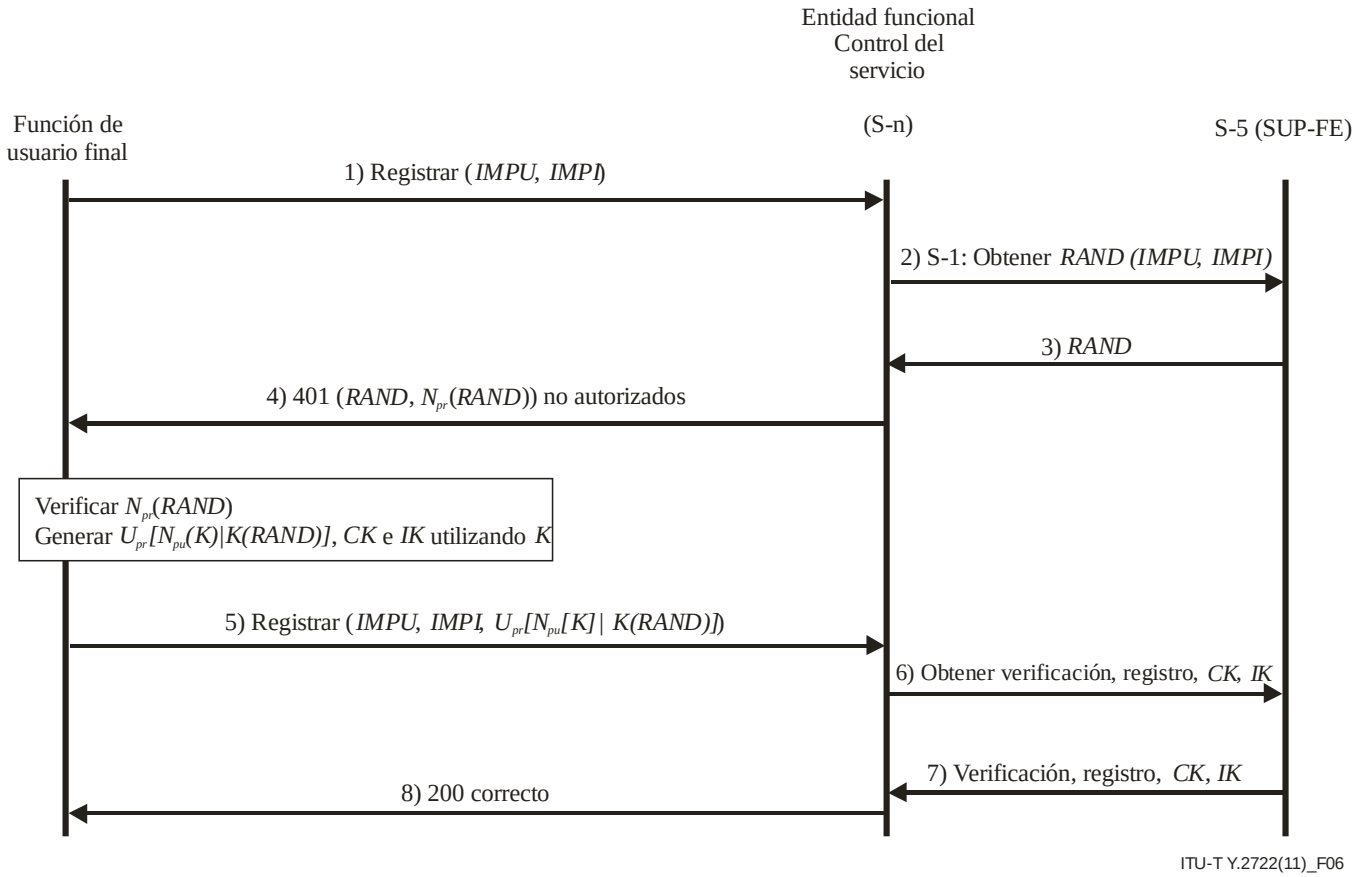


Figura 6 – Integración del mecanismo de autentificación IMS con el basado en PKI (opción 2)

6.2.6.5 Comparación entre las opciones 1 y 2

En el cuadro 1 se ofrece una comparación entre los mecanismos correspondientes a las opciones 1 y 2.

Cuadro 1 – Comparación entre las opciones 1 y 2 para el acuerdo de claves entre la función de usuario final y S-5 sobre *CK* e *IK*

	Opción 1 (con secreto compartido previamente)	**Opción 2 (sin secreto compartido previamente)**
Ventajas	Reutiliza por completo el mecanismo AKA para establecer el acuerdo sobre las claves *CK* e *IK*	No requiere la habilitación de un secreto compartido entre la función de usuario final y S-5
Inconvenientes	Requiere la habilitación de un secreto compartido entre la función de usuario final y S-5	Requiere modificaciones a las aplicaciones que se ejecutan en la función de usuario final (por ejemplo, en una tarjeta inteligente) y S-5 para conseguir el acuerdo sobre *CK* e *IK*

Para simplificar el acuerdo de claves sobre *CK* e *IK* cuando la función de usuario final y S-5 comparten un secreto, debe seleccionarse la opción 1. Cuando la función de usuario final y S-5 no tengan un secreto compartido, deberá seleccionarse la opción 2.

Las realizaciones de este mecanismo de integración deben soportar ambas opciones.

Requisitos de la entidad funcional S-5

Además de las capacidades especificadas en [ATIS 33102], S-5 debe ser capaz de:

- Guardar los certificados de la red y de los usuarios en un depósito de certificados y recuperarlos del mismo.
- Ejecutar la desencriptación basada en PKI descrita en el paso 6 (para ambas opciones).
- Ejecutar el protocolo Diameter modificado para transmitir la información descrita en el paso 6 (para ambas opciones) y la información necesaria para la negociación con la función de usuario final de la autentificación basada en PKI.
- Negociar con la función de usuario final un acuerdo sobre el método de autentificación basado en PKI.

6.2.6.6 Requisitos de la función de usuario final

La función de usuario final debe ser capaz de:

- Guardar de forma segura la clave privada del usuario U_{pr}.
- Guardar de forma segura el secreto compartido K_s con la red (sólo en la opción 1).
- Guardar un certificado de red UIT-T X.509 con la clave pública de la red N_{pu}.
- Generar aleatoriamente una clave de sesión única *K* y ejecutar la encriptación de clave simétrica con *K*.
- Generar las claves *CK* e *IK* utilizando el secreto compartido K_s especificado en [ATIS 33102] (sólo en la opción 1).
- Generar las claves *CK* e *IK* descritas en el paso 6 para la opción 2.
- Ejecutar la encriptación y desencriptación basadas en PKI descritas en los pasos 4 y 5 para ambas opciones.
- Ejecutar un cliente SIP con un protocolo SIP modificado para permitir la comunicación de la información descrita en los pasos 4 y 5.
- Negociar con S-2 un acuerdo sobre la utilización de la autentificación basada en PKI.

6.2.6.7 Requisitos de S-1

Los requisitos adicionales de S-1 son los siguientes:

- Debe ser capaz de construir mensajes SIP con la información descrita en el paso 4 (para ambas opciones).
- Debe ser capaz de recuperar de los mensajes SIP la información descrita en el paso 5 y volverla a empaquetar en los mensajes Diameter descritos en el paso 6 (para ambas opciones).
- Debe ser capaz de ejecutar la encriptación basada en PKI descrita en el paso 4 (para ambas opciones).
- Debe ser capaz de entender la notificación de S-5 relativa al empleo de la autentificación basada en PKI.

6.2.6.8 Requisitos de las interfaces SIP entre entidades participantes

La función de usuario final y S-1 se comunican mediante las entidades funcionales S-2 y S-3. Las entidades S-2 y S-3 no son indispensables para la autentificación descrita y no se representan en la figura 5 ni en la 6.

Hay interfaces SIP entre:

- La función de usuario final y S-2.
- S-2 y S-3.
- S-1 y S-3.

Estas interfaces deben poder negociar el empleo de la autentificación basada en PKI (incluida la opción específica de generación de claves) y transmitir la información descrita en los pasos 4 y 5 (para ambas opciones).

6.2.6.9 Requisitos de las interfaces Diameter entre entidades participantes

Hay interfaces Diameter entre:

- S-1 y S-5.
- S-3 y S-5.

Estas interfaces deben poder negociar el empleo de la autentificación basada en PKI (incluida la opción específica de generación de claves) y transmitir la información descrita en el paso 6 (para ambas opciones).

6.2.7 Integración de la autentificación basada en PKI con los mecanismos de aserción SAML

El SAML permite que una entidad (por ejemplo, un IdSP) autentique y que sea otra (la parte confiante, tal como el Proveedor de servicios de aplicaciones) la que utilice los resultados de la autentificación. En este supuesto, el IdSP puede aplicar los métodos de autentificación múltiple, mientras que el proveedor de servicios de aplicaciones (ASP) confía en las aserciones SAML del IdSP. Este supuesto es ventajoso tanto para los IdSP como para los ASP. Las ventajas para estos últimos son las siguientes:

- El ASP no tiene que poner en práctica muchos métodos de autentificación.
- El ASP podría dar soporte a una gran variedad de servicios de aplicaciones con diferentes requisitos de garantía de autentificación.

Las ventajas para el IdSP son las siguientes:

- Puede ofrecer servicios IdM, en particular la autentificación, a múltiples ASP.
- El IdSP (especialmente cuando es proveedor de red NGN) puede utilizar la infraestructura de autentificación que tiene desplegada para ofrecer servicios IdM a otros proveedores.

En esta cláusula se especifica un mecanismo de autentificación del cliente en el que se utilizan aserciones SAML y la autentificación basada en PKI. Este mecanismo, junto con el descrito en la cláusula 6.2.6 *Integración de la autentificación basada en PKI con el IMS*, permite que los operadores de las NGN utilicen su infraestructura basada en PKI.

Este mecanismo se basa en la vinculación Redirect de HTTP del SAML especificada en [UIT-T X.1141].

6.2.7.1 Entidades implicadas en la autentificación y el flujo de información

- Función de usuario final – Entidad capaz de ejecutar un cliente web y soportar la autentificación basada en PKI [UIT-T X.509].
- Servidor de aplicaciones (AS) – Entidad que presta un servicio web. Desempeña la función de parte confiante. Se comporta como un peticionario SAML con arreglo a la definición de [UIT-T X.1141].
- A-2 – Entidad funcional pasarela de aplicaciones (APL-GW-FE), habilitada para ejecutar la autentificación basada en PKI y actuar como un respondedor SAML con arreglo a la definición de [UIT-T X.1141].
- S-5 – Entidad funcional Perfil de usuario del servicio (SUP-FE).

El flujo de información del procedimiento de autentificación se representa en la figura 7. Los pasos básicos del intercambio de datos para la autentificación basada en PKI con aserción SAML se describen a continuación. En [ITU-T Y.2012] se describen las entidades funcionales de NGN (función de usuario final, AS, A-2 y S-5).

6.2.7.2 Convenios

En la descripción se utilizan los siguientes convenios:

– "|" indica concatenación de cadenas de caracteres

– *K()* indica una encriptación de clave simétrica

– K_s indica un secreto compartido entre A-2 y AS

– N_{pr} *[]* indica encriptación con clave de red privada N_{pr}

– N_{pu} *[]* indica encriptación con clave de red pública N_{pu} obtenida del certificado de red

– U_{pr} *[]* indica encriptación con la clave de usuario privada U_{pr}

– *RAND* indica una pregunta generada aleatoriamente.

6.2.7.3 Parámetros del mecanismo

En esta cláusula se especifican los parámetros específicos del mecanismo. La lista de parámetros es la siguiente:

`pki-auth-challenge` – Parámetro para transmitir el valor de *RAND*

`pki-auth-challenge-encrypted` – Parámetro para transmitir el valor de N_{pr}*[RAND]*

`pki-auth-user-signature` – Parámetro para transmitir el valor de $U_{pr}[N_{pu}[K]|K(RAND)]$

`pki-auth-keyinfo` – Parámetro para transmitir el valor de $K_s(K)$

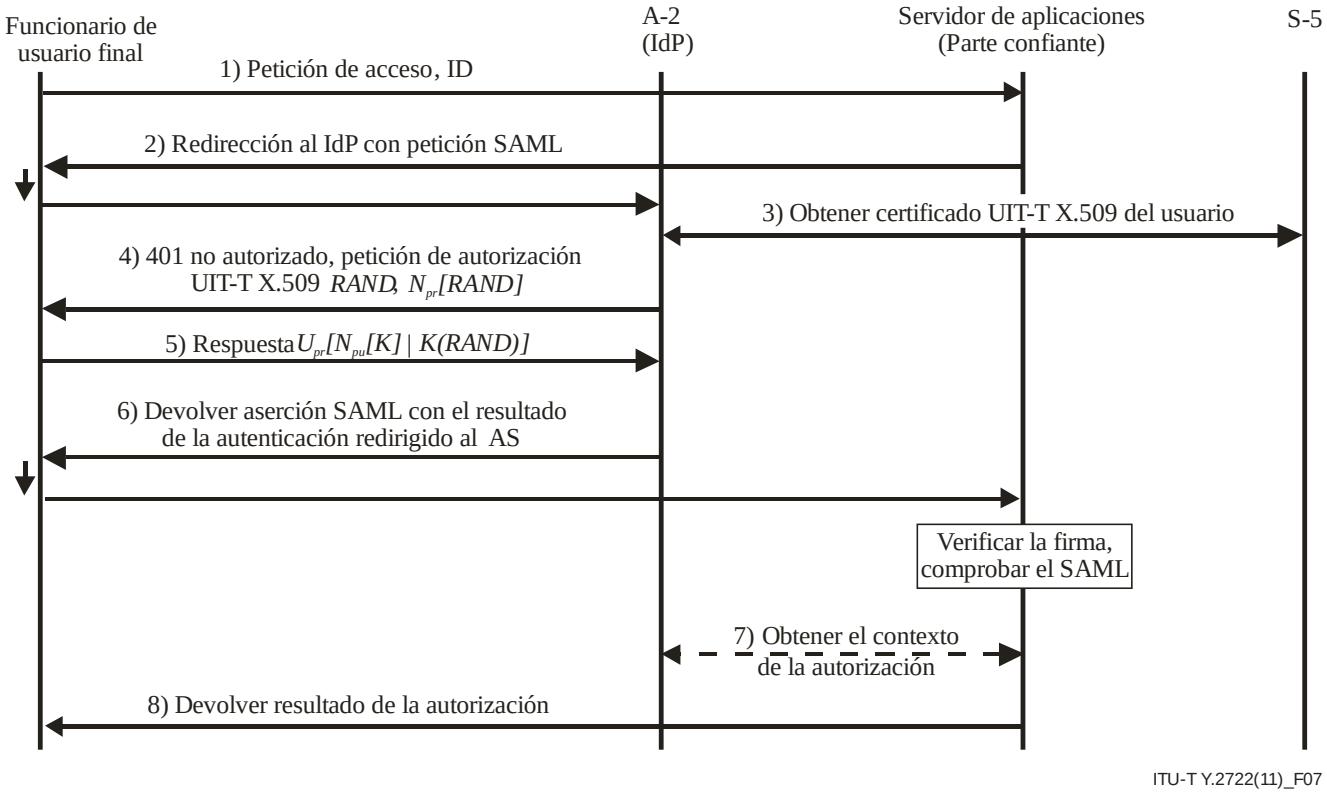


Figura 7 – Pasos básicos del intercambio de datos para la autentificación basada en PKI con aserción SAML

La autentificación mutua entre la función de usuario final y A-2 es semejante al procedimiento utilizado por el mecanismo de integración de la autentificación basada en PKI con IMS, descrito en la cláusula 6.2.6.

Los pasos básicos del procedimiento que confía en la autentificación basada en PKI y las aserciones SAML son los siguientes:

1) Un cliente web de la función de usuario final emite una petición de acceso HTTP dirigida al Servidor de aplicaciones (AS). Esta petición contiene un identificador de usuario y la URL de A-2.

2) El servidor de aplicaciones se comporta como un peticionario SAML y responde a la petición HTTP enviando una petición SAML. La petición SAML se codifica en el encabezamiento de ubicación de respuesta HTTP, con un valor del estado de HTTP igual a 302 ó 303. El agente de la función de usuario final entrega la petición SAML emitiendo una petición HTTP GET con destino a A-2 que se comporta como un respondedor SAML. Este procedimiento de redireccionamiento SAML, denominado *vinculación Redirect de HTTP*, viene especificado en [UIT-T X.1141]. Para conseguir la autentificación y la integridad del mensaje codificado en la URL, debe estar firmado como se especifica en la cláusula 10.2.4.5.2 *Consideraciones de seguridad* de [UIT-T X.1141]. Para la firma debe utilizarse el secreto compartido K_S.

3) Tras la validación de la firma, A-2 obtiene de S-5 el certificado de usuario final y comprueba su validez. El certificado contiene la clave pública de la función de usuario final.

4) A-2 responde a la función de usuario final con un mensaje de respuesta HTTP que indica que se requiere una autentificación con el certificado X.509 [UIT-T X.509]. Esto se lleva a cabo asignando al encabezamiento de respuesta `WWW-Authenticate` [b-IETF RFC 2616] el valor "`pki-auth`". El cuerpo del mensaje contiene los parámetros `pki-auth-challenge` y

`pki-auth-challenge-encrypted` con los valores de la pregunta *RAND* generada aleatoriamente y su encriptación $N_{pr}[RAND]$, respectivamente. Al encabezamiento `Content-Type` debe asignársele el valor `application/x-www-form-urlencoded`.

5) La función de usuario final:

- recupera los valores de *A*, supuestamente igual a *RAND*, y *B*, supuestamente igual a $N_{pr}[RAND]$;
- recupera la clave pública de red N_{pu};
- desencripta *B* con N_{pu} y compara el resultado con *A*. Si los valores coinciden, la red queda autenticada, de lo contrario el procedimiento de autentificación se suspende;
- se genera una clave secreta *K*;
- se genera el valor $U_{pr}[N_{pu}[K]|K(RAND)]$, se asigna al parámetro `pki-auth-user-signature` dicho valor y se envía en el cuerpo de un mensaje HTTP POST a A-2. Al encabezamiento `Content-Type` del mensaje se le debe asignar el valor `application/x-www-form-urlencoded`.

Tras este paso, A-2 comprueba la validez de la respuesta. Para ello, A-2 ejecuta las siguientes operaciones:

- consulta el certificado del usuario para obtener la clave pública de éste U_{pu};
- desencripta con U_{pu} el valor recibido *C*, supuestamente igual a $U_{pr}[N_{pu}[K]|K(RAND)]$ para recuperar el valor *D*|*E*, siendo *D* supuestamente igual a $N_{pu}[K]$ y *E* supuestamente igual a *K(RAND)*;
- desencripta el valor *D* con la clave privada de la red N_{pr} para obtener *K'*;
- desencripta el valor *E* con *K'* para obtener *RAND'*;
- compara *RAND'* con *RAND*. Si concuerdan, el usuario queda autenticado y $K' = K$. O sea, la función de usuario final y A-2 comparten ahora la clave K.

6) Si todos los pasos anteriores se completan con éxito, A-2 ejecuta las siguientes operaciones:

- genera una aserción SAML asignando al atributo método del elemento <SubjectConfirmation> el valor sender-vouches (el remitente lo garantiza);
- calcula el valor $K_s(K)$;
- introduce la aserción en una respuesta SAML. A continuación, entrega la respuesta SAML y el valor calculado $K_s(K)$ por HTTP igual que la petición SAML descrita en el paso 2 (o sea, como parte de la cadena de caracteres de una consulta). El valor $K_s(K)$ se introduce en el parámetro `pki-auth-keyinfo`;
- para lograr la autentificación en origen y la integridad del mensaje codificado en la URL, A-2 lo firma con arreglo a lo especificado en la cláusula 10.2.4.5.2 *Consideraciones de seguridad* de [UIT-T X.1141]. Para la firma debe utilizarse el secreto compartido K_s.

  Tras validar la URL firmada, el AS sabe que la aserción SAML es de A-2. El AS verifica la aserción por sí mismo (por ejemplo, para verificar que se cumplen las *condiciones*). Tras ello, el AS recupera el valor $K_s(K)$ y lo desencripta utilizando el secreto compartido K_s para obtener *K*. En este momento el AS ha autenticado la función de usuario final y ambas entidades comparten la clave *K*, que puede utilizarse para establecer comunicaciones seguras entre ellas.

7) Si lo requiere la política de adopción de las decisiones de autorización, el AS obtiene información del contexto de autentificación. En este caso, A-2 responde con la información especificada en la clase de contexto de autentificación clave pública – UIT-T X.509 de [UIT-T X.1141].

8) El AS envía a la función de usuario final el resultado de la decisión de autorización.

6.2.7.4 Requisitos adicionales de las entidades que participan en la autentificación

Para soportar el mecanismo descrito, las entidades participantes deben cumplir los siguientes requisitos:

6.2.7.4.1 Requisitos de la función de usuario final

La función de usuario final debe poder:

- Ejecutar el cliente HTTP.
- Guardar de forma segura su clave privada U_{pr} (por ejemplo, en una tarjeta inteligente).
- Obtener la clave pública de la red N_{pu}.
- Realizar la encriptación y la desencriptación.
- Generar una clave K.

6.2.7.4.2 Requisitos del servidor de aplicaciones (AS)

- El AS debe soportar SAML [UIT-T X.1141].
- El AS debe tener un secreto compartido (K_s) con A-2.

6.2.7.4.3 Requisitos de la entidad funcional A-2

La entidad funcional A-2 debe poder:

- Soportar el protocolo HTTP.
- Guardar de forma segura su clave privada N_{pr}.
- Obtener la clave pública del usuario U_{pu}.
- Realizar la encriptación y la desencriptación.
- Generar una pregunta aleatoria RAND.
- Soportar SAML [UIT-T X.1141].
- Tener un secreto compartido (K_s) con el AS.

6.2.7.4.4 Requisitos de la entidad funcional S-5

La entidad funcional S-5 debe guardar de forma segura los certificados UIT-T X.509 de los usuarios en un depósito o recuperarlos del mismo (o ambas cosas).

6.2.7.5 Requisitos adicionales de las interfaces entre las entidades participantes

Los requisitos de las interfaces son los siguientes:

- La interfaz entre la función de usuario final y el servidor de aplicaciones debe soportar el protocolo HTTP [b-IETF RFC 2616].
- Las interfaces entre la función de usuario final y las entidades funcionales A-2 deben soportar el protocolo HTTP [b-IETF RFC 2616].
- La interfaz entre A-2 y el servidor de aplicaciones debe soportar SAML [UIT-T X.1141].
- La interfaz entre las entidades funcionales A-2 y S-5 debe soportar un mecanismo de consulta-respuesta gracias al cual A-2 pueda obtener de S-5 los certificados X.509 de los usuarios.

6.2.8 Integración de la autentificación basada en *Open-ID* con autentificación AKA

La integración de la autentificación AKA con la autentificación basada en *Open-ID* permite combinar las capacidades IdM centradas en la red y las centradas en el usuario. Este mecanismo de integración:

- Permite a los proveedores de red prestar servicios de identidad a los usuarios que acceden a aplicaciones web.
- Puede emplearse para ofrecer a los usuarios un inicio de sesión único (SSO) para todo el entorno de servicios web y de red IMS con una aplicación ISIM existente, así como con otras aplicaciones SIM que confían en AKA.
- Permite a los usuarios controlar sus identificadores públicos en la web, conforme a lo especificado en [b-OpenID v.2] y, a su vez, potenciar los servicios NGN.
- Mejora la seguridad del usuario al utilizar un proveedor de red de confianza para el usuario que controle el acceso a las aplicaciones web.

En el Informe Técnico [b-3GPP TR 33.924] se describen varias soluciones de integración de *OpenID* y AKA que utilizan la Función de servidor de inicialización (BSF).

En esta cláusula se describe otro mecanismo adicional que permite la integración de OpenID y AKA. Para realizarlo, la especificación OpenID requiere una diversidad de mecanismos de autentificación.

OpenID puede funcionar con otras tecnologías tales como OAuth, como se indica en el apéndice III.

6.2.8.1 Entidades implicadas en la autentificación y el flujo de información

- Función de usuario final – Entidad capaz de ejecutar un cliente web y comunicarse con la aplicación ISIM.
- Servidor de aplicaciones (AS) – Entidad que presta el servicio web. Desempeña la función de parte confiante.
- A-2 – Entidad funcional pasarela de aplicaciones (APL-GW-FE) que está habilitada para hacer de Proveedor de identidad *OpenID* [b-OpenID v.2]. (A-2 puede compartir opcionalmente un secreto a corto plazo con el servidor de aplicaciones especificado en [b-OpenID v.2]).
- S-5 – Entidad funcional perfil de usuario del servicio (SUP-FE).

El flujo de información del procedimiento de autentificación se representa en la figura 8. No se muestra el procedimiento de establecer la clave de firma a corto plazo entre el servidor de aplicaciones y A-2. En esta figura se representan los pasos básicos del procedimiento para ambas opciones *OpenID*:

a) A-2 y el servidor de aplicaciones comparten un secreto.

b) A-2 y el servidor de aplicaciones no comparten ningún secreto.

Los pasos 1 a 6 son comunes a ambas opciones. El paso 7a sólo corresponde a la opción a).

Los pasos 7b, 8b y 9b sólo corresponden a la opción b).

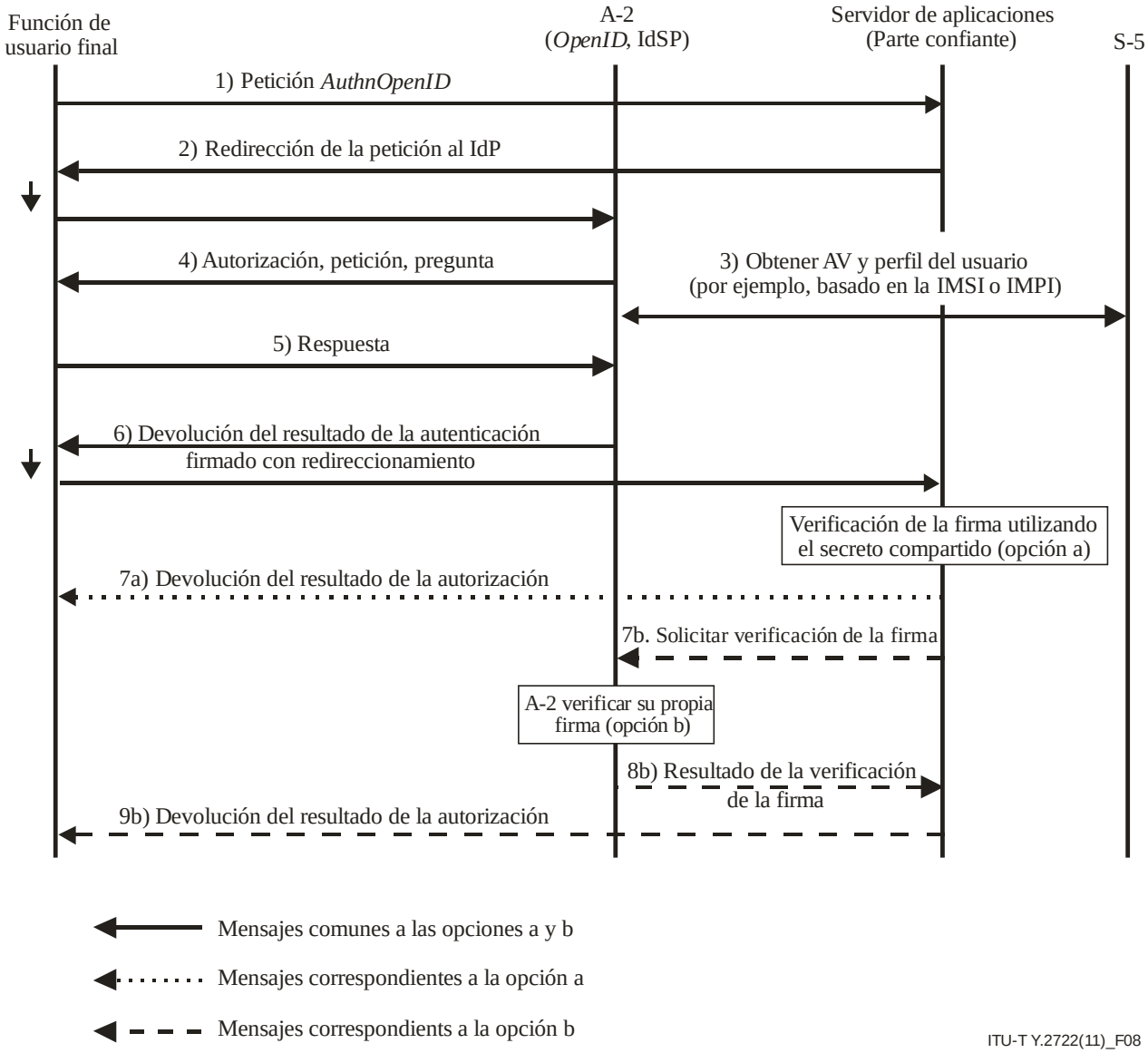


Figura 8 – Integración del mecanismo de autentificación AKA con OpenID

Los pasos básicos son los siguientes:

1) Un cliente web de la función de usuario final emite una petición de autentificación *AuthnOpenID* dirigida al servidor de aplicaciones. La solicitud contiene un identificador *OpenID*.

2) Gracias al identificador OpenID, el servidor de aplicaciones descubre la URL de A-2, que sirve de proveedor de identidad OpenID, y redirecciona la petición de autentificación del usuario a dicha URL.

   Tras este paso, A-2 establece una correlación entre el identificador del usuario y la identidad adecuada (por ejemplo, IMSI o IMPI).

3) A-2 obtiene de S-5 el vector de autentificación del AKA, AV, y el perfil del usuario basado en la IMPI.

4) A-2 envía a la función de usuario final la petición de autentificación utilizando el método del AKA de compendio HTTP [b-IETF RFC 4169] o [b-IETF RFC 3310]. Esta petición consta de una pregunta y una cantidad gracias a las cuales la función de usuario final puede autenticar la red.

Tras este paso, la función de usuario final autentica la red con arreglo a lo especificado en [b-IETF RFC 4169] o [b-IETF RFC 3310].

5) La función de usuario final envía a A-2 la respuesta a la pregunta con arreglo a lo especificado en [b-IETF RFC 4169] o [b-IETF RFC 3310].

Tras este paso, A-2 autentica la función de usuario final con arreglo a lo especificado en [b-IETF RFC 4169] o [b-IETF RFC 3310].

6) A-2 envía a la función de usuario final un mensaje firmado en el que se indica que el identificador OpenID pertenece al usuario. Este mensaje está firmado utilizando el secreto compartido con el servidor de aplicaciones para la opción a). Para la opción b), el mensaje se firma con la clave secreta de A-2. El mensaje contiene una petición de redireccionamiento del cliente web de la función de usuario final al servidor de aplicaciones. Los detalles del procedimiento de firma y redireccionamiento se describen en [b-OpenID v.2], donde también se especifican las medidas para evitar ataques que consisten en reutilizar la aserción de autentificación firmada.

Pasos específicos de la opción a)

7a) Tras verificar la firma de la respuesta recibida en el paso 6, el servidor de aplicaciones notifica a la función de usuario final el resultado de la autentificación. El servidor de aplicaciones utiliza el secreto compartido con A-2 para esta verificación.

De haber un fallo en alguno de los pasos: 1 a 6, o 7a, el procedimiento de autentificación se detiene.

Pasos específicos de la opción b)

7b) El servidor de aplicaciones envía a A-2 una copia del mensaje recibido en el paso 6 con una solicitud de verificación de la firma.

8b) Tras verificar su propia firma, A-2 informa al servidor de aplicaciones del resultado de la verificación.

9b) El servidor de aplicaciones informa del resultado de la autentificación a la función de usuario final.

Si se produce un fallo en alguno de los pasos: 1 a 6, 7b, 8b o 9b, el procedimiento de autentificación se detiene.

6.2.8.2 Requisitos adicionales de las entidades que participan en la autentificación

Para soportar el mecanismo descrito, las entidades participantes deben satisfacer los siguientes requisitos:

6.2.8.2.1 Requisitos de la función de usuario final

La función de usuario final debe ser capaz de:

- Autenticar utilizando el método AKA de compendio HTTP.
- Comunicarse con la aplicación SIM adecuada.

6.2.8.2.2 Requisitos del servidor de aplicaciones.

El servidor de aplicaciones debe soportar la versión 2.0 de la especificación OpenID [b-OpenID v.2].

6.2.8.2.3 Requisitos de la entidad funcional A-2

La entidad funcional A-2 debe poder:

- Ejecutar la autentificación AKA de compendio HTTP.

- Correlacionar el identificador OpenID con su identificador adecuado (tal como IMSI o IMPI).
- Servir de proveedor de identidad OpenID.

6.2.8.2.4 Requisitos de la entidad funcional S-5

No hay más requisitos para la entidad funcional S-5 que los especificados en [UIT-T Y.2012].

6.2.8.3 Requisitos adicionales para las interfaces existentes entre las entidades participantes

Los requisitos de estas entidades son los siguientes:

- La interfaz entre la función de usuario final y el servidor de aplicaciones debe soportar la autentificación OpenID con arreglo a lo especificado en su versión 2.0 [b-OpenID v.2].
- Las interfaces entre la función de usuario final y las entidades funcionales A-2 deben soportar el protocolo AKA de compendio HTTP [b-IETF RFC 4169] o [b-IETF RFC 3310].
- La interfaz entre las entidades funcionales A-2 y S-5 no tiene requisitos específicos de ningún mecanismo.

6.2.8.4 Mecanismo de interfuncionamiento de OpenID y AKA para el caso del terminal de usuario dividido

El mecanismo que se describe en esta cláusula también sirve para el caso de terminal dividido, descrito en [b-3GPP TR 33.924]. El caso del terminal de usuario dividido corresponde a la situación en la que el agente autenticador (una entidad con acceso a la tarjeta UICC) y el agente explorador no están situados en el mismo terminal de usuario.

Considerando que en la solución de AKA directo especificada en esta cláusula el IdSP corresponde a una NAF/BSF caída, los escenarios descritos en [b-3GPP TR 33.924] están completamente soportados por esta solución. Este mecanismo utiliza la autentificación AKA directa en vez de la basada en la autentificación GBA.

6.2.9 La arquitectura de inicialización genérica

La arquitectura de inicialización genérica (GBA, *generic bootstrap architecture*) especifica un marco para inicializar la autentificación y establecer un acuerdo de claves que favorezca el mecanismo de acuerdo de claves (AKA) y autentificación 3GPP. La GBA facilita la autentificación de usuarios finales ante la función de aplicación de red y puede utilizarse en la gestión de identidades de la NGN para habilitar:

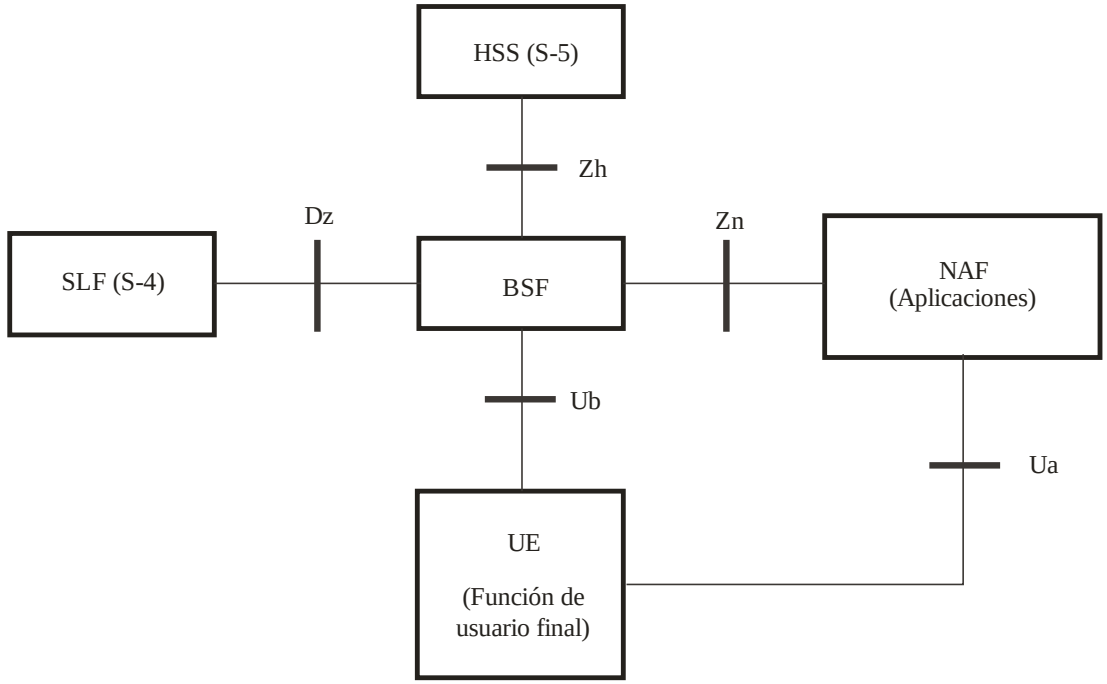
- El acuerdo de autentificación y clave.
- La protección de la privacidad.
- El inicio de sesión único.

La GBA es un sistema de autentificación que consta de tres partes:

- El usuario final que intenta obtener los servicios de red mediante el equipo de usuario (UE).
- El servidor de aplicaciones (denominado Función de aplicación de red o NAF).
- Una entidad de confianza (denominada Función de servidor de inicialización o BSF) implicada en la autentificación e intercambio de claves entre las otras dos entidades.

La GBA permite la autentificación del usuario final que utiliza el UE ante un servidor de aplicaciones (NAF) sin revelarle los secretos y credenciales a largo plazo del usuario final, gracias a la utilización de la BSF de una entidad de confianza.

En la figura 9 se describe GBA [b ETSI TS 133 220] y se muestra la correspondencia de las entidades definidas en 3GPP con las entidades funcionales especificadas en [ITU T Y.2012].



NOTA – Las etiquetas entre paréntesis representan entidades especificadas en [UIT-T Y.2012].

ITU-T Y.2722(11)_F09

Figura 9 – Modelo inicialización de una red sencilla

Pasos básicos del procedimiento GBA:

1) NAF solicita la autentificación y negocia la utilización de GBA en el punto de referencia Ua.

2) El cliente BSF que se ejecuta en la UE inicia el procedimiento de inicialización en el punto de referencia Ub. La BSF busca la información de autentificación y los valores de seguridad del usuario GBA en el HSS a través de Zh. La UE y la BSF se autentican mutuamente utilizando http Compendio AKA. Este procedimiento consigue que el UE reciba el identificador de la transacción de inicialización (B-TID) de la BSF y se establezca una clave compartida (Ks) entre el UE y la BSF.

3) El UE obtiene Ks_NAF de Ks y envía B-TID (junto con los datos específicos de la aplicación) a la NAF.

4) La NAF envía B-TID a la BSF a través del punto de referencia Zn.

5) La BSF determina a partir de B-TID la Ks que debe utilizarse, calcula Ks_NAF a partir de ésta y envía Ks_NAF a la NAF.

6) Por último, el UE y la NAF pueden autenticarse recíprocamente mediante la clave compartida Ks_NAF. El procedimiento de autentificación exacto depende del protocolo entre el UE y la NAF. Por ejemplo, la GBA especifica que las aplicaciones basadas en HTTP pueden utilizar o bien la autentificación compendio HTTP [b-IETF RFC 2617] o los conjuntos de cifrado de claves previamente compartidas [b-IETF RFC 4279].

NOTA – La BSF consulta la SLF a través del punto de referencia Dz para obtener el nombre del HSS que contiene los datos específicos del abonado. Cuando la BSF está configurada para que utilice un HSS predefinido, se puede prescindir de la SLF.

Correspondencia de las entidades GBA con las entidades NGN especificadas en [UIT-T Y.2012], *Requisitos funcionales y arquitectura de la red de próxima generación*.

- La NAF se corresponde con la entidad "Aplicaciones" de la figura 3, *Arquitectura funcional generalizada para las NGN*, de [UIT-T Y.2012].
- El HSS se corresponde con la FE S-5, perfil de usuario del servicio.
- La SLF se corresponde con la FE S-4, localizador de la suscripción.
- El UE se corresponde con la función de usuario final.

6.2.10 La autentificación basada en la IMSI

Dependiendo del nivel de garantía requerido, puede utilizarse la identidad de abonado móvil internacional (IMSI) en la red del protocolo de aplicaciones inalámbricas (WAP) para la autentificación, ofreciendo compatibilidad ascendente. Dado que la IMSI es una cadena de caracteres alfabéticos única, puede utilizarse como identidad de la entidad para un servicio particular.

Este enfoque:

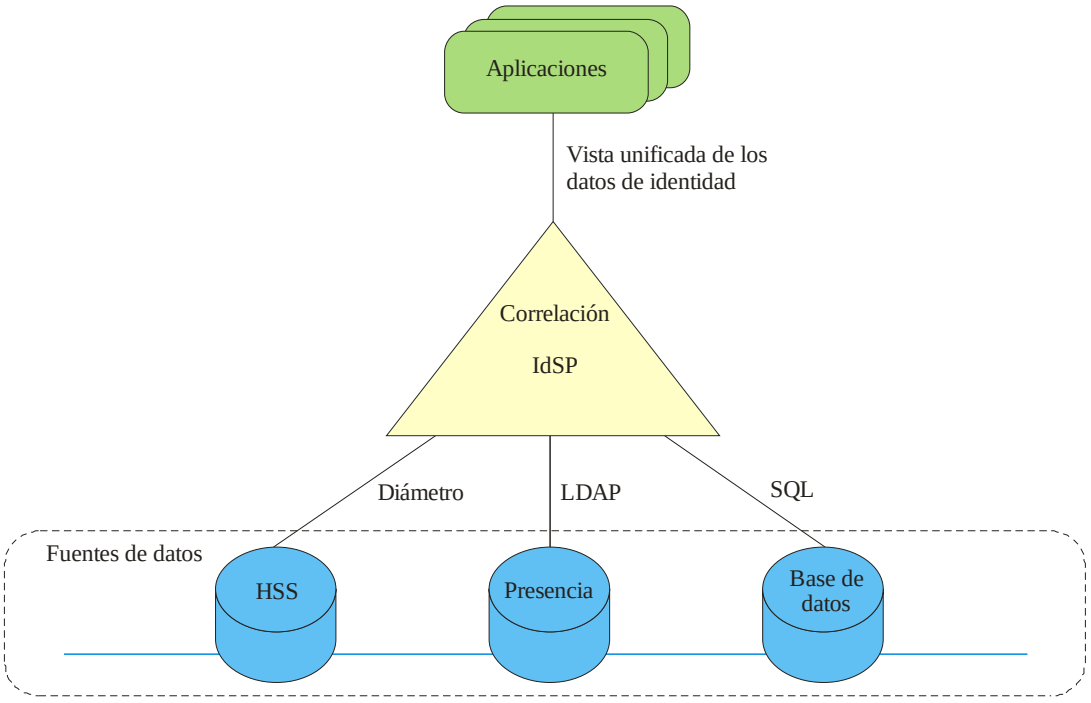
- utiliza el código IMSI como identidad de la entidad en las aplicaciones inalámbricas;
- ofrece un canal de servicio fiable a condición de que el punto de extremo tenga un IMSI legítimo cuando el punto de extremo solicite la autentificación;
- parte del supuesto de que todos los sistemas confían en el resultado de la autentificación de la pasarela WAP y prestan servicios a dicha entidad;
- puede utilizarse para proporcionar la función de inicio de sesión único en virtud de la unicidad de la IMSI para el mismo punto de extremo entre GPRS/CMDA 1 x y la aplicación inalámbrica (por ejemplo, buzón inalámbrico, etc.);
- proporciona seguridad al código IMSI.

6.3 Correlación y vinculación

La [UIT-T Y.2720], *Marco general para la gestión de identidades en las redes de la próxima generación* estipula que *la información de identidad (por ejemplo, identificadores, credenciales y atributos) puede correlacionarse para establecer un vínculo que garantice la identidad de una entidad.*

Uno de los objetivos de una solución de correlación es reunir diversos tipos de información de identidad de distintas fuentes y presentarlas a las aplicaciones con un formato unificado que ella pueda entender.

El concepto de una solución de este tipo se ilustra en la figura 10. En esta figura se representan tres ejemplos de fuentes de información de identidad: el HSS, el servidor de presencia y la base de datos con los datos del usuario específicos de la aplicación. Para la adopción de decisiones en materia de autentificación y autorización, la aplicación quizá necesite los tres tipos de información. En el ejemplo representado, para obtener datos de las respectivas fuentes, el mecanismo de correlación emplea los protocolos Diameter, LDAP y SQL. A continuación, estos datos se presentan a la aplicación en un formato inteligible para ésta. Así pues, el mecanismo de correlación descarga a las aplicaciones del peso que supone tener que soportar varios protocolos para comunicarse con fuentes de información de identidad diferentes.



NOTA – El mecanismo de correlación ofrece aplicaciones con una vista unificada de todos los datos de ident idad.

ITU-T Y.2722(11)_F10

Figura 10 – Correlación de la información de identidad

6.4 Descubrimiento

La [UIT-T Y.2721] *Requisitos de gestión de identidad en las NGN y ejemplos de utilización* especifica que *se requiere del IdSP de la NGN que soporte funciones y capacidades que le permitan descubrir las fuentes de identidad de la información* de un dominio NGN/IdSP y de varios dominios NGN/IdSP diferentes.

En esta cláusula se brindan ejemplos de mecanismos corrientes que soportan estos requisitos y las referencias a las especificaciones pertinentes.

6.4.1 Descubrimiento dentro de la red

En [UIT-T Y.2012], *Requisitos funcionales y arquitectura de la red de próxima generación,* se define una entidad especial, la entidad funcional localizador de la suscripción (SL-FE), que proporciona una dirección de la entidad funcional perfil del usuario del servicio (SUP-FE) que guarda la información de identidad de un abonado concreto. SL-FE permite la detección de SUP-FE, responsable de guardar los perfiles de usuario, los datos posicionales de los abonados, y los datos de su situación presencial. Si consultan SUP-FE, las entidades de la red pueden obtener esta información de identidad. De acuerdo con lo especificado en [UIT-T Y.2012], las siguientes entidades de red pueden consultar en la SL-FE la dirección de la oportuna SUP-FE:

- la entidad funcional soporte de la aplicación (AS-FE);
- la entidad funcional control de sesión de la llamada interrogante (I-CSC-FE);
- la entidad funcional control de sesión de la llamada servidora (S-CSC-FE).

En [3GPP TS 23.228] se especifica un mecanismo que permite que estas entidades encuentren en la red de un operador la dirección de SUP-FE donde se guarda la información de identidad de un determinado usuario. Obsérvese que la correspondencia entre las entidades de [UIT-T Y.2012] y las de [3GPP TS 23.228] es la siguiente:

- AS-FE se corresponde con AS.
- I-CSC-FE se corresponde con I-CSCF.

- S-CSC-FE se corresponde con S-CSCF.
- SL-FE se corresponde con SLF.

6.4.2 Detección entre redes

Entre los ejemplos de mecanismos de detección del IdSP entre redes se encuentran los especificados en SAML [UIT-T X.1141] e ID-WSF [B-LA WSF]. Estos mecanismos dependen de los acuerdos establecidos entre las entidades interesadas (por ejmplo, IdSP y la parte confiante) o entre los miembros de una federación.

Otro ejemplo es OpenID [b-OpenID v.2], que especifica un mecanismo de detección que permite a la parte confiante localizar el IdSP del usuario a partir del identificador OpenID suministrado por el usuario.

6.5 Comunicaciones e intercambio de información del IdM

En esta cláusula se recomiendan protocolos y mecanismos de comunicación e intercambio de la información de identidad.

6.5.1 Seguridad de las comunicaciones y el intercambio de información del IdM

En esta cláusula se recomiendan mecanismos de protección de la integridad y confidencialidad de las comunicaciones IdM.

6.5.1.1 Soluciones basadas en SAML 2.0 [UIT-T X.1141]

Para proteger la confidencialidad y la integridad, en SAML 2.0 se recomienda utilizar un protocolo de canal seguro o de red segura tal como TLS o IPSec y configurarlo para proteger los paquetes que se transmiten por la conexión de red.

Para proteger la integridad a nivel de mensaje, además de disponer del canal de comunicaciones seguro, puede utilizarse la firma XML. Cuando se utiliza la firma XML es necesario aplicar lo especificado en la cláusula 8.4 "Sintaxis y procesamiento de firmas XML y SAML" de [UIT-T X.1141].

Para proteger la confidencialidad a nivel de mensaje, además de disponer del canal de comunicaciones seguro, puede utilizarse la encriptación XML. Cuando se utiliza la encriptación XML es necesario aplicar lo especificado en la cláusula 8.4 de [UIT-T X.1141].

6.5.1.2 Marco de servicios web de identidad (denominado ID-WSF)

Para utilizar ID-WSF es necesario que la confidencialidad e integridad de las comunicaciones y mensajes entre el remitente y el destinatario estén protegidas. Al igual que con SAML 2.0, se recomienda utilizar un canal seguro o protocolo de red seguro tal como TLS o IPsec configurado para proteger los paquetes que se transmiten por la conexión de red [b-LA ID-WSF security].

1) Protección del canal de la capa de transporte

Cuando se utiliza SSL o TLS como protocolo de red seguro para ID-WSF, es necesario utilizar SSL 3.0, TLS 1.0 o versiones posteriores. Es necesario que una entidad que termine una conexión SSL (3.0) o TLS (1.0) ofrezca o acepte conjuntos de cifrado adecuados durante la toma de contacto. Entre los conjuntos de cifrado TLS 1.0 recomendados (o sus equivalentes SSL 3.0) se encuentran los siguientes (la relación no es exhaustiva):

- TLS_RSA_WITH_RC4_128_SHA
- TLS_RSA_WITH_3DES_EDE_CBC_SHA
- TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
- TLS_RSA_WITH_AES_CBC_SHA
- TLS_DHE_DSS_WITH_AES_CBC_SHA

Para la firma y verificación de los mensajes de protocolo, se recomienda que las entidades comunicantes utilicen certificados y claves privadas que sean distintos de los aplicados en la protección del canal SSL o TLS.

También pueden utilizarse otros protocolos de seguridad tales como IPSec o Kerberos, siempre que implementen medidas de seguridad equivalentes.

2) Protección de la confidencialidad de los mensajes

En presencia de intermediarios, las entidades comunicantes necesitan garantizar que la información confidencial no se revela a entidades no autorizadas. En este caso, es necesario que, para encriptar el contenido <S:Body> del sobre SOAP, estas entidades utilicen los mecanismos de confidencialidad especificados por OASIS en Seguridad de los mensajes SOAP de Seguridad de los servicios web (WSS) [b-OASIS WSS SOAP].

3) Reglas de integridad de los mensajes

Las reglas de integridad de los mensajes de esta cláusula sólo son aplicables cuando se utiliza la Seguridad de mensajes SOAP de Seguridad de los servicios web (WSS) por OASIS [b-OASIS WSS SOAP] para mensajes de protocolo ID-WSF dirigidos a SOAP con arreglo a la vinculación SOAP de libertad versión 2.0 [b-LA SOAP binding].

En este caso, es necesario que el remitente cree una única <ds:Signature> dentro del encabezamiento <wsse Security>, siendo necesaria esta firma para referenciar todos los componentes de mensaje que haya que firmar.

En particular, esta firma es necesaria para referenciar el elemento Cuerpo SOAP (el propio elemento), el testigo de seguridad asociado a la firma y todos los encabezamientos del mensaje que se hayan definido en la versión 2.0 de la vinculación SOAP de Libertad [b-LA SOAP binding], incluidos los bloques de encabezamiento obligatorios y los opcionales.

Un ejemplo de testigo de seguridad es un elemento <saml2:Assertion> transportado en el encabezamiento <wsse:Security>. El encabezamiento wsu:Timestamp del bloque de encabezamiento wsse:Security, los bloques de encabezamiento wsa:MessageID, wsa:RelatesTo, sb:Framework, sb:Sender y sb:InvocationIdentity son ejemplos de elementos de encabezamiento que se referenciarían en una firma.

Obsérvese que deben construirse con cuidado los elementos contenidos en los parámetros de referencia de las referencias de los puntos extremos ya que éstas se promoverán a bloques de encabezamiento SOAP. Deben tomarse las medidas adecuadas para evitar atributos de identificadores duplicados o en conflicto, por ejemplo, utilizando técnicas de generación de identificadores únicos.

Si el mensaje va firmado, en necesario que el remitente introduzca la firma XML resultante en un elemento <ds:Signature> como hijo del encabezamiento <wsse:Security>.

Es necesario que el elemento <ds:Signature> haga referencia a la clave de confirmación del sujeto con un elemento <ds:KeyInfo>. A su vez, es necesario que el elemento <ds:KeyInfo> contenga un elemento <wsse:SecurityTokenReference> para poder localizar la clave de confirmación del sujeto en el encabezamiento <wsse:Security>. Se recomienda la inclusión de la referencia para cumplir la directriz especificada en la cláusula 3.4.2 de la *Seguridad de los servicios web: Perfil del testigo SAML 1.1* de OASIS [b-OASIS SAML token].

i) Reglas de procesamiento del remitente

- La construcción y decoración del elemento de encabezamiento <wsse:Security> son necesarias para cumplir las normas especificadas en la *Seguridad de los servicios web: Perfil del testigo SAML 1.1* de OASIS [b-OASIS SAML token].
- El elemento de encabezamiento <wsse:Security> es necesario para tener un atributo mustUnderstand con el valor lógico verdadero.

- Es necesario que el remitente coloque el testigo de seguridad de autentificación del mensaje como hijo directo del elemento <wsse:Security>.
- Es necesario que el remitente se atenga a las normas de integridad de los mensajes dirigidas a remitentes y destinatarios cuando se utilizan mecanismos de autentificación de los mensajes.

Las siguientes consideraciones no son aplicables a los testigos de Portador:

- Para valores de despliegue que requieran autentificación de mensajes independiente, es necesario que se satisfaga la obligación firmando el cuerpo del mensaje y partes del encabezamiento y colocando la <ds:Signature> como hijo directo del encabezamiento <wsse:Security>.
- Para valores de despliegue que no requieran autentificación de mensajes independiente, puede satisfacerse la obligación de confirmación del sujeto correlacionando el certificado y la clave utilizada para la autentificación de la entidad par con el certificado y la clave descritos en el testigo de autentificación del mensaje. Para satisfacer esto, es necesario que la autoridad que expide la aserción la construya de forma que pueda verificarse sin ambigüedad que la clave de confirmación sea el mismo certificado y clave que se han utilizado para establecer la autentificación de la entidad par. Esto es necesario para mitigar la amenaza de un ataque de sustitución de certificado. Se recomienda vincular el certificado o cadena de éstos a la clave de confirmación del sujeto.

ii) Reglas de procesamiento del destinatario

- Es necesario que el destinatario localice el elemento <wsse:Security> del que es objetivo. Éste DEBE cumplir las normas especificadas en *Seguridad de mensajes SOAP de Seguridad de los servicios web (WSS)* por OASIS [b-OASIS WSS SOAP] y los perfiles de testigos WSS aplicables (por ejemplo, *Seguridad de los servicios web: Perfil del testigo SAML 1.1* de OASIS [b-OASIS SAML token] para los testigos SAML).
- Es necesario que el elemento de encabezamiento <wsse:Security> tenga un atributo mustUnderstand con el valor lógico verdadero y que el destinatario pueda procesar este bloque de encabezamiento con arreglo a *Seguridad de mensajes SOAP de Seguridad de los servicios web (WSS)* por OASIS [b-OASIS WSS SOAP] y los perfiles de testigos WSS adecuados (por ejemplo, *Seguridad de los servicios web: Perfil del testigo SAML 1.1* de OASIS [b-OASIS SAML token] para los testigos SAML).
- Es necesario que el destinatario localice el testigo de seguridad y que determine si confía en la autoridad que ha expedido el testigo.
- Es necesario que el destinatario valide la firma del expedidor que lleva el testigo. Esta validación es necesaria para ajustarse a las reglas de validación básicas descritas en la *Sintaxis y procesamiento de las firmas XML* (segunda edición) del World Wide Web Consortium (W3C) [b-W3C XML signature]. Se recomienda que el destinatario valide la semántica de confianza de la clave firmante adecuada al riesgo de autentificación incorrecta.
- Si el mensaje ha sido firmado, el destinatario necesitará localizar el elemento <ds:Signature> transportado en el encabezamiento <wsse:Security>.
- Salvo que el mecanismo de seguridad sea peerSAMLV2 será necesario que el destinatario resuelva el contenido del elemento <ds:KeyInfo> transportado en la <ds:Signature> y utilice la clave allí descrita para validar los elementos firmados. Cuando el mecanismo de seguridad sea peerSAMLV2, la clave será la clave del cliente utilizada en la autentificación del cliente SSL/TLS.

- Es necesario que el destinatario cumpla las normas de integridad de los mensajes dirigidas a remitentes y destinatarios cuando se utilizan mecanismos de autentificación de los mensajes.

4) Procesamiento de mensajes con el testigo WSS UIT-T X.509

En esta cláusula se describen la semántica y reglas de procesamiento de los mecanismos cuyo MENSAJE tiene el valor UIT-T X509. En el apéndice I se presenta un ejemplo.

Estos URI soportan autentificación unilateral (remitente) de los mensajes y son de la forma:

- *urn:liberty:security:2003-08:PEER:X509* pudiendo variar PEER en función del mecanismo de autentificación de par desplegado (por ejemplo, puede ser nulo, TLS, etc.).

El mecanismo de autentificación de mensajes WSS UIT-T X509 utiliza el perfil de testigo certificado X.509 de Seguridad de servicios web [b-OASIS WSS UIT-T X.509 profile] como medio de autentificación del destinatario por parte del remitente del mensaje. Estos mecanismos de autentificación de mensajes son unilaterales, o sea, sólo se autentica el remitente del mensaje. Queda fuera del alcance de esta Recomendación proponer cuándo deben autenticarse los mensajes de respuesta pero es interesante observar que podría confiarse en este mecanismo para autenticar también los mensajes de respuesta. Es interesante reconocer, no obstante, que la autentificación independiente de los mensajes de respuesta no ofrece la misma semántica de protección del tren binario de mensajes que el que puede ofrecer el mecanismo de autentificación mutua de la entidad par.

Para valores de implantación que requieran la autentificación de los mensajes con independencia de la autentificación de la entidad par, es necesario que la entidad par remitente ejecute la autentificación de los mensajes demostrando que posee la clave asociada al testigo UIT-T X.509. Es necesario que esta clave sea reconocida por el destinatario como perteneciente a la entidad par remitente.

Cuando el remitente tiene la clave de confirmación del sujeto para firmar elementos del mensaje, la firma garantiza la autenticidad e integridad de los elementos que ampara. Sin embargo, esto por sí solo no mitigaría la amenaza de reproducción, inserción ni ciertas clases de ataques de modificación de mensajes. Para proteger los mensajes contra estos ataques, puede utilizarse uno de los mecanismos que soportan la autentificación de la entidad par o requerir que el modelo de procesamiento de peticiones de vinculación SOAP subyacente contemple estas amenazas.

i) Reglas de procesamiento del remitente

Las reglas de esta cláusula son complementarias a las de procesamiento de la autentificación de mensajes genéricos especificadas en esta Recomendación.

- Es necesario que el remitente demuestre estar en posesión de la clave privada asociada a la firma generada conjuntamente con el perfil del testigo WSS UIT-T X509.
- Para valores de despliegue que requieran autentificación de mensajes independiente, es necesario que se satisfaga la obligación firmando las partes del mensaje pertinentes y grabando información en el encabezamiento <wsse:Security> (como se indica en [b-OASIS WSS SOAP]).
- Para valores de despliegue que no requieran autentificación de mensajes independiente, es necesario que el remitente cumpla esta obligación decorando el encabezamiento de seguridad con un elemento <ds:KeyInfo> que lleve el certificado.

Es necesario que se verifique sin lugar a duda que se trata del mismo certificado y clave utilizados para establecer la autentificación de la entidad par. Esto es necesario para mitigar la amenaza de ataque de sustitución del certificado. Obsérvese asimismo que esta optimización sólo se aplica a los mecanismos *ClientTLS:X509*.

ii) Reglas de procesamiento del destinatario

- Si la política de validación relativa a la autentificación de la entidad par se considera suficiente a efectos de autentificación, es necesario que el destinatario establezca la correspondencia del certificado y la clave utilizados para establecer la autentificación par con la correspondiente información de clave que lleva el mensaje. Gracias a ello, el destinatario del mensaje puede determinar si el remitente del mensaje pretendía utilizar alguna identidad de transporte autenticada en particular. La información que relaciona la clave SSL/TLS con el mensaje MAY puede transportarse en el mensaje utilizando un testigo Seguridad de Mensaje UIT-T X.509 SOAP de OASIS.

6.6 Protección de la información personal identificable (PII)

De acuerdo con [UIT-T Y.2720], *Marco general para la gestión de identidades en las redes de la próxima generación*, la protección de la PII está sujeta a reglamentos nacionales y regionales. Aunque los mecanismos y procedimientos empleados en la protección de la PII pueden variar dependiendo de dichos reglamentos, se basan en principios comunes.

En esta cláusula se presenta un breve resumen de los procedimientos para la protección de la PII especificados en el Informe Especial del NIST *Guide to Protecting the Confidentiality of Personally Identifiable Information (PII)* [b-NIST-SP 800-122] (Guía para la protección de la confidencialidad de la información personal identificable). Las especificaciones sobre *salvaguardas de la confidencialidad de la PII* presentadas en este documento pueden utilizarse como guía para los diseñadores de sistemas IdM. Se definen las siguientes categorías de salvaguardas:

- Salvaguardas operacionales:
  - Política y creación del procedimiento.
  - Concienciación, formación y educación.
- Salvaguardas específicas de la privacidad:
  - Minimización de la utilización, captura y conservación de PII.
  - Evaluación de la repercusión de la privacidad.
  - Desidentificación de la información.
  - Transformación en anónima de la información.
- Controles de seguridad:

  En la cláusula *Controles de seguridad* se ofrecen orientaciones sobre los mecanismos y procedimientos de seguridad que no son específicos de la PII aunque pueden emplearse para la protección de la PII. Análogamente, pueden utilizarse los mecanismos de seguridad de la PII especificados en [UIT-T Y.2704] para proteger la PII.

6.7 Funciones de identidad federada

En [UIT-T Y.2721] *Requisitos de gestión de identidad en las NGN y ejemplos de utilización* se estipula que *la finalidad general de la federación es permitir que cada miembro de una federación siga siendo independiente, al paso que se facilita el intercambio de información de identidad específica para permitir la prestación de servicios federados.*

En la presente cláusula se recomienda utilizar dos mecanismos tipo ampliamente utilizados que permiten a los usuarios acceder a varios servicios sin tener que abonarse a cada uno de ellos.

La Recomendación SAML [UIT-T X.1141] ofrece una solución de federación tipo que se utiliza principalmente en empresas, organizaciones gubernamentales y sus proveedores de servicios.

OpenID [b-OpenID v.2] especifica una solución centrada en el usuario que ha alcanzado bastante difusión para acceder a servicios web en Internet.

6.7.1 Puenteo e interfuncionamiento

En esta Recomendación se describen varios mecanismos de soporte del puenteo y el interfuncionamiento entre diversas federaciones y soluciones IdM. Las importantes se describen en las cláusulas siguientes:

- Integración de la autentificación basada en PKI con el IMS (cláusula 6.2.6).
- Integración de la autentificación basada en PKI con los mecanismos de aserción SAML (cláusula 6.2.7).
- Integración de la autentificación basada en OpenID con el IMS (cláusula 6.2.8).
- La Arquitectura de inicialización genérica (cláusula 6.2.9).
- Correlación y vinculación (cláusula 6.3).
- Funciones de identidad federada (cláusula 6.7).

6.7.2 Descubrimiento del IdSP en un entorno federado

En la cláusula 11.4.3 de la Recomendación SAML [UIT-T X.1141] se define el *perfil de detección del proveedor de identidad*, que permite que el proveedor de un servicio descubra los proveedores de identidad de un usuario. El perfil se especifica en apoyo del *perfil SSO del navegador web SAML* (definido en la cláusula 11.4.1 de [UIT-T X.1141]).

En OpenID [b-OpenID v.2] se especifica un mecanismo de detección que permite que una parte confiante localice a un usuario del IdSP a partir del identificador *OpenID* facilitado por el usuario.

6.8 Control de acceso a la información de identidad

En [UIT-T Y.2721] se exige que la información de identidad sólo sea accesible para las entidades autorizadas con sujeción a la reglamentación y política aplicables. En esta cláusula se describen los mecanismos que pueden utilizarse para verificar los privilegios de autorización.

6.8.1 Mecanismo de compartición de atributos basado en SAML

Las aserciones SAML que contengan las declaraciones de los atributos pueden utilizarse como mecanismo para gestionar privilegios. El mecanismo descrito en la cláusula 6.2.1 puede utilizarse para distribuir los testigos SAML.

6.8.2 Infraestructura de gestión de privilegios basada en UIT-T X.509

El marco de certificados de atributos definido en [UIT-T X.509] puede utilizarse como mecanismo para la infraestructura de gestión de privilegios.

6.9 Inicio de sesión único

El inicio de sesión único (SSO, *single sign-on*) es una capacidad de la red que permite a un usuario abrir una sesión una sola vez y acceder a los diversos servicios de aplicación de la red sin tener que someterse a peticiones reiteradas de facilitar sus credenciales de autentificación para cada servicio de aplicación. Esta capacidad mejora significativamente la experiencia del usuario al permitirle recibir varios servicios sin tener que mantener varias credenciales de autentificación (por ejemplo, pares de nombres de usuario y contraseñas). Dado que el SSO permite a un usuario tener un conjunto de credenciales de autentificación para acceder a varios servicios de aplicaciones, facilita a los proveedores de servicios la imposición de normas más estrictas para establecer credenciales. Esto contribuye a la mejora de la seguridad de la red.

Por otra parte, si las credenciales del usuario están en peligro, la repercusión sobre las redes habilitadas para SSO podría ser mayor que la de los sistemas que no soportan SSO. Para ello, es indispensable que el SSO emplee mecanismos seguros. En esta cláusula se presenta un resumen de los diversos mecanismos que pueden utilizarse para soportar el SSO.

6.9.1 El mecanismo basado en GBA

En la cláusula 6.2.9 se describe la utilización de la GBA para autenticar a un usuario ante cualquier Función de aplicación de red (NAF). De este modo, la GBA proporciona efectivamente un único mecanismo para conectar a un usuario a todas las NAF de una red habilitadas para GBA. Efectivamente, si un usuario se ha conectado a una NAF, la BSF y el UE ya se han autenticado recíprocamente y han establecido una clave compartida (Ks). A continuación, el procedimiento de conectar al usuario a la siguiente NAF consistirá en ejecutar los pasos 1, 3, 4, 5 y 6 (saltándose el 2) descritos en la cláusula 6.2.9. Una vez más, el procedimiento conlleva la compartición de una clave secreta (Ks_NAF) entre el UE y la nueva NAF. Este secreto compartido puede utilizarse para la autentificación entre el UE y la NAF.

Se recomienda utilizar el SSO basado en GBA en aquellos entornos en que se haya desplegado la GBA.

6.9.2 El mecanismo basado en SAML

El mecanismo de inicio de sesión único basado en SAML se especifica en la cláusula 11.4 *Perfiles SSO de SAML* de [UIT-T X.1141], en la que se define un conjunto de perfiles SAML que soportan SSO, y que comprende el *Perfil de desconexión único* (cláusula 11.4.4 de [UIT-T X.1141]). Este perfil especifica un procedimiento que permite que un usuario se desconecte de todas las aplicaciones a las que se haya conectado mediante SSO.

SAML v.2 supone que se han establecido previamente relaciones de confianza entre un IdSP y los RP. También soporta identificadores seudónimos entre un IdSP y un RP. Está recomendable para aplicaciones en los que haya un acuerdo contractual tal como el SLA o transacciones e información de gran valor.

6.9.3 El mecanismo basado en OpenID

La autentificación OpenID 2.0 soporta la capacidad de inicio de sesión único permitiendo a los usuarios finales acceder a más de una parte confiante (RP) una vez autenticados satisfactoriamente. No es necesario que exista una relación de confianza entre el IdSP y la parte confiante. Como sólo soporta el formato basado en URL/URI para identificar a los usuarios, en necesario el DNS para utilizarlo. Por ello, resulta adecuado para aplicaciones de servicios web en los que entran en juego información y transacciones de valor relativamente escaso.

6.10 La desconexión única

El *Protocolo de fin de sesión único del SAML*, cláusula 8.2.7 de [UIT-T X.1141], permite que el usuario final se desconecte casi simultáneamente de las múltiples sesiones participantes. Se denominan sesiones participantes a las establecidas a través de un IdSP (es decir, que el IdSP ha aseverado la identidad del usuario para las sesiones entre un usuario y las aplicaciones). El IdSP controla el estado de todas las sesiones autenticadas con diversas partes confiantes que un usuario haya establecido a través de él. Esto comprende la invalidación de las credenciales de autentificación (por ejemplo, cookies y aserciones) de las sesiones terminadas. Este protocolo puede utilizarse en los casos siguientes:

1) El usuario se desconecta de una de las sesiones e indica que desea terminar todas las sesiones que haya iniciado el IdSP.
2) El usuario indica directamente al IdSP que desea terminar todas las sesiones.
3) El IdSP desconecta a un usuario sin que éste lo haya solicitado (por ejemplo, debido al agotamiento de un plazo).

Este protocolo define las entidades participantes, su comportamiento, el flujo de los mensajes y el formato de los mensajes intercambiados. En las subcláusulas siguientes se describe la utilización de este protocolo en los casos anteriormente relacionados.

6.10.1 El usuario se desconecta de una de las sesiones e indica que desea terminar todas las sesiones iniciadas por el IdSP

En la figura 11 se representan los pasos básicos del flujo del mensaje que se describen a continuación.

6.10.1.1 Entidades implicadas en el procedimiento y el flujo de información

Las entidades implicadas son las siguientes:

- Función de usuario final.
- Servidor de aplicaciones 1 (AS1) – Entidad que presta un servicio. Desempeña la función de parte confiante. Hace de peticionario y respondedor SAML de acuerdo con la definición de [UIT-T X.1141].
- Servidor de aplicaciones 2 (AS2) – Entidad que presta un servicio. Desempeña la función de parte confiante. Hace de peticionario y respondedor SAML de acuerdo con la definición de [UIT-T X.1141].
- A-2 – Entidad funcional pasarela de la aplicación (APL-GW-FE), sirve de IdSP y hace de peticionario y respondedor SAML de acuerdo con la definición de [UIT-T X.1141].

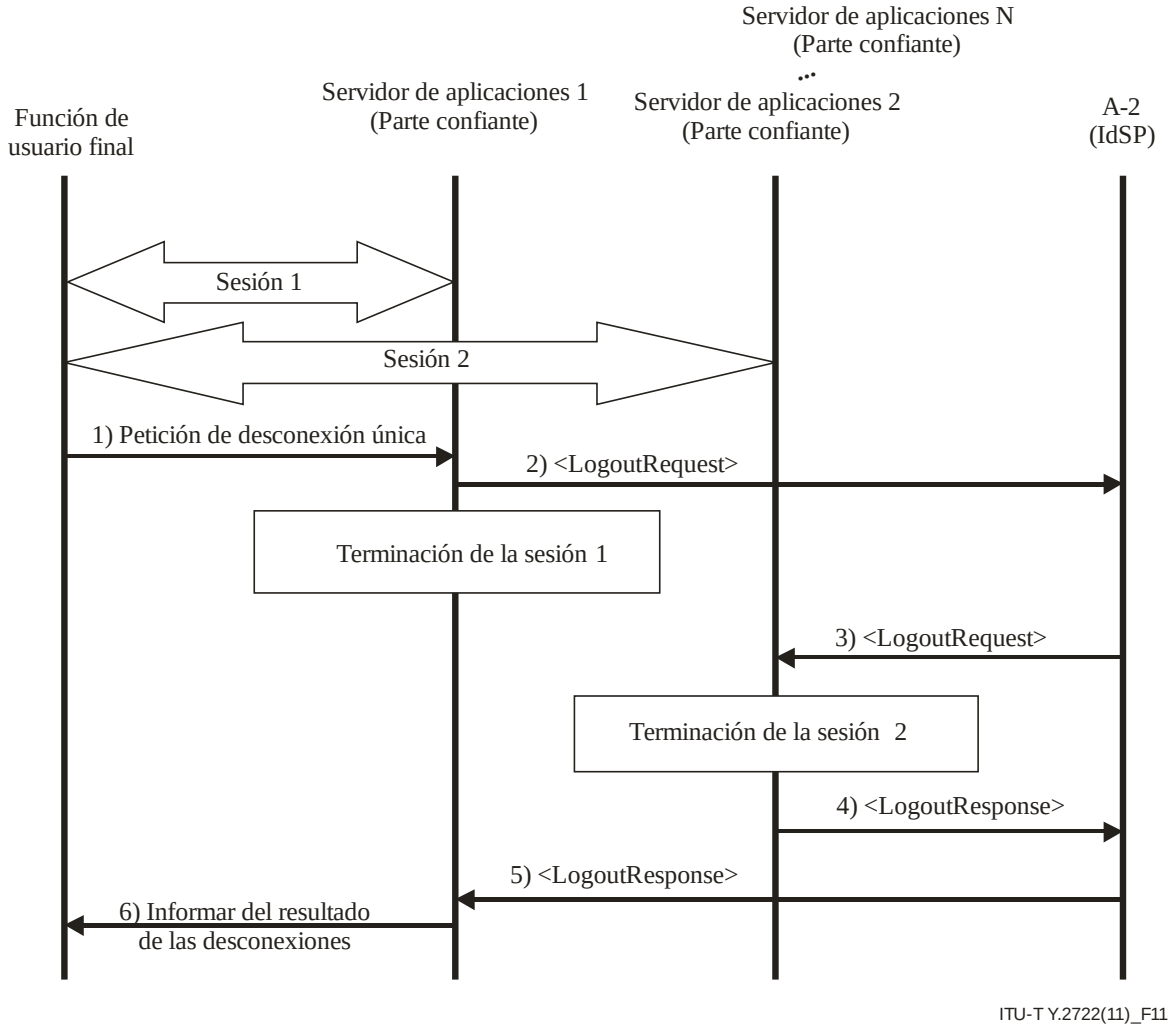


Figura 11 – Desconexión única basada en SAML solicitada por un usuario en una sesión participante

Los pasos básicos del procedimiento de la desconexión única (también llamada terminación de sesión única) son los siguientes:

1) La función de usuario final invoca una petición de desconexión en el servidor de aplicaciones 1 (AS1) indicando que desea desconectarse de todas las sesiones participantes.

2) El AS1 solicita una desconexión de todas las sesiones participantes mediante el envío de <LogoutRequest> a A-2. Esta petición debe estar firmada por motivos de autentificación y protección de la integridad con arreglo a lo especificado en la cláusula 8.2.7 de [UIT-T X.1141].

   Tras este paso, AS1 intenta terminar la sesión 1. Para ello, AS1 invalida las credenciales de autentificación de la sesión (por ejemplo, las aserciones y las cookies), lo que obligará a la función de usuario final a someterse al procedimiento de autentificación si presenta otra petición a AS1.

3) Tras validar la petición de AS1, A-2 envía a todas la partes confiantes (aunque en la figura 11 sólo se muestra AS2) los mensajes <LogoutRequest>. Estas peticiones deben estar firmadas con arreglo a lo especificado en la cláusula 8.2.7 de [UIT-T X.1141].

   Tras validar la petición de desconexión, AS2 intenta terminar la sesión 2.

4) AS2 informa al remitente de la petición de desconexión (A-2) acerca del resultado del intento de desconexión mediante el envío de <LogoutResponse>, que debe estar firmado.

5) A-2 envía <LogoutResponse>, al remitente inicial de la petición de desconexión (AS1) comunicando el resultado de la desconexión única (por ejemplo, satisfactoria o desconexión parcial). La respuesta tiene que estar firmada.

   Tras este paso A-2 actualiza su lista de sesiones activas e invalida las credenciales de autentificación (por ejemplo, las aserciones y las cookies) de las sesiones que deben terminar.

6) AS1 responde a la petición de la función de usuario final del paso 1 comunicándole el resultado de la desconexión.

6.10.2 El usuario indica directamente al IdSP que desea desconectarse de todas las sesiones

En la figura 12 se muestran los pasos básicos del flujo de mensajes que se describen a continuación.

6.10.2.1 Entidades implicadas en el procedimiento y el flujo de información

Las entidades implicadas en el procedimiento de desconexión coinciden con las descritas en la cláusula 6.10.1.1. En este caso, la función de usuario final tiene una sesión separada (la sesión 3) con A-2 (el IdSP). Utiliza la sesión para enviar una única petición de desconexión en el paso 1. Los demás pasos del procedimiento son semejantes a los descritos en la cláusula 6.10.1.1.

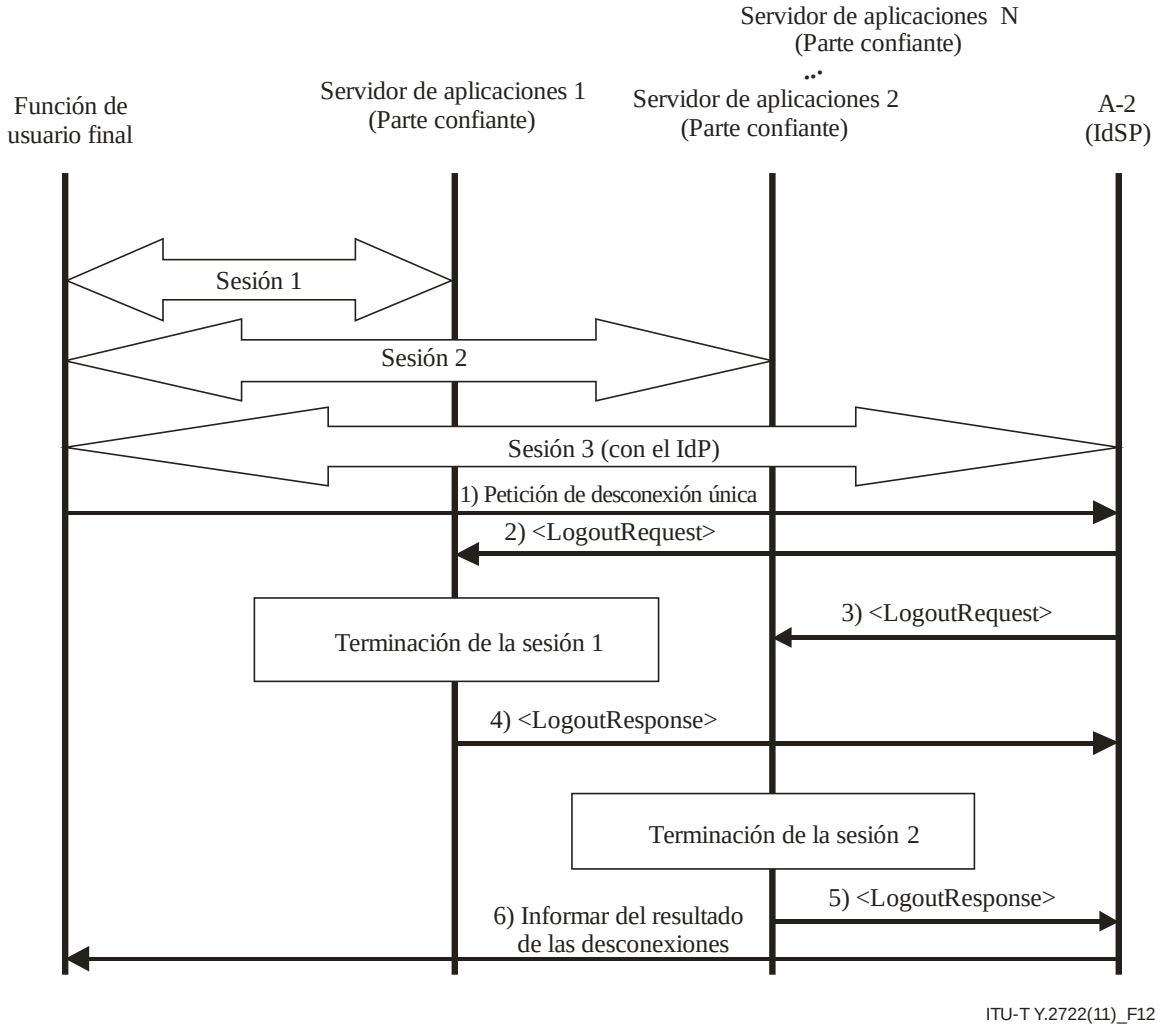


Figura 12 – Desconexión única basada en SAML solicitada por un usuario a un IdSP

Los pasos básicos de este procedimiento de desconexión única son los siguientes:

1) La función de usuario final invoca directamente una petición de desconexión única en A-2.

2) A-2 envía <LogoutRequest> a AS1. Esta petición debe estar firmada por motivos de autentificación y protección de la integridad con arreglo a lo especificado en la cláusula 8.2.7 de [UIT-T X.1141].

   Tras validar esta petición, AS1 intenta terminar la sesión 1. Para ello, AS1 invalida las credenciales de autentificación de la sesión (por ejemplo, las aserciones y las cookies), lo que obligará a la función de usuario final a someterse al procedimiento de autentificación si presenta otra petición a AS1.

3) A-2 envía <LogoutRequest> a AS2 (también envía esta petición a los demás servidores de las sesiones participantes). Este paso es semejante al 2.

   Tras validar la petición de desconexión, AS2 intenta terminar la sesión 2.

4) AS1 informa al remitente de la petición de desconexión (A-2) acerca del resultado del intento de desconexión mediante el envío de <LogoutResponse>, que debe estar firmado.

5) Al igual que en el paso anterior, AS2 informa al remitente de la petición de desconexión (A-2) acerca del resultado del intento de desconexión mediante el envío de un <LogoutResponse> firmado.

Tras este paso A-2 actualiza su lista de sesiones activas e invalida las credenciales de autentificación (por ejemplo, las aserciones y las cookies) de las sesiones que deban terminar.

Tras validar todas las respuestas de desconexión, A-2 informa a la función de usuario final del resultado de la desconexión única. Ésta es la respuesta a la petición de la función de usuario final del paso 1.

7 Seguridad

Los mecanismos contemplados en la presente Recomendación junto con los especificados en [UIT-T Y.2704] se han diseñado para satisfacer los requisitos de seguridad de [UIT-T Y.2721].

Apéndice I

Autentificación del mensaje WSS X.509 v3

(Este apéndice no forma parte integrante de la presente Recomendación)

En el siguiente ejemplo se muestra una forma de procesar los mensajes con testigo WSS X.509 como se explica en la cláusula 6.5.1.2.

```
<?xml version="1.0" encoding="UTF-8"?>
<s:Envelope xmlns:s="http://schemas.xmlsoap.org /soap/envelope/"
     xmlns:sb="urn:liberty:sb:2006-08"
     xmlns:pp="urn:liberty:id-sis-pp:2003-08"
     xmlns:sec="urn:liberty:security:20 06-08"
     xmlns:wsse="http://docs.oasis-open.org/wss/2004 /01/oasis-200401-wss-wssecurity-
     secext-1.0.xsd"
     xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-
     utility-1.0.xsd"
     xmlns:wsa="http://www.w3.org/2005/08/addressing">

<s:Header>
<!-- see Liberty SOAP Binding Specification for which headers are required and optional -
->

<wsa:MessageID wsu:Id="mid">...</wsa:MessageID>

<wsa:To wsu:Id="to">...</wsa:To>

<wsa:Action wsu:Id="action">...</wsa:Action>

<wsse:Security mustUnderstand="1">

       <wsu:Timestamp wsu:Id="ts">
       <wsu:Created>2005-06-17T04:49:17Z</ wsu:Created >
       </wsu:Timestamp>

       <wsse:BinarySecurityToken
              ValueType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-x509-
token-profile-1.0#X509v3 "
              wsu:Id="X509Token"
              EncodingType="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-soap-
message-security-1.0#Base64Binar y">
              MIIB9zCCAWSgAwIBAgIQ...
       </wsse:BinarySecurityToken>

       <ds:Signature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
              <ds:SignedInfo>

                     <!-- in general include a ds:Reference for each wsa: header added
according to SOAP binding -->

                     <!-- include the MessageID in the signature -->
                     <ds:Reference URI="#mid">...</ds:Reference>

                     <!-- include the To in the signature -->
                     <ds:Reference URI="#to">...</ds:Reference>

                     <!-- include the Action in the signature -->
                     <ds:Reference URI="#action">...</ds:Reference>

                     <!-- include the Timestamp in the signature -->
                     <ds:Reference URI="#ts">...</ds:Reference>

                     <!-- bind the security token (thwart cert substitution attacks) -->
                     <ds:Reference URI="#X509Token">
                            <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/x
mldsig#sha1"/>
```

```
                        <ds:DigestValue>Ru4cAfeBABE...</ ds:DigestValue>
                    </ds:Reference>

                    <!-- bind the body of the message -->
                    <ds:Reference URI="#MsgBody">
                        <ds:DigestMethod Algorithm="http://www.w3.org/2000/09/
                        xmldsig# sha1"/>
                        <ds:DigestValue>YgGfS0pi56pu...</ds:DigestValue>
                    </ds:Reference>
                </ds:SignedInfo>
                <ds:KeyInfo>
                    <wsse:SecurityTokenReference>
                        <wsse:Reference URI="#X509Token" />
                    </wsse:SecurityTokenReference>
                </ds:KeyInfo>
                <ds:SignatureValue>
                    HJJWbvqW9E84vJVQkjjLLA6nNvBX7mY00TZhwBdFNDElgscS XZ5Ekw==
                </ds:SignatureValue>
        </ds:Signature>
</wsse:Security>
</s:Header>

<s:Body wsu:Id="MsgBody">
        <pp:Modify>
            <!-- this is an ID-SIS-PP Modify message -->
        </pp:Modify>
</s:Body>


</s:Envelope>
```

Apéndice II

Mecanismo de control de acceso basado en "OpenID + OAuth"

(Este apéndice no forma parte integrante de la presente Recomendación)

En la cláusula 6.2.8 se explica la utilización de OpenID para autenticar a un usuario de cualquier función de aplicación de red (NAF). Así pues, se propone introducir OAuth basado en OpenID para proteger la PII y controlar el acceso.

II.1 OAuth [b-IETF RFC 5849]

OAuth es un protocolo abierto gracias al cual una aplicación puede acceder a la información de un usuario final facilitada por un servicio web siempre que la aplicación cuente con la autorización del usuario final. La información del usuario final se transmite de forma segura sin revelar la identidad del usuario.

La meta de OAuth es adquirir un testigo de acceso del servidor web, que puede utilizarse para intercambiar datos específicos del usuario con un servicio web (tal como información de calendario o una lista de direcciones). El proceso ordinario de OAuth consiste en una secuencia de cuatro pasos:

1) Solicitar un testigo de la "petición".
2) Solicitar la autorización del testigo, lo que dará lugar a la autorización del usuario.
3) Intercambiar el testigo de la petición autorizado por un testigo de "acceso".
4) Utilizar el testigo de acceso para interaccionar con los datos del servicio web acerca del usuario.

Para mayor información sobre OAuth, puede consultarse [b-IETF RFC 5849].

II.2 Uso conjunto de OpenID y OAuth

Si bien OpenID puede emplearse como mecanismo IdM de autentificación de usuarios, también puede utilizarse OAuth para autorizar el acceso a datos de usuario sensibles. En tal caso, el IdSP ejerce las funciones de proveedor de identidad OpenID (OP) y de proveedor de servicios OAuth.

III.3 Flujo de la autorización OpenID + OAuth

Con OpenID + OAuth esta secuencia sigue siendo prácticamente la misma. La diferencia es que la obtención de un testigo autorizado de la petición OAuth (pasos 1 y 2) está integrada en la petición de autentificación OpenID. De este modo, el usuario puede aprobar simultáneamente el inicio de sesión y el acceso al servicio.

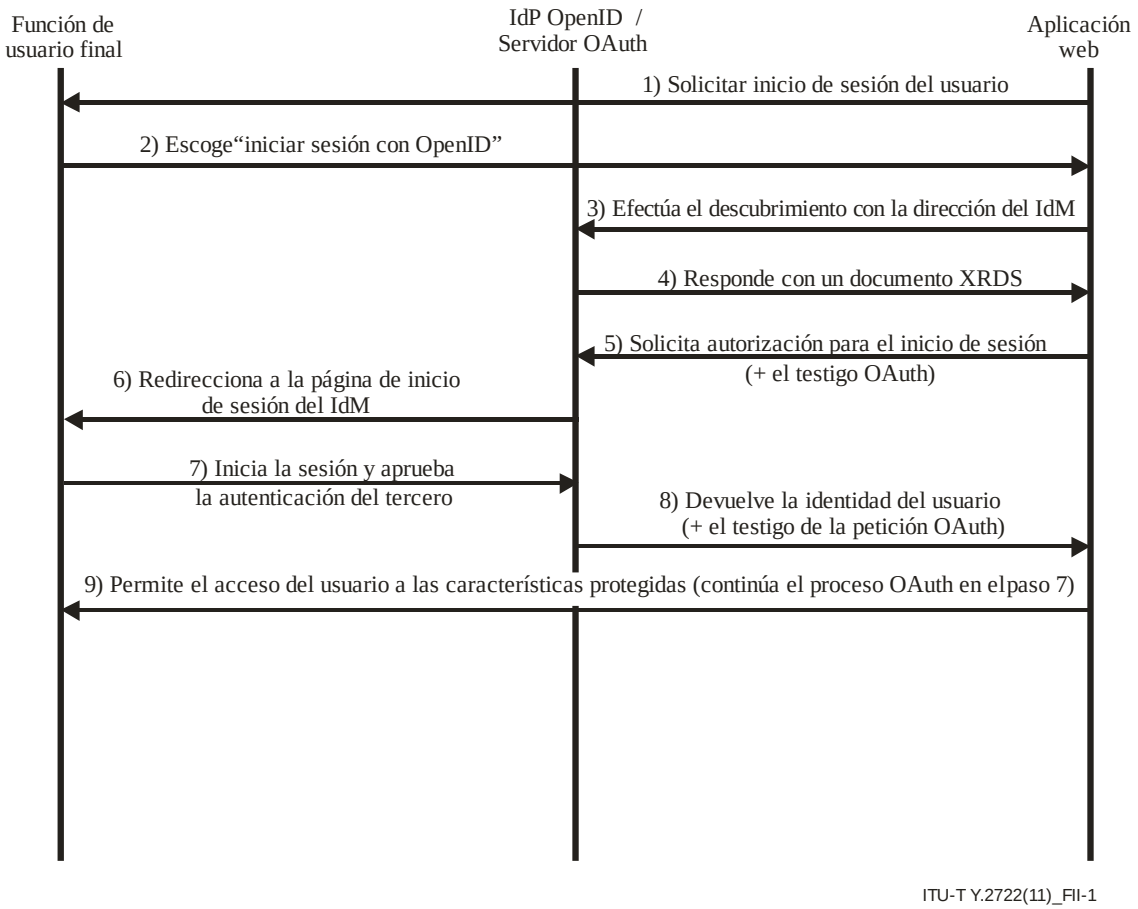


Figura II.1 – Autentificación basada en OpenID + OAuth

Los pasos básicos son los siguientes:

1) La aplicación web pide al usuario final que inicie la sesión presentándole un conjunto de alternativas, entre ellas la de utilizar su cuenta OpenID.
2) El usuario selecciona la opción "Iniciar sesión con OpenID".
3) La aplicación web envía una petición "detección" al IdSP para obtener información del punto de extremo de autentificación del inicio de sesión del IdSP.
4) El IdSP devuelve un documento XRDS donde figura la dirección del punto extremo.
5) La aplicación web envía una petición de autentificación del inicio de sesión a la dirección del punto de extremo.
6) Esta acción ocasiona el redireccionamiento del usuario a una página de inicio de sesión federada del IdSP, ya sea en la misma ventana del explorador o en una ventana emergente, y se solicita al usuario que introduzca sus datos de conexión.
7) Una vez el usuario a iniciado la sesión, el IdSP muestra una página de confirmación y notifica al usuario que la aplicación de un tercero está solicitando la autentificación. Esta página solicita al usuario que confirme o rechace la vinculación del inicio de sesión en su cuenta IdSP al inicio de sesión de la aplicación web. A continuación se solicita al usuario que apruebe el acceso a un conjunto especificado de servicios. Tanto el inicio de sesión como la compartición de la información del usuario, deben estar aprobados por el usuario para que pueda continuar la autentificación.

8) Si el usuario aprueba la autentificación, el IdSP devuelve al usuario a la URL especificada en el parámetro `openid.return_to` de la petición original. Si el identificativo suministrado por el IdSP no guarda relación con nombre de la cuenta IdSM o contraseña actual del usuario, se añade como el parámetro de consulta `openid.claimed_id`. Si la petición también comprende el intercambio de atributos, puede añadirse información de usuarios adicionales. Para OpenID + OAuth se devuelve también un testigo de la petición OAuth.

9) La aplicación web utiliza el identificador proporcionado por el IdSP para reconocer al usuario y facilitar el acceso a las características y datos de la aplicación. Para OpenID + OAuth, la aplicación web utiliza el testigo de la petición para continuar la secuencia de OAuth y obtener acceso a los servicios IdSP del usuario.

Bibliografía

[b-ETSI TS 133 220] ETSI TS 133 220 V6.3.0 (2004), *Universal Mobile Telecommunications System (UMTS); Generic Authentication Architecture (GAA); Generic bootstrapping architecture*.

[b-IETF RFC 2616] IETF RFC 2616 (1999), *Hypertext Transfer Protocol – HTTP/1.1*. <http://datatracker.ietf.org/doc/rfc2616/>

[b-IETF RFC 2617] IETF RFC 2617 (1999), *HTTP Authentication: Basic and Digest Access Authentication*. <http://datatracker.ietf.org/doc/rfc2617/>

[b-IETF RFC 3310] IETF RFC 3310 (2002), *Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA)*. <http://www.rfc-editor.org/rfc/rfc3310.txt>

[b-IETF RFC 4169] IETF RFC 4169 (2005), *Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA) Version-2*. <http://www.ietf.org/rfc/rfc4169.txt?number=4169>

[b-IETF RFC 4279] IETF RFC 4279 (2005), *Pre-Shared Key Ciphersuites for Transport Layer Security (TLS)*. <http://datatracker.ietf.org/doc/rfc4279/>

[b-IETF RFC 5849] IETF RFC 5849 (2010), *The OAuth 1.0 Protocol*. <http://tools.ietf.org/html/rfc5849>

[b-LA WSF] Liberty Alliance (2008), *Web Services Framework: A Technical Overview*. <http://www.projectliberty.org/liberty/content/download/4120/27687/file/idwsf-intro-v1.0.pdf>

[b-LA ID-WSF security] Liberty Alliance Project (2007), *Liberty ID-WSF Security Mechanisms Core version 2.0-errata version 1.0*.

[b-LA SOAP binding] Liberty Alliance Project Web Services Security (WSS) (2006), *Liberty SOAP Binding Version 2.0*.

[b-NIST-SP 800-122] NIST Special Publication SP 800-122 (2010), *Guide to Protecting the Confidentiality of Personally Identifiable Information (PII)*. <http://csrc.nist.gov/publications/nistpubs/800-122/sp800-122.pdf>

[b-OASIS SAML token] OASIS (2006), *Web Services security: SAML Token Profile 1.1, and its Approved Errata 1*.

[b-OASIS WSS SOAP] OASIS (2004), *Web Services Security: SOAP Message Security 1.1 (WS-Security 2004)*.

[b-OASIS WSS X.509 profile] OASIS (2006), *Web Services Security X.509 Certificate Token Profile 1.1*.

[b-OpenID v.2] *OpenID Authentication 2.0*. <http://openid.net/specs/openid-authentication-2_0.html>

[b-W3C XML signature] World Wide Web Consortium (W3C) (2008), *XML Signature Syntax and Processing (second edition)*.

[b-3GPP TR 33.924] 3GPP TR 33.924 Release 9 (2009), 3rd Generation Partnership Project, *Identity management and 3GPP security interworking; Identity management and Generic Authentication Architecture (GAA) interworking (Release 9)*.

SERIES DE RECOMENDACIONES DEL UIT-T

Serie A	Organización del trabajo del UIT-T
Serie D	Principios generales de tarificación
Serie E	Explotación general de la red, servicio telefónico, explotación del servicio y factores humanos
Serie F	Servicios de telecomunicación no telefónicos
Serie G	Sistemas y medios de transmisión, sistemas y redes digitales
Serie H	Sistemas audiovisuales y multimedia
Serie I	Red digital de servicios integrados
Serie J	Redes de cable y transmisión de programas radiofónicos y televisivos, y de otras señales multimedia
Serie K	Protección contra las interferencias
Serie L	Construcción, instalación y protección de los cables y otros elementos de planta exterior
Serie M	Gestión de las telecomunicaciones, incluida la RGT y el mantenimiento de redes
Serie N	Mantenimiento: circuitos internacionales para transmisiones radiofónicas y de televisión
Serie O	Especificaciones de los aparatos de medida
Serie P	Terminales y métodos de evaluación subjetivos y objetivos
Serie Q	Conmutación y señalización
Serie R	Transmisión telegráfica
Serie S	Equipos terminales para servicios de telegrafía
Serie T	Terminales para servicios de telemática
Serie U	Conmutación telegráfica
Serie V	Comunicación de datos por la red telefónica
Serie X	Redes de datos, comunicaciones de sistemas abiertos y seguridad
Serie Y	**Infraestructura mundial de la información, aspectos del protocolo Internet y redes de la próxima generación**
Serie Z	Lenguajes y aspectos generales de soporte lógico para sistemas de telecomunicación

Impreso en Suiza
Ginebra, 2011