UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T H.235

(02/98)

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

SÉRIE H: SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Infrastructure des services audiovisuels – Aspects système

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

(Antérieurement Recommandation du CCITT)

RECOMMANDATIONS UIT-T DE LA SÉRIE H

SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Caractéristiques des canaux de transmission pour des usages autres que téléphoniques	H.10–H.19
Emploi de circuits de type téléphonique pour la télégraphie à fréquence vocale	H.20–H.29
Circuits et câbles téléphoniques utilisés pour les divers types de transmission télégraphique et de transmissions simultanées	H.30–H.39
Circuits de type téléphonique utilisés en bélinographie	H.40–H.49
Caractéristiques des signaux de données	H.50–H.99
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.399

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

RECOMMANDATION UIT-T H.235

SECURITE ET CRYPTAGE DES TERMINAUX MULTIMEDIAS DE LA SERIE H (TERMINAUX H.323 ET AUTRES TERMINAUX DE TYPE H.245)

Résumé

La présente Recommandation décrit des améliorations apportées dans le cadre de la série des Recommandations H.3xx afin d'y introduire des services de sécurité tels que *l'authentification* et le *secret des communications* (cryptage des données). Le procédé qui est proposé est applicable aussi bien aux simples conférences point à point qu'aux conférences point à multipoint, à partir de tous les terminaux faisant appel au protocole de commande décrit dans la Recommandation H.245.

Par exemple, les systèmes H.323 fonctionnent sur des réseaux en mode paquet qui n'offrent pas une qualité de service garantie. La sûreté et la qualité du service offert par le réseau de base sont absentes pour les mêmes raisons techniques. Des communications sûres et en temps réel sur des réseaux non sûrs soulèvent généralement deux grands types de préoccupation: *l'authentification* et le *secret des communications*.

La présente Recommandation décrit l'infrastructure de sécurité et les techniques spécifiques de secret des communications que les terminaux multimédias conformes à la série H.3xx doivent utiliser. La présente Recommandation traite les questions relatives aux conférences interactives, c'est-à-dire, entre autres domaines, l'authentification et le secret des communications de tous les flux médias échangés en temps réel au cours d'une conférence. La présente Recommandation indique le protocole et les algorithmes nécessaires entre les entités H.323.

La présente Recommandation fait appel aux capacités générales qui sont décrites dans la Recommandation H.245: toute norme d'exploitation liée à ce protocole de commande pourra donc utiliser ce cadre de sécurité. L'on prévoit que, dans la mesure du possible, d'autres terminaux selon la série H pourront interfonctionner et utiliser directement les méthodes décrites dans la présente Recommandation. Dans un premier temps, la présente Recommandation n'assurera pas une mise en œuvre complète dans tous les domaines. Elle développera spécifiquement l'authentification des points d'extrémité et le secret des communications multimédias.

La présente Recommandation prévoit la possibilité de négocier les services et les capacités de façon générique. Elle prévoit également la possibilité de sélectionner les techniques et capacités cryptographiques utilisées. Leur mode d'emploi particulier dépend des capacités des systèmes, des exigences d'application et des contraintes propres aux politiques de sécurité. La présente Recommandation prend en compte divers algorithmes cryptographiques, avec diverses options appropriées à différents objectifs, comme les longueurs des clés. Certains algorithmes cryptographiques peuvent être attribués à des services de sécurité spécifiques (par exemple un algorithme pour un chiffrement rapide du flux média et un autre pour le codage de la signalisation).

Il convient également de noter que certains des algorithmes ou mécanismes cryptographiques dont on dispose pourront être réservés à l'exportation ou à d'autres fins nationales (par exemple avec des clés de longueur soumise à contrainte). La présente Recommandation prend en compte la signalisation d'algorithmes notoires, en plus de celle d'algorithmes cryptographiques non normalisés ou privatifs. Aucun algorithme n'est spécifiquement prescrit mais il est fortement conseillé que les points d'extrémité prennent en charge autant d'algorithmes applicables que possible afin de réaliser l'interopérabilité. Ce conseil est à rapprocher de l'idée que la conformité à la Recommandation H.245 ne garantit pas l'interopérabilité de deux codecs d'entité.

Source

La Recommandation UIT-T H.235, élaborée par la Commission d'études 16 (1997-2000) de l'UIT-T, a été approuvée le 6 février 1998 selon la procédure définie dans la Résolution n° 1 de la CMNT.

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.

La Conférence mondiale de normalisation des télécommunications (CMNT), qui se réunit tous les quatre ans, détermine les thèmes d'études 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 n° 1 de la CMNT.

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.

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 H.235

SECURITE ET CRYPTAGE DES TERMINAUX MULTIMEDIAS DE LA SERIE H (TERMINAUX H.323 ET AUTRES TERMINAUX DE TYPE H.245)

(Genève, 1998)

1 Domaine d'application

L'objectif principal de la présente Recommandation est d'assurer l'authentification, le secret des communications et l'intégrité dans le cadre du protocole actuel de la série H. Le texte de la présente Recommandation (1998) donne des détails sur la mise en œuvre avec le protocole H.323. On prévoit que ce cadre fonctionnera en liaison avec d'autres protocoles de la série H utilisant le protocole de commande H.245.

Les objectifs complémentaires de la présente Recommandation sont les suivants:

1) il y a lieu de développer une architecture de sécurité sous la forme d'un cadre extensible et flexible, permettant de mettre en œuvre un système de sécurité pour terminaux conformes à la série H. Ce cadre devra être fourni au moyen des capacités offertes par des services flexibles et indépendants, telles que la capacité de négocier et de sélectionner les techniques cryptographiques utilisées, ainsi que la façon de les utiliser;

2) assurer la sécurité de toutes les communications résultant de l'utilisation du protocole H.3xx, ce qui implique les questions d'établissement des connexions, de commande d'appel et d'échange de médias entre toutes les entités. Cette exigence comporte l'emploi de communications confidentielles (capacité de secret des communications) où l'on peut exploiter des fonctions d'authentification d'homologue ainsi que de protection de l'environnement de l'utilisateur contre les attaques qu'il pourrait subir;

3) la présente Recommandation ne doit pas interdire l'intégration d'autres fonctions de sécurité dans des entités H.3xx, pouvant les protéger contre des attaques issues du réseau;

4) la présente Recommandation ne doit pas limiter l'échelonnement de quelconques terminaux de la série de Recommandations H.3xx, selon les nécessités. Il peut s'agir aussi bien du nombre d'utilisateurs protégés que des niveaux de sécurité procurés;

5) le cas échéant, tous les mécanismes et toutes les capacités doivent être fournis indépendamment des couches ou topologies de transport sous-jacentes. D'autres moyens, hors du domaine d'application de la présente Recommandation, peuvent être requis pour contrer les menaces de ce type;

6) des dispositions doivent être prises pour le fonctionnement en environnement mixte (entités protégées et non protégées);

7) la présente Recommandation doit offrir la possibilité de distribuer des clés de session associées à la méthode cryptographique utilisée. (Ce qui n'implique pas que la gestion de clés publiques fondées sur des certificats doive faire partie de la présente Recommandation.)

L'architecture de sécurité décrite dans la présente Recommandation ne part pas du principe que les participants se connaissent déjà. Elle suppose cependant que des précautions appropriées ont été prises pour protéger physiquement les points d'extrémité conformes à la série H. Le principal risque pour les communications est donc supposé être une indiscrétion dans le réseau ou une autre méthode de détournement de flux médias.

La Recommandation H.323 (1996) donne la possibilité de conduire une conférence en mode audio, vidéo ou données entre plusieurs correspondants; mais elle ne donne pas à chaque participant la possibilité d'authentifier l'identité des autres participants. Elle ne permet pas non plus de privatiser les communications (c'est-à-dire de coder les flux).

Les terminaux de type H.323, H.324 et H.310 font appel aux procédures de signalisation par canal logique selon la Recommandation H.245, dans laquelle le contenu de chaque canal logique est décrit dès son ouverture. Des procédures sont prévues pour exprimer les capacités du récepteur et de l'émetteur. Les transmissions sont limitées à ce que les récepteurs peuvent décoder et ces derniers peuvent demander aux émetteurs un mode préférentiel particulier. Les capacités de sécurité de chaque entité terminale sont communiquées de la même façon que toutes les autres capacités de communication.

Certains terminaux de la série H (H.323) peuvent être utilisés en configuration multipoint. Le mécanisme de sécurité décrit dans la présente Recommandation permettra un fonctionnement sûr dans les environnements mettant en œuvre une exploitation par ponts de conférence (MCU) aussi bien centralisés que décentralisés.

2 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.

– Recommandation UIT-T H.225.0 (1998), *Protocoles de signalisation d'appel et mise en paquets des flux médias pour systèmes de communications multimédias en mode paquet.*

– Recommandation UIT-T H.245 (1998), *Protocole de commande pour communications multimédias*.

– Recommandation UIT-T H.323 (1998), *Systèmes de communications multimédias en mode paquet*.

– Recommandation UIT-T Q.931 (1993), *Spécification de la couche 3 de l'interface usager-réseau RNIS pour la commande de l'appel de base.*

– Recommandation UIT-T X.509 (1997) | ISO/CEI 9594-8:1997, *Technologies de l'information – Interconnexion des systèmes ouverts – L'annuaire: cadre d'authentification.*

– Recommandation X.800 du CCITT (1991), *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT*.

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

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

– 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 9798-2:1994, *Technologies de l'information – Techniques de sécurité – Authentification d'entité – Partie 2: Mécanismes utilisant des algorithmes de chiffrement symétriques.*

– ISO/CEI 9798-3:1993, *Technologies de l'information – Techniques de sécurité – Mécanismes d'authentification d'entité – Partie 3: Authentification d'entité utilisant un algorithme à clé publique.*

– ISO/CEI 9798-4:1995, *Technologies de l'information – Techniques de sécurité – Mécanismes d'authentification d'entité – Partie 4: Mécanismes utilisant une fonction cryptographique de vérification.*

– ATKINSON (R.): Security Architecture for the Internet Protocol (Architecture de sécurité pour le protocole Internet), RFC 1825, *Internet Engineering Task Force*, 1995.

KRAWCZYK (H.), BELLARE (M.), CANETTI (R.): HMAC: Keyed-Hashing for Message Authentification (Adressage dispersé sur clés calculées pour authentification de messages) RFC 2104, *Internet Engineering Task Force*, 1997.

3 Définitions

Pour les besoins de la présente Recommandation, les définitions figurant au paragraphe 3 des Recommandations H.323, H.225.0 et H.245 s'appliquent, en plus de celles que contient le présent paragraphe. Certains des termes suivants sont utilisés selon la définition donnée dans la Rec. X.800 du CCITT | ISO 7498-2 et dans les Recommandations X.803, X.810 et X.811.

3.1 contrôle d'accès: précaution prise contre l'utilisation non autorisée d'une ressource, y compris l'utilisation d'une ressource d'une façon non autorisée (X.800).

3.2 authentification: attestation de l'identité revendiquée par une entité (X.811).

3.3 autorisation: octroi d'une permission sur la base d'une identité authentifiée.

3.4 agression: activités entreprises pour contourner ou exploiter des déficiences constatées dans les mécanismes de sécurité d'un système. Une agression directe d'un système exploite des déficiences dans les algorithmes, principes ou propriétés sous-tendant un mécanisme de sécurité. Des agressions indirectes consistent à contourner le mécanisme ou à en provoquer une utilisation incorrecte par le système.

3.5 certificat: ensemble de données relatives à la sécurité, émis par une autorité de sécurité ou par un tiers de confiance en même temps que des informations de sécurité qui sont utilisées pour fournir les services d'intégrité et d'authentification d'origine des données (X.810). Dans la présente Recommandation, ce terme vise des certificats "à clé publique" qui sont des valeurs représentant une clé publique de détenteur (et d'autres informations facultatives), ces valeurs ayant été vérifiées et signées par une autorité de confiance sous une forme infalsifiable.

3.6 chiffre: algorithme cryptographique ou transformée mathématique.

3.7 confidentialité: caractéristique qui empêche la divulgation des informations à des individus, entités ou processus non autorisés.

3.8 algorithme cryptographique: fonction mathématique qui calcule un résultat à partir d'une ou de plusieurs valeurs d'entrée.

3.9 chiffrement; cryptage: processus consistant à rendre des données illisibles par des entités non autorisées après application d'un algorithme cryptographique (ou de cryptage). Le déchiffrement (décryptage) est l'opération inverse par laquelle le texte chiffré est transformé en texte clair.

3.10 intégrité: caractéristique de données qui n'ont pas été altérées de façon non autorisée.

3.11 gestion de clé: production, stockage, distribution, suppression, archivage et application de clés conformément à une politique de sécurité (X.800).

3.12 flux média: flux audio, vidéo ou de données, ou combinaison quelconque de ces types de flux. Les flux médias acheminent des données d'utilisateur ou d'application (capacité utile) mais pas de données de commande.

3.13 non-répudiation: protection contre le déni, par une des entités impliquées dans une communication, d'avoir participé à tout ou partie de celle-ci.

3.14 secret des communications: mode de communication dans lequel seules les parties explicitement habilitées peuvent interpréter la communication. Le secret des communications est normalement réalisé par cryptage et par partage de clé(s) pour accéder au chiffre.

3.15 canal privé: dans la présente Recommandation, un canal privé est celui qui résulte d'une négociation préalable par canal sécurisé et qui peut servir à acheminer des flux médias.

3.16 cryptographie à clés publiques: système de cryptage qui fait appel (pour le cryptage et le décryptage) à des clés asymétriques liées par une relation mathématique qui ne peut logiquement pas être calculée.

3.17 algorithme cryptographique symétrique (à clés secrètes): algorithme permettant de réaliser le chiffrement ou le déchiffrement correspondant, dans lequel la même clé est requise à la fois pour le chiffrement et pour le déchiffrement (X.810).

3.18 menace: violation potentielle de la sécurité (X.800).

4 Symboles et abréviations

La présente Recommandation utilise les abréviations suivantes:

DSS norme de signature numérique (*digital signature standard*)

IPSEC sécurité du protocole Internet (*Internet protocol security*)

QS qualité de service

RSA algorithme à clé publique de Rivest, Shamir et Adleman

SDU unité de données de service (*service data unit*)

TLS sécurité de la couche transport (*transport level security*)

5 Conventions

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

– l'auxiliaire "doit/doivent" indique une prescription impérative;

– l'auxiliaire "devrait/devraient" indique une mesure suggérée mais facultative;

– l'auxiliaire "peut/peuvent" indique une possibilité d'action plutôt qu'une recommandation de résultat.

Sauf énumération explicite d'une autre Recommandation, les références aux paragraphes, sous-paragraphes, annexes et appendices se rapportent à ceux de la présente Recommandation. Par

exemple, la référence "1.4" correspond au sous-paragraphe 1.4 de la présente Recommandation. La référence "6.4/H.245" correspond au sous-paragraphe 6.4 de la Recommandation H.245.

La présente Recommandation décrit l'utilisation de "n" types de message différents: H.245, RAS, Q.931, etc. Pour établir une distinction entre ces différents types de message, la convention suivante est utilisée: les messages et noms de paramètres H.245 se composent de plusieurs mots concaténés qui sont mis en valeur par un caractère gras (**maximumDelayJitter**); les noms de message RAS sont représentés par des abréviations à trois lettres (**ARQ**); les noms de message Q.931 se composent d'un ou de deux mots dont la première lettre est en majuscule (**Call Proceeding**).

6 Introduction du système

6.1 Résumé

1) le canal de signalisation d'appel peut être sécurisé au moyen du protocole TLS [**TLS**] ou IPSEC [**13/IPSEC**] à un accès dont la sûreté est bien établie (H.225.0);
2) les utilisateurs peuvent être authentifiés soit au cours de la connexion d'appel initiale, soit au cours du processus de sécurisation du canal H.245 et/ou par échange de certificats sur le canal H.245;
3) les capacités de cryptage d'un canal média sont déterminées par des extensions du mécanisme existant de négociation de capacité;
4) la distribution initiale par l'entité maîtresse des données relatives aux clés s'effectue par les messages H.245 **OpenLogicalChannel** ou **OpenLogicalChannelAck**;
5) la redéfinition des clés peut s'effectuer par les commandes H.245: **EncryptionUpdateRequest** et **EncryptionUpdate**;
6) la distribution des données relatives aux clés est protégée soit par l'exploitation du canal H.245 en tant que canal privé ou par protection spécifique des données de clé par échange des certificats sélectionnés;
7) les protocoles de sécurité présentés sont conformes soit à des normes ISO publiées ou à des normes proposées par le groupe IETF.

6.2 Authentification

Le processus d'authentification vérifie que ceux qui répondent sont bien ceux qu'ils disent être. L'authentitication peut être réalisée dans le cadre de l'échange de certificats à clé publique ou dans celui d'un échange faisant appel à une information secrète, partagée entre les entités en cause. Il peut s'agir d'un mot de passe statique ou d'un autre type d'information arbitraire.

La présente Recommandation décrit le protocole d'échange des certificats mais ne spécifie pas les critères permettant de les vérifier et de les accepter les uns en fonction des autres. En général, les certificats donnent au vérificateur une certaine garantie que celui qui présente le certificat est la personne qu'il déclare être. L'échange de certificats a pour objet d'authentifier *l'utilisateur* du point d'extrémité et non simplement le point d'extrémité physique. L'utilisation de certificats numériques permet de prouver, par protocole d'authentification, que ceux qui répondent possèdent les clés privées correspondant aux clés publiques contenues dans les certificats. Cette authentification protège contre les attaques par entremetteur mais ne prouve pas automatiquement l'identité de ceux qui répondent. Pour cela, il faut normalement qu'une certaine politique s'applique au reste du contenu des certificats. Pour les certificats d'autorisation par exemple, le certificat contiendra normalement l'identification du fournisseur de service ainsi qu'une certaine identification du compte d'utilisateur, prescrite par le fournisseur de service.

Le cadre d'authentification présenté dans la présente Recommandation ne prescrit pas le contenu des certificats (c'est-à-dire qu'il ne spécifie pas de politique relative aux certificats) au-delà de ce qui est requis par le protocole d'authentification. Une application utilisant ce cadre pourra toutefois imposer des prescriptions politiques de haut niveau comme la présentation du certificat à l'utilisateur pour approbation. Cette politique de niveau supérieur pourra soit être automatisée au sein de l'application soit nécessiter une interaction humaine.

Pour l'authentification qui ne fait pas appel à des certificats numériques, la présente Recommandation indique la signalisation permettant de réaliser divers scénarios d'interrogation/réponse. Cette méthode d'authentification nécessite une coordination préalable entre les entités communicantes, de façon qu'un secret partagé soit obtenu. Un exemple de cette méthode serait celui d'un client abonné à un service.

Une troisième option permet de réaliser l'authentification dans le contexte d'un protocole de sécurité distinct tel que la sécurité TLS [**TLS**] ou IPSEC [**13/IPSEC**].

Des entités homologues peuvent prendre en charge une authentification aussi bien bidirectionnelle qu'unidirectionnelle. Cette authentification peut se produire sur tout ou partie des voies de communication.

Tous les mécanismes d'authentification spécifiques qui sont décrits dans la présente Recommandation sont identiques aux algorithmes mis au point par l'ISO (ou en sont dérivés), comme spécifié dans les Parties 2 à 3 de l'ISO/CEI 9798 ou sont fondés sur des protocoles IETF.

6.2.1 Certificats

La normalisation des certificats, y compris leur production, leur administration et leur distribution, est hors du domaine d'application de la présente Recommandation. Les certificats utilisés pour établir des canaux sûrs (signalisation d'appel et/ou commande d'appel) doivent être conformes aux prescriptions de tout protocole qui a été négocié pour sécuriser ces canaux.

Il y a lieu de noter que, pour l'authentification utilisant des certificats à clé publique, les points d'extrémité sont appelés à fournir des signatures numériques utilisant la valeur de clé privée associée. Le seul échange de certificats à clé publique ne suffit pas à protéger contre les attaques par entremetteur. Les protocoles H.235 sont conformes à cette exigence.

6.3 Sécurité lors de l'établissement d'appel

Il y a au moins deux raisons pour sécuriser le canal d'établissement d'appel (par exemple par message H.323 utilisant Q.931). La première est l'exécution d'une authentification simple, avant d'accepter l'appel. La deuxième vise à permettre une autorisation d'appel. Si cette fonction est souhaitée dans le terminal conforme à la série H, il y a lieu d'utiliser un mode de communication sécurisé (tel que TLS/IPSEC pour H.323) avant l'échange des messages de connexion d'appel. En variante, l'autorisation peut être donnée sur la base d'une authentification de service spécifique, dont les contraintes de politique sont hors du domaine d'application de la présente Recommandation.

6.4 Sécurité de la commande d'appel (H.245)

Le canal de commande d'appel (H.245) devrait également être sécurisé de quelque façon, afin d'offrir ensuite un média secret. Le canal H.245 doit être sécurisé par un quelconque mécanisme de secret des communications (dont la négociation comporte l'option "aucun"). Les messages H.245 sont utilisés pour signaler les algorithmes et clés de cryptage utilisés dans les canaux de médias partagés et privés. Cette capacité permet de crypter, canal logique par canal logique, différents canaux de média au moyen de différents mécanismes. Par exemple, lors de conférences multipoint centralisées, différentes clés peuvent être utilisées pour les différents flux destinés à chaque point d'extrémité.

Cela permet de sécuriser les flux médias destinés à chaque point d'extrémité de la conférence. Pour utiliser les messages H.245 de manière sûre, l'ensemble du canal H.245 (canal logique 0) devrait être ouvert après sécurisation négociée.

Le mécanisme par lequel un canal H.245 est sécurisé dépend des terminaux série H utilisés. La seule exigence imposée à tous les systèmes utilisant cette structure de sécurité est que chacun d'eux possède une certaine capacité permettant de négocier et/ou de signaler que le canal H.245 doit être exploité d'une certaine manière sécurisée avant d'être effectivement initialisé. Par exemple, le protocole H.323 utilisera les messages de signalisation de connexion H.225.0 pour réaliser cette condition.

6.5 Secret des communications par flux médias

La présente Recommandation décrit le secret des communications multimédias pour des flux médias acheminés par transport en mode paquet. Ces canaux peuvent être unidirectionnels dans le cadre de la définition des canaux logiques H.245. Il n'est pas prescrit que ces canaux soient unidirectionnels dans la couche Physique ou Transport.

Une première étape pour réaliser le secret des communications multimédias devrait être la fourniture d'un canal de commande privé permettant d'établir des bases de construction de clés et/ou l'établissement des canaux logiques devant transporter les flux médias cryptés. A cette fin, lors du fonctionnement en conférence sécurisée, tout point d'extrémité participant peut utiliser un canal H.245 crypté. Cette procédure permet de protéger la sélection des algorithmes cryptographiques et les clés de cryptage transmises dans la commande H.245 **OpenLogicalChannel**.

Le canal H.245 sécurisé peut être exploité avec des caractéristiques différentes des canaux médias privés, dans la mesure où il procure un niveau de secret des communications acceptable par les deux parties. Il permet aux mécanismes de sécurité protégeant les flux médias et tous canaux de commande de fonctionner de manière totalement indépendante, en fournissant des niveaux de robustesse et de complexité totalement différents.

S'il est prescrit que le canal H.245 soit exploité de manière non cryptée, les clés spécifiques de cryptage de média peuvent être chiffrées séparément par les parties engagées, de la manière qui a été signalée et convenue. Un canal logique de type **h235Control** peut être utilisé pour fournir les données protégeant les clés de cryptage de média. Ce canal logique peut être exploité par tout mode négocié à cette fin.

Le secret (cryptage) des communications de données acheminées dans les canaux logiques doit avoir la forme spécifiée par la commande **OpenLogicalChannel**. Les informations d'en-tête spécifiques à la couche Transport ne doivent pas être chiffrées. Le secret des données doit être fondé sur un cryptage de bout en bout.

6.6 Eléments crédibilisés

La base de l’authentification (confiance) et du secret des communications est définie par les terminaux du canal de communication. Pour un canal d'établissement de connexion, ces terminaux peuvent être ceux de l'appelant et d'un élément du réseau d'accueil. Par exemple, un poste téléphonique "escompte" que le commutateur du réseau le connectera au poste dont le numéro a été composé. C'est pourquoi toute entité à laquelle aboutit un canal de commande H.245 crypté ou un canal logique de type **encryptedData** doit être considérée comme un élément crédibilisé de la connexion. Ces entités peuvent être des ponts de conférence ou des têtes de ligne (passerelles). Le résultat de la crédibilisation d'un élément est l'assurance de pouvoir révéler en confiance à cet élément le mécanisme de secret des communications (algorithme et clé).

Compte tenu de ce qui précède, il incombe aux participants du chemin de communication d'authentifier tout un chacun des éléments "crédibilisés". Pour cela, on procédera normalement à un échange de certificats comme dans le cas de l'authentification de bout en bout normale. La présente Recommandation ne prescrira aucun niveau spécifique d'authentification et se limitera à suggérer que ce niveau soit acceptable par toutes les entités faisant appel aux éléments crédibilisés. Les détails relatifs à un modèle et à une politique de certificats applicables à la crédibilisation feront l'objet d'un complément d'étude.

Le secret des communications ne peut être assuré entre les deux points d'extrémité que si les connexions entre éléments crédibilisés sont démontrées avoir été protégées contre les attaques par entremetteur.

6.6.1 Dépôt de clé

Bien que cela ne soit pas spécifiquement requis pour le fonctionnement, la présente Recommandation contient des dispositions pour conférer aux entités utilisant le protocole H.235 une capacité de récupération de clé dans le cadre des éléments de signalisation.

La possibilité de récupérer les clés de codage de média perdues doit être prise en charge par les installations lorsqu'une telle capacité est souhaitable ou requise.

Le dépôt de clé est une fonctionnalité qui est souvent désignée par le terme "tiers de confiance" (TTP, *trusted third party*). Cette fonctionnalité reste à étudier.

6.7 Non-répudiation

A étudier.

7 Procédures d'établissement de connexion

7.1 Introduction

Comme indiqué dans le paragraphe Introduction du système, aussi bien le canal de connexion d'appel (H.225.0 pour terminaux série H.323) que le canal de commande d'appel (H.245) doivent fonctionner dans le mode négocié, sécurisé ou non sécurisé, à partir du premier échange de messages. Pour le canal de connexion d'appel, le mode de sécurité est déterminé *a priori* [pour un terminal H.323, un point TSAP sécurisé par TLS (accès 1300) doit être utilisé pour les messages Q.931]. Pour le canal de commande d'appel, le mode de sécurité est déterminé par les informations transmises dans le protocole d'établissement de connexion initial, utilisé par le terminal série H.

Lorsqu'il n'y a pas de chevauchement entre capacités de sécurité, le terminal appelé peut refuser la connexion. L'erreur renvoyée ne devrait pas contenir de renseignements sur un quelconque défaut de correspondance entre messages de sécurité. Le terminal appelant devra déterminer l'origine du problème par d'autres moyens. Lorsque le terminal appelant reçoit un message "d'acquittement de connexion" CONNECT AKNOWLEDGE sans capacités de sécurité suffisantes, il y a lieu qu'il mette fin à l'appel.

Si les terminaux appelant et appelé sont des capacités de sécurité compatibles, chaque extrémité doit partir du principe que le canal H.245 doit fonctionner dans le mode sécurisé qui a été négocié. L'échec d'établissement du mode H.245 sécurisé qui est défini ici devrait être considéré comme une erreur de protocole et la connexion devrait être fermée.

8 Signalisation et procédures H.245

En général, les aspects relatifs au secret des communications par canaux médias sont commandés de la même façon que tout autre paramètre de codage: chaque terminal indique ses capacités, la source des données choisit un format à utiliser et le récepteur acquitte ou refuse le mode. Tous les aspects indépendants du mécanisme qui sont indépendants du transport, comme la sélection de l'algorithme, sont indiqués par des éléments génériques de canal logique. Les caractéristiques de transport telles que la synchronisation des algorithmes de clé ou de cryptage sont acheminées dans des structures propres à la couche Transport.

8.1 Fonctionnement avec canal H.245 sécurisé

En supposant que les procédures de connexion indiquées dans le paragraphe précédent (Procédures d'établissement de connexion) indiquent un mode de fonctionnement sécurisé, le dialogue négocié et l'authentification doivent être effectués pour le canal logique H.245 avant l'échange d'éventuels autres messages H.245. S'il a été négocié, tout échange de certificats doit intervenir au moyen de tout mécanisme approprié pour les terminaux conformes à la série H. Après sécurisation du canal H.245, les terminaux utilisent le protocole H.245 comme ils le feraient en mode non sécurisé.

8.2 Fonctionnement avec canal H.245 non sécurisé

En variante, le canal H.245 peut fonctionner en mode non sécurisé et les deux entités ouvrent un canal logique sécurisé avec lequel l'authentification et le calcul du secret partagé sont effectués. Par exemple, une commande TLS ou IPSEC peut être utilisée afin d'ouvrir un canal logique dont le champ **dataType** contient une valeur pour le paramètre **encryptionData**. Ce canal pourra ensuite être utilisé pour calculer un secret partagé protégeant d'éventuelles clés de session média ou pour transporter le message **EncryptionSync**.

8.3 Echange de capacités

Conformément aux procédures indiquées au 8.3/H.245 (Procédures d'échange de capacités) et conformément à la Recommandation de la série H applicable au système, les points d'extrémité échangent leurs capacités au moyen de messages H.245. Ces ensembles de capacités peuvent maintenant contenir des définitions indiquant des paramètres de sécurité et de cryptage. Par exemple, un point d'extrémité peut signaler des capacités d'émission et de réception de données vidéo H.261, normales ou cryptées.

Chaque algorithme de cryptage utilisé avec un codec média particulier implique une nouvelle définition de capacité. Comme pour toute autre capacité, les points d'extrémité peuvent indiquer, au cours de leur échange de capacité, des codecs cryptés aussi bien indépendants que dépendants. Cela permettra aux points d'extrémité de dimensionner leurs capacités de sécurité en fonction des charges et des ressources disponibles.

Une fois l'échange de capacités effectué, les points d'extrémité peuvent ouvrir des canaux logiques sécurisés pour médias de la même façon qu'ils le feraient en mode non sécurisé.

8.4 Rôle de maître

La relation maître-esclave H.245 est utilisée pour établir l'entité maîtresse en vue du fonctionnement en canal bidirectionnel et de la résolution d'autres conflits. Ce rôle de maître est également utilisé dans les méthodes de sécurité. Bien que le ou les modes de sécurité d'un flux média soient fixés par la source (en fonction des capacités du récepteur), le maître est le point d'extrémité qui produit la clé de cryptage. Cette production est effectuée sans tenir compte du fait que le maître est le récepteur ou

l'émetteur du média crypté. Pour permettre le fonctionnement de canaux à destinations multiples avec clés partagées, le pont (qui est également le maître) doit normalement produire les clés.

8.5 Signalisation par canal logique

Les points d'extrémité ouvrent des canaux logiques pour médias en mode sécurisé de la même façon qu'ils le feraient pour des canaux logiques de médias en mode non sécurisé. Chaque canal peut fonctionner de manière totalement indépendante des autres canaux – en particulier pour ce qui est de la sécurité. Le mode particulier doit être défini dans le message **OpenLogicalChannel** du champ **dataType**. La clé de cryptage initiale doit être transmise dans le message **OpenLogicalChannel** ou **OpenLogicalChannelAck** selon la relation maître/esclave de l'expéditeur du message **OpenLogicalChannel**.

Le message **OpenLogicalChannelAck** doit faire fonction de confirmation du mode de cryptage. Si la commande **openLogicalChannel** n'est pas acceptable par le destinataire, le paramètre **dataTypeNotSupported** ou **dataTypeNotAvailable** (condition transitoire) doit être renvoyé dans le champ de cause du message **OpenLogicalChannelReject**.

Au cours de l'échange protocolaire qui établit le canal logique, la clé de cryptage doit être transmise du maître à l'esclave (sans tenir compte de l'expéditeur du message **OpenLogicalChannel**). Pour les canaux médias ouverts par un point d'extrémité (autre que le maître), le maître doit renvoyer la clé de cryptage initiale et le point de synchronisation initial dans le message **OpenLogicalChannelAck** (dans le champ **encryptionSync**). Pour les canaux médias ouverts par le maître, le message **OpenLogicalChannel** doit comporter la clé de cryptage initiale et le point de synchronisation dans le champ **encryptionSync**.

9 Procédures multipoint

9.1 Authentification

L'authentification doit être effectuée entre un point d'extrémité et de pont de conférence de la même façon qu'elle le serait dans une conférence point à point. Le pont doit déterminer la politique concernant le niveau et la sévérité de l'authentification. Comme indiqué au 6.6, le pont est un élément qui doit être crédibilisé. Les points d'extrémité d'une conférence peuvent être limités par le niveau d'authentification employé par le pont de conférence. De nouvelle commandes **ConferenceRequest/ConferenceResponse** permettent aux points d'extrémité d'obtenir du pont les certificats d'autres participants à la conférence. Comme indiqué dans les procédures H.245, les points d'extrémité d'une conférence multipoint peuvent demander d'autres certificats de point d'extrémité via le pont mais ne sont pas toujours en mesure d'effectuer une authentification cryptographique directe à l'intérieur du canal H.245.

9.2 Secret des communications

Un pont de conférence doit remporter tous les échanges maître/esclave et doit donc fournir la ou les clés de cryptage aux participants à une conférence multipoint. Le secret des communications pour des sources individuelles au cours d'une session commune (dans l'hypothèse de destinations multiples) peut être obtenu avec des clés individuelles ou avec des clés communes. Ces deux modes peuvent être arbitrairement choisis par le pont. Ils ne doivent pas pouvoir être commandés à partir d'un point d'extrémité particulier, sauf dans les modes autorisés par la politique des ponts de conférence. En d'autres termes, une clé commune peut être utilisée sur de multiples canaux logiques qui ont été ouverts par des sources différentes.

10 Signalisation et procédures d'authentification

10.1 Introduction

L'on peut utiliser deux types d'authentification. Le premier type est fondé sur un cryptage symétrique qui n'exige aucun contact préliminaire entre les entités communicantes. Le second type est fondé sur la capacité de disposer d'un secret partagé préalable (ci-après dénommé "sur abonnement"). Deux formes d'authentification sur abonnement sont fournies: par mot de passe et par certificat.

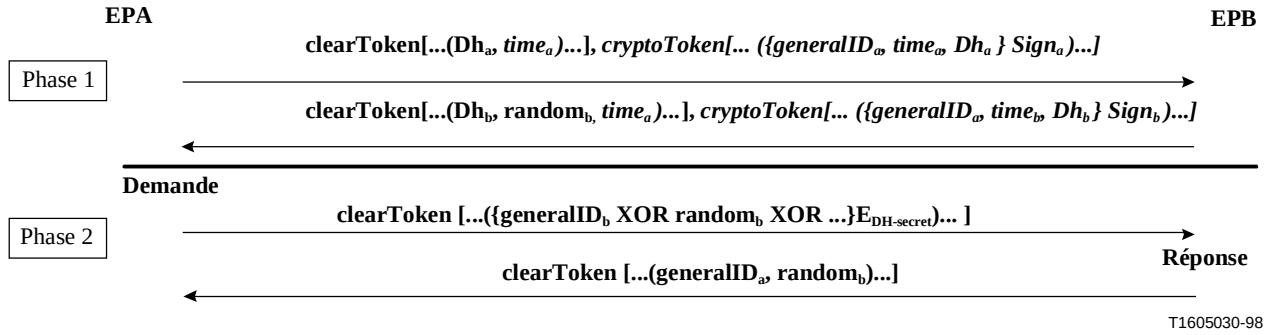
10.2 Méthode de Diffie-Hellman avec authentification facultative

Cette méthode ne vise pas à réaliser une authentification absolue au niveau de l'utilisateur. Elle assure une signalisation permettant de constituer un secret partagé entre deux entités, pouvant conduire à des données de calcul de clés pour des communications privées.

A l'issue de cet échange, les deux entités posséderont une clé à secret partagé ainsi qu'un algorithme sélectionné qui leur permettra d'utiliser cette clé. Cette clé à secret partagé pourra ensuite être utilisée dans tout échange ultérieur de type demande/réponse. Il convient de noter que, dans de rares cas, l'échange Diffie-Hellman peut produire des clés notoirement *faibles* pour certains algorithmes. Dans ces cas, chaque entité devrait se déconnecter et se reconnecter afin d'établir un nouveau jeu de clés.

La première phase de la Figure 1 montre les données échangées lors d'une authentification Diffie-Hellman. La deuxième phase permet au répondant d'authentifier des messages de demande propres à une application ou à un protocole. On notera qu'une nouvelle valeur aléatoire peut être renvoyée avec chaque réponse.

NOTE – Un élément facultatif de signature (indiqué ci-dessous en ***italiques***) peut aussi être fourni.



[... ...] indique une séquence de jetons

() indique un jeton particulier, qui peut contenir des éléments multiples

{}$\mathbf{E}_{\mathbf{DH\text{-}Secret}}$ indique que les valeurs contenues sont cryptées au moyen de la méthode de secret Diffie-Hellman

Le point d'extrémité B (EPB) connaît la clé à secret partagé qu'il faut utiliser pour déchiffrer l'identificateur **generalID$_b$** en l'associant à l'identificateur **generalID$_a$** qu'il convient de transmettre également dans le message. On notera que la valeur cryptée dans la phase 2 est transmise dans le champ d'identificateur général d'un **jeton en clair**, afin de simplifier le codage.

Figure 1/H.235

10.3 Authentification sur abonnement

10.3.1 Introduction

Bien que les procédures décrites ici (ainsi que les algorithmes ISO dont elles sont issues) soient de nature bidirectionnelle, elle ne peuvent être utilisées que dans un seul sens si l'authentification n'est requise que dans ce sens. Ces échanges partent du principe que chaque extrémité possède un certain identificateur notoire (comme un identificateur en mode texte) qui l'identifie sans équivoque. Une autre hypothèse est faite pour supposer qu'il existe une référence temporelle acceptable de part et d'autre (permettant de calculer les pointeurs temporels). La grandeur de la dérive temporelle acceptable relève d'une décision de mise en œuvre locale.

Il existe trois variantes différentes de mise en œuvre, selon les exigences:

1) authentification par mot de passe avec cryptage symétrique;

2) authentification par mot de passe avec dispersion d'adresse;

3) authentification par certificat avec signatures.

Dans tous les cas, le jeton contiendra les informations décrites dans les sous-paragraphes suivants, selon la variante choisie. On notera que, dans tous les cas, **l'identificateur général** peut être connu par examen de la configuration ou d'un répertoire, plutôt que par échange protocolaire dans la bande.

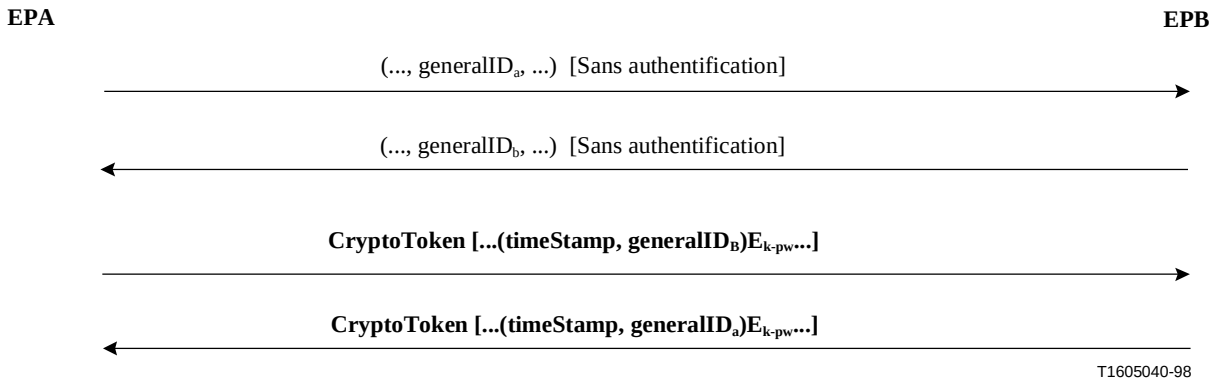
10.3.2 Authentification par mot de passe avec cryptage symétrique

La Figure 2 montre le format du jeton et l'échange de messages requis pour exécuter ce type d'authentification. Ce protocole est fondé sur le 5.2.1 de l'ISO/CEI 9798-2. On fait l'hypothèse qu'un identificateur et le mot de passe associé sont échangés lors de l'abonnement. La clé de cryptage a une longueur de N octets (comme indiqué par l'identificateur d'algorithme). Elle est formée comme suit:

– si la longueur du mot de passe = N, clé = mot de passe;

– si la longueur du mot de passe < N, la clé est bourrée de zéros;

– si la longueur du mot de passe > N, les N premiers octets sont attribués à la clé, puis le N + M^e octet du mot de passe est combiné par un OU exclusif avec le $Mmod(N)^e$ octet (pour tous les octets au-delà de N). (En d'autres termes, tous les octets "surnuméraires" du mot de passe sont successivement repliés sur la clé par application de la fonction OUX.)



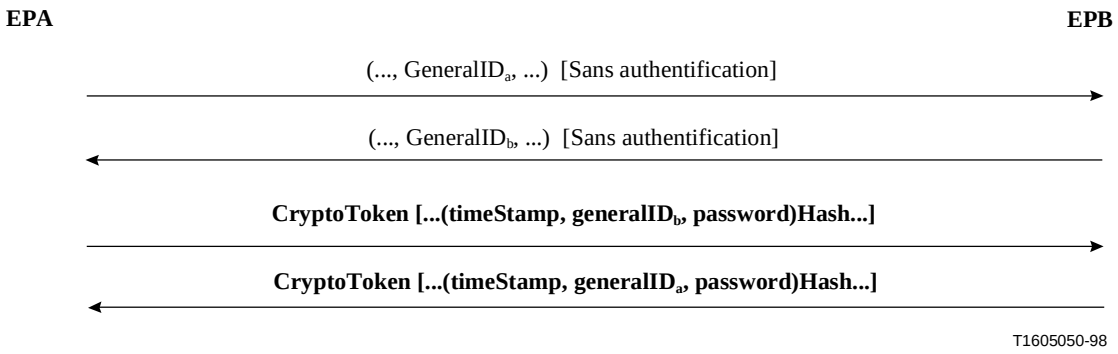
NOTE 1 – Le jeton de retour de l'extrémité EPB est facultatif; s'il est omis, seule l'authentification à sens unique est réalisée.

NOTE 2 – La variable $\mathbf{E_{k-pw}}$ indique des valeurs qui sont cryptées au moyen de la clé "k" calculée d'après le mot de passe "pw".

Figure 2/H.235

10.3.3 Authentification par mot de passe avec dispersion d'adresse

La Figure 3 montre le format du jeton et l'échange de messages requis pour exécuter ce type d'authentification. Ce protocole est fondé sur le 5.2.1 de l'ISO/CEI 9798-4. On fait l'hypothèse qu'un identificateur et le mot de passe associé sont échangés lors de l'abonnement.



NOTE 1 – Le jeton de retour de l'extrémité EPB est facultatif; s'il est omis, seule l'authentification à sens unique est réalisée.

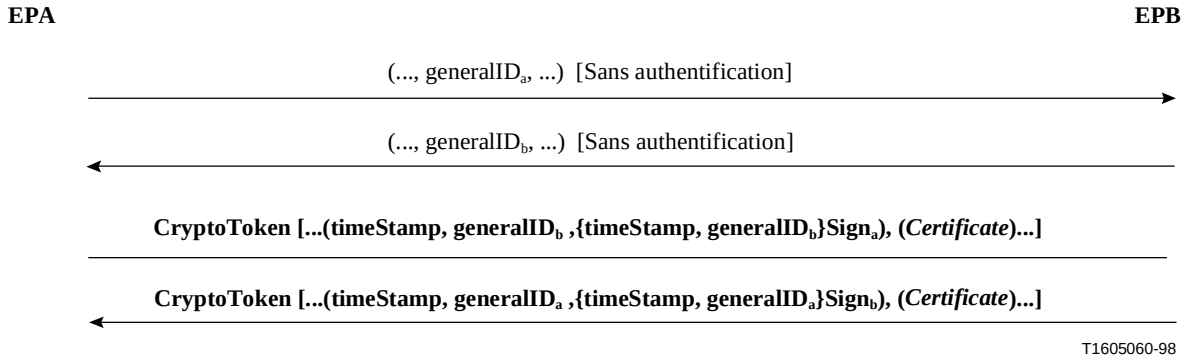
NOTE 2 – La variable **Hash** indique une fonction de dispersion d'adresse qui s'applique aux valeurs contenues.

Figure 3/H.235

10.3.4 Authentification par certificat avec signatures

La Figure 4 montre le format du jeton et l'échange de messages requis pour exécuter ce type d'authentification. Ce protocole est fondé sur le 5.2.1 de l'ISO/CEI 9798-3. On fait l'hypothèse qu'un identificateur et le mot de passe associé sont échangés lors de l'abonnement.

NOTE – Un élément facultatif de certificat (indiqué ci-dessous en ***italiques***) peut aussi être fourni.



NOTE 1 – Le jeton de retour de l'extrémité EPB est facultatif; s'il est omis, seule l'authentification à sens unique est réalisée.

NOTE 2 – Un certificat de type "paiement" peut, facultativement, être inclus par l'émetteur situé au point EPA.

NOTE 3 – La variable **Sign** indique une fonction de signature (issue du certificat associé) qui est exécutée sur les valeurs contenues.

Figure 4/H.235

11 Procédures de cryptage de flux médias

Les flux médias doivent être codés au moyen de l'algorithme et de la clé qui sont présentés dans le canal H.245. Les Figures 5 et 6 montrent le flux général. On notera que l'en-tête de transport est attaché à l'unité SDU de transport une fois que cette unité a été cryptée. Les segments opaques indiquent le secret des communications. Au fur et à mesure que de nouvelles clés sont reçues par l'émetteur et utilisées dans le cryptage, l'en-tête d'unité SDU doit indiquer d'une façon ou d'une autre au récepteur que la nouvelle clé est désormais en usage. Par exemple, dans un flux H.323, l'en-tête (SDU) de protocole RTP modifiera son type de charge utile pour indiquer le commutateur à la nouvelle clé.

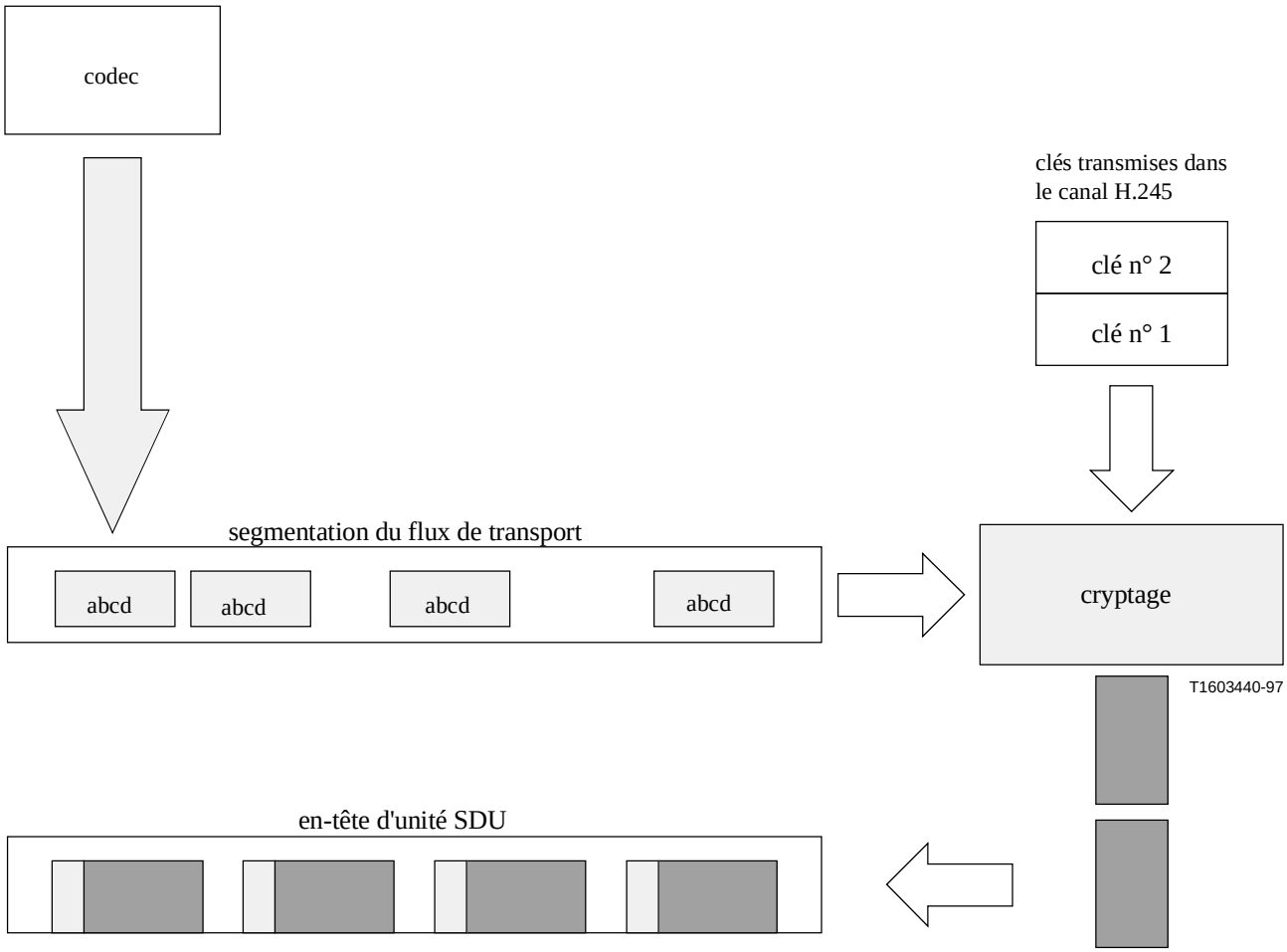


Figure 5/H.235 – Cryptage du média

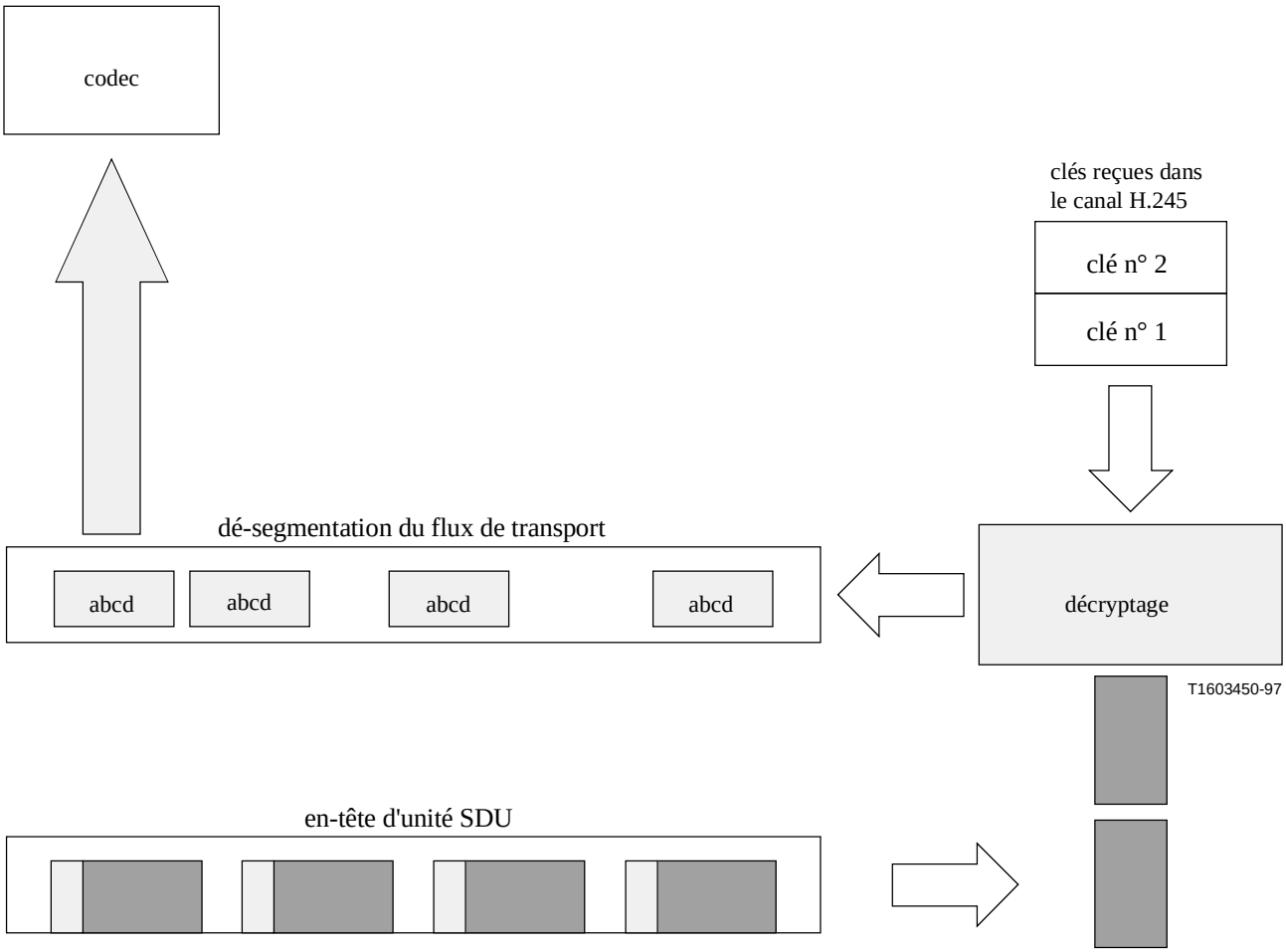


Figure 6/H.235 – Décryptage du média

11.1 Clés de session média

Le message **encryptionUpdate** comporte le champ de clé **h235key**, qui est codé en notation ASN.1 dans le contexte de l'arbre ASN.1 de la Recommandation H.235 et qui est transmis sous forme d'une chaîne d'octets opaque par rapport au flux H.245. Cette clé peut être protégée au moyen d'un des trois mécanismes possibles, au fur et à mesure de leur passage entre deux points d'extrémité.

- si le canal H.245 est sécurisé, aucune protection additionnelle n'est appliquée aux données de clé. Celle-ci est transmise "en clair" dans ce champ; la valeur de choix ASN.1 **secureChannel** est alors utilisée;
- si une clé et un algorithme de secret ont été établis en dehors du canal H.245 dans son ensemble (c'est-à-dire hors du flux H.323 ou sur un canal logique de type **h235Control**), le secret partagé est utilisé pour chiffrer les données de clé et la clé chiffrée résultante est insérée dans ce champ. Dans ce cas, la valeur de choix ASN.1 **sharedSecret** est utilisée;
- des certificats peuvent être utilisés lorsque le canal H.245 n'est pas sécurisé; mais ils peuvent aussi être employés en complément d'un canal H.245 sécurisé. Lorsque des certificats sont utilisés, les données de clé sont chiffrées au moyen de la clé publique du certificat et de la structure ASN.1 **certProtectedKey**.

A tout point d'une conférence, un récepteur (ou un émetteur) peut demander une nouvelle clé (par une demande de type **encryptionUpdateRequest**), par exemple parce qu'il suppose qu'il a perdu la synchronisation de l'un des canaux logiques. Le point maître qui reçoit cette demande doit produire

une ou des nouvelles clés en réponse à cette commande. Le maître peut également décider, de manière asynchrone, de distribuer une ou des nouvelles clés: il doit dans ce cas utiliser le message **encryptionUpdate**.

Après avoir reçu une demande **encryptionUpdateRequest**, un maître doit envoyer une mise à jour **encryptionUpdate**. Si la conférence est de type multipoint, le pont (en tant que maître) doit normalement distribuer la nouvelle clé à tous les récepteurs avant de la donner à l'émetteur. L'émetteur des données sur le canal logique doit utiliser la nouvelle clé au plus tôt après avoir reçu le message.

Un émetteur (supposé autre que le maître) peut également demander une nouvelle clé. Si l'émetteur fait partie d'une conférence multipoint, la procédure doit être la suivante:

- l'émetteur doit envoyer au pont (maître) la demande **encryptionUpdateRequest;**
- le pont doit produire une ou des nouvelles clés et envoyer un message **encryptionUpdate** à tous les participants de la conférence, sauf à l'émetteur;
- après avoir distribué les nouvelles clés à tous les autres participants, le pont doit envoyer le message **encryptionUpdate** à l'émetteur. Celui-ci doit alors utiliser la nouvelle clé.

12 Reprise sur erreur de sécurité

La présente Recommandation ne spécifie ni ne préconise de méthodes permettant aux points d'extrémité de surveiller le secret absolu de leurs communications. Elle recommande cependant des mesures à prendre lors de la détection d'une perte du secret des communications.

Si l'un des points d'extrémité détecte une brèche dans la sécurité du canal de connexion d'appel (par exemple un canal H.225.0 pour flux H.323), il doit immédiatement fermer la connexion conformément aux procédures protocolaires appropriées au point d'extrémité en question [pour un flux, selon 8.5/H.323, à l'exception de l'étape 5)].

Si l'un des points d'extrémité détecte une brèche dans la sécurité du canal H.245 ou du canal logique à données sécurisées (**h235Control**), il doit immédiatement fermer la connexion conformément aux procédures protocolaires appropriées au point d'extrémité en question [pour un flux H.323, selon le 8.5/H.323 à l'exception de l'étape 5)].

Si l'un des points d'extrémité détecte une perte du secret des communications sur l'un des canaux logiques, il doit immédiatement demander une nouvelle clé (par une demande **encryptionUpdateRequest**) et/ou fermer le canal logique. A la discrétion du pont de conférence, une perte de secret sur un canal logique peut causer la fermeture de tous les autres canaux logiques et/ou le recalcul de leurs clés. Le pont de conférence doit envoyer une demande de mise à jour **encryptionUpdateRequest** et une mise à jour **encryptionUpdate** à tous les points d'extrémités affectés.

A la discrétion du pont de conférence, une erreur de sécurité sur un canal individuel peut provoquer la fermeture des connexions à tous les points d'extrémité de la conférence – ce qui met fin à celle-ci.

ANNEXE A

Notation ASN.1 du protocole H.235

```
H235-SECURITY-MESSAGES DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

-- EXPORTS All

ChallengeString     ::= OCTET STRING (SIZE(8..128))
TimeStamp           ::= INTEGER(1..4294967295) -- seconds since 00:00 1/1/1970 UTC
RandomVal           ::= INTEGER
Password            ::= BMPString (SIZE (1..128))
Identifier          ::= BMPString (SIZE (1..128))
KeyMaterial         ::= BIT STRING(SIZE(1..2048))

NonStandardParameter ::= SEQUENCE
{
        nonStandardIdentifier     OBJECT IDENTIFIER,
        data                      OCTET STRING
}

-- if local octet representations of these bit strings are used they shall
-- utilize standard Network Octet ordering (e.g. Big Endian)
DHset ::= SEQUENCE
{
        halfkey             BIT STRING (SIZE(0..2048)), -- = g^x mod n
        modSize             BIT STRING (SIZE(0..2048)), --  n
        generator           BIT STRING (SIZE(0..2048)), -- g
        ...
}

TypedCertificate ::= SEQUENCE
{
        type                OBJECT IDENTIFIER,
        certificate         OCTET STRING,
        ...
}

AuthenticationMechanism ::=CHOICE
{
        dhExch              NULL, -- Diffe-Hellman
        pwdSymEnc           NULL, -- password with symmetric encryption
        pwdHash             NULL, -- password with hashing
        certSign            NULL, -- Certificate with signature
        ipsec               NULL, -- IPSEC based connection
        tls                 NULL,
        nonStandard         NonStandardParameter, -- something else.
        ...
}

ClearToken          ::= SEQUENCE  -- a "token" may contain multiple value types.
{
        timeStamp           TimeStamp OPTIONAL,
        password            Password OPTIONAL,
        dhkey               DHset OPTIONAL,
        challenge           ChallengeString OPTIONAL,
        random              RandomVal OPTIONAL,
        certificate         TypedCertificate OPTIONAL,
        generalID           Identifier OPTIONAL,
```

```
        nonStandard         NonStandardParameter OPTIONAL,
        ...
}

--
-- Start all the cryptographic parameterized types here...
--


SIGNED { ToBeSigned } ::= SEQUENCE {
  toBeSigned        ToBeSigned,
  algorithmOID      OBJECT IDENTIFIER,
  paramS            Params,       -- any "runtime" parameters
  signature         BIT STRING
} ( CONSTRAINED BY { -- Verify or Sign Certificate -- } )


ENCRYPTED { ToBeEncrypted } ::= SEQUENCE {
  algorithmOID              OBJECT IDENTIFIER,
  paramS                    Params,       -- any "runtime" parameters
  encryptedData             OCTET STRING
} ( CONSTRAINED BY { -- Encrypt or Decrypt -- ToBeEncrypted } )

HASHED { ToBeHashed } ::= SEQUENCE {
  algorithmOID              OBJECT IDENTIFIER,
  paramS                    Params,       -- any "runtime" parameters
  hash                      BIT STRING
} ( CONSTRAINED BY { -- Hash -- ToBeHashed } )

IV8 ::= OCTET STRING (SIZE(8))

-- signing algorithm used must select one of these types of parameters
-- needed by receiving end of signature.

Params ::= SEQUENCE {
        ranInt          INTEGER OPTIONAL, -- some integer value
        iv8             IV8 OPTIONAL,     -- 8 octet initialization vector
        ...
}

EncodedGeneralToken ::= TYPE-IDENTIFIER.&Type (ClearToken -- general usage token
PwdCertToken ::= ClearToken (WITH COMPONENTS {..., timeStampPRESENT, generalIDPRESENT})
EncodedPwdCertToken ::= TYPE-IDENTIFIER.&Type (PwdCertToken)

CryptoToken::= CHOICE
{

        cryptoEncryptedToken SEQUENCE -- General purpose/application specific token
        {
                tokenOID        OBJECT IDENTIFIER,
                token           ENCRYPTED { EncodedGeneralToken }
        },
        cryptoSignedToken  SEQUENCE -- General purpose/application specific token
        {
                tokenOID        OBJECT IDENTIFIER,
                token           SIGNED { EncodedGeneralToken }
        },
        cryptoHashedToken SEQUENCE -- General purpose/application specific token
        {
                tokenOID            OBJECT IDENTIFIER,
                hashedVals          ClearToken,
```

```
            token HASHED { EncodedGeneralToken }
        },
        cryptoPwdEncr       ENCRYPTED { EncodedPwdCertToken },
        ...
}

-- These allow the passing of session keys within the H.245 OLC structure.
-- They are encoded as standalone ASN.1 and based as an OCTET STRING within H.245
H235Key      ::=CHOICE  -- this is used with the H.245 "h235Key"' field
{
        secureChannel       KeyMaterial,
        sharedSecret        ENCRYPTED {EncodedKeySyncMaterial},
        certProtectedKey            SIGNED { EncodedKeySignedMaterial },
        ...
}

KeySignedMaterial ::= SEQUENCE {
        generalId           Identifier, -- slave's alias
        mrandom     RandomVal, -- master's random value
        srandom     RandomVal OPTIONAL, -- slave's random value
        timeStamp           TimeStamp OPTIONAL, -- master's timestamp for unsolicited EU
        encrptval           ENCRYPTED {EncodedKeySyncMaterial }
}
EncodedKeySignedMaterial ::= TYPE-IDENTIFIER.&Type (KeySignedMaterial)

H235CertificateSignature  ::=SEQUENCE
{
        certificate                 TypedCertificate,
        responseRandom              RandomVal,
        requesterRandom             RandomVal OPTIONAL,
        signature                   SIGNED { EncodedReturnSig },
        …
}

ReturnSig ::= SEQUENCE {
        generalId                   Identifier, -- slave's alias
        responseRandom              RandomVal,
        requestRandom               RandomVal OPTIONAL,
        certificate                 TypedCertificate OPTIONAL -- requested certificate
}

EncodedReturnSig ::= TYPE-IDENTIFIER.&Type (ReturnSig)
KeySyncMaterial    ::=SEQUENCE
{
        generalID           Identifier,
        keyMaterial         KeyMaterial,
        …
}
EncodedKeySyncMaterial ::=TYPE-INDENTIFIER.&Type (KeySyncMaterial)



END   -- End of H235-SECURITY-MESSAGES DEFINITIONS
```

ANNEXE B

Points spécifiques de la Recommandation H.323

B.1 Rappel

La Figure B.1 donne un aperçu général du domaine d'application de la présente Recommandation dans le cadre de la Recommandation H.323.

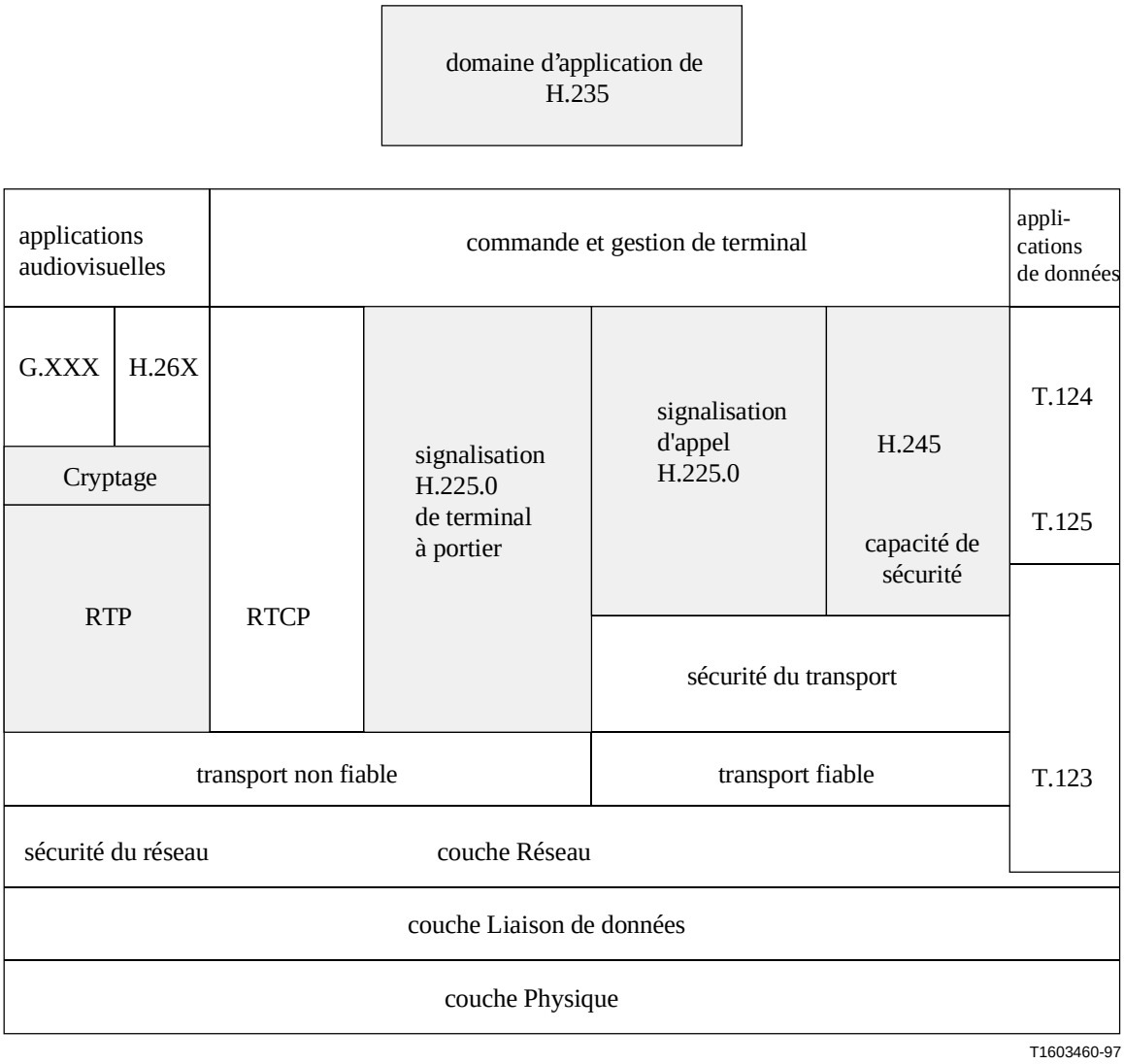


Figure B.1/H.235

Pour les flux H.323, la signalisation de l'utilisation du protocole TLS, IPSEC ou d'un mécanisme privé sur le canal de commande H.245 doit être effectuée sur le canal H.225.0 sécurisé ou non sécurisé, pendant l'échange initial de messages Q.931.

B.2 Signalisation et procédures

Les procédures décrites au paragraphe 8/H.323 (procédures de signalisation d'appel) doivent être suivies. Les points d'extrémité H.323 doivent avoir la capacité de coder et de reconnaître la présence (ou l'absence) de prescriptions de sécurité (pour le canal H.245) signalées dans les messages H.225.0.

Si le canal H.225.0 lui-même doit être sécurisé, les mêmes procédures qu'au paragraphe 8/H.323 doivent être suivies. La différence d'exploitation est que les communications ne doivent avoir lieu qu'après connexion à l'identificateur de point TSAP sécurisé et au moyen des modes de sécurité prédéterminés (TLS par exemple). Etant donné que les messages H.225.0 sont les premiers échangés lors de l'établissement de communications H.323, il ne peut pas y avoir de négociation de sécurité "dans la bande" pour les messages H.225.0. En d'autres termes, les deux parties doivent savoir *a priori* qu'elles vont utiliser un mode de sécurité particulier. Pour les flux H.323 en protocole IP, un autre accès notoire (1300) est utilisé pour les communications sécurisées par la méthode TLS.

Un des résultats des échanges H.225.0, dans la mesure où ils concernent la sécurité des flux H.323, est d'offrir un mécanisme permettant d'installer le canal H.245 sécurisé. Facultativement, l'authentification peut se produire pendant l'échange de messages H.225.0. Cette authentification peut être fondée sur des certificats ou sur des mots de passe, avec cryptage et/ou dispersion d'adressage (c'est-à-dire signature). Les particularités de ces modes de fonctionnement sont décrits aux 10.2 à 10.3.4.

Un point d'extrémité H.323 qui reçoit un message "d'établissement" SETUP avec la capacité **h245SecurityCapability** activée doit répondre en indiquant le mode acceptable correspondant (**h245SecurityMode)** dans le message de "connexion" CONNECT. Lorsqu'il n'y a pas de capacités correspondantes, le terminal appelé peut refuser la connexion en envoyant un message **Release Complete** avec le code de cause mis à ***SecurityDenied***. Cette erreur est destinée à n'acheminer aucune information sur une éventuelle discordance de sécurité: le terminal appelant devra déterminer le problème par un autre moyen. Lorsque le terminal appelant reçoit un message de "connexion" sans mode de sécurité suffisant ou acceptable, ce terminal peut mettre fin à l'appel par un message **Release Complete** avec le code de cause ***SecurityDenied***. Lorsque le terminal appelant reçoit un message de "connexion" sans aucune capacité de sécurité, ce terminal peut mettre fin à l'appel par un message **Release Complete** avec la cause ***undefinedReason***.

Si le terminal appelant reçoit un mode **h245Security** acceptable, il doit ouvrir et exploiter le canal H.245 dans le mode de sécurité indiqué. L'échec d'établissement du canal H.245 dans le mode de sécurité déterminé ici doit être considéré comme une erreur de protocole et la connexion doit être fermée.

B.2.1 Compatibilité avec la Révision 1

Un point d'extrémité possédant la capacité de sécurité ne doit pas renvoyer, à un point d'extrémité ne possédant pas la capacité de sécurité, de champs, d'indications ou d'états liés à la sécurité. Si un appelé reçoit un message "d'établissement" qui ne contient pas les capacités de sécurité **H245Security** ni/ou un jeton d'authentification, cet utilisateur peut renvoyer un message **ReleaseComplete** afin de refuser la connexion; mais dans ce cas il doit utiliser le code de cause ***UndefinedReason***. De manière analogue, si un appelant reçoit un message de "connexion" sans indication **H245SecurityMode** et/ou un jeton d'authentification ayant envoyé un message "d'établissement" avec **H245Security** et/ou un jeton d'authentification, cet utilisateur peut également mettre fin à la connexion en émettant un message **ReleaseComplete** avec le code de cause ***UndefinedReason***.

B.3 Liaisons avec les protocoles RTP/RTCP

L'utilisation du cryptage dans un flux en protocole de transport en temps réel RTP suivra la méthode générale qui a été recommandée dans le document indiqué par la commande [**RTP**]. Le cryptage du média doit être assuré sur une base indépendante, paquet par paquet[1]. L'en-tête RTP (y compris l'en-tête de charge utile) ne doit pas être crypté. La synchronisation de nouvelles clés et du texte chiffré est fondée sur une charge utile de type dynamique.

La clé de cryptage initiale est présentée par le maître en même temps que le numéro de charge utile dynamique (par un message **EncryptionSync** dans un canal H.245). Le ou les récepteurs du flux média doit commencer à utiliser la clé dès la réception de ce numéro de charge utile dans l'en-tête RTP. Une ou plusieurs nouvelles clés peuvent être distribuées à tout moment par le point d'extrémité maître. La synchronisation de la toute nouvelle clé avec le flux média doit être indiquée par la transition du type de charge utile à une nouvelle valeur dynamique. On notera que les valeurs spécifiques n'ont pas d'importance du moment qu'elle changent à chaque distribution d'une nouvelle clé.

L'on part du principe que le cryptage n'est appliqué qu'à la charge utile dans chaque paquet RTP; les en-têtes RTP restent en clair. On suppose que tous les paquets RTP sont un multiple entier d'octets. La façon dont les paquets RTP sont encapsulés dans la couche Transport ou Réseau est hors du domaine d'application de la présente Recommandation. Tous les modes doivent prévoir la perte (ou le déclassement) de paquets, ainsi que le bourrage de paquets pour qu'ils comportent un multiple approprié d'octets.

Le décryptage du flux doit être effectué sans tenir compte du contexte des états afin de tenir compte du fait que des paquets peuvent être perdus; chaque paquet doit donc être déchiffrable isolément. Deux exigences du mode algorithmique par blocs doivent s'appliquer comme suit:

a) vecteurs d'initialisation

La plupart des modes par blocs impliquent un certain "chaînage"; chaque cycle de cryptage dépend d'une certaine manière de la sortie du cycle précédent. Au début d'un paquet, une certaine valeur initiale de bloc [généralement appelée vecteur d'initialisation (IV, *initialization vector*)] doit donc être fournie afin de commencer le processus de cryptage. Quel que soit le nombre d'octets de flux qui sont traités à chaque cycle de cryptage, la longueur du vecteur d'initialisation est toujours égale à celle d'un bloc. Tous les modes, sauf le mode dictionnaire (ECB, *electronic code book*), nécessitent un vecteur d'initialisation. Dans tous les cas, un vecteur d'initialisation doit être construit à partir des B premiers octets (où B est la longueur de bloc) de la séquence (Seq# + pointeur temporel). Ce motif doit être répété jusqu'à ce qu'un nombre d'octets suffisant ait été produit. Il convient de noter que le vecteur d'initialisation produit de cette façon peut faire apparaître un motif de clé qui est considéré comme "faible" pour un algorithme particulier.

b) bourrage

Les modes ECB (dictionnaire électronique) et CBC (chaînage de blocs chiffrants) traitent toujours le flux d'entrée bloc par bloc. Alors que les modes CFB (rebouclage du cryptogramme) et OFB (rebouclage autoclave sur la sortie) peuvent traiter un nombre N ($\leq$ B) quelconque d'octets du flux d'entrée, il est recommandé que N = B.

Deux méthodes permettent de traiter les paquets dont la charge utile n'est pas un multiple de blocs:

---

[1] Il convient de noter que, si la longueur d'un paquet de protocole RTP est supérieure à celle d'une unité MTU, une perte partielle (d'un fragment) provoquera l'indéchiffrabilité de l'ensemble du paquet RTP.

1) l’emprunt cryptographique pour les modesECB et CBC; le bourrage à zéro pour les modes CFB et OFB;

2) le bourrage de la façon prescrite par la commande [**RTP**] (section 5.1).

Le protocole [**RTP**] (section 5.1) décrit une méthode de bourrage dans laquelle la charge utile est bourrée jusqu'à un multiple des blocs, le dernier octet indiquant le nombre d'octets de bourrage (y compris ce dernier octet) et le bit P étant activé dans l'en-tête RTP. La valeur du bourrage doit être déterminée par la convention normale de l'algorithme cryptographique.

Toutes les mises en œuvre conformes à la H.235 doivent prendre en charge les deux méthodes. La méthode utilisée peut être déduite comme suit: si le bit P est activé dans l'en-tête RTP, le paquet est bourré; si le paquet n'est pas un multiple de la longueur B et que le bit P ne soit pas activé, la méthode d'extraction cryptographique s'applique. Sinon, le paquet est un multiple de B et le bourrage ne s'applique pas.

La protection du flux RTP contre les atteintes à l'intégrité et les réexécutions fera l'objet d'un complément d'étude.

L'application des techniques cryptographiques aux éléments du protocole de commande en temps réel (RTCP) fera l'objet d'un complément d'étude.

B.4 Procédures et signalisation des messages d'enregistrement, admission et état (RAS) pour l’authentification

B.4.1 Introduction

La présente annexe n'indiquera explicitement aucune forme de secret des messages échangés entre portiers et points d'extrémité. Il existe deux types d’authentification pouvant être utilisés. Le premier type est fondé sur un cryptage symétrique ne nécessitant aucun contact préalable entre le point d'extrémité et le portier. Le deuxième type est fondé sur un abonnement et aura deux formes: mot de passe ou certificat. Toutes ces formes sont issues des procédures indiquées aux 10.2, 10.3.2, 10.3.3 et 10.3.4. Dans la présente annexe, les étiquettes génériques (des points EPA et EPB), indiquées dans les sous-paragraphes précédents, représenteront respectivement le point d'extrémité et le portier.

B.4.2 Authentification entre point d'extrémité et portier (non fondée sur abonnement)

Ce mécanisme peut fournir au portier une indication cryptographique selon laquelle un point d'extrémité particulier, qui s'est préalablement enregistré, est bien celui qui émettra les messages RAS ultérieurs. Il y a lieu de noter que ce procédé peut ne pas fournir au point d'extrémité une quelconque authentification du portier, à moins que l'élément facultatif de signature soit inclus. L'établissement de la relation d'identité s'effectue lorsque le terminal émet la demande **GRQ**, comme indiqué au 7.2.1/H.323. L'échange selon la méthode Diffie-Hellman s'effectue conjointement avec les messages **GRQ** et **GCF**, comme indiqué dans la première phase du 10.2. Cette clé à secret partagé doit ensuite être utilisée pour toute demande **RRQ/URQ** subséquente, envoyée par le terminal au portier. Si un portier fonctionne dans ce mode et reçoit une demande **GRQ** sans jeton contenant la valeur *DHset* ou une valeur algorithmique acceptable, ce portier doit renvoyer, dans le message de rejet **DRJ**, le code de cause **securityDenial**.

La clé à secret partagé qui a été créée par la méthode Diffie-Hellman au cours de l'échange de messages **GRQ/GCF** peut être utilisée pour l’authentification dans d'ultérieures messages de type **xRQ**. Les procédures suivantes doivent être utilisées pour réaliser ce mode d’authentification.

Terminal (**xRQ**):

1) le terminal doit fournir toutes les informations contenues dans le message, comme décrit dans les sous-paragraphes appropriés de la Recommandation H.225.0;
2) le terminal doit chiffrer l'identificateur du portier **GatekeeperIdentifier** (renvoyé dans le message **GCF**) au moyen de la clé à secret partagé qui a été négociée. Ce cryptogramme doit être transmis dans un jeton cryptographique **cryptoToken** en tant qu'identificateur général **generalID**.

Les 16 bits du nombre aléatoire **random** puis du numéro **requestSeqNum** doivent être combinés par un opérateur OUX avec chacun des 16 bits de l'identificateur de portier **GatekeeperIdentifier**. Si cet identificateur **GatekeeperIdentifier** ne se termine pas par une limite paire à la 16e position, les 8 derniers éléments binaires de l'identificateur de portier **GatekeeperIdentifier** doivent être combinés par un opérateur OUX avec l'octet de plus faible poids de la valeur aléatoire puis avec le numéro **requestSeqNum**. L'identificateur de portier **GatekeeperIdentifier** doit être chiffré au moyen de l'algorithme sélectionné dans le message **GCF** (intégrité) et au moyen de l'ensemble du secret partagé.

Afin de relier cryptographiquement ce message et les messages ultérieurs avec l'entité qui s'est enregistrée initialement (le point d'extrémité qui a émis la demande **RRQ**), la plus récente valeur aléatoire **random** renvoyée doit être utilisée (cette valeur peut être plus récente que celle qui a été renvoyée dans le message **RCF** faisant suite à un message **xCF** ultérieur).

Portier (**xCF/xRJ**)

1) le portier doit chiffrer son identificateur **GatekeeperIdentifier**(conformément à la procédure ci-dessus) avec la clé à secret partagé qui est associée à l'alias du point d'extrémité; il doit ensuite le comparer à la valeur contenue dans la demande **xRQ**;
2) le portier doit renvoyer un message de rejet **xRJ** si les deux valeurs chiffrées ne correspondent pas;
3) si son identificateur **GatekeeperIdentifier** correspond à la valeur demandée, le portier doit appliquer toute logique locale éventuelle puis répondre par un message **xCF** ou **xRJ**;
4) si un message **xCF** est envoyé par le portier, ce message doit contenir un identificateur de point d'extrémité **EndpointIdentifier** assigné et une nouvelle valeur aléatoire dans le champ **random** d'un paramètre **clearToken**.

Pour la représentation graphique de cet échange, voir la phase 2 de la Figure 1 contenue au 10.2. Le portier connaît la clé à secret partagé qu'il faut utiliser pour déchiffrer l'identificateur de portier indiqué dans le nom d'alias du message.

B.4.3 Authentification entre point d'extrémité et portier (fondée sur abonnement)

Tous les messages RAS autres que GRQ/GCF doivent normalement contenir les jetons d'authentification requis par le mode de fonctionnement spécifique. Il existe trois variantes différentes qui peuvent être mise en œuvre, selon les exigences et l'environnement:

1) authentification par mot de passe avec cryptage symétrique;
2) authentification par mot de passe avec dispersion d'adresse;
3) authentification par certificat avec signatures.

Dans tous les cas, le jeton contiendra les informations décrites dans les sous-paragraphes suivants, selon la variante choisie. Si un portier fonctionne en mode sécurisé et reçoit un message RAS sans valeur de jeton acceptable, il doit renvoyer un code de cause **securityDenial** dans le message de rejet. Dans tous les cas, le jeton renvoyé par le portier est facultatif: s'il est omis, seule une authentification à sens unique est effectuée.

B.4.3.1 Mot de passe avec cryptage symétrique

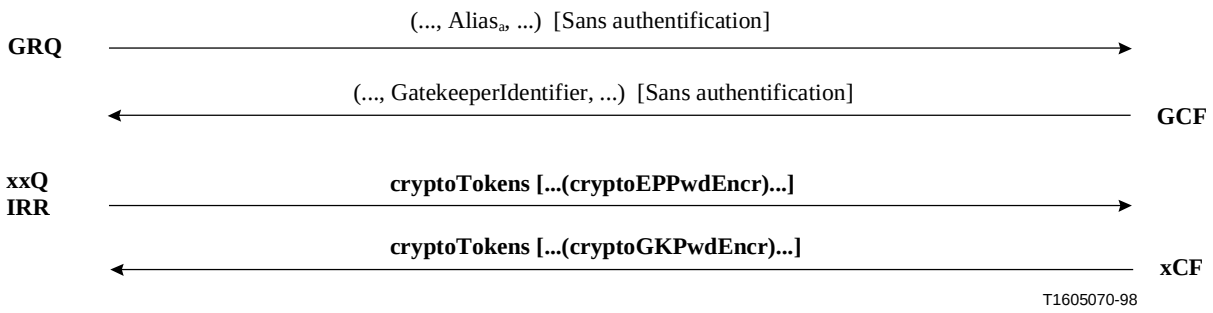


Figure B.2/H.235

B.4.3.2 Mot de passe avec dispersion d'adresse

L'on part du principe qu'un alias et le mot de passe associé sont échangés hors bande pour cet échange de messages particulier.

Figure B.3/H.235

B.4.3.3 Authentification par certificat avec signatures

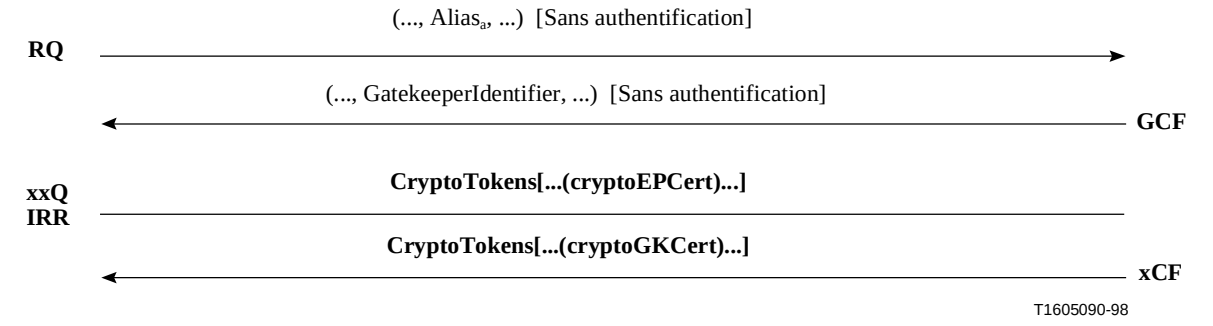


Figure B.4/H.235

B.5 Interactions non terminales

B.5.1 Passerelle

Comme indiqué au 6.6, une passerelle H.323 doit être considérée comme un élément crédibilisé. Cela comprend les passerelles entre protocoles (H.323-H.320, etc., ...) et les passerelles de sécurité (serveurs tampons/pare-feu). Le secret des communications multimédias peut être assuré entre le point d'extrémité communicant et la passerelle tête de ligne. Mais ce qui se produit au-delà de la passerelle doit être considéré *a priori* comme non sécurisé.

ANNEXE C

Points spécifiques de la Recommandation H.324

A étudier.

APPENDICE I

Détails de mise en œuvre H.323

I.1 Méthodes de bourrage cryptographique

Il existe une description d'emprunt cryptographique dans [**Schneier**], p. 191 et 196. Les Figures I.1 à I.5 illustrent cette technique.

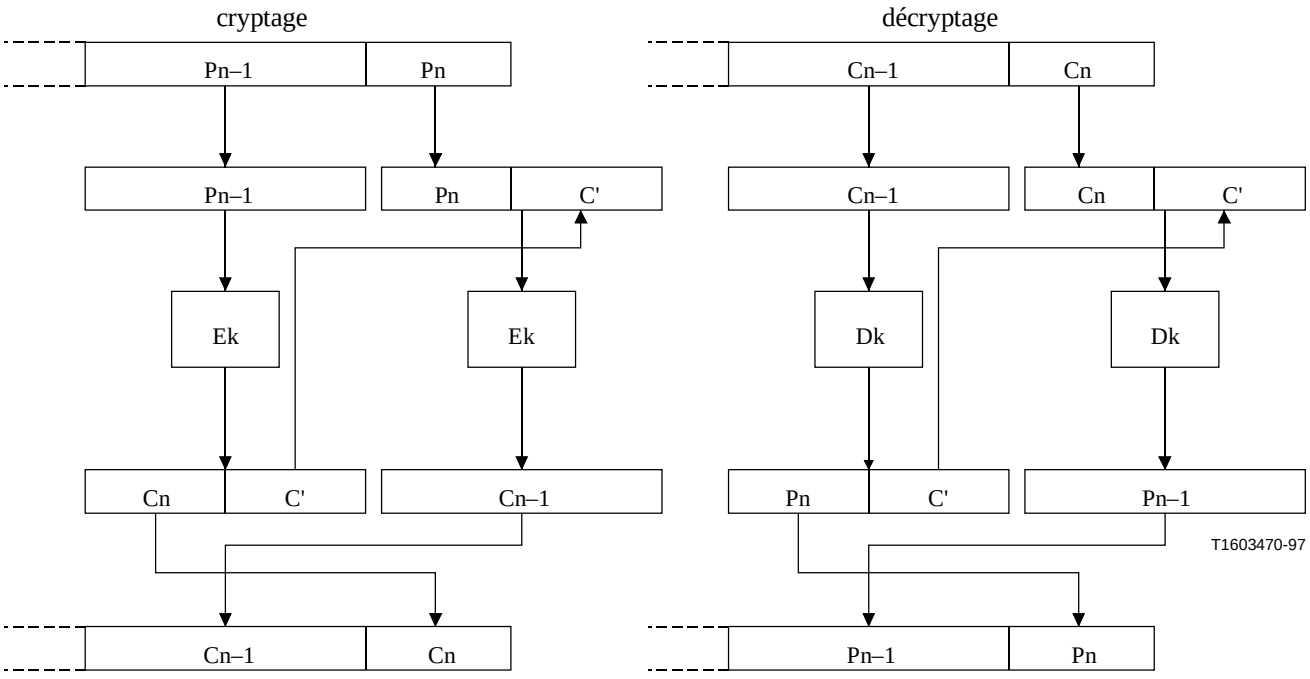


Figure I.1/H.235 – Interception cryptographique en mode ECB

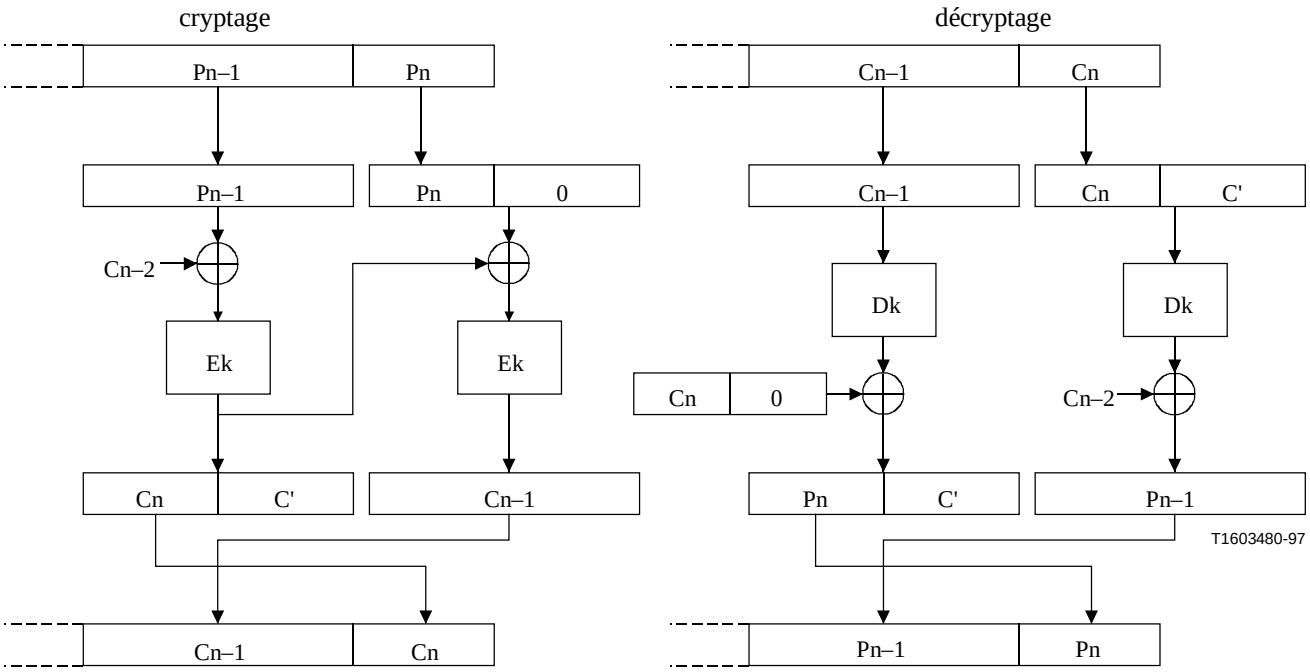


Figure I.2/H.235 – Interception cryptographique en mode CBC

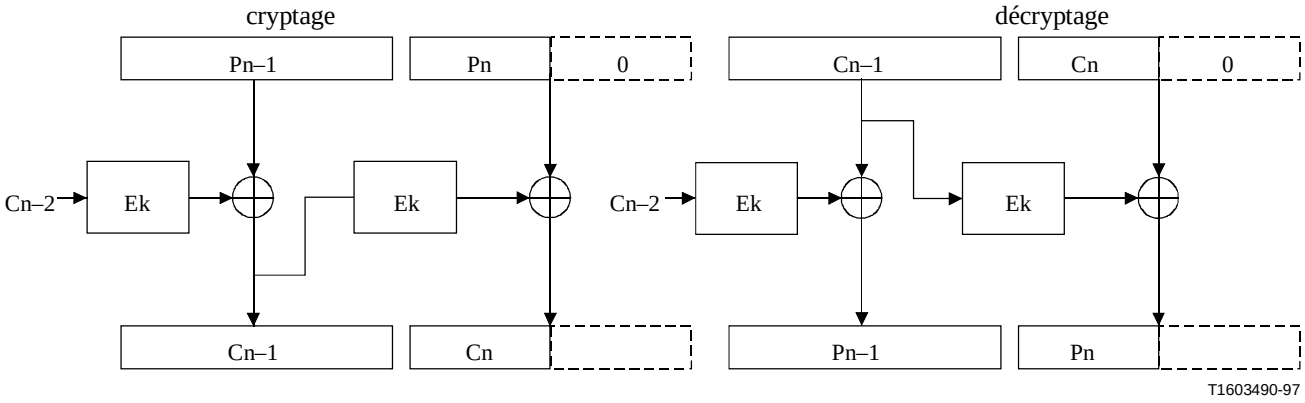
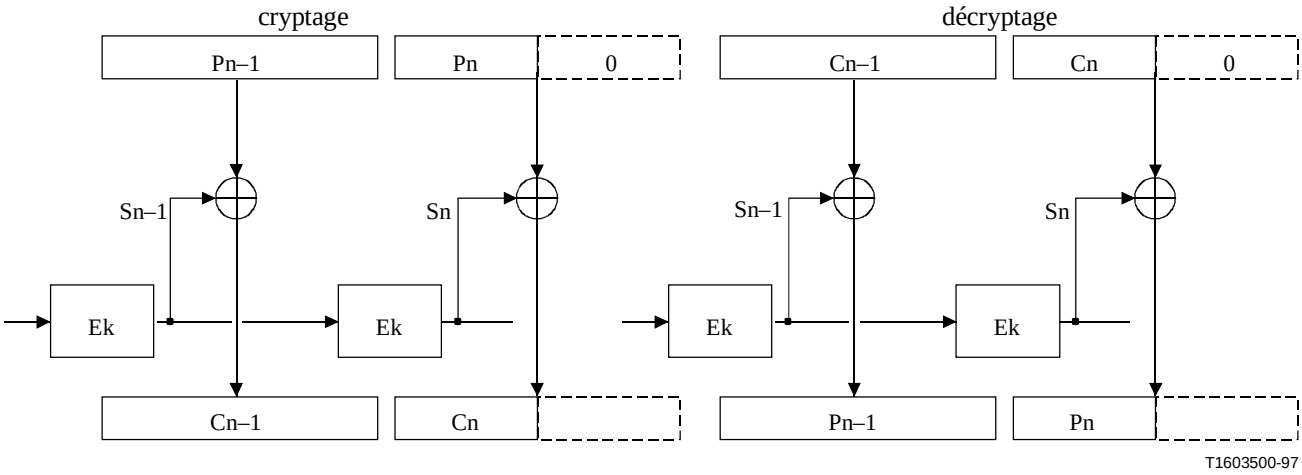


Figure I.3/H.235 – Bourrage de zéros en mode CFB



NOTE – Le signal Si est le résultat de cryptages répétitifs (c'est-à-dire de permutations) du vecteur d'initialisation.

Figure I.4/H.235 – Bourrage de zéros en mode OFB

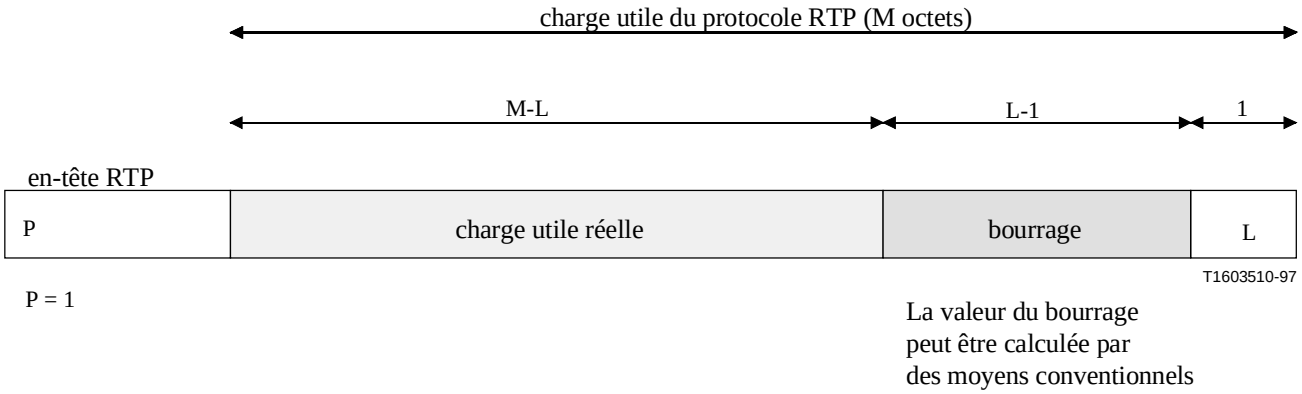


Figure I.5/H.235 – Bourrage tel que prescrit par le protocole RTP

I.2 Nouvelles clés

Les procédures décrites au 8.5/H.323 sont appliquées par un pont de conférence afin d'éjecter un participant d'une conférence. Le point maître peut produire de nouvelles clés de chiffrage pour les canaux logiques (et ne pas les distribuer au correspondant éjecté); cette méthode peut être utilisée afin d'empêcher le correspondant éjecté de surveiller les flux médias.

I.3 Eléments crédibilisés H.323

En général, les ponts de conférence, les passerelles et les portiers (s'ils mettent en œuvre le modèle acheminé par portier) sont des éléments crédibilisés pour ce qui est du secret des communications par le canal de commande. Si le canal d'établissement des connexions (H.225.0) est sécurisé *et* acheminé par l'entremise du portier, l'on doit également s'y fier. Si l'un de ces éléments H.323 doit fonctionner avec les flux médias (c'est-à-dire pour un mixage, un transcodage), ils doivent alors, par définition, être aussi considérés comme crédibles pour le secret des communications médias.

Les serveurs tampons ou pare-feu (bien que ne constituant pas des éléments spécifiques de la Recommandation H.323) peuvent aussi être crédibilisés, car ils terminent des connexions et peuvent tout à fait avoir à manipuler les messages et les flux médias.

I.4 Exemples de mise en œuvre

Les sous-paragraphes suivants décrivent des exemples de mise en œuvre qui pourraient être développés dans le cadre de H.235. Ils ne sont pas destinés à prendre le pas sur les nombreuses autres possibilités proposées dans la présente Recommandation. Ces paragraphes visent plutôt à donner des exemples plus concrets d'utilisation dans le cadre de la Recommandation H.323.

I.4.1 Jetons

Le présent sous-paragraphe décrira un exemple d'utilisation de jetons de sécurité afin d'occulter ou de masquer les informations d'adressage de destination. Le scénario donné en exemple est un point d'extrémité qui souhaite établir une communication avec un autre point d'extrémité utilisant son alias notoire. Plus précisément, le réseau H.323 se compose d'un point d'extrémité A, d'un portier, d'une passerelle avec le RTC et d'un poste téléphonique B, comme illustré ci-dessous.

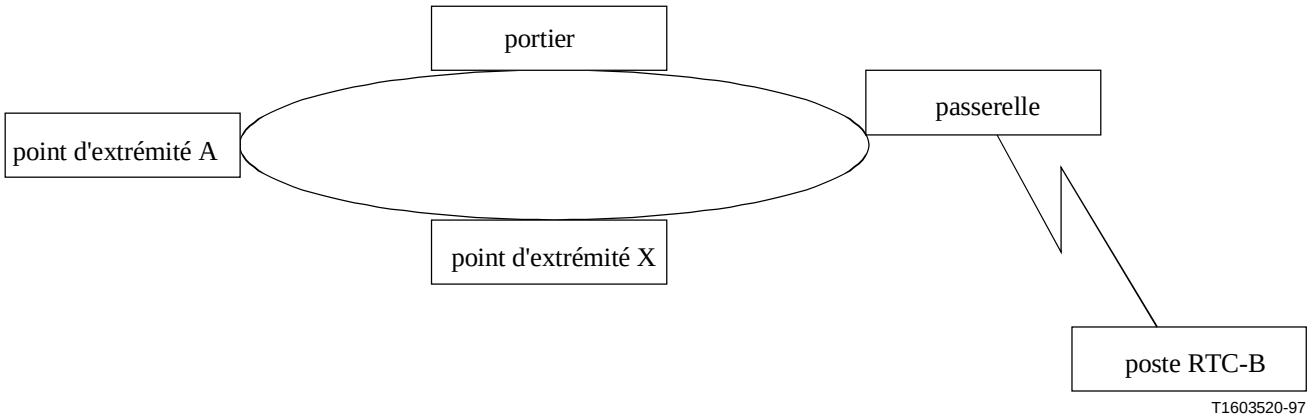


Figure I.6/H.235

Actuellement, un réseau H.323 peut fonctionner de façon analogue à un réseau téléphonique avec identification de l'appelant. Ce scénario illustre une situation dans laquelle l'*appelé* ne souhaite pas divulguer son adresse physique tout en acceptant l'établissement de l'appel. Cela peut être important dans les passerelles RTC-H.323, lorsque le numéro téléphonique de destination peut devoir rester privé.

Supposons que le point A essaye d'appeler le point RTC-B et que celui-ci ne souhaite pas divulguer au point A son numéro de téléphone selon le plan E.164. (La façon dont cette politique est établie est hors du domaine d'application de cet exemple.)

- le point EPA enverra une demande ARQ à son portier pour résoudre l'adresse du poste RTC, tel que représenté par son alias/passerelle. Le portier reconnaîtra cette adresse comme un alias "privé", sachant que pour réaliser la connexion il doit renvoyer l'adresse de la passerelle avec le RTC. (Ce cas est analogue à celui du renvoi d'adresse d'une passerelle H.320 si un point d'extrémité H.320 est appelé par un point H.323.);

- dans le message ACF renvoyé, le portier renvoie l'adresse de la passerelle avec le RTC, comme prévu. Les informations d'adressage qui sont nécessaires pour appeler le poste distant (c'est-à-dire le numéro de téléphone) sont renvoyées dans un jeton codé dans le message ACF. Ce jeton crypté contient le numéro E.164 réel (de téléphone) du poste, qui ne peut pas être déchiffré ni compris par l'appelant (c'est-à-dire le point EPA);
- le point d'extrémité envoie à la passerelle tête de ligne (dont l'adresse de signalisation d'appel a été renvoyée par le message ACF) un message SETUP contenant le ou les jetons opaques qu'il a reçus dans l'ACF.
- Dès qu'elle reçoit le message SETUP, la passerelle envoie sa demande ARQ à son portier, y compris tous jetons reçus dans le message SETUP;
- le portier est en mesure de déchiffrer le ou les jetons et de renvoyer le numéro de téléphone dans le message ACF.

Une partie de la notation ASN.1 d'une structure de jeton est montrée ci-dessous à titre d'exemple, avec description du contenu des champs. L'on suppose que l'on utilise le paramètre **cryptoEncodedGeneralToken** pour y insérer le numéro de téléphone crypté.

Une mise en œuvre peut choisir un identificateur d'objet jeton, **tokenOID**, pour indiquer que ce jeton contient le numéro de téléphone E.164. La méthode particulière qui sera utilisée pour coder ce numéro de téléphone (par exemple une norme DES à 56 bits) sera incluse dans la définition "ENCRYPT' contenue dans l'identificateur d'algorithme, **algorithmOID**.

```
CryptoToken::= CHOICE
{
        cryptoEncodedGeneralToken SEQUENCE   -- Jeton à usage général ou à application spécifique
        {
                tokenOID  OBJECT IDENTIFIER,
                ENCRYPTED { EncodedGeneralToken }
        },
.
.
. [abbreviated text]
.

}
```

Le message **CryptoToken** sera transmis dans le message SETUP (du point EPA à la passerelle) et les messages **ARQ** (de la passerelle au portier) seront transmis comme indiqué ci-dessus. Après avoir déchiffré le jeton (numéro de téléphone), le portier en transmet la version en clair dans le paramètre **clearToken**.

I.4.2 Mot de passe

Dans cet exemple, l'on suppose que l'utilisateur est abonné au service de portier (c'est-à-dire qu'il se trouve dans la zone de celui-ci) et qu'il possède un identificateur d'abonnement et un mot de passe correspondants. Cet utilisateur va s'enregistrer auprès du portier en utilisant son identificateur d'abonnement (tel qu'il a été transmis dans un identificateur d'alias H.323) et en chiffrant une chaîne d'interrogation rédhibitoire qui lui sera présentée par le portier. Ce processus suppose que le portier connaît également le mot de passe associé à l'identificateur d'abonnement. Le portier authentifiera l'utilisateur en vérifiant que la chaîne d'interrogation a été correctement chiffrée.

La procédure d'enregistrement avec authentification par portier sera la suivante pour cet exemple:

1) si le point d'extrémité utilise une demande **GRQ** pour découvrir un portier, un des alias contenus dans le message se trouvera dans l'identificateur d'abonnement (sous forme d'identificateur **H323ID**). Le message **authenticationcapability** contiendra un mécanisme d'authentification (**AuthenticationMechanism**) par codage de mot de passe (**pwdSymEnc)** et les identificateurs d'algorithme (**algorithmOID**) seront paramétrés de façon à indiquer l'ensemble complet des algorithmes de cryptage pris en charge par le point d'extrémité. (Par exemple, l'un de ces algorithmes sera la norme DES à 56 bits en mode EBC);

2) le portier répondra à ce message par une confirmation **GCF** (en supposant qu'il reconnaît l'alias) acheminant un élément **tokens** contenant un seul jeton en clair, **ClearToken**. Celui-ci se composera de deux parties: une interrogation rédhibitoire (**challenge**) et un pointeur temporel **TimeStamp**. L'interrogation **challenge** sera codée sur 16 octets (pour prévenir les attaques par réexécution, le jeton en clair **clearToken** contiendra un élément **timeStamp**). Le mode d'authentification **authenticationmode** sera mis à **pwdSymEnc** et l'identificateur d'algorithme **algorithmOID** indiquera l'algorithme de cryptage requis par le portier (par exemple norme DES à 56 bits en mode EBC).

   Si le portier ne prend en charge aucun des identificateurs d'algorithme **algorithmOID** indiqués dans la demande **GRQ**, il répondra par un rejet **GRJ** contenant une cause **GatekeeperRejectReason** égale à **ressourceUnavailable**;

3) l'application du point d'extrémité tentera alors de s'enregistrer auprès du (d'un des) portier(s) ayant répondu par une confirmation **GCF**, en envoyant une demande **RRQ** contenant un élément **cryptoEPPwdEncr** dans le paramètre **cryptoTokens**. Cet élément **cryptoEPPwdEncr** contiendra l'identificateur de l'algorithme de cryptage **algorithmOID** convenu lors de l'échange de messages **GRQ/GCF**, ainsi que l'interrogation rédhibitoire cryptée.

   La clé de cryptage est construite sur la base du mot de passe de l'utilisateur, au moyen de la procédure décrite au 10.3. La "chaîne" d'octets résultante sera alors utilisée comme clé DES pour chiffrer l'interrogation rédhibitoire;

4) lorsque le portier reçoit l'interrogation rédhibitoire dans la demande **RRQ**, il la compare à une interrogation rédhibitoire déjà chiffrée à l'identique, afin d'authentifier l'utilisateur requérant. Si les deux chaînes chiffrées ne correspondent pas, le portier répond par un message de rejet **RRJ** avec la cause **RegistrationRejectReason** mise à la valeur **securityDenial**. Si les chaînes correspondent, le portier envoie une confirmation **RCF** au point d'extrémité;

5) si le portier reçoit une demande **RRQ** qui ne contient pas d'élément **cryptoTokens** acceptable, il doit répondre par un rejet **RRJ** avec la cause **GatekeeperRejectReason** mise à la valeur **discoveryRequired**. Le point d'extrémité, dès qu'il reçoit ce message **RRJ**, peut exécuter la recherche qui permettra au couple portier/point d'extrémité d'échanger une nouvelle interrogation rédhibitoire. On notera que le message **GRQ** peut être envoyé en mode point à point au portier.

I.4.3 Sécurité IPSEC

En général, la méthode IPSEC [**13/IPSEC**] peut être utilisée pour assurer l'authentification et, facultativement, la confidentialité (c'est-à-dire le cryptage) dans la couche IP de façon transparente à tout protocole (applicatif) exploité dans les couches supérieures. Le protocole applicatif n'a pas besoin d'être mis à jour pour permettre cette opération; seule la politique de sécurité à chaque extrémité doit correspondre.

Par exemple, pour tirer le meilleur parti de la sécurité IPSEC pour une simple communication point à point, le scénario ci-après peut être suivi:

1) le point d'extrémité appelant et son portier détermineront par le protocole RAS la politique prescrivant l'utilisation de la sécurité IPSEC (authentification et, facultativement, confidentialité). Avant l'envoi du premier message RAS du point d'extrémité au portier, la routine ISAKMP/Oakley située au point d'extrémité négociera les services de sécurité à utiliser pour les paquets à destination et en provenance de l'accès notoire du canal RAS. Une fois la négociation achevée, le canal RAS fonctionne exactement comme s'il n'avait pas été sécurisé. Au moyen de ce canal sécurisé, le portier informera le point d'extrémité de l'adresse et du numéro d'accès du canal de signalisation d'appel se trouvant au point d'extrémité appelé;

2) après avoir obtenu l'adresse et le numéro d'accès du canal de signalisation d'appel, le point d'extrémité appelant met à jour dynamiquement sa politique de sécurité afin de demander la sécurité IPSEC souhaitée à cette adresse pour cette paire protocole/accès. Ensuite, lorsque le point d'extrémité appelant tentera de se mettre en contact avec cette paire adresse/accès, les paquets seront mis en file d'attente pendant l'exécution d'une négociation par routine ISAKMP/Oakley entre les points d'extrémité. A l'achèvement de cette négociation, une association de sécurité IPSEC existera pour cette paire adresse/accès et la signalisation Q.931 pourra commencer;

3) lors de l'échange des messages Q.931 SETUP et CONNECT, les points d'extrémité peuvent négocier l'utilisation de la sécurité IPSEC pour le canal H.245. Cela permettra aux points d'extrémité de remettre à jour dynamiquement leurs bases de données pour politique de sécurité IPSEC et d'imposer l'utilisation de cette politique sur cette connexion;

4) comme dans le canal de signalisation d'appel, une négociation ISAKMP/Oakley transparente se déroulera avant qu'un quelconque paquet H.245 soit émis. L'authentification effectuée par cet échange ISAKMP/Oakley sera la tentative initiale d'une authentification d'utilisateur à utilisateur. Elle établira un canal (probablement) sécurisé entre les deux utilisateurs, permettant de négocier les caractéristiques du canal audio. Si, à la suite d'un dialogue interpersonnel, l'un des utilisateurs n'est pas satisfait de l'authentification, différents certificats peuvent être choisis et l'échange ISAKMP/Oakley peut être répété;

5) après chaque authentification ISAKMP/Oakley H.245, de nouvelles données de clé sont échangées pour le canal audio en protocole RTP. Ces données sont distribuées par le maître sur le canal H.245 sécurisé. Comme le protocole H.245 est défini de façon que le maître distribue les données de clés multimédias sur le canal H.245 (afin de permettre des communications multipoints), il n'est pas recommandé d'utiliser la méthode IPSEC pour le canal RTP.

Un canal H.245 crypté peut poser un problème pour les serveurs tampons ou les pare-feu NAT car les numéros d'accès attribués dynamiquement sont acheminés dans le protocole H.245. Pour fonctionner correctement, de tels pare-feu devront déchiffrer, modifier et rechiffrer le protocole. C'est pourquoi le canal logique "de sécurité" a été introduit dans la Recommandation H.245. Si ce canal est utilisé, le canal H.245 peut rester non sécurisé; l'authentification et la production de clés seront effectuées avec le canal logique "de sécurité". La signalisation par canal logique permettra de protéger ce canal par la méthode IPSEC et la clé à secret utilisée dans le canal logique "de sécurité" servira à protéger la synchronisation (**EncryptionSync**) distribuée par le maître sur le canal H.245.

APPENDICE II

Détails de mise en œuvre H.324

A étudier.

APPENDICE III

Autres détails de mise en œuvre pour la série H

A étudier.

APPENDICE IV

Bibliographie

[Daemon]

– DAEMON (J.): Cipher and Hash function design, Ph.D. Thesis, Katholieke Universiteit Leuven, mars 1995.

[IPSEC]

– ORMAN (H.K.): The Oakley Key Determination Protocol, draft-ietf-ipsec-oakley-02.txt, *Internet Engineering Task Force*, 1997.

– MAUGHAN (D.), SCHERTLER (M.), SCHNEIDER (M.), TURNER (J.): Internet Security Association and Key Management Protocol (ISAKMP), draft-ietf-ipsec-isakmp-08.text, *Internet Engineering Task Force*, 1997.

– KENT (S.), ATKINSON (R.): IP Authentication Header, draft-ietf-ipsec-auth-header-01.txt, *Internet Engineering Task Force*, 1997.

– HARKINS (D.), CARREL (D.): The resolution os ISAKMP with Oakley, draft-ietf-ipsec-isakmp-oakley-04.txt, *Internet Engineering Task Force*, 1997.

[RTP]

– SCHULZRINNE (H.), CASNER (S.), FREDERICK (R.), JACOBSON (V.): RTP: A Transport Protocol for Real-Time Applications, RFC 1889, *Internet Engineering Task Force*, 1996.

[Schneier]

– SCHNEIER (B.): Applied Cryptography: Protocols, Algorithms, and Source Code in C, 2nd Edition, John Wiley & Sons, Inc., 1995.

[TLS]

– DIEKS (T.), ALLEN (C.): The TLS Protocol Version 1.0, draft-ietf-tls-protocol-03.txt, *Internet Engineering Task Force*, 1997.

SERIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série B	Moyens d'expression: définitions, symboles, classification
Série C	Statistiques générales des télécommunications
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	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	RGT et maintenance des réseaux: systèmes de transmission, de télégraphie, de télécopie, circuits téléphoniques et circuits loués internationaux
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 pour données et communication entre systèmes ouverts
Série Y	Infrastructure mondiale de l'information
Série Z	Langages de programmation