Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

H.235.2

(09/2005)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Infrastructure des services audiovisuels – Aspects système

Cadre de sécurité H.323: profil de sécurité avec signature

Recommandation UIT-T H.235.2

Union internationale des télécommunications

RECOMMANDATIONS UIT-T DE LA SÉRIE H

SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

CARACTÉRISTIQUES DES SYSTÈMES VISIOPHONIQUES	H.100–H.199
INFRASTRUCTURE DES SERVICES AUDIOVISUELS	
Généralités	H.200–H.219
Multiplexage et synchronisation en transmission	H.220–H.229
Aspects système	**H.230–H.239**
Procédures de communication	H.240–H.259
Codage des images vidéo animées	H.260–H.279
Aspects liés aux systèmes	H.280–H.299
Systèmes et équipements terminaux pour les services audiovisuels	H.300–H.349
Architecture des services d'annuaire pour les services audiovisuels et multimédias	H.350–H.359
Architecture de la qualité de service pour les services audiovisuels et multimédias	H.360–H.369
Services complémentaires en multimédia	H.450–H.499
PROCÉDURES DE MOBILITÉ ET DE COLLABORATION	
Aperçu général de la mobilité et de la collaboration, définitions, protocoles et procédures	H.500–H.509
Mobilité pour les systèmes et services multimédias de la série H	H.510–H.519
Applications et services de collaboration multimédia mobile	H.520–H.529
Sécurité pour les systèmes et services multimédias mobiles	H.530–H.539
Sécurité pour les applications et services de collaboration multimédia mobile	H.540–H.549
Procédures d'interfonctionnement de la mobilité	H.550–H.559
Procédures d'interfonctionnement de collaboration multimédia mobile	H.560–H.569
SERVICES À LARGE BANDE ET MULTIMÉDIAS TRI-SERVICES	
Services multimédias à large bande sur VDSL	H.610–H.619

Pour plus de détails, voir la Liste des Recommandations de l'UIT-T.

Recommandation UIT-T H.235.2

Cadre de sécurité H.323: profil de sécurité avec signature

Résumé

La présente Recommandation décrit un profil de sécurité facultatif visant à utiliser des signatures numériques pour sécuriser la signalisation H.225.0.

Dans les anciennes versions de la sous-série H.235, ce profil était défini dans l'Annexe E/H.235. Les Appendices IV, V et VI/H.235.0 donnent le mappage entre tous les paragraphes, toutes les figures et tous les tableaux de la version 3 et tous ceux de la version 4 de la Rec. UIT-T H.235.

Source

La Recommandation UIT-T H.235.2 a été approuvée le 13 septembre 2005 par la Commission d'études 16 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8.

Mots clés

Authentification, certificat, chiffrement, gestion de clés, intégrité, profil de sécurité, sécurité multimédia, signature numérique.

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Il est chargé de l'étude des questions techniques, d'exploitation et de tarification, et émet à ce sujet des Recommandations en vue de la normalisation des télécommunications à l'échelle mondiale.

L'Assemblée mondiale de normalisation des télécommunications (AMNT), qui se réunit tous les quatre ans, détermine les thèmes d'étude à traiter par les Commissions d'études de l'UIT-T, lesquelles élaborent en retour des Recommandations sur ces thèmes.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution 1 de l'AMNT.

Dans certains secteurs des technologies de l'information qui correspondent à la sphère de compétence de l'UIT-T, les normes nécessaires se préparent en collaboration avec l'ISO et la CEI.

NOTE

Dans la présente Recommandation, l'expression "Administration" est utilisée pour désigner de façon abrégée aussi bien une administration de télécommunications qu'une exploitation reconnue.

Le respect de cette Recommandation se fait à titre volontaire. Cependant, il se peut que la Recommandation contienne certaines dispositions obligatoires (pour assurer, par exemple, l'interopérabilité et l'applicabilité) et considère que la Recommandation est respectée lorsque toutes ces dispositions sont observées. Le futur d'obligation et les autres moyens d'expression de l'obligation comme le verbe "devoir" ainsi que leurs formes négatives servent à énoncer des prescriptions. L'utilisation de ces formes ne signifie pas qu'il est obligatoire de respecter la Recommandation.

DROITS DE PROPRIÉTÉ INTELLECTUELLE

L'UIT attire l'attention sur la possibilité que l'application ou la mise en œuvre de la présente Recommandation puisse donner lieu à l'utilisation d'un droit de propriété intellectuelle. L'UIT ne prend pas position en ce qui concerne l'existence, la validité ou l'applicabilité des droits de propriété intellectuelle, qu'ils soient revendiqués par un Membre de l'UIT ou par une tierce partie étrangère à la procédure d'élaboration des Recommandations.

A la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux responsables de la mise en œuvre de consulter la base de données des brevets du TSB.

TABLE DES MATIÈRES

Recommandation UIT-T H.235.2

Cadre de sécurité H.323: profil de sécurité avec signature

1 Domaine d'application

La présente Recommandation décrit un profil de sécurité facultatif visant à utiliser des signatures numériques pour sécuriser la signalisation H.225.0.

2 Références

2.1 Références normatives

La présente Recommandation se réfère à certaines dispositions des Recommandations UIT-T et textes suivants qui, de ce fait, en sont partie intégrante. Les versions indiquées étaient en vigueur au moment de la publication de la présente Recommandation. Toute Recommandation ou tout texte étant sujet à révision, les utilisateurs de la présente Recommandation sont invités à se reporter, si possible, aux versions les plus récentes des références normatives suivantes. La liste des Recommandations de l'UIT-T en vigueur est régulièrement publiée. La référence à un document figurant dans la présente Recommandation ne donne pas à ce document, en tant que tel, le statut d'une Recommandation.

– Recommandation UIT-T H.225.0 (2003), *Protocoles de signalisation d'appel et paquétisation des flux monomédias pour les systèmes de communication multimédias en mode paquet*.

– Recommandation UIT-T H.235 (1998), *Sécurité et cryptage des terminaux multimédias de la série H (terminaux H.323 et autres terminaux de type H.245)*.

– Recommandation UIT-T H.235.0 (2005), *Cadre de sécurité H.323: cadre de sécurité pour les systèmes multimédias de la série H (systèmes H.323 et autres systèmes de type H.245)*.

– Recommandation UIT-T H.235.1 (2005), *Cadre de sécurité H.323: profil de sécurité de base*.

– Recommandation UIT-T H.235.6 (2005), *Cadre de sécurité H.323: profil pour le chiffrement vocal avec gestion de clés H.235/H.245 native*.

– Recommandation UIT-T H.245 (2005), *Protocole de commande pour communications multimédias*.

– Recommandation UIT-T H.323 (2003), *Systèmes de communication multimédia en mode paquet*.

– Recommandation UIT-T Q.931 (1998), *Spécification de la couche 3 de l'interface utilisateur-réseau RNIS pour la commande de l'appel de base*.

– Recommandation UIT-T X.509 (2005) | ISO/CEI 9594-8:2005, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: cadre général des certificats de clé publique et d'attribut*.

– Recommandation UIT-T X.800 (1991), *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT*.

– Recommandation UIT-T X.803 (1994) | ISO/CEI 10745:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de sécurité pour les couches supérieure*s.

– Recommandation UIT-T X.810 (1995) | ISO/CEI 10181-1:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: aperçu général*.

– Recommandation UIT-T X.811 (1995) | ISO/CEI 10181-2:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: cadre d'authentification*.

– ISO/CEI 7498-2:1989, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Modèle de référence de base – Partie 2: Architecture de sécurité*.

– ISO/CEI 9798-3:1998, *Technologies de l'information – Techniques de sécurité – Authentification d'entité – Partie 3: Mécanismes utilisant des techniques de signature numériques*.

– IETF RFC 3280 (2002), *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*.

2.2 Références informatives

[ISO/CEI 14888-3] ISO/CEI 14888-3:1998, *Technologies de l'information – Techniques de sécurité – Signatures digitales avec appendice – Partie 3: Mécanismes fondés sur certificat*.

[PKCS] PKCS #1 v2.0: *RSA Cryptography Standard;* RSA Laboratories; 1er octobre 1998; http://www.rsa.com/rsalabs/pubs/PKCS/index.html.

[PKCS] PKCS #7: *Cryptographic Message Syntax Standard*, An RSA Laboratories Technical Note, version 1.5, Revised 1er novembre 1993; http://www.rsa.com/rsalabs/pubs/PKCS/index.html

[RFC1321] IETF RFC 1321 (1992), *The MD5 Message-Digest Algorithm.*

[RFC3447] IETF RFC 3447 (2003), *Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1*.

3 Termes et définitions

Dans la présente Recommandation, les définitions figurant au § 3/H.323, au § 3/H.225.0 et au § 3/H.245 s'appliquent, en plus de celles du présent paragraphe. Certains des termes suivants sont utilisés selon la définition donnée dans les Recommandations UIT-T X.800 | ISO 7498-2, X.803 | ISO/CEI 10745, X.810 | ISO/CEI 10181-1 et X.811 | ISO/CEI 10181-2.

3.1 autorité de certification: utilisée dans un contexte de signature électronique, une Autorité de certification (CA) certifie les clés de vérification publique en émettant des "certificats".

3.2 registre de certificats: un dépôt de certificat (par exemple un annuaire X.500) contient des certificats d'utilisateur et des listes de révocation de certificats (CRL). Il est fiable pour ce qui est de donner accès à ces informations mais il n'est responsable ni du contenu ni de l'exactitude des informations qu'il reçoit des entités CA et RA.

3.3 signature numérique: transformation cryptographique (au moyen d'une technique cryptographique asymétrique) de la représentation numérique d'un message de données, telle que toute personne ayant le message signé et la clé publique appropriée peut déterminer:

i) si la transformation a été faite au moyen de la clé privée correspondant à la clé publique en question;

ii) si le message signé n'a pas été altéré depuis la transformation cryptographique.

3.4 fournisseur de statut de certificat en ligne: le protocole de statut de certificat en ligne (OCSP) permet à des applications de déterminer l'état de révocation d'un certificat identifié. Le protocole OCSP peut être utilisé pour satisfaire certaines conditions opérationnelles visant à fournir les informations de révocation plus rapidement que cela n'est possible avec les listes CRL. On peut considérer les fournisseurs de statut de certificat en ligne comme une alternative à l'emploi des listes CRL hors ligne.

3.5 proxy: le proxy est une entité H.323 intermédiaire analogue à un portier. Il peut être un nœud de réseau séparé ou peut être situé au même endroit que la fonctionnalité d'une entité H.323, par exemple celle d'un portier. Le proxy peut effectuer des tâches de sécurité telles que la vérification de signatures et de certificats ainsi que le contrôle d'accès.

3.6 autorité d'enregistrement: les autorités d'enregistrement agissent comme des intermédiaires entre les utilisateurs et les autorités de certification. Elles reçoivent des demandes émanant des utilisateurs et les transmettent aux autorités de certification sous une forme appropriée.

3.7 autorité d'horodatage: les autorités d'horodatage sont obligatoires pour la non-répudiation en cas de perte ou de corruption de clé. Dans la pratique, elles fournissent à quiconque une contre-signature, avec une heure fiable, en plus d'une valeur de hachage et d'un identificateur de hachage.

3.8 fournisseur de services de confiance: entité qui peut être utilisée par d'autres entités comme un intermédiaire de confiance dans un processus de communication ou de vérification, ou comme un fournisseur de services d'information de confiance.

La présente Recommandation utilise les termes suivants dans le contexte de la fourniture des services de sécurité.

3.9 authentification seulement: ce service de sécurité du profil de sécurité avec signature prend en charge l'authentification de l'utilisateur lorsque celui-ci s'authentifie par la signature numérique correcte de données au moyen de la clé privée. On notera que ce service de sécurité n'offre pas de contre-mesures en cas d'opération "couper & coller" arbitraire, de manipulation de message ou d'attaque par altération. L'authentification seulement peut être utile pour les proxys de sécurité qui vérifient l'authenticité du message (authentification de l'origine des données) lors de la retransmission de ce message à une autre destination (un portier, par exemple).

NOTE – La retransmission modifie généralement certaines parties du message; il n'est donc pas possible d'assurer l'intégrité de bout en bout.

Néanmoins, l'authentification seulement peut également être appliquée bond par bond. La procédure III définit ce service de sécurité dans le cas de bout en bout alors que la procédure II le définit dans le cas bond par bond.

3.10 authentification et intégrité: double service de sécurité prenant en charge l'intégrité de message en plus de l'authentification de l'utilisateur. L'utilisateur s'authentifie par la signature numérique correcte de données au moyen de la clé privée. En outre, le message est protégé contre les altérations. Les deux services de sécurité sont fournis par le même mécanisme de sécurité. L'authentification et l'intégrité combinées ne sont possibles que dans le cas bond par bond. Ce double service de sécurité est défini dans la procédure II.

NOTE – L'utilisation de signatures numériques permet éventuellement de prendre en charge un service de sécurité de non-répudiation; cela dépend aussi de la valeur des bits d'utilisation de la clé de signature dans le certificat (voir également RFC 3280).

4 Symboles et abréviations

La présente Recommandation utilise les abréviations suivantes:

ARQ demande d'admission (*admission request*)

ASN.1 notation de syntaxe abstraite numéro un (*abstract syntax notation one*)

CA autorité de certification (*certification authority*)

CRL liste de révocation de certificats (*certificate revocation list*)

DH Diffie-Hellman

DNS système de dénomination de domaine (*domain name system*)

EP point d'extrémité (*endpoint*)

EPID identificateur de point d'extrémité (*endpoint identifier*)

GK portier (*gatekeeper*)

GKID identificateur de portier (*gatekeeper identifier*)

GRQ demande de portier (*gatekeeper request*)

ICV valeur de contrôle d'intégrité (*integrity check value*)

IP protocole Internet (*Internet protocol*)

LDAP protocole rapide d'accès à l'annuaire (*light-weight directory access protocol*)

LRQ demande de localisation (*location request*)

MCU pont de conférence (*multipoint control unit*)

MD5 résumé de message numéro 5 (*message digest 5*)

NAT traduction d'adresse de réseau (*network address translation*)

OCSP protocole de statut de certificat en ligne (*online certificate status protocol*)

OID identificateur d'objet (*object identifier*)

PKCS système cryptographique à clé publique (*public-key crypto system*)

RA autorité d'enregistrement (*registration authority*)

RAS enregistrement, admission et statut (*registration, admission and status*)

RSA Rivest, Shamir, Adleman

RTP protocole de transport en temps réel (*real-time protocol*)

SHA algorithme de hachage sécurisé (*secure hash algorithm*)

UIT Union internationale des télécommunications

URL identificateur uniforme de ressource (*uniform resource locator*)

5 Conventions

Dans la présente Recommandation, les conventions suivantes s'appliquent:

– la forme "doit/doivent" indique une disposition obligatoire;

– la forme "devrait/devraient" indique une mesure suggérée mais facultative;

– la forme "peut/peuvent" indique une action possible plutôt qu'une action recommandée.

Le profil de sécurité avec signature peut utiliser le **profil de sécurité pour le chiffrement vocal** de la Rec. UIT-T H.235.1 pour assurer la confidentialité vocale si nécessaire.

Les procédures II et III spécifient la manière d'implémenter les services de sécurité pour divers scénarios – bond par bond et de bout en bout – avec des mécanismes de sécurité différents tels que les techniques de cryptographie asymétrique (signature numérique).

Si le service d'intégrité des messages fournit toujours l'authentification des messages, l'inverse n'est pas toujours vrai. En mode d'authentification seulement, l'intégrité assurée porte uniquement sur un sous-ensemble donné de champs de messages. Cela s'applique aux services d'intégrité assurés par des moyens asymétriques (par exemple les signatures numériques). Donc, en pratique, le double service d'authentification et d'intégrité exploite les mêmes données de clé sans introduire de faiblesse au niveau de la sécurité.

Par ailleurs, toutes les informations de sécurité bond par bond sont introduites dans l'élément **CryptoSignedToken**. Ces informations sont recalculées à chaque bond, conformément à la procédure II.

Quant aux informations de sécurité de bout en bout (uniquement possibles en cas d'utilisation d'un proxy H.323 et de la procédure III), il s'agit essentiellement d'informations analogues à celles placées dans **CryptoSignedToken** mais elles sont enregistrées dans un autre jeton **CryptoToken** du message. Ces informations ne sont pas modifiées pendant le transit. Un identificateur d'objet distinct permet de faire la distinction entre les jetons **CryptoToken** bond par bond et de bout en bout.

Des techniques asymétriques à signatures numériques peuvent s'appliquer bond par bond et/ou de bout en bout.

6 Aperçu général

La présente Recommandation décrit un profil de sécurité facultatif visant à utiliser des signatures numériques pour sécuriser la signalisation H.225.0. Les entités de sécurité H.323 (terminaux, portiers, passerelles, ponts MCU, etc.) peuvent implémenter ce profil de sécurité avec signature pour améliorer la sécurité ou chaque fois que cela est nécessaire.

Le profil de sécurité avec signature, pour lequel le modèle à routage par portier est obligatoire, est fondé sur les techniques de tunnellisation H.245. La prise en charge de modèles autres que le modèle à routage par portier nécessite un complément d'étude.

Le profil de sécurité avec signature est applicable à la téléphonie IP "mondiale" évolutive; ce profil de sécurité n'est pas exposé aux limitations du profil de sécurité de base, simple, de la Rec. UIT-T H.235.1. Par exemple, le profil de sécurité avec signature ne dépend pas de l'administration des secrets partagés mutuels des bonds dans différents domaines. Il assure la tunnellisation des messages H.245 pour l'intégrité de ceux-ci et offre également la non-répudiation des messages. Le profil de sécurité avec signature offre ainsi une sécurité bond par bond ainsi que l'authentification vraie de bout en bout avec l'utilisation simultanée de proxys H.235 ou de portiers intermédiaires.

Les fonctionnalités offertes par ce profil sont les suivantes, pour les messages RAS, H.225.0 et H.245:

- authentification de l'utilisateur auprès de l'entité voulue, indépendamment du nombre de bonds au niveau application franchis par ce message;

  NOTE 1 – Par "bond", on entend dans le cas présent un élément de réseau H.235 de confiance (tel que portier, passerelle, pont MCU, proxy ou pare-feu). En conséquence, la sécurité bond par bond au niveau application, lorsqu'elle est utilisée avec des techniques symétriques, n'assure pas une sécurité vraie de bout en bout entre les terminaux.
- intégrité de tous les champs ou des champs critiques d'un message arrivant à une entité, indépendamment du nombre de bonds au niveau application franchis par le message. L'intégrité du message assurée au moyen d'un nombre aléatoire fort est proposée en option;
- l'authentification, l'intégrité et la non-répudiation d'un message bond par bond au niveau application couvrent la totalité du message;

- la non-répudiation d'un message échangé entre deux entités, indépendamment du nombre de bonds au niveau application franchis par le message, peut également être assurée. Plus précisément, la non-répudiation est assurée pour les champs critiques du message. Cela peut par exemple être le cas lorsqu'un point d'extrémité EP envoie un message SETUP à son portier et que ceux-ci (le point EP et le portier) sont séparés par un ou plusieurs proxys.

Plusieurs types d'attaque sont combattus au moyen des services de sécurité ci-dessus, utilisés de manière appropriée. Il s'agit:

- des attaques de type déni de service: un contrôle rapide des signatures numériques peut prévenir de telles attaques;
- des attaques par intercepteur: l'authentification et l'intégrité des messages bond par bond au niveau application protègent contre de telles attaques lorsque l'intercepteur, un routeur hostile par exemple, se trouve entre deux bonds au niveau application. Lorsque l'intercepteur est une entité au niveau application, de telles attaques sont empêchées par l'authentification de l'utilisateur de bout en bout et par l'intégrité de certains champs du message;
- des attaques par réexécution: l'emploi d'horodates et de numéros de séquence protège contre de telles attaques;
- des mystifications: l'authentification de l'utilisateur protège contre de telles attaques;
- du détournement de connexions: l'utilisation de l'authentification/intégrité pour chaque message de signalisation empêche de telles attaques.

Ce profil de sécurité est applicable dans les environnements pouvant avoir de nombreux terminaux et où l'attribution des mots de passe/clés symétriques n'est pas réalisable, par exemple dans un scénario à grande échelle, voire à l'échelle mondiale. Le profil de sécurité avec signature fournit des services de sécurité additionnels pour la non-répudiation en utilisant des signatures numériques et des certificats. Les signatures numériques, qui peuvent être fondées sur un hachage SHA1 ou MD5, permettent d'assurer l'authentification et/ou l'intégrité (voir les procédures II et III).

Les entités H.323 utilisant l'authentification et l'intégrité, ou l'authentification seulement, bond par bond doivent utiliser la procédure II. Les entités H.323 n'utilisant que l'authentification seulement n'implémentent pas l'intégrité. Les entités H.323 utilisant l'authentification seulement doivent utiliser la procédure III pour réaliser l'authentification vraie de bout en bout.

La présente Recommandation permet d'assurer une protection de l'intégrité couvrant la totalité de chaque message. Pour un message RAS H.225.0, la protection d'intégrité couvre la totalité du message RAS; pour un message de signalisation d'appel, cette protection couvre la totalité du message de signalisation d'appel H.225.0 y compris les en-têtes Q.931.

Le profil de sécurité avec signature permet de tunnelliser, en toute sécurité, les unités PDU de commande d'appel H.245 dans des messages facility H.225.0. Les mécanismes de mise à jour et de synchronisation des clés H.245, qui sont par exemple utiles pour les très longues communications, nécessitent une tunnellisation.

NOTE 2 – La mise à jour des clés pour le codage de la parole G.711 sécurisé devrait intervenir au plus tard après la transmission de 2^{30} blocs de 64 bits, soit plus de 12 jours de conversation continue.

La zone hachurée verticalement – en bleu dans la version électronique – du Tableau 1 représente le domaine du profil de sécurité avec signature. Lorsqu'on omet l'intégrité, signalée dans la zone hachurée verticalement – en vert dans la version électronique, on obtient le profil de sécurité par authentification seulement. Une option dans le profil de sécurité avec signature consiste à faire un choix entre les signatures numériques RSA-SHA1 et RSA-MD5. Le profil de sécurité pour le chiffrement vocal de la Rec. UIT-T H.235.6 (voir § 6.1/H.235.6) peut facultativement être utilisé en combinaison avec le profil de sécurité avec signature.

Tableau 1/H.235.2 – Profil de sécurité avec signature

Services de sécurité	Fonctions d'appel			
	RAS	H.225.0	H.245 (Note)	RTP
Authentification	SHA1/ MD5	SHA1/ MD5	SHA1/ MD5	
	Signature numérique	Signature numérique	Signature numérique	
Non-répudiation	SHA1/ MD5	SHA1/ MD5	SHA1/ MD5	
	Signature numérique	Signature numérique	Signature numérique	
Intégrité	SHA1/ MD5	SHA1/ MD5	SHA1/ MD5	
	Signature numérique	Signature numérique	Signature numérique	
Confidentialité				
Contrôle d'accès				
Gestion de clés	Attribution d'un certificat	Attribution d'un certificat		

NOTE – Message H.245 tunnellisé ou message H.245 imbriqué dans le cadre de la connexion rapide H.225.0.

NOTE 3 – Le profil de sécurité avec signature doit aussi être pris en charge par d'autres entités H.235 (telles que portiers, passerelles et proxys H.235).

NOTE 4 – Les bits d'utilisation de clé disponibles dans le certificat peuvent également déterminer le service de sécurité fourni par un terminal (par exemple: non-répudiation déclarée par assertion).

Pour l'authentification, l'utilisateur devrait utiliser un système de signature à clé publique ou privée. Un tel système offre généralement une intégrité et une non-répudiation meilleures de l'appel.

La présente Recommandation **ne** définit **pas** de procédure:

- pour l'enregistrement, la certification et l'attribution de certificat à partir d'un centre de confiance, ni pour l'attribution de clés privées/publiques, pour les services d'annuaire, les paramètres CA spécifiques, la révocation de certificats, la mise à jour/récupération de paires de clés, ni d'autres procédures d'exploitation ou de gestion des certificats, par exemple la remise de certificats ou de clés publiques/privées et de certificats ainsi que l'installation dans les terminaux.

De telles procédures peuvent être exécutées par des moyens qui ne font pas partie de la présente Recommandation.

Les entités de communication concernées ont la capacité de déterminer implicitement l'utilisation du profil de sécurité de base H.235.1 ou de ce profil de sécurité avec signature en évaluant les identificateurs d'objet de sécurité signalés dans les messages (**tokenOID** et **algorithmOID**; voir également § 20).

Les procédures destinées à être utilisées dans ce profil sont les suivantes:

la procédure II est fondée sur des signatures numériques au moyen d'une paire de clés privée/publique pour assurer l'authentification, l'intégrité et la non-répudiation des messages RAS, Q.931 et H.245. Les terminaux peuvent utiliser cette méthode si la non-répudiation et une intégrité élaborée sont requises.

Selon la politique de sécurité, l'authentification peut être unilatérale ou bilatérale, l'authentification/intégrité étant alors appliquée dans les deux sens, ce qui accroît la sécurité. La politique de sécurité d'un terminal peut permettre de procéder à une authentification seulement sans calculer l'intégrité cryptographique (voir § 9).

Lorsque les portiers détectent un échec de validation de l'authentification et/ou de l'intégrité dans un message RAS ou un message de signalisation d'appel reçu d'un terminal ou d'un portier homologue, ils répondent par un message de rejet correspondant indiquant l'absence de sécurité en mettant le motif de rejet à **securityDenial** ou tout autre code d'erreur de sécurité approprié, conformément au § 11.1/H.235.0. En fonction de sa capacité à reconnaître des attaques et de la façon la plus appropriée de réagir à ces attaques, un portier qui reçoit un message **xRQ** sécurisé contenant des identificateurs d'objet non définis (**tokenOID**, **algorithmOID**) devrait répondre par un message **xRJ** non sécurisé ou peut ignorer le message. L'événement de sécurité rencontré devrait être journalisé. Par ailleurs, le point d'extrémité doit éliminer le message non sécurisé reçu, temporiser et peut ensuite procéder à un nouvel essai en envisageant de choisir des identificateurs OID différents. De même, un portier qui reçoit un message sécurisé SETUP H.225.0 contenant des identificateurs d'objet non définis (**tokenOID**, **algorithmOID**) devrait répondre par un message non sécurisé RELEASE COMPLETE avec le motif **securityDenied** ou peut éliminer ce message. L'événement de sécurité rencontré devrait être aussi journalisé.

Une signalisation H.235 implicite permet d'indiquer l'utilisation de la procédure II et du mécanisme de sécurité appliqué, sur la base de la valeur des identificateurs d'objet (voir également § 20) et du contenu des champs de message. Les identificateurs d'objets sont désignés symboliquement par des lettres (par exemple "A") dans le présent texte.

Ce profil n'utilise pas les champs ICV H.235; en effet, les valeurs de contrôle d'intégrité cryptographique sont placées dans le champ **signature** du jeton **token** du **cryptoSignedToken**.

6.1 Prescriptions H.323

Les entités H.323 qui implémentent ce profil avec signature sont supposées prendre en charge les caractéristiques H.323 suivantes:

- la connexion rapide;
- le modèle à routage par portier.

7 Signatures numériques avec paires de clés publiques/privées (procédure II)

Il est nécessaire de se conformer aux procédures suivantes si la procédure II est utilisée pour la sécurité bond par bond:

- il convient d'utiliser l'algorithme SHA1 ou MD5 avec l'algorithme RSA pour produire la signature numérique. A cet égard, la conformité aux systèmes PKCS n° 1 et PKCS n° 7 favorise l'interopérabilité.

  Le champ **CryptoH323Token** de chaque message RAS/H.225.0 doit contenir les champs suivants:

  - **nestedCryptoToken** contenant un **CryptoToken** qui contient à son tour le champ **cryptoSignedToken** avec les champs suivants:
    - **tokenOID** mis à:
      - "A" pour indiquer que le calcul d'authentification/intégrité englobe tous les champs du message H.225.0 RAS ou de signalisation d'appel (voir § 11);
      - "B" pour indiquer que le calcul d'authentification/intégrité englobe seulement un sous-ensemble de champs (voir § 10) du message RAS/H.225.0 pour l'authentification seulement;

- **token** contenant les champs:
  - **toBeSigned** contenant le champ **EncodedGeneralToken**, qui est en fait un **ClearToken** avec les champs suivants:
    - **tokenOID** mis à "S" pour indiquer que **ClearToken** est en cours d'utilisation pour l'authentification/intégrité/non-répudiation d'un message;
    - **timeStamp** contenant l'horodate;
    - **random** contenant un numéro de séquence croissant monotone;
    - **generalID** contenant l'identificateur du destinataire (seulement en cas de messages envoyés à un seul destinataire);
    - **sendersID** contenant l'identificateur de l'expéditeur;
    - **dhkey**, utilisé pour transmettre les paramètres Diffie-Hellman comme défini dans la présente Recommandation au cours de **Setup** à **Connect**:
      - **halfkey** contenant la clé publique aléatoire de l'un des correspondants;
      - **modsize** contenant le nombre **premier** DH (voir Tableau 4/H.235.6);
      - **generator** contenant le groupe DH (voir Tableau 4/H.235.6).

      NOTE 1 – Lorsque le profil de sécurité avec signature est utilisé sans le profil de sécurité pour le chiffrement vocal, aucun paramètre Diffie-Hellman ne devrait être envoyé et **dhkey** devrait être absent; les champs **halfkey**, **modsize** et **generator** peuvent être mis à {'0'B,'0'B,'0'B}.
    - **certificate** contenant le certificat numérique de l'expéditeur dans lequel **type** indique le type de certificat ("V" pour des certificats MD5-RSA ou "W" pour des certificats SHA1-RSA) et **certificate** achemine le certificat proprement dit (voir § 14).
    - **algorithmOID** mis à:
      - "V" pour indiquer l'utilisation de la signature MD5-RSA;
      - "W" pour indiquer l'utilisation de la signature SHA1-RSA.
    - **params** mis à NULL.
    - **signature** contenant la signature calculée au moyen de l'algorithme SHA1 ou MD5 RSA sur l'ensemble des champs (si **tokenOID** vaut "A", voir § 11) ou certains champs critiques (si **tokenOID** vaut "B", voir § 10) du message H.225.0 RAS ou de signalisation d'appel.

  Lorsque l'identificateur tokenOID "A" est utilisé pour la protection d'unités H323-UU-PDU tunnellisées comprenant tout le contenu du message H.245, le calcul des signatures doit être fait sur la totalité du message H.225.0 de signalisation d'appel avec l'ensemble des champs conformément à la procédure décrite au § 11. Lorsque l'identificateur tokenOID "B" est utilisé, l'authentification seulement de **CryptoToken** est réalisée par l'application de la procédure III (voir § 10).
- L'entité à laquelle la signature est destinée (elle peut être distante d'un ou de plusieurs bonds au niveau application) vérifie cette signature.

NOTE 2 – Le destinataire a la capacité de détecter l'utilisation de la procédure II en évaluant l'identificateur **algorithmOID** dans le jeton de **cryptoSignedToken** (par détection de la présence de "V" ou de "W").

8 Procédures de conférence multipoint

Les ponts MCU doivent prendre en charge la distribution sécurisée des certificats à la demande des terminaux par les commandes H.245 **ConferenceRequest** et **ConferenceResponse** tunnellisées, comme indiqué au § 8.8.1/H.235.6. Cela permet aux terminaux de demander des certificats à d'autres terminaux dans un contexte de conférence multipoint et d'obtenir ainsi avec certitude l'identité des autres participants à la conférence.

ConferenceRequest achemine **requestTerminalCertificate** avec les champs suivants:

- **terminalLabel**: utilisé comme moyen d'adressage du terminal distant via le pont MCU;
- **certSelectionCriteria**: l'expéditeur peut demander uniquement des certificats de types donnés;
- **sRandom**: épreuve aléatoire produite par l'expéditeur demandeur.

ConferenceResponse achemine **terminalCertificateResponse** avec les champs suivants:

- **terminalLabel**: permet d'associer le certificat renvoyé avec le terminal
- **CertificateResponse**: achemine la réponse du pont MCU avec les champs suivants:
  - **terminalLabel**: identification du terminal distant
  - **certificateResponse**: il s'agit en fait d'une chaîne d'octets codée ASN.1 à partir de **EncodedReturnSig** comme suit:
    - **generalID**: identification du terminal de destination;
    - **responseRandom**: valeur d'épreuve aléatoire produite par le pont MCU;
    - **requestRandom**: **sRandom** reproduit:
    - **certificate**: achemine le certificat renvoyé dans lequel **type** indique le type de certificat sous forme d'identificateur OID et **certificate** achemine le certificat numérique (voir § 14).

9 Authentification de bout en bout (procédure III)

La Figure 1 représente un scénario dans lequel des proxys séparent les portiers GK et les points EP et dans lequel deux jetons CryptoToken différents sont utilisés pour l'authentification bond par bond ainsi que pour l'authentification de bout en bout et/ou l'intégrité bond par bond. Le jeton **CryptoToken** pour l'authentification bond par bond s'applique uniquement au tronçon compris entre deux entités et doit être recalculé pour chaque nouveau tronçon. Par ailleurs, le jeton **CryptoToken** pour l'authentification de bout en bout est produit une seule fois par le point d'extrémité expéditeur et n'est pas modifié pendant le transit par les nœuds intermédiaires. Ceux-ci peuvent valider des signatures et des certificats acheminés dans des jetons **CryptoTokens** de bout en bout et devraient retransmettre le jeton **CryptoToken** en transit.

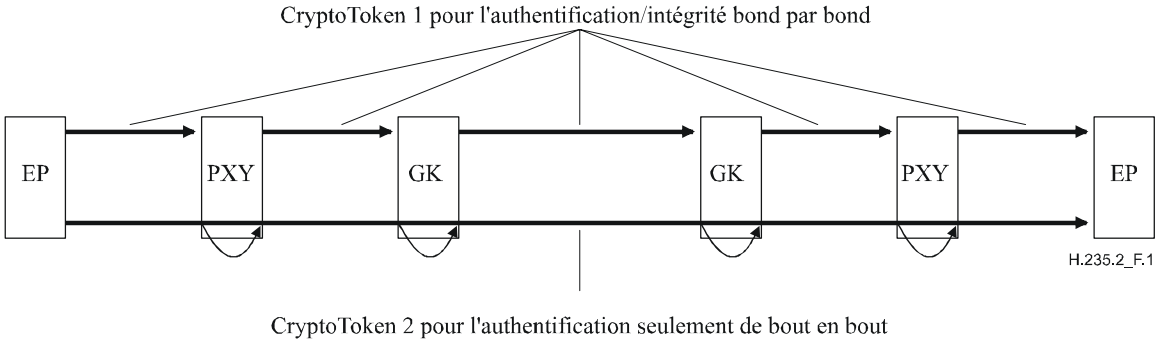

Figure 1/H.235.2 – Utilisation simultanée de la sécurité bond par bond et de l'authentification de bout en bout

NOTE 1 – Le proxy peut être un nœud de réseau distinct comme indiqué sur la Figure 1 ou peut être situé au même endroit que la fonctionnalité d'une entité H.323, par exemple faire partie du portier.

NOTE 2 – Selon l'identificateur **tokenOID** signalé, le proxy a la capacité de déterminer si le jeton **CryptoToken** reçu est destiné au proxy ("S") ou à un autre destinataire ("R").

NOTE 3 – Vu que les entités intermédiaires modifient le contenu du message de signalisation pour chaque tronçon, l'intégrité de bout en bout n'est pas possible.

Pour assurer une authentification vraie de bout en bout, lorsque les points d'extrémité sont séparés par des proxys H.323 et des éléments de réseau intermédiaires, le point d'extrémité/terminal expéditeur doit calculer une signature numérique de la manière suivante:

Le champ **CryptoH323Token** de chaque message RAS/H.225.0 doit contenir les champs suivants:

- **nestedCryptoToken** contenant un **CryptoToken** qui lui-même contient **cryptoSignedToken** avec les champs suivants:
  - **tokenOID** mis à:
    - "A" pour indiquer que le calcul de l'authentification/intégrité bond par bond englobe tous les champs du message RAS/H.225.0 (voir § 11);
    - "B" pour indiquer que le calcul d'authentification englobe uniquement un sous-ensemble de champs (voir § 10) du message H.225.0 RAS ou de signalisation d'appel en vue de l'authentification seulement.
- **token** contenant les champs:
  - **toBeSigned** contenant le champ **ClearToken** utilisé avec les champs suivants:
    - **tokenOID** mis à "R" pour indiquer que **ClearToken** est en cours d'utilisation pour l'authentification seulement/non-répudiation de bout en bout;

      NOTE 4 – Le service de sécurité qui sera effectivement appliqué dépend également des bits d'utilisation de la clé dans le certificat.
    - **random** contenant un numéro de séquence croissant monotone;
    - **timeStamp**, facultativement pour une sécurité améliorée uniquement lorsque les entités terminales sont synchronisées;
    - **generalID** contenant l'identificateur de point d'extrémité du destinataire (uniquement pour un message destiné à un seul destinataire). Dans le cas bond par bond, il s'agit de l'identificateur du bond suivant; dans le cas de bout en bout, il s'agit de l'identificateur du point d'extrémité distant;
    - **sendersID** contient l'identificateur de point d'extrémité de l'expéditeur;

- **certificate** contenant le certificat numérique de l'expéditeur, où **type** indique le type de certificat ("V" pour un certificat MD5-RSA ou "W" pour un certificat SHA1-RSA) et **certificate** achemine le certificat proprement dit (voir § 14);
- **dhkey**, utilisé pour acheminer les paramètres Diffie-Hellman spécifiés dans la présente Recommandation au cours de **Setup** à **Connect**:
  - **halfkey** contenant la clé publique aléatoire de l'un des correspondants;
  - **modsize** contenant le nombre premier DH (voir Tableau 4/H.235.6);
  - **generator** contenant le groupe DH (voir Tableau 4/H.235.6).

NOTE 5 – Lorsque le profil de sécurité avec signature est utilisé sans le profil de sécurité pour le chiffrement vocal, aucun paramètre Diffie-Hellman ne devrait être envoyé et **dhkey** devrait être absent; **halfkey**, **modsize** et **generator** peuvent être mis à {'0'B,'0'B,'0'B}.

- **algorithmOID** mis à:
  - "V" pour indiquer l'utilisation de la signature MD5-RSA;
  - "W" pour indiquer l'utilisation de la signature SHA1-RSA.
- **params** mis à NULL;
- **signature** contenant la signature calculée au moyen de l'algorithme SHA1-RSA ou MD5-RSA sur l'ensemble des champs (si **tokenOID** vaut "A") ou seulement sur certains champs critiques (si **tokenOID** vaut "B") du message H.225.0 RAS ou de signalisation d'appel.

Le proxy peut vérifier toute signature numérique et/ou certificat reçu et peut ignorer le message s'il le juge inapproprié compte tenu de la politique locale, ou bien faire suivre le **CryptoToken** qu'il a reçu. Le proxy doit produire de nouveaux éléments d'information de signalisation H.235 pour la sécurité bond par bond conformément à la procédure II ou III.

L'entité qui termine le tronçon – un terminal, par exemple – devrait vérifier les informations de sécurité reçues dans le **CryptoToken** et, selon la présence d'éléments de sécurité de bout en bout, peut évaluer aussi l'information **CryptoToken** de bout en bout. Les procédures de vérification exactes à faire dans un terminal ou une entité H.323 intermédiaire peuvent varier en fonction de la politique locale.

10 Authentification seulement

Les terminaux peuvent décider d'implémenter l'authentification seulement (en utilisant l'identificateur OID "B"). Dans ce cas, l'authentificateur n'est calculé que sur un sous-ensemble (**ClearToken** de **CryptoToken**) du message RAS/H.225.0. L'authentification seulement peut être utile pour l'authentification vraie de bout en bout (voir § 9). Les champs suivants de la structure **ClearToken** sont utilisés en tant que sous-ensemble:

- **tokenOID**: identificateur d'objet jeton séparé (tokenOID "B") pour l'implémentation de l'authentification seulement.
- **random**: numéro de séquence croissant monotone.
- **timeStamp**: horodate.
- **generalID**: identificateur du destinataire (uniquement pour un message destiné à un seul destinataire). Dans le cas bond par bond, il s'agit de l'identificateur du bond suivant; dans le cas de bout en bout, il s'agit de l'identificateur du point d'extrémité distant.
- **sendersID**: identificateur de l'expéditeur.
- **dhkey**: paramètres Diffie-Hellman. Ce champ et ses sous-champs sont utilisés uniquement au cours des messages **Setup** à **Connect**.

L'authentificateur est calculé sur **ClearToken** à l'intérieur de **EncodedGeneralToken** (c'est-à-dire **ClearToken**) du champ **token** de **cryptoSignedToken**. La signature numérique sera calculée sur la chaîne binaire codée ASN.1 de **ClearToken**. Avant de calculer la signature numérique, le champ **tokenOID** de **ClearToken** doit être mis à {0 0}.

11 Authentification et intégrité

La procédure pour l'authentification et l'intégrité du message sur l'ensemble des champs de message codés ASN.1 (identificateur OID "A") est la suivante:

l'expéditeur du message doit calculer la signature de la manière suivante:

1) mettre la valeur de signature à une séquence par défaut spécifique de longueur fixe (par exemple 1024 bits). Dans cette étape, il faut réserver de l'espace pour la longueur maximale possible d'une signature numérique, compte tenu d'un certificat donné. La séquence binaire exacte importe peu, mais il est préférable de choisir une séquence qui ne survient pas dans le reste du message;
2) coder l'ensemble du message en ASN.1; pour un message RAS, cette opération doit porter sur la totalité du message H.225.0 RAS; pour un message de signalisation d'appel, cette opération doit porter sur la totalité du message de signalisation d'appel H.225.0.
3) localiser la séquence par défaut dans le message codé, annuler la séquence binaire trouvée et la remplacer entièrement par des bits zéro;

   NOTE 1 – Cela peut sous-entendre quelques essais et quelques erreurs au cas, très rare, où la séquence par défaut survient plusieurs fois dans le message.
4) calculer la signature numérique à partir du message codé ASN.1 par la méthode indiquée par **algorithmOID** à savoir "V" ou "W" (voir § 12);
5) substituer la séquence par défaut dans le message codé par la valeur de la signature numérique calculée. Si la signature numérique est plus courte que l'espace réservé, des zéros sont placés devant les bits de plus fort poids de la valeur de signature.

Le destinataire qui reçoit le message procède alors de la manière suivante:

1) décoder le message ASN.1;
2) extraire la valeur de la signature numérique reçue et la conserver dans une variable locale SV;
3) rechercher et localiser la valeur de signature SV dans le message codé reçu;

   NOTE 2 – Dans le cas rare où la sous-chaîne de la valeur de signature survient plusieurs fois dans l'ensemble du message, il convient d'itérer les étapes 3 à 6 avec des positions de départ de la recherche différentes.
4) annuler la séquence binaire du message codé et la remplacer entièrement par des zéros;
5) calculer la signature numérique à partir du message codé ASN.1 par la méthode indiquée par **algorithmOID** à savoir "V" ou "W" (voir § 12);
6) comparer la valeur SV avec la valeur de signature calculée. Le message est considéré exempt d'erreur et authentique seulement si les deux valeurs de signature sont identiques; dans ce cas, l'authentification a abouti et la procédure s'arrête;
7) sinon, répéter les étapes 3 à 7 en recherchant d'autres concordances après avoir mis la variable SV à l'emplacement précédent. Si aucune des concordances ne donne une comparaison satisfaisante des valeurs de signature, l'authentification échoue et le message a été altéré (accidentellement ou intentionnellement) au cours du transit ou pour toute autre raison.

12 Calcul de la signature numérique

Au départ du processus de production de la signature numérique, il y une chaîne binaire codée ASN.1 ainsi que le résultat du processus de calcul du résumé de message et la clé privée du signataire. Les détails de la production de la signature numérique dépendent de l'algorithme de signature utilisé; le certificat détermine l'algorithme de signature qu'il convient d'appliquer; lorsque l'extension relative à l'utilisation de la clé figure dans le certificat, le bit **digitalSignature** doit être mis à 1 de manière à pouvoir utiliser la clé pour la signature. La valeur de signature produite par le signataire est codée sous forme d'une chaîne binaire et acheminée dans le champ **signature**.

Il faudra utiliser la méthode décrite dans [PKCS #1, section E.8.1.1] pour calculer une signature numérique de type RSA avec appendice (RSASSA-PKCS1-v1_5-SIGN) conjointement avec les procédures OS2IP, RSASP1, I2OSP et la méthode EMSA-PKCS1-v1_5-ENCODE.

13 Vérification de la signature numérique

Le départ du processus de vérification de la signature est le résultat du processus de calcul du résumé de message et la clé publique du signataire. Le destinataire peut obtenir la clé publique correcte du signataire par n'importe quel moyen, mais la méthode préférée est celle d'un certificat obtenu dans le champ **certificate** et ensuite validé au moyen du hachage du certificat du signataire. La validation de la clé publique du signataire peut être basée sur le traitement du trajet de certification (RFC 3280). Les détails de la vérification de la signature dépendent de l'algorithme de signature employé.

Il faudra utiliser la méthode décrite dans [PKCS #1, section E.8.1.2] pour vérifier une signature numérique de type RSA avec appendice (RSASSA-PKCS1-v1_5-VERIFY) conjointement avec les procédures OS2IP, RSAVP1, I2OSP et la méthode EMSA-PKCS1-v1_5-ENCODE.

14 Traitement des certificats

Pour la vérification des signatures numériques, l'entité de réception doit avoir accès au certificat de l'expéditeur, qui est signé par une autorité de certification reconnue (CA, *certification authority*). Le destinataire dispose de plusieurs possibilités pour accéder au certificat de l'expéditeur:

- le certificat est inclus dans l'échange de messages comme décrit dans les procédures II et III; dans ce cas, **certificate** contient le certificat réel et **type** contient l'identificateur OID "V" ou "W";
- le destinataire connaît le certificat; celui-ci a éventuellement été enregistré localement lors d'un échange antérieur;
- plutôt que d'inclure le certificat proprement dit, l'expéditeur fournit une adresse URL où le certificat peut être trouvé. A cet effet, **certificate** contient l'URL et **type** est mis à l'identificateur OID "P";
- le destinataire obtient le certificat par un autre moyen qui ne relève pas de la présente Recommandation (par exemple consultation d'annuaire par protocole LDAP).

Chaque fois qu'un certificat numérique est acheminé dans un message, l'entité réceptrice (portier, point d'extrémité) doit vérifier que l'identité de l'expéditeur (portier, point d'extrémité) correspond bien à l'identité du certificat afin d'empêcher toute attaque par intercepteur.

Pour les messages à signature numérique envoyés du portier au point d'extrémité, celui-ci dispose de différentes possibilités pour vérifier l'identité du portier:

– si le nom d'hôte est disponible, par exemple dans l'attribut de noms communs du champ subjet ou du champ subjectAltName du certificat, le point d'extrémité peut vérifier que ce nom d'hôte correspond bien à l'identificateur de portier. En outre, le point d'extrémité peut

interroger le système DNS pour obtenir l'adresse IP associée et vérifier s'il s'agit bien de l'adresse IP du portier telle que présentée dans le message de réponse signé du portier;

– par exemple, l'identificateur de portier peut être élaboré à partir de l'adresse IP (représentée comme une valeur sur 4 octets dans l'ordre des octets réseau) concaténée avec une autre information d'identification de l'identificateur de portier, puis être tronqué à la longueur maximale du champ sendersID, qui achemine l'identité du portier. Le point d'extrémité peut en outre vérifier que l'adresse IP appartenant au nom d'hôte correspond bien à l'adresse IP présentée dans l'en-tête IP de la réponse du portier;

  NOTE – Cette méthode ne fonctionne pas comme prévu en présence de dispositifs NAT (*network address translation*).

– si le nom d'hôte n'est pas disponible dans le certificat, l'adresse IP – qui devrait faire partie du certificat (*iPAddress subjectAltName*) – doit être extraite directement pour effectuer les vérifications indiquées ci-dessus.

Les utilisateurs devraient examiner soigneusement le certificat présenté par le portier pour déterminer s'il répond à leurs attentes. Si le point d'extrémité dispose d'informations externes concernant l'identité attendue du portier, la vérification du nom d'hôte peut être omise. Donnons l'exemple d'un point d'extrémité qui se connecte à un portier dont l'adresse et le nom d'hôte sont dynamiques, mais qui connaît le certificat que le portier présentera. En pareil cas, il est important de réduire la portée des certificats acceptables autant que possible afin d'éviter des attaques par intercepteur. Dans certains cas, il peut être utile que le point d'extrémité ignore simplement l'identité du portier, mais il faut bien comprendre que cela laisse la connexion ouverte à des attaques actives.

Si le nom d'hôte ne correspond pas à l'identité du certificat, les points d'extrémité orientés vers l'utilisateur doivent soit en notifier l'utilisateur (les points d'extrémité peuvent donner à l'utilisateur la possibilité de continuer avec la connexion en tout état de cause) soit mettre fin à la connexion avec une erreur de type certificat erroné. Des points d'extrémité automatisés doivent journaliser l'erreur dans un journal d'audit approprié (s'il en existe un) et mettre fin à la connexion (avec une erreur de type certificat erroné).

Les points d'extrémité automatisés peuvent disposer d'une commande permettant de désactiver cette vérification mais doivent disposer d'une commande permettant de l'activer.

De même, il est recommandé que le portier exécute un contrôle d'identité pour tout message à signature numérique qu'un point d'extrémité lui envoie. Les modalités exactes d'implémentation d'une telle vérification par le portier sont considérées comme étant une question locale et dépendent de l'implémentation de la politique de sécurité du portier. Par exemple, on peut imaginer que le nom d'utilisateur acheminé dans le certificat peut également faire partie de l'identificateur H.323. Par la suite, le portier peut effectuer une contre-vérification de cette information d'identité avec les données d'utilisateur localement administrées/configurées, si celles-ci sont disponibles, et peut fonder une décision de politique à partir du résultat.

Si le portier dispose d'informations externes concernant l'identité attendue du point d'extrémité, le contrôle du nom d'hôte peut être omis. Donnons l'exemple d'un portier qui se connecte à un point d'extrémité dont l'adresse est le nom d'hôte sont dynamiques, mais qui connaît le certificat que le point d'extrémité va présenter. En pareil cas, il est important de réduire la portée des certificats acceptables autant que possible afin d'éviter des attaques par intercepteur. Dans certains cas, il peut être utile que le portier ignore simplement l'identité du point d'extrémité, mais il faut bien comprendre que cela laisse la connexion ouverte à des attaques actives.

Si le nom d'hôte ne correspond pas à l'identité figurant dans le certificat, le portier doit journaliser l'erreur dans un journal d'audit approprié (s'il en existe un) et doit mettre fin à la connexion (avec une erreur de type certificat erroné).

Si une extension subjectAltName du type dNSName est présente, cette extension doit être utilisée comme identité. Dans les autres cas, le champ Common Name (le plus spécifique) dans le champ Subject du certificat doit être utilisé. Bien que l'utilisation de Common Name est une pratique actuelle, elle est déconseillée et les autorités de certification sont encouragées à utiliser le nom dNSName.

Le contrôle doit être exécuté en utilisant les règles de concordance spécifiées dans la norme RFC 3280. Si plusieurs identités d'un type donné sont présentes dans le certificat (par exemple plusieurs noms dNSName), la concordance avec l'un quelconque des éléments de l'ensemble est considérée comme acceptable. Les noms peuvent contenir le caractère générique * qui est censé correspondre à une seule composante quelconque de nom de domaine ou à un seul fragment de composante. Par exemple, *.a.com englobe foo.a.com mais pas bar.foo.a.com. f*.com englobe foo.com mais pas bar.com.

Les procédures II et III offrent des moyens pour acheminer un certificat numérique. Pour des raisons d'efficacité, les certificats numériques des entités doivent être transmis au maximum une seule fois, à moins qu'ils ne soient déjà disponibles dans les entités par d'autres moyens qui ne relèvent pas de la présente Recommandation. L'échange de certificats devrait donc survenir uniquement au début de l'établissement d'une communication: pour la procédure RAS, il se produit durant la découverte du portier ou, si cette phase est omise, pendant l'enregistrement du portier. Il en est de même pour la procédure de connexion rapide, où le certificat peut être inclus dans les messages de signalisation d'appel initiaux mais peut être omis en toute sécurité dans les messages de signalisation d'appel ultérieurs.

Pour ce profil de sécurité, il faut utiliser le certificat X.509v3 (1997). D'autres formats de certificat feront l'objet d'un complément d'étude.

15 Exemple d'utilisation de la procédure II

Considérons le cas de la Figure 2 dans lequel chaque entité a sa propre paire de clés publique-privée/son propre certificat. Une entité peut également disposer de plusieurs paires de clés. Sur la figure, un proxy H.323 sépare le point EP1 du portier GK1.

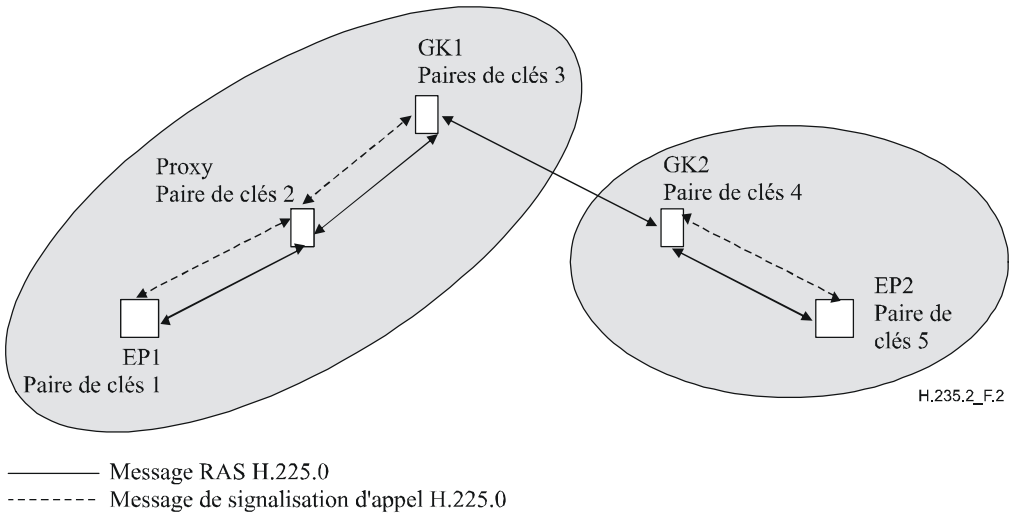

Figure 2/H.235.2 – Exemple de l'utilisation de clés publiques dans un modèle routé de portier à portier

Le proxy H.323 a un comportement double: d'une part, il termine l'authentification et l'intégrité sur chacun de ses tronçons. De manière analogue à ce qui est décrit dans la procédure I de la Rec. UIT-T H.235.1, il ajoute activement, dans les messages RAS sortants, les informations d'authentification/intégrité venant d'être calculées. Par ailleurs, le proxy laisse passer les informations de sécurité de bout en bout sans modification. Il peut toutefois vérifier les certificats et/ou signatures numériques reçus en transit.

Les détails des procédures pour l'authentification, l'intégrité et la non-répudiation des messages RAS, de signalisation d'appel H.225.0 et H.245 sont présentés ci-après.

15.1 Authentification, intégrité et non-répudiation des messages RAS

Considérons le cas d'une communication bond par bond dans laquelle le point EP1 souhaite envoyer un message RAS – par exemple un message **ARQ –** au portier GK1. Le point EP1 produit une horodate et un numéro de séquence, qu'il inclut respectivement dans les champs **timeStamp** et **random**, avec le pseudonyme du proxy dans le champ **generalID** et l'identificateur du point EP1 dans **sendersID**. Ces champs sont présents dans le champ **ClearToken** de **EncodedGeneralTokens** présent dans le **token** de **cryptoSignedToken** du champ **CryptoToken** de **cryptoH323Token** du message **ARQ**. Ce **cryptoH323Token** est l'un des (pour le moins) nombreux jetons de la séquence **cryptoTokens**. Le champ **tokenOID** de **cryptoSignedToken** est mis à "A" pour indiquer que tous les champs du message **ARQ** sont signés. Le **token** de **cryptoSignedToken** a son champ **algorithmOID** mis à "V" pour indiquer l'utilisation de l'algorithme MD5-RSA ou à "W" pour indiquer l'utilisation de l'algorithme SHA1-RSA, et le champ **params** mis à NULL. Le point EP1 calcule ensuite la signature sur la base de l'algorithme de signature en question en utilisant sa propre clé privée. La signature est calculée sur l'ensemble des champs du message **ARQ** lorsque **tokenOID** est mis à "A". Le point EP1 place la signature calculée dans **signature** du champ **token** du champ **cryptoSignedToken** de **CryptoToken** qui se trouve dans **cryptoH323Token** du message **ARQ**, et place son certificat dans le champ **certificate**.

De manière analogue, pour une communication de bout en bout passant par un proxy, le point EP1 produit un autre **CryptoToken** contenant une signature numérique qui couvre certains champs critiques (voir § 9) dans **ClearToken** du message **ARQ**. Le champ **tokenOID** de **CryptoSignedToken** est mis à "B" pour indiquer l'authentification seulement de ce **ClearToken**; il met le champ **tokenOID** de **ClearToken** à "R" pour indiquer l'authentification de bout en bout, remplit les champs **timeStamp**, **random**, **sendersID**, **generalID** et, s'il s'agit d'un message **SETUP/CONNECT,** également le champ **dhkey**, ainsi que les champs suivants dans **token**: **algorithmOID** à "V" ou "W" pour indiquer l'algorithme de signature, **params** à NULL et **signature** à la signature numérique calculée à partir des champs **ClearToken**. Le champ **certificate** achemine le certificat numérique du point EP1. Le message **ARQ** est ensuite envoyé au proxy.

Lorsqu'il reçoit le message **ARQ**, le proxy vérifie la signature des jetons qui lui sont adressés (dans ce cas, par exemple, ceux ayant le **tokenOID** "A") sur la base de plusieurs critères, notamment:

- l'actualité de l'horodate et l'unicité de **random**;
- l'identité de **generalID** et son propre identificateur;
- les autorisations d'accès pour **sendersID**;
- la concordance de la signature du message **ARQ** et de celle calculée par le portier GK1;
- la vérification des paramètres Diffie-Hellman, par exemple vérifier si le nombre premier à 1024 bits et le générateur sont corrects. La vérification des paramètres DH est une opération longue qui n'aura lieu que si la politique locale l'exige;
- la vérification du certificat reçu.

Si la vérification de la signature est positive, le proxy calcule une nouvelle signature qu'il substitue à l'ancienne dans le message **ARQ** avant d'envoyer celui-ci au portier GK1 de la manière suivante: le proxy remplace les champs **timeStamp**, **random**, **sendersID** et **generalID** de **ClearToken** (**toBeSigned**) par des valeurs s'appliquant au tronçon compris entre le proxy et le portier GK1. Le champ **timeStamp** contient l'horodate courante, le champ **random** contient le numéro de séquence croissant monotone suivant pour le tronçon proxy-portier GK1, le champ **sendersID** contient l'identité du proxy et le champ **generalID** contient le pseudonyme du portier GK1. Le proxy calcule ensuite une nouvelle signature pour ce message **ARQ** en utilisant sa clé privée et l'algorithme de signature, l'introduit dans **signature** de **token** et ajoute son certificat dans le champ **certificate**. Le proxy introduit aussi le **CryptoToken** de bout en bout reçu et son **ClearToken** dans le nouveau message sortant et transmet le message **ARQ** au portier GK1. La signature calculée par le point EP1 sur la base d'une sélection de champs du message **ARQ** (**tokenOID** de "B") et qui n'était pas destinée au proxy est également transmise sans modification dans le message **ARQ** au portier GK1.

Lorsqu'il reçoit le message **ARQ**, le portier GK1 vérifie les signatures, calcule une nouvelle signature après avoir modifié les champs **ClearToken** de **toBeSigned** comme il convient, l'introduit dans le champ **signature**, ajoute son certificat dans le champ **certificate** et transmet le message **Setup** au portier EP2. Ici aussi, le portier GK1 devrait envoyer toute information de bout en bout reçue dans les **CryptoToken** séparés au portier homologue GK2 en plaçant ces informations sans les modifier dans un **CryptoToken** séparé.

15.2 Authentification seulement des messages RAS

Considérons le cas d'une communication bond par bond dans laquelle le point EP1 souhaite envoyer un message RAS – par exemple un message **ARQ** – au portier GK1. Le point EP1 produit une horodate et un numéro de séquence, qu'il inclut respectivement dans les champs **timeStamp** et **random**, avec le pseudonyme du proxy dans le champ **generalID** et l'identificateur du point EP1 dans **sendersID**. Ces champs sont présents dans le champ **ClearToken** de **toBeSigned** présent dans le **token** de **cryptoSignedToken** du champ **CryptoToken** de **cryptoH323Token** du message **ARQ**. Le champ **tokenOID** de **cryptoSignedToken** est mis à "B" pour indiquer que seul le sous-ensemble spécifié de champs du message de **ClearToken** est signé. Le **token** de **cryptoSignedToken** a son champ **algorithmOID** mis à "V" pour indiquer l'utilisation de l'algorithme MD5-RSA ou à "W" pour indiquer l'utilisation de l'algorithme SHA1-RSA, et le champ **Params** mis à NULL. Le point EP1 calcule ensuite la signature sur la base de l'algorithme de signature en question en utilisant sa clé privée. La signature est calculée sur les champs **ClearToken** spécifiés du message **ARQ**. Le point EP1 place la signature calculée dans **signature** du champ **token** du champ **cryptoSignedToken** de **CryptoToken** qui se trouve dans **cryptoH323Token** du message **ARQ**, et ajoute son certificat dans le champ **certificate**.

D'une manière analogue, le point EP1 produit une autre signature numérique pour l'authentification de bout en bout qui couvre certains champs **ClearToken** dans un **CryptoToken** distinct du message **ARQ**. Cette signature numérique (identifiée par le **tokenOID** "V" ou "W") est incluse. Le message **ARQ** est ensuite envoyé au proxy.

Lorsqu'il reçoit le message **ARQ**, le proxy vérifie la signature des jetons qui lui sont adressés (dans ce cas, par exemple, ceux ayant le **tokenOID** "B") sur la base de plusieurs critères, notamment:

- l'actualité de l'horodate et l'unicité de **random**;
- l'identité de **generalID** et son propre identificateur;
- les autorisations d'accès pour **sendersID**;
- la concordance de la signature du message **ARQ** et de celle calculée par le portier GK1;
- la vérification du certificat reçu.

Si la vérification de la signature est positive, le proxy calcule une nouvelle signature qu'il substitue à l'ancienne dans le message **ARQ** avant d'envoyer celui-ci au portier GK1 de la manière suivante: le proxy remplace les champs **timeStamp**, **random, sendersID** et **generalID** de **ClearToken** de **toBeSigned** par des valeurs s'appliquant au tronçon compris entre le proxy et le portier GK1. Le champ **timeStamp** contient l'horodate courante, le champ **random** contient le numéro de séquence croissant monotone suivant pour le tronçon proxy-portier GK1, et le champ **generalID** contient le pseudonyme du portier GK1. Le proxy calcule ensuite une nouvelle signature pour ce **ClearToken** en utilisant sa clé privée et l'algorithme de signature MD5-RSA ou SHA1-RSA (**algorithmOID** à "V" ou à "W"), l'introduit dans **signature** de **token** de **cryptoSignedToken**, ajoute son certificat dans le champ **certificate** et transmet le message **ARQ** au portier GK1. La signature calculée par le point EP1 sur la base d'une sélection de champs **ClearToken** du message **ARQ** (**tokenOID** de "B") et qui n'était pas destinée au proxy est également transmise sans modification dans le message **ARQ** au portier GK1.

Lorsqu'il reçoit le message **ARQ**, le portier GK1 vérifie la signature, calcule une nouvelle signature après avoir modifié les champs **ClearToken** de **toBeSigned** comme il convient, l'introduit dans le champ **signature** et transmet le message **Setup** au point EP2. Les informations de signature de bout en bout du point EP1 sont incluses sans modification dans le message **Setup**.

15.3 Authentification, intégrité et non-répudiation des messages H.225.0

La procédure s'appliquant aux messages H.225.0 est la même que celle utilisée pour les messages RAS, à ceci près que pour chaque message de signalisation d'appel H.225.0, il faut identifier l'ensemble de champs qu'il convient de signer lorsque **tokenOID** est mis à "B".

15.4 Authentification et intégrité des messages H.245

Considérons le cas où le point EP1 souhaite envoyer un message H.245 – un message **TerminalCapabilitySet** par exemple – au point EP2. Le point EP1 vérifie si un message H.225.0 est en attente d'envoi au proxy. Si c'est le cas, le message H.245 est tunnellisé dans ce message H.225.0. Les champs du message H.225.0 sont mis aux valeurs indiquées précédemment pour la transmission d'un message H.225.0. Etant donné que le message H.245 est tunnellisé, les champs de **h323-uu-pdu** du message **h323-UserInformation** sont remplis comme suit:

- **h323-message-body** est mis au type de message H.225.0 en cours de transmission.
- **h245Tunnelling** est mis à TRUE.
- **h245Control** contient la chaîne d'octets PDU H.245.

Toutefois, si aucun message H.225.0 n'est en attente d'envoi, le message H.245 est tunnellisé dans un message H.225.0 **facility** ad hoc. Les champs de **h323-uu-pdu** du message **h323-UserInformation** sont remplis comme suit:

- **h323-message-body** est mis à **facility** qui contient:
  - **reason** mis à **undefinedReason**;
  - **tokens** et **cryptoTokens** comme pour tout message H.225.0.
- **h245Tunnelling** est mis à TRUE.
- **h245Control** contient la chaîne d'octets PDU H.245.

Le message **facility** est ensuite transmis par le point EP1 au proxy.

Dans les deux cas (message H.225.0 en attente d'envoi ou utilisation d'un message H.225.0 **facility** ad hoc), le proxy vérifie la signature destinée qui lui est destinée (dans ce cas, avec **tokenOID** "A") lorsqu'il reçoit le message. Ensuite, si un message H.225.0 est en attente d'envoi pour le tronçon proxy-portier GK1, le message H.245 est tunnellisé dans ce message; sinon, il est tunnellisé dans un message H.225.0 **facility** ad hoc. Comme c'est le cas pour la transmission de tout message de

signalisation d'appel H.225.0, une nouvelle signature est calculée pour le message en question avant qu'il ne soit transmis du proxy au portier GK1. La signature qui avait été envoyée du point EP1 au proxy et qui n'était pas destiné au proxy est transmise sans modification par le proxy au portier GK1.

Le présent paragraphe donne un résumé sur les méthodes que le profil de signature utilise pour assurer les différénts messages de signalisation H.323.

16 Compatibilité avec le contexte H.235 version 1

Bien que ces profils de sécurité soient mis au point dans le contexte H.235 version 2 (Rec. UIT-T H.235v2), il est possible de les appliquer dans un contexte H.235 version 1 (Rec. UIT-T H.235v1) moyennant quelques modifications mineures. Un destinataire est en mesure de détecter la présence de la version du protocole H.235 de l'expéditeur en évaluant les identificateurs d'objet du profil de sécurité (voir § 20).

Implémentations H.235 version 1 (Rec. UIT-T H.235v1):

- ne pas attribuer de valeur à ou ne pas évaluer **sendersID** de **ClearToken**.

17 Comportement pour les messages multidestinatires

Les messages H.225.0 multidestinataires tels que **GRQ** et **LRQ** doivent comporter un **CryptoToken** conformément aux procédures II et III lorsque aucune valeur n'est attribuée à **generalID**. Lorsque de tels messages sont envoyés à un seul destinataire, ils doivent comporter un **CryptoToken**.

18 Liste des messages de signalisation sécurisés

18.1 Messages RAS H.225.0

Message RAS H.225.0	Champs de signalisation H.235	Authentification seulement	Authentification et intégrité	Non-répudiation
Tous	cryptoTokens	Procédure II/III	Procédure II/III	Procédure II/III

NOTE – Dans le cas des messages envoyés à un seul destinataire, la procédure II ou III doit être appliquée avec les champs de sécurité de **CryptoToken** définis.

18.2 Messages de signalisation d'appel H.225.0

Message de signalisation d'appel H.225.0	Champs de signalisation H.235	Authentification seulement	Authentification et intégrité	Non-répudiation
Alerting-UUIE, CallProceeding-UUIE, Connect-UUIE, Setup-UUIE, Facility-UUIE, Progress-UUIE, Information-UUIE, ReleaseComplete-UUIE, Status-UUIE, StatusInquiry-UUIE, SetupAcknowledge-UUIE, Notify-UUIE	cryptoTokens	Procédure II/III	Procédure II/III	Procédure II/III

19 Utilisation des identificateurs sendersID et generalID

Le paramètre **ClearToken** contient les champs **sendersID** et **generalID**. Lorsque les informations d'identification sont disponibles, le champ **sendersID** doit contenir l'identificateur du portier (GKID) pour les messages provenant du portier, l'identificateur du point d'extrémité (EPID) pour les messages provenant du point d'extrémité. Lorsque les informations d'identification sont disponibles, le champ **generalID** doit contenir l'identification du portier (GKID) pour les messages provenant du point d'extrémité, et l'identificateur du point d'extrémité (EPID) pour les messages provenant du portier. Lorsque les informations d'identification ne sont pas disponibles ou lorsque la diffusion générale/multidiffusion est ambiguë, le champ est absent ou contient une chaîne néant. Le Tableau 2 résume la situation.

Tableau 2/H.235.2 – Utilisation des identificateurs sendersID et generalID

Message	sendersID	generalID
GRQ envoyé à un seul destinataire	**EPID** si disponible, autrement **NULL**	**GKID**
GRQ envoyé à plusieurs destinataires	**EPID** si disponible, autrement **NULL**	
GCF, GRJ	**GKID**	**EPID** si disponible, autrement **NULL**
RRQ initial		**GKID**
RCF	**GKID**	**EPID**
RRJ	**GKID**	
URQ, UCF, URJ, BRQ, BCF, BRJ, DRQ, DCF, DRJ, NSM, RIP, SCI, SCR, XRS (EP vers GK)	**EPID**	**GKID**
URQ, UCF, URJ, BRQ, BCF, BRJ, DRQ, DCF, DRJ, NSM, RIP, SCI, SCR, XRS (GK vers EP)	**GKID**	**EPID**
ARQ, IRQ, RAI	**EPID**	**GKID**
ACF, ARJ, BCF, LCF, LRJ, IRR, IRQ, RAC, LCF, LRJ, IACK, INAK	**GKID**	**EPID**
LRQ envoyé à un seul destinataire (EP vers GK)	**EPID**	**GKID**
LRQ envoyé à un seul destinataire (GK vers GK)	**GKID**	**GKID**
LRQ envoyé à plusieurs destinataires	**EPID**	
NOTE – GKID désigne l'identificateur du portier, EPID désigne l'identificateur du point d'extrémité. L'espace vide indique une chaîne d'identification manquante ou néant.		

20 Liste des identificateurs d'objet

Le Tableau 3 contient tous les identificateurs OID qui ont été mentionnés (voir aussi [OIW] et [WEBOIDs]. Il y a des identificateurs d'objet pour H.235v1 et pour H.235v2.

Tableau 3/H.235.2 – Identificateurs d'objet

Désignation de l'identificateur d'objet	Valeur(s) de l'identificateur d'objet	Description
"A"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 1} {itu-t (0) recommendation (0) h (8) 235 version (0) 1 1}	Utilisé dans la procédure II pour l'identificateur CryptoToken-tokenOID, indiquant que la signature englobe **tous** les champs du message H.225.0 RAS ou du message de signalisation d'appel (authentification et intégrité).
"B"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 2} {itu-t (0) recommendation (0) h (8) 235 version (0) 2 2} {itu-t (0) recommendation (0) h (8) 235 version (0) 1 2}	Utilisé dans la procédure II pour l'identificateur CryptoToken-tokenOID, indiquant que la signature englobe un **sous-ensemble** des champs du message RAS/H.225.0 (ClearToken) pour terminaux à authentification seulement (sans intégrité). Utilisé dans la procédure IA/H.235.1 pour l'identificateur CryptoToken-tokenOID, indiquant que le hachage englobe un sous-ensemble des champs du message RAS/H.225.0 (ClearToken) pour terminaux à authentification seulement (sans intégrité).
"P"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 4} {itu-t (0) recommendation (0) h (8) 235 version (0) 1 4}	Utilisé dans la procédure II ou III pour indiquer que le champ **certificate** achemine une adresse URL.
"R"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 3} {itu-t (0) recommendation (0) h (8) 235 version (0) 1 3}	Utilisé dans la procédure II pour l'identificateur ClearToken-tokenOID, indiquant que le ClearToken est en cours d'utilisation pour l'authentification/intégrité de bout en bout.
"S"	{itu-t (0) recommendation (0) h (8) 235 version (0) 2 7} {itu-t (0) recommendation (0) h (8) 235 version (0) 1 7}	Utilisé dans la procédure II, cet identificateur token OID indique l'authentification, l'intégrité et la non-répudiation du message.

Tableau 3/H.235.2 – Identificateurs d'objet

Désignation de l'identificateur d'objet	Valeur(s) de l'identificateur d'objet	Description
"V"	{iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 4}	Utilisé dans la procédure II ou III pour l'identificateur algorithm OID, indiquant l'utilisation de la signature numérique MD5-RSA.
"W"	{iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 5}	Utilisé dans la procédure II ou III pour l'identificateur algorithm OID, indiquant l'utilisation de la signature numérique SHA1-RSA.

SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série D	Principes généraux de tarification
Série E	Exploitation générale du réseau, service téléphonique, exploitation des services et facteurs humains
Série F	Services de télécommunication non téléphoniques
Série G	Systèmes et supports de transmission, systèmes et réseaux numériques
Série H	**Systèmes audiovisuels et multimédias**
Série I	Réseau numérique à intégration de services
Série J	Réseaux câblés et transmission des signaux radiophoniques, télévisuels et autres signaux multimédias
Série K	Protection contre les perturbations
Série L	Construction, installation et protection des câbles et autres éléments des installations extérieures
Série M	Gestion des télécommunications y compris le RGT et maintenance des réseaux
Série N	Maintenance: circuits internationaux de transmission radiophonique et télévisuelle
Série O	Spécifications des appareils de mesure
Série P	Qualité de transmission téléphonique, installations téléphoniques et réseaux locaux
Série Q	Commutation et signalisation
Série R	Transmission télégraphique
Série S	Equipements terminaux de télégraphie
Série T	Terminaux des services télématiques
Série U	Commutation télégraphique
Série V	Communications de données sur le réseau téléphonique
Série X	Réseaux de données, communication entre systèmes ouverts et sécurité
Série Y	Infrastructure mondiale de l'information, protocole Internet et réseaux de prochaine génération
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication

Imprimé en Suisse
Genève, 2006