Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

H.235.4

(09/2005)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Infrastructure des services audiovisuels – Aspects système

Cadre de sécurité H.323: sécurité des appels à routage direct et des appels à routage sélectif

Recommandation UIT-T H.235.4

RECOMMANDATIONS UIT-T DE LA SÉRIE H

SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

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

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

Recommandation UIT-T H.235.4

Cadre de sécurité H.323: sécurité des appels à routage direct et des appels à routage sélectif

Résumé

L'objet de la présente Recommandation est de recommander des procédures de sécurité pour l'utilisation de la signalisation des appels à routage direct en association avec les profils de sécurité H.235.1 et H.235.3. Ce profil de sécurité est proposé en option et peut venir compléter les profils de sécurité des Recommandations UIT-T H.235.1 et H.235.3. En outre, la présente Recommandation donne des détails sur l'implémentation du § 8.4/H.235.0 avec des techniques de gestion de clés symétriques.

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

Source

La Recommandation UIT-T H.235.4 a été approuvée le 13 septembre 2005 par la Commission d'études 16 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8.

Mots clés

Authentification, chiffrement, gestion de clés, intégrité, profil de sécurité, sécurité multimedia, sécurité des appels à routage direct, sécurité des appels à routage sélectif.

AVANT-PROPOS

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

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

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

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

NOTE

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

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

DROITS DE PROPRIÉTÉ INTELLECTUELLE

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

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

TABLE DES MATIÈRES

Recommandation UIT-T H.235.4

Cadre de sécurité H.323: sécurité des appels à routage direct et des appels à routage sélectif

1 Domaine d'application

L'objet de la présente Recommandation est de recommander des procédures de sécurité pour l'utilisation de la signalisation des appels à routage direct en association avec les profils de sécurité H.235.1 et H.235.3.

Ce profil de sécurité est proposé en option et peut venir compléter les profils de sécurité H.235.1 et H.235.3. En outre, la présente Recommandation donne des détails sur l'implémentation du § 8.4/H.235.0 avec des techniques de gestion de clés symétriques.

2 Références

2.1 Références normatives

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

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

– Recommandation UIT-T H.235 (2003), *Sécurité et chiffrement pour les terminaux multimédias de la série H (terminaux H.323 et autres terminaux de type H.245)*, Corrigendum 1 (2005), plus Erratum 1 (2005).

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

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

– Recommandation UIT-T H.235.3 (2005), *Cadre de sécurité H.323: profil de sécurité hybride*.

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

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

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

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

– ISO/CEI 10118-3:2004, *Technologies de l'information – Techniques de sécurité – Fonctions de brouillage – Partie 3: Fonctions de brouillage dédiées*.

2.2 Références informatives

– Recommandation UIT-T H.235.2 (2005), *Cadre de sécurité H.323: profil de sécurité avec signature*.

– IETF RFC 4120 (2005), *The Kerberos Network Authentication Service (V5)*.

3 Termes et définitions

Dans la présente Recommandation, les définitions figurant au § 3 des Recommandations UIT-T H.323, H.225.0, H.235.0 et X.800 | ISO/CEI 7498-2 s'appliquent, en plus de celles du présent paragraphe.

4 Symboles et abréviations

La présente Recommandation utilise les abréviations suivantes:

CT	ClearToken
DH	Diffie-Hellman
DRC	appel à routage direct (*direct-routed call*)
EK_{AG}	clé de chiffrement partagée entre le point d'extrémité A et le portier G
EK_{BH}	clé de chiffrement partagée entre le point d'extrémité B et le portier H
EK_{GH}	clé de chiffrement partagée entre le portier G et le portier H
$ENC_{K;\,S,\,IV}(M)$	chiffrement EOFB de *M* au moyen de la clé secrète *K*, de la clé de salage secrète *S* et du vecteur initial *IV*
EPID	identificateur de point d'extrémité (*endpoint identifier*)
GK	portier (*gatekeeper*)
GKID	identificateur de portier (*gatekeeper identifier*)
g^x, g^y	demi-clé Diffie-Hellman du portier G, du portier H
K_{AB}	clé de chiffrement partagée entre le point d'extrémité A et le point d'extrémité B
K_{AG}	secret partagé (H.235.1, H.235.3) entre le point d'extrémité A et le portier G
K_{BH}	secret partagé (H.235.1, H.235.3) entre le point d'extrémité B et le portier H
K_{GH}	secret partagé (H.235.1, H.235.3) entre le portier G et le portier H
KS_{AG}	clé de salage partagée secrète entre le point d'extrémité A et le portier G
KS_{BH}	clé de salage partagée secrète entre le point d'extrémité B et le portier H
KS_{GH}	clé de salage partagée secrète entre le portier G et le portier H
PRF	fonction pseudo-aléatoire (*pseudo-random function*)

5 Conventions

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

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

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

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

Les identificateurs d'objet sont indiqués par un symbole dans le texte (par exemple "I11"); le § 14 donne la liste des valeurs numériques réelles correspondant aux symboles d'identificateur d'objet (voir également le § 5/H.235.0).

6 Introduction

Pour la mise en œuvre de la Rec. UIT-T H.323, on utilise souvent le modèle de routage par portier (par exemple, pour tirer parti de meilleures fonctionnalités de facturation). La large utilisation des modèles d'appel à routage par portier est aussi la raison pour laquelle différents profils de sécurité reposant précisément sur ce type de modèle d'appel sont définis dans la Rec. UIT-T H.235.0 (tels que les profils H.235.1, H.235.2, H.235.3).

Toutefois, étant donné la nécessité de prendre en charge un nombre croissant de canaux parallèles, le modèle d'appel à routage direct avec un portier peut conduire à de meilleures performances et à de meilleures propriétés d'évolutivité. L'avantage de ce modèle tient à ce qu'un portier est utilisé pour l'enregistrement, l'admission, la résolution des adresses et la commande de largeur de bande mais que les appels sont établis directement entre les points d'extrémité de bout en bout.

La présente Recommandation décrit les améliorations à apporter au profil de sécurité de base H.235.1 et au profil de sécurité hybride H.235.3 afin de pouvoir prendre en charge des appels à routage direct avec portier(s).

7 Aperçu général

Le profil de sécurité de base H.235.1 ainsi que le profil de sécurité hybride H.235.3 appliquent un secret partagé (après la première prise de contact) pour assurer l'authentification et/ou la protection de l'intégrité des messages bond par bond, le portier étant utilisé comme hôte intermédiaire de confiance. Si on utilise le modèle d'appel à routage direct, on ne peut pas supposer qu'il existe un secret partagé entre deux points d'extrémité. En outre, il n'est pas commode d'utiliser un secret partagé préétabli pour sécuriser la communication car dans ce cas, il faudrait que tous les points d'extrémité sachent à l'avance quel autre point d'extrémité sera appelé.

La présente Recommandation traite du scénario représenté sur la Figure 1, dans lequel les points d'extrémité sont rattachés à un portier et utilisent une signalisation d'appel à routage direct. Dans ce scénario, on considère un réseau IP non sécurisé dans la zone du portier.

On suppose que chaque point d'extrémité a une relation de communication et une association de sécurité avec son portier et que chaque point d'extrémité s'est enregistré de manière sécurisée auprès du portier en utilisant le profil de sécurité de base ou le profil de sécurité hybride.

Ainsi, le portier du point d'extrémité d'origine (DRC1) ou le portier du point d'extrémité de destination (DRC2) est en mesure d'offrir un secret partagé pour des points d'extrémité en communication directe en utilisant une approche de type Kerberos (voir RFC 4120).

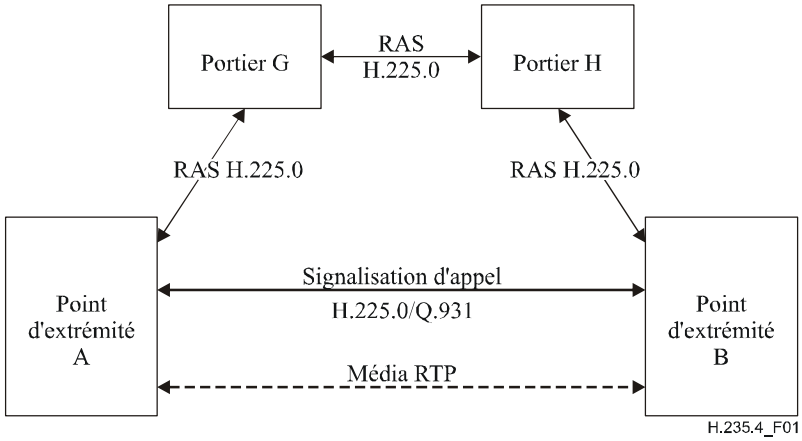

Figure 1/H.235.4 – Scénario d'un appel à routage direct

La présente Recommandation décrit deux procédures, DRC1 et DRC2, pour des environnements différents.

La procédure DRC1 (§ 9) est applicable dans des environnements d'entreprise où les portiers sont situés dans des sites (locaux) différents mais où les sites adhèrent à une politique de sécurité d'entreprise commune. Dans de tels environnements, on considère comme acceptable que le portier d'origine G détermine la politique de sécurité effective à suivre pour un appel à établir; par conséquent, le portier d'origine G sélectionne et choisit les paramètres de sécurité appliqués. Le portier de destination H acceptera les paramètres de sécurité choisis.

Les procédures DRC2 (§ 10) et DRC3 (§ 11) sont applicables dans des environnements interdomaines où les portiers sont situés dans des domaines administratifs différents et où chaque domaine peut employer une politique de sécurité différente.

La procédure DRC2 est applicable dans les cas où le point d'extrémité appelant ou les portiers ne prennent pas en charge l'algorithme Diffie-Hellman. En pareils cas, on considère comme acceptable que le portier de destination H détermine la politique de sécurité effective à suivre pour un appel à établir; par conséquent, le portier de destination H sélectionne et choisit les paramètres de sécurité appliqués. Le portier d'origine G acceptera les paramètres de sécurité choisis.

La procédure DRC3 est applicable dans les cas où le point d'extrémité appelant ne prend pas en charge l'algorithme Diffie-Hellman tandis que les portiers se trouvant dans le domaine appelant et dans le domaine appelé prennent tous deux en charge l'algorithme Diffie-Hellman.

Au début de l'enregistrement de l'appel, les procédures offrent des moyens de signalisation permettant de négocier la procédure DRC1, DRC2 ou DRC3 à appliquer.

8 Limitations

La présente Recommandation ne traite pas des scénarios à routage direct sans portier. Ces scénarios appellent un complément d'étude.

9 Procédure DRC1 (environnement d'entreprise)

La procédure décrite dans le présent paragraphe est applicable dans des environnements d'entreprise où les portiers sont situés dans des sites (locaux) différents mais où les sites adhèrent à une politique de sécurité d'entreprise commune. Dans ces environnements, on considère comme acceptable que le portier d'origine G détermine la politique de sécurité effective à suivre pour l'appel à établir; par conséquent, le portier d'origine sélectionne et choisit les paramètres de sécurité appliqués. Le portier de destination H acceptera les paramètres de sécurité choisis.

9.1 Phase GRQ/RRQ

Les points d'extrémité en mesure de prendre en charge ce profil de sécurité l'indiquent pendant l'envoi des messages **GRQ** et/ou **RRQ** en incluant un ClearToken distinct avec le champ **tokenOID** mis à "I10"; les autres champs de ce ClearToken ne devraient pas être utilisés. Les portiers disposant des capacités H.235.4 souhaitant offrir cette fonctionnalité répondent respectivement par un message **GCF** ou **RCF** contenant un ClearToken distinct avec le champ **tokenOID** mis à "I10", les autres champs du ClearToken n'étant pas utilisés.

9.2 Phase ARQ

Avant qu'un point d'extrémité A commence à envoyer directement des messages de signalisation d'appel à un point d'extrémité B, le point d'extrémité A ou B demande son admission au portier G ou H au moyen d'un message **ARQ**. Le point d'extrémité A inclut dans le message **ARQ** un ClearToken distinct avec le champ **tokenOID** mis à "I10", les autres champs de ClearToken n'étant pas utilisés.

9.3 Phase LRQ

Cette procédure s'applique aussi bien au cas d'un seul portier commun à plusieurs points d'extrémité qu'au cas de plusieurs portiers en chaîne. Dans le cas de plusieurs portiers, le portier G – dans la zone duquel l'appel provient – devrait localiser le portier H au moyen du mécanisme **LRQ** (multidestinataire) comme décrit au § 8.1.6/H.323 "Signalisation facultative par l'extrémité appelée". La communication entre deux portiers est sécurisée conformément à la Rec. UIT-T H.235.1. Pour cela, on part du principe qu'un secret partagé commun K_{GH} est disponible. Etant donné que le message **LRQ** entre portiers est généralement un message multidestinataire, le secret partagé K_{GH} ne peut pas en principe être un secret partagé par une paire mais est censé être en fait un secret partagé par un groupe à l'intérieur du nuage potentiel de portiers.

NOTE – Cette hypothèse limite l'évolutivité dans le cas général et ne permet pas l'authentification de la source. Cependant, on estime que dans les réseaux d'entreprise dont le nombre de portiers bien établis est petit et limité, ces obstacles à la sécurité sont encore acceptables. On pourrait surmonter ces derniers en sécurisant les communications multidestinataires entre portiers au moyen de signatures numériques; cette question appelle toutefois un complément d'étude.

Si le mécanisme **LRQ** est utilisé pour localiser le portier distant, le message **LRQ** achemine alors un jeton ClearToken distinct avec le champ **tokenOID** mis à "I10"; les autres champs de ce ClearToken ne devraient pas être utilisés. Dans le cas multidestinataire, le champ **generalID** du jeton ClearToken du message **LRQ** n'est pas utilisé. La communication entre portiers fondée sur les Recommandations UIT-T H.501 et/ou H.510 fera l'objet d'un complément d'étude.

9.4 Phase LCF

EK_{BH} désigne la clé de chiffrement et KS_{BH} désigne la clé de salage qui sont partagées entre le point d'extrémité B et le portier H. Comme indiqué ci-dessous, le portier H et le point d'extrémité B calculent séparément ces données de clé à partir du secret partagé K_{BH} au moyen d'une fonction PRF.

Le portier H génère un élément Challenge-B aléatoire puis il génère les données de clé de chiffrement EK_{BH} et les données de clé de salage KS_{BH} à partir du secret partagé K_{BH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par l'élément Challenge-B et CT_{HG}**→h235Key→secureSharedSecret→keyDerivationOID** contenant "Annex I-HMAC-SHA1-PRF" (voir le § 14).

EK_{GH} désigne la clé de chiffrement et KS_{GH} désigne la clé de salage qui sont partagées entre le portier G et le portier H. Le portier H génère un élément Challenge-G aléatoire puis il génère les

données de clé de chiffrement EK_{GH} et les données de clé de salage KS_{GH} à partir du secret partagé K_{GH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 14, l'élément **challenge** étant remplacé par l'élément Challenge-G. CT_{HG} →**challenge** contient l'élément challenge-G. L'identificateur du point d'extrémité B est placé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**generalID**.

Le portier H transmet les clés EK_{BH} et KS_{BH} chiffrées au portier G. Le mode de chiffrement OFB amélioré (EOFB) (voir le § 8.4/H.235.6) est utilisé avec la clé de salage secrète KS_{GH} propre au point d'extrémité. Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier H génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1"et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**params**→**iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

Le portier H inclut $ENC_{EK_{GH},\ KS_{GH},\ IV}(EK_{BH})$ et $ENC_{EK_{GH},\ KS_{GH},\ IV}(KS_{BH})$ dans le ClearToken CT_{HG} avec le champ **tokenOID** mis à "I13". Le texte chiffré obtenu $ENC_{EK_{GH},\ KS_{GH},\ IV}(EK_{BH})$ est acheminé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**; le texte chiffré obtenu $ENC_{EK_{GH},\ KS_{GH},\ IV}(KS_{BH})$ est acheminé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSaltingKey**. L'algorithme de chiffrement est indiqué dans CT_{HG}→**h235Key**→**algorithmOID** ("X1", "Y1", "Z1" ou "Z2"). L'élément Challenge-B est placé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**clearSaltingKey**. L'identificateur du portier G est placé dans CT_{HG}→**generalID** et l'identificateur du portier H est placé dans CT_{HG}→**sendersID**.

L'élément Challenge-B est acheminé au point d'extrémité B par l'inclusion d'un élément **profileInfo** dans le jeton **ClearToken** CT_{HG}→**profileInfo**→**elementID** = 0 qui identifie cet élément de profil particulier;

CT_{HG}→**profileInfo**→**paramS** n'est pas utilisé et CT_{HG}→**profileInfo**→**element**→**octets** contient l'élément Challenge-B.

La réponse **LCF** contient le ClearToken CT_{HG}.

9.5 Phase ACF

Constatant que les points d'extrémité A et B prennent en charge la présente Recommandation, le portier G génère les données de clé et les ClearToken comme spécifié ci-dessous.

Le portier est en mesure de calculer un secret partagé K_{AB} fondé sur l'appel, à partir du message **ARQ** normal. Ce secret est ensuite propagé aux deux points d'extrémité au moyen de jetons ClearToken. Ces derniers sont acheminés dans le message **ACF** et envoyés à l'appelant.

Deux ClearToken sont inclus, un CT_A pour l'appelant A et un CT_B pour l'appelé B. Chaque **ClearToken** contient un identificateur OID ("I11" ou "I12") dans le champ **tokenOID**, qui indique si le jeton est destiné à l'appelant (OID "I11" pour CT_A) ou à l'appelé (OID "I12" pour CT_B).

Le portier G déchiffre CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** pour obtenir la clé EK_{BH} et déchiffre CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSaltingKey** pour obtenir la clé KS_{BH}.

Le **ClearToken** défini dans la présente Recommandation peut être utilisé en association avec d'autres profils de sécurité (par exemple H.235.1 ou H.235.3) qui mettent aussi en œuvre des **ClearToken**. En pareil cas, un ClearToken conforme à la présente Recommandation doit aussi utiliser les champs de ces autres **ClearToken**. Par exemple, pour pouvoir utiliser la présente Recommandation conjointement avec la Rec. UIT-T H.235.1, les champs **timestamp**, **random**, **generalID**, **sendersID** et **dhkey** doivent être présents et être utilisés comme décrit dans le profil de sécurité H.235.1.

L'identificateur du portier G est inséré dans CT_A→**sendersID** et dans CT_B→**sendersID** tandis que CT_A→**generalID** contient l'identificateur du point d'extrémité A et CT_B→**generalID** l'identificateur du point d'extrémité B.

Le portier G génère les données de clés de salage KS_{GH} et les données de clé de chiffrement EK_{GH} à partir du secret K_{GH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par CT_{HG} →**challenge**.

Les clés de chiffrement EK_{AG} et EK_{BH} pour la clé chiffrée de bout en bout K_{AB} sont calculées à partir du secret partagé entre le portier et les points d'extrémité (EK_{AG} ou EK_{BH}) au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, CT_A→**h235Key**→**secureSharedSecret**→**keyDerivationOID** CT_B→**h235Key**→**secureSharedSecret**→**keyDerivationOID** contenant "AnnexI-HMAC-SHA1-PRF" (voir § 14) et CT_A→**challenge** contenant l'élément Challenge-A.

Le portier G copie l'élément Challenge-B de CT_{HG}→**h235Key**→**secureSharedSecret**→**clearSaltingKey** dans CT_B→**challenge**.

CT_B→**profileInfo** contient l'élément de profil qui a été acheminé dans CT_{HG}→**profile Info** afin que le point d'extrémité B finisse par obtenir l'élément Challenge-B.

Ce secret de session K_{AB} est chiffré par EK_{AG} (pour le jeton CT destiné au point d'extrémité A) ou par EK_{BH} (pour le jeton CT destiné au point d'extrémité B) au moyen d'un algorithme de chiffrement.

Le mode de chiffrement OFB amélioré (EOFB) (voir § 8.4/H.235.6) est utilisé avec la clé de salage secrète KS_{AG} ou KS_{BH} propre au point d'extrémité. Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier G génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1" et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv8** et dans CT_B→**h235Key**→**secureSharedSecret**→**params**→**iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv16** et dans CT_B→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

Le texte chiffré obtenu $ENC_{EK_{AG},\ KS_{AG},\ IV}(K_{AB})$ est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** et le texte chiffré obtenu $ENC_{EK_{BH},\ KS_{BH},\ IV}(K_{AB})$ est acheminé dans CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**. L'algorithme de chiffrement est indiqué dans CT_A→**h235Key**→**secureSharedSecret**→**algorithmOID** et dans CT_B→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" ou "Z2").

Pour le ClearToken destiné au point d'extrémité A, l'identificateur du point d'extrémité B ($EPID_B$) est inséré dans CT_A→**h235Key**→**secureSharedSecret**→**generalID**. De même, pour le ClearToken destiné au point d'extrémité B, l'identificateur du point d'extrémité A ($EPID_A$) est inséré dans CT_B→**h235Key**→**secureSharedSecret**→**generalID**.

Pour les algorithmes de chiffrement EOFB, l'élément **encryptedSaltingKey** n'est pas être utilisé.

Le portier G inclut à la fois les ClearToken CT_A et CT_B dans le message **ACF** destiné au point d'extrémité A.

9.6 Phase SETUP

Le point d'extrémité A identifie le jeton CT_A en inspectant l'identificateur **tokenOID** "I11" dans ClearToken.

Le point d'extrémité A vérifie que le jeton CT_A est tout nouveau en contrôlant l'horodate **timestamp**. D'autres contrôles de sécurité sont opérés pour vérifier les champs **generalID** et **sendersID** de ClearToken et le champ **generalID** de **V3KeySyncMaterial**. Si après vérification, il s'avère que le jeton CT_A reçu est tout nouveau, le point d'extrémité A récupère le vecteur IV et calcule EK_{AG} et KS_{AG} comme décrit ci-dessus pour le portier G. Le point d'extrémité A déchiffre l'information **encryptedSessionKey** qui se trouve dans **secureSharedSecret** du CT_A pour obtenir K_{AB}.

Si après vérification, il s'avère que le jeton CT_A est tout nouveau, le point d'extrémité A est en mesure d'envoyer un message SETUP au point d'extrémité B. Ce message SETUP inclut le jeton CT_B et est sécurisé (il est authentifié et/ou son intégrité est protégée) au moyen du profil H.235.1 ou du profil H.235.3 par l'application du secret partagé K_{AB}. A cette fin, le champ **generalID** du jeton ClearToken haché H.235.1 (pas le CT_B!) n'est pas utilisé sauf si le point d'extrémité A dispose déjà d'un identificateur $EPID_B$ (par exemple, par configuration ou mémorisé à partir d'une ancienne communication). S'il utilise une valeur $EPID_B$ pour le champ generalID du message SETUP, le point d'extrémité A doit accepter la valeur du champ sendersID dans le message de signalisation d'appel renvoyé en tant qu'identificateur $EPID_B$ vrai.

Le point d'extrémité B identifie le jeton CT_B par inspection de l'identificateur **tokenOID** "I12" dans le ClearToken.

Le point d'extrémité B vérifie que le jeton CT_B obtenu est tout nouveau en contrôlant l'horodate **timestamp**. D'autres contrôles de sécurité sont opérés pour vérifier le champ **sendersID** de ClearToken et le champ **generalID** de **secureSharedSecret**. Si après vérification, il s'avère que le jeton CT_B reçu est tout nouveau, le point d'extrémité B récupère l'élément Challenge-B de CT_{HG}→**profileInfo**→**element**→**octets** et le vecteur IV et calcule EK_{BH} et KS_{BH}, l'élément **challenge** du § 12 étant remplacé par l'élément Challenge-B, comme décrit ci-dessus pour le portier. Le point d'extrémité B déchiffre l'information **encryptedSessionKey** se trouvant dans **secureSharedSecret** du jeton CT_B pour obtenir K_{AB}.

Si après vérification, il s'avère que le jeton CT_B est tout nouveau, le point d'extrémité B est en mesure de poursuivre la signalisation d'appel en répondant par un message CALL-PROCEEDING, ALERTING ou CONNECT, etc. selon le cas. Si après vérification, il s'avère que le jeton CT_B n'est pas tout nouveau ou si le contrôle de sécurité du message SETUP révèle un problème, le point d'extrémité B répond par un message RELEASE-COMPLETE, l'élément **ReleaseCompleteReason** étant mis à une erreur de sécurité définie au § 11.1/H.235.0.

Lorsque la sécurité de média doit être appliquée (voir § 6.1/H.235.6), les points d'extrémité A et B s'échangent des demi-clés Diffie-Hellman conformément au § 8.5/H.235.6 et établissent une clé maître dynamique fondée sur la session à partir de laquelle des clés de session propres au média peuvent être déduites.

Le point d'extrémité B inclut le champ **generalID** mis à l'identificateur $EPID_A$ et le champ **sendersID** mis à l'identificateur $EPID_B$ pour la protection de tout message de signalisation d'appel H.225.0 destiné au point d'extrémité A (par exemple, Call Proceeding, Alerting ou Connect).

La Figure 2 illustre le flux de communication de base:

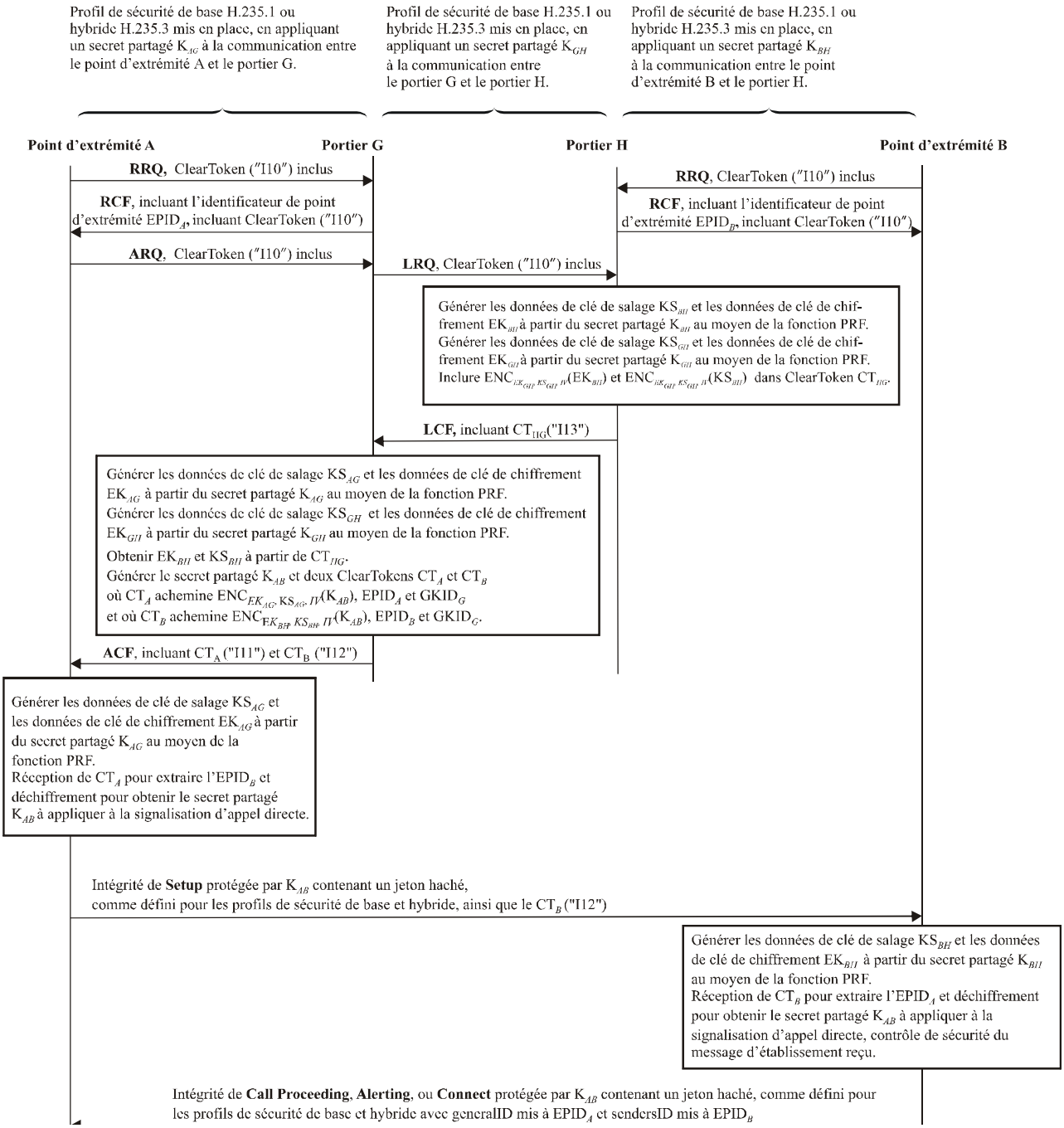

Figure 2/H.235.4 – Flux de communication de base

10 Procédure DRC2 (environnement interdomaines)

La procédure décrite dans le présent paragraphe est applicable dans des environnements interdomaines où les portiers sont situés dans domaines administratifs différents et où chaque domaine peut employer une politique de sécurité différente. La procédure DRC2 est applicable dans les cas où le point d'extrémité appelant ou les portiers ne prennent pas en charge l'algorithme Diffie-Hellman.

Dans ces environnements, on considère comme acceptable que le portier de destination H détermine la politique de sécurité effective à suivre pour l'appel à établir; par conséquent, le portier de destination H sélectionne et choisit les paramètres de sécurité appliqués. Le portier d'origine G acceptera les paramètres de sécurité choisis.

10.1 Phase GRQ/RRQ

Les points d'extrémité en mesure de prendre en charge ce profil de sécurité l'indiquent pendant l'envoi des messages **GRQ** et/ou **RRQ** en incluant un ClearToken distinct avec le champ **tokenOID** mis à "I20"; les autres champs de ce ClearToken ne devraient pas être utilisés. Les portiers disposant des capacités H.235.4 souhaitant offrir cette fonctionnalité répondent respectivement par un message **GCF** ou **RCF** contenant un ClearToken distinct avec le champ **tokenOID** mis à "I20", les autres champs du ClearToken n'étant pas utilisés.

10.2 Phase ARQ

Avant qu'un point d'extrémité A commence à envoyer directement des messages de signalisation d'appel à un point d'extrémité B, le point d'extrémité A ou B demande son admission au portier G ou H au moyen d'un message **ARQ**. Le point d'extrémité A inclut dans le message **ARQ** un ClearToken distinct avec le champ **tokenOID** mis à "I20", les autres champs de ClearToken n'étant pas utilisés.

10.3 Phase LRQ

Cette procédure s'applique aussi bien au cas d'un seul portier commun aux points d'extrémité qu'au cas de plusieurs portiers en chaîne. Dans le cas de plusieurs portiers, le portier G – dans la zone duquel l'appel provient – devrait localiser le portier H au moyen du mécanisme **LRQ** (multidestinataire) comme décrit au § 8.1.6/H.323 "signalisation facultative par l'extrémité appelée". La communication entre deux portiers doit être sécurisée conformément à la Rec. UIT-T H.235.1. Pour cela, on part du principe qu'un secret partagé commun K_{GH} est disponible. Etant donné que le message **LRQ** entre les portiers est généralement un message multidestinataire, le secret partagé K_{GH} ne peut pas en principe être un secret partagé par une paire mais est censé être en fait un secret partagé par un groupe à l'intérieur du nuage potentiel de portiers.

NOTE – Cette hypothèse limite l'évolutivité dans le cas général et ne permet pas l'authentification de la source. Cependant, on estime que dans les réseaux d'entreprise dont le nombre de portiers bien établis est petit et limité, ces obstacles à la sécurité sont encore acceptables. On pourrait surmonter ces derniers en sécurisant les communications multidestinataires entre portiers au moyen de signatures numériques; cette question appelle toutefois un complément d'étude.

Si le mécanisme **LRQ** est utilisé pour localiser le portier distant, le message **LRQ** doit acheminer un jeton ClearToken distinct avec le champ **tokenOID** mis à "I20"; les autres champs de ce ClearToken ne devraient pas être utilisés. Dans le cas multidestinataire, le champ **generalID** du jeton ClearToken du message **LRQ** n'est pas utilisé. La communication entre portiers fondée sur les Recommandations UIT-T H.501 et/ou H.510 fera l'objet d'un complément d'étude.

10.4 Phase LCF

Constatant que les points d'extrémité A et B prennent en charge la présente Recommandation, le portier H génère les données de clé et les ClearToken dans le message **LCF** comme spécifié ci-dessous.

K_{BH} désigne le secret partagé entre le point d'extrémité B et le portier H. EK_{BH} désigne la clé de chiffrement et KS_{BH} désigne la clé de salage qui sont partagées entre le point d'extrémité B et le portier H. Le portier H génère un élément Challenge-B aléatoire. Il génère ensuite les données de clé de chiffrement EK_{BH} à partir du secret partagé K_{BH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par l'élément Challenge-B et CT_B→**h235Key**→**secureSharedSecret**→**keyDerivationOID** contenant "AnnexI-HMAC-SHA1-PRF" (voir § 14).

Le portier H génère une clé de salage KS_{BH} à partir de K_{BH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par l'élément Challenge-B.

EK_{GH} désigne la clé de chiffrement et KS_{GH} désigne la clé de salage qui sont partagées entre le portier G et le portier H. Le portier H génère un élément Challenge-G aléatoire. Il génère ensuite les données de clé de chiffrement EK_{GH} à partir du secret partagé K_{GH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par l'élément Challenge-G et CT_{HG}→**h235Key**→**secureSharedSecret**→**keyDerivationOID** contenant "AnnexI-HMAC-SHA1-PRF" (voir § 14).

Le portier H génère la clé KS_{GH} à partir du secret partagé K_{GH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par l'élément Challenge-G.

Le portier H crée deux ClearToken dans le message **LCF**, un CT_{HG} pour le portier G et un CT_B pour l'appelé B. CT_{HG}→**tokenOID** contient l'identificateur OID "I23" tandis que CT_B→**tokenOID** contient l'identificateur OID "I12".

L'élément Challenge-G est mis dans CT_{HG}→**challenge**, l'identificateur du portier H est mis dans CT_{HG}→**sendersID** et l'identificateur du portier G (copié du message **LRQ**) est mis dans CT_{HG}→**generalID**.

L'élément Challenge-B est mis dans CT_B→**challenge**, l'identificateur du portier H est mis dans CT_B→**sendersID** et l'identificateur du point d'extrémité B est mis dans CT_B→**generalID**. Si le champ endpointIdentifier du message **LRQ** contient l'identificateur du point d'extrémité A, le portier H le copie dans CT_B→**h235Key**→**secureSharedSecret**→**generalID** ainsi que dans CT_{HG}→**h235Key**→**secureSharedSecret**→**generalID**.

La réponse **LCF** contient les ClearToken CT_{HG} et CT_B si le portier H et le point d'extrémité B prennent aussi en charge la procédure DRC2 de la présente Recommandation.

Après avoir reçu le message **LCF** du portier H, le portier G vérifie les ClearToken CT_B et $CT_{HG.}$ Il utilise l'élément Challenge-G comme élément **challenge** et la fonction PRF comme décrit au § 12 pour calculer KS_{GH} et EK_{GH} à partir de K_{GH} puis pour déchiffrer CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** et obtenir le secret K_{AB} partagé par les points d'extrémité A et B.

10.5 Phase ACF

Le portier H calcule un secret K_{AB} fondé sur l'appel, partagé par les points d'extrémité A et B. Ce secret est ensuite propagé aux deux points d'extrémité au moyen d'un ClearToken. Le ClearToken est d'abord envoyé au portier d'origine G, qui envoie ensuite l'information à l'appelant dans le message **ACF**.

Le portier H chiffre le secret K_{AB} à partir de la clé EK_{GH} sous la forme $ENC_{EK_{HG},\ KS_{HG},\ IV}(K_{AB})$ puis place le secret K_{AB} chiffré dans CT_{HG}→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**.

Le mode de chiffrement OFB amélioré (EOFB) (voir le § 8.4/H.235.6) est utilisé avec la clé de salage secrète KS_{GH} propre au point d'extrémité. Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier H génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1" et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_{HG}→**h235Key**→**secureSharedSecret**→ **params**→**iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_{HG}→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

L'algorithme de chiffrement est indiqué dans CT_{HG}→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" ou "Z2"). Pour les algorithmes de chiffrement EOFB, **encryptedSaltingKey** n'est pas utilisé.

De même, le portier H chiffre le secret K_{AB} à partir de EK_{BH} sous la forme $ENC_{EK_{BH},\ KS_{BH},\ IV}(K_{AB})$ et place le secret K_{AB} chiffré dans CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**.

Le mode de chiffrement OFB amélioré (EOFB) (voir le § 8.4/H.235.6) est utilisé avec la clé de salage secrète KS_{BH} propre au point d'extrémité pour le point d'extrémité B (CT_B). Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier H génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1" et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_B→**h235Key**→**secureSharedSecret**→ **params**→**iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_B→**h235Key**→**secureSharedSecret**→**params**→**iv16**.

L'algorithme de chiffrement est indiqué dans CT_B→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" ou "Z2"). Pour les algorithmes de chiffrement EOFB, **encryptedSaltingKey** n'est pas utilisé.

Pour la réponse **ACF** au point d'extrémité A, deux ClearToken sont inclus, un CT_A pour l'appelant A et un CT_B pour l'appelé B. Le **ClearToken** CT_A→**tokenOID** contient l'identificateur OID "I11".

Le portier G génère un élément Challenge-A puis il génère les données de clé de chiffrement EK_{AG} à partir du secret partagé K_{AG} au moyen de la procédure de calcul de clé fondée sur la fonction PRF définie au § 12, l'élément **challenge** étant remplacé par l'élément Challenge-A, CT_A→**h235Key**→**secureSharedSecret**→**keyDerivationOID** contenant "AnnexI-HMAC-SHA1-PRF" (voir § 14) et l'élément Challenge-A étant placé dans CT_A→**challenge**.

Le portier G chiffre le secret K_{AB} au moyen de la clé EK_{AG} sous la forme $ENC_{EK_{AG},\ KS_{AG},\ IV}(K_{AB})$ au moyen d'un algorithme de chiffrement et place le secret K_{AB} chiffré dans CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey**.

Le mode de chiffrement OFB amélioré (EOFB) (voir § 8.4/H.235.6) est utilisé avec la clé de salage secrète KS_{AG} propre au point d'extrémité. Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier G génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1" et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv16**. L'algorithme de chiffrement est indiqué dans CT_A→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" ou "Z2").

L'identificateur du portier G est mis dans CT_A→**sendersID**, l'identificateur du point d'extrémité A est mis dans CT_A→**generalID** et l'identificateur du point d'extrémité B est copié de CT_B→**generalID** dans CT_A→**h235Key**→**secureSharedSecret**→**generalID.**

Si le portier G n'a pas mis l'identificateur du point d'extrémité A dans le champ endpointIdentifier du message **LRQ**, il le met dans CT_B→**h235Key**→**secureSharedSecret**→**generalID**.

Pour les algorithmes de chiffrement EOFB, **encryptedSaltingKey** n'est pas utilisé.

Le **ClearToken** défini dans la présente Recommandation peut être utilisé conjointement avec d'autres profils de sécurité (par exemple H.235.1 ou H.235.3) qui mettent aussi en œuvre des ClearToken. Dans ce cas, le ClearToken de la présente Recommandation doit aussi utiliser les champs de ces autres **ClearToken**. Par exemple, pour pouvoir utiliser la présente Recommandation conjointement avec la Rec. UIT-T H.235.1, les champs **timestamp**, **random**, **generalID**, **sendersID** et **dhkey** doivent être présentés et utilisés comme décrit dans le profil de sécurité H.235.1.

L'identificateur du portier G est placé dans CT_A→**sendersID** tandis que CT_A→**generalID** contient l'identificateur du point d'extrémité A.

Le point d'extrémité A identifie le CT_A en inspectant CT_A→**tokenOID** "I21". Il vérifie que le CT_A obtenu est tout nouveau en contrôlant l'horodate **timestamp**. D'autres contrôles de sécurité sont opérés pour vérifier les champs **generalID** et **sendersID** du ClearToken et le champ **generalID** de **secureSharedSecret**. Si après vérification, il s'avère que le CT_A reçu est tout nouveau, le point d'extrémité A récupère le vecteur IV et calcule les clés EK_{AG} et KS_{AG} comme décrit ci-dessus pour le portier G en utilisant CT_A→**challenge** comme élément Challenge-A employé à la place de l'élément **challenge** décrit au § 12. Le point d'extrémité A déchiffre CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** pour obtenir K_{AB}.

10.6 Phase SETUP

Le point d'extrémité A identifie le jeton CT_A en inspectant CT_A →**tokenOID** "I11". Il vérifie que le CT_A obtenu est tout nouveau en contrôlant l'horodate. D'autres contrôles de sécurité sont opérés pour vérifier les champs **generalID** et **sendersID** du ClearToken et le champ **generalID** de **secureSharedSecret**. Si après vérification, il s'avère que le CT_A reçu est tout nouveau, le point d'extrémité A récupère le vecteur IV et calcule les clés EK_{AG} et KS_{AG} comme décrit ci-dessus pour le portier G en utilisant CT_A→**challenge** comme élément Challenge-A. Le point d'extrémité A déchiffre CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** pour obtenir K_{AG}.

Si après vérification, il s'avère que le CT_A reçu est tout nouveau, le point d'extrémité A est en mesure d'envoyer au point d'extrémité B un message SETUP contenant le CT_B. Le message SETUP est sécurisé (il est authentifié et/ou son intégrité est protégée) conformément au profil H.235.1 ou au profil H.235.3 par l'application du secret partagé K_{AB}. Pour cela, le champ **generalID** du ClearToken haché H.235.1 (pas le CT_B!) n'est pas utilisé sauf si le point d'extrémité A dispose déjà d'un identificateur $EPID_B$ (par exemple par configuration ou mémorisé à partir d'une ancienne communication). Si le point d'extrémité A utilise une valeur $EPID_B$ pour le champ generalID du message SETUP, il doit accepter la valeur du champ sendersID du message de signalisation d'appel renvoyé en tant qu'identificateur $EPID_B$ vrai.

Le point d'extrémité B identifie le jeton CT_B en inspectant l'identificateur **tokenOID** "I12" dans le ClearToken.

Le point d'extrémité B vérifie que le CT_B obtenu est tout nouveau en contrôlant l'horodate **timestamp**. D'autres contrôles de sécurité sont opérés pour vérifier le champ **sendersID** du ClearToken et le champ **generalID** de **secureSharedSecret**. Si après vérification, il s'avère que le CT_B reçu est tout nouveau, le point d'extrémité B récupère le vecteur IV et calcule les clés EK_{BH} et KS_{BH} en utilisant CT_B→**challenge** comme élément Challenge-B employé à la place de l'élément **challenge** décrit au § 12, comme décrit ci-dessus pour le portier H. Le point d'extrémité B déchiffre CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** pour obtenir K_{AB}.

Si après vérification, il s'avère que le CT_B est tout nouveau, le point d'extrémité B est en mesure de poursuivre la signalisation d'appel en répondant par un message CALL-PROCEEDING, ALERTING ou CONNECT etc., selon le cas. S'il s'avère que le CT_B n'est pas tout nouveau ou si le contrôle de sécurité du message SETUP révèle un problème, le point d'extrémité B répond par un message RELEASE-COMPLETE avec le motif **ReleaseCompleteReason** mis à une erreur de sécurité définie au § 11.1/H.235.0.

Lorsque la sécurité de média doit être mise en œuvre (voir le § 6.1/H.235.6), les points d'extrémité A et B s'échangent des demi-clés Diffie-Hellman conformément au § 8.5/H.235.6 et établissent une clé maître dynamique fondée sur la session à partir de laquelle des clés de session propres au média peuvent être déduites.

Le point d'extrémité B inclut le champ **generalID** mis à l'identificateur $EPID_A$ et le champ **sendersID** mis à l'identificateur $EPID_B$ pour la protection de tout message de signalisation d'appel H.225.0 destiné au point d'extrémité A (par exemple Call Proceeding, Alerting ou Connect).

La Figure 3 montre le flux de communication de base:

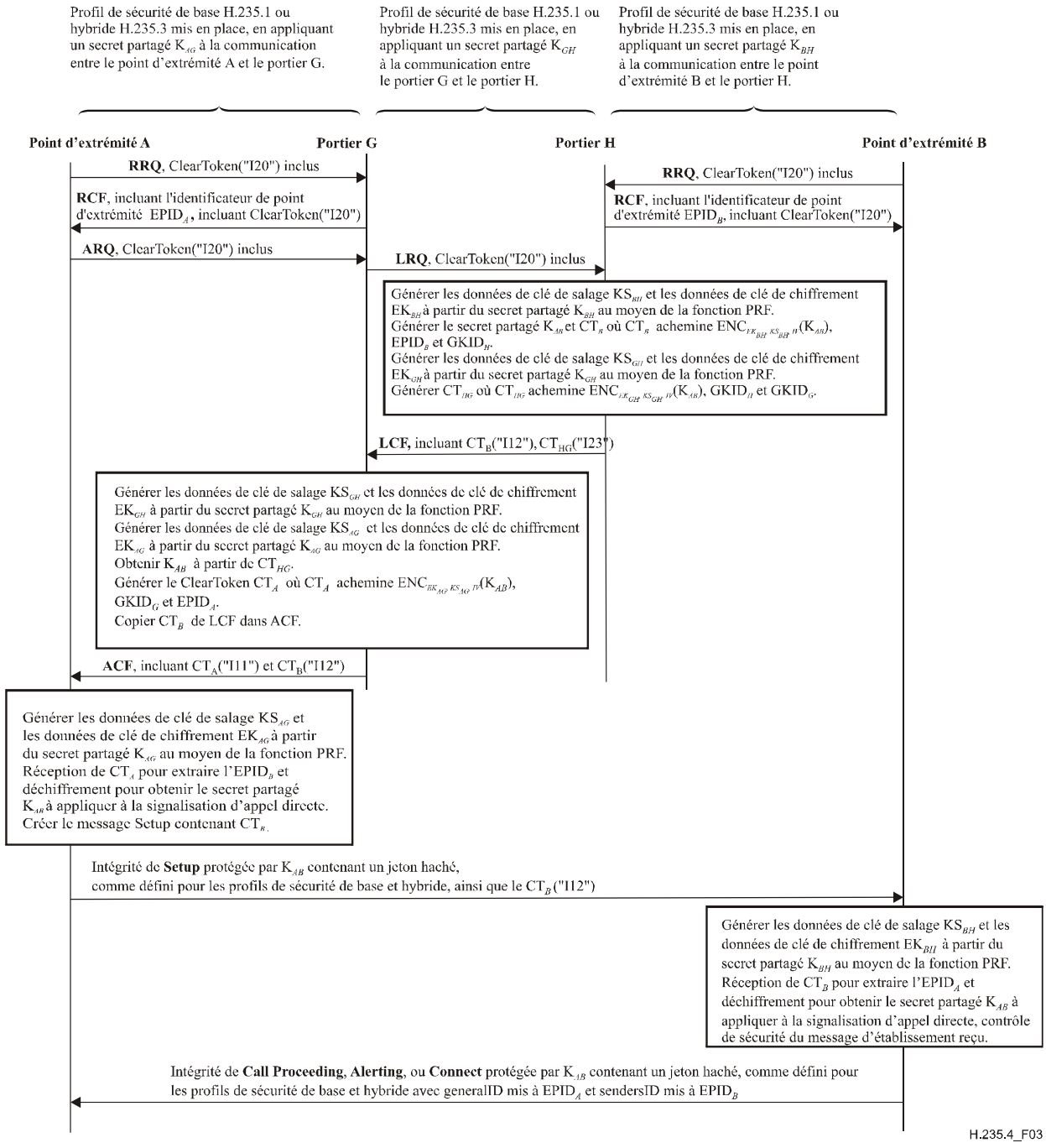

Figure 3/H.235.4 – Flux de communication de base (DRC2)

11 Procédure DRC3 (environnement interdomaines)

La procédure décrite dans le présent paragraphe est applicable dans des environnements interdomaines où le point d'extrémité appelant ne prend pas en charge l'algorithme Diffie-Hellman tandis que les portiers se trouvant dans le domaine appelant et dans le domaine appelé sont tous deux capables de calculer et d'échanger des paramètres DH. Dans ces environnements, la clé de session est calculée grâce à l'échange de paramètres DH entre le portier d'origine et le portier de destination.

11.1 Phase GRQ/RRQ

Ce scénario englobe plusieurs portiers en chaîne. Les points d'extrémité en mesure de prendre en charge ce profil de sécurité l'indiquent pendant l'envoi des messages **GRQ** et/ou **RRQ** en incluant un ClearToken distinct avec le champ **tokenOID** mis à "I30", les autres champs de ce ClearToken n'étant pas utilisés. Les portiers disposant des capacités H.235.4 souhaitant offrir cette fonctionnalité répondent respectivement par un message **GCF** ou **RCF** contenant un ClearToken distinct avec le champ **tokenOID** mis à "I30", les autres champs du ClearToken n'étant pas utilisés.

11.2 Phase ARQ

Avant que le point d'extrémité A n'appelle le point d'extrémité B au moyen de la procédure DRC3, il envoie au portier G un message **ARQ** contenant un ClearToken distinct avec le champ **tokenOID** mis à "I30", les autres champs n'étant pas utilisés.

11.3 Phase LRQ

Dès qu'il reçoit le message **ARQ** envoyé par le point d'extrémité A, le portier G envoie un message **LRQ** au portier H pour obtenir l'adresse du point d'extrémité B étant donné que le point d'extrémité B n'appartient pas au domaine du portier G. Le portier G vérifie le ClearToken acheminé dans le message **ARQ** et constate que le champ **tokenOID** est mis à "I30"; si le portier G prend en charge l'algorithme DH, il applique certaines règles préconfigurées qui déterminent que la procédure DRC3 devrait être choisie.

Le portier G génère ensuite un message **LRQ** contenant un ClearToken (dans le CryptoHashedToken) avec le champ **tokenOID** mis à "I30" pour indiquer au portier H qu'une négociation de clé DH est nécessaire. Le champ **dhkey** du ClearToken est rempli avec les paramètres DH de l'appelant (g, p, g^x) générés par le portier G, les autres champs n'étant pas utilisés.

Le portier G envoie ensuite ce message **LRQ** au portier H. Dans le cas d'un nuage de portiers, le portier G envoie le message **LRQ** au portier qui est son voisin immédiat, lequel retransmet le message **LRQ** au portier qui est son propre voisin immédiat. La retransmission se poursuit jusqu'à ce que le message **LRQ** finisse par atteindre le portier H.

Dans le cas multidestinataire, le champ **generalID** du CryptoToken du message **LRQ** n'est pas utilisé. Si le portier G n'a pas été en mesure de localiser le point d'extrémité distant B, il retourne le message **ARJ** au point d'extrémité A. La communication entre deux portiers est sécurisée conformément au profil de la Rec. UIT-T H.235.1.

Si le portier G ne prend pas en charge le profil, il est libre de choisir un repli sur la procédure DRC2 ou de retourner un message ARJ au point d'extrémité A; si la procédure DRC2 est choisie, la phase **LRQ** et les phases suivantes sont identiques à celles de la procédure DRC2.

11.4 Phase LCF

Après avoir reçu le message **LRQ** du portier G, le portier H, constatant que les deux points d'extrémité A et B prennent en charge cette procédure, génère la clé de session K_{AB} comme spécifié ci-dessous.

Tout d'abord, le portier H produit un élément Challenge-B aléatoire, qui est mis dans CT_B→**challenge**, et CT_B→**h235Key→secureSharedSecret→keyDerivationOID** contient "AnnexI-HMAC-SHA1-PRF". Le portier H utilise ensuite la clé partagée K_{GH} et l'élément Challenge-B pour calculer les éléments de clé EK_{GH} et la clé de salage KS_{GH} au moyen de la procédure de calcul de clé fondée sur la fonction PRF.

L'élément Challenge-B est mis dans CT_B→**challenge**, l'identificateur du portier H est mis dans CT_B→**sendersID** et l'identificateur du point d'extrémité B est mis dans CT_B→**generalID**. Si le champ endpointIdentifier du message **LRQ** contient l'identificateur du point d'extrémité A, le portier H le copie dans CT_B→**h235Key→secureSharedSecret→generalID** ainsi que dans CT_{HG}→**h235Key→secureSharedSecret→generalID**.

Le portier H crée ensuite deux ClearToken dans le message **LCF**, un CT_{HG} avec le champ **tokenOID** mis à "I33" pour le portier G et un CT_B avec le champ **tokenOID** mis à "I12" pour le point d'extrémité B. Le portier H génère les paramètres DH de l'appelé (g, p, g^y). Il utilise ensuite les paramètres DH de l'appelant obtenus à partir du message **LRQ** pour calculer la clé de session $K_{AB} = g^{xy} \bmod p$.

Enfin, le portier H chiffre la clé K_{AB} au moyen des clés EK_{BH} et KS_{BH} sous la forme $ENC_{EK_{BH}, KS_{BH}, IV}(K_{AB})$, place la clé K_{AB} chiffrée dans CT_B→**h235Key→secureSharedSecret→encryptedSessionKey** et place les paramètres DH de l'appelé dans le champ **dhkey** de CT_{HG}. Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier H génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1" et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_B→**h235Key→secureSharedSecret→ params→iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_B→**h235Key→secureSharedSecret→params→iv16**.

L'algorithme de chiffrement est indiqué dans CT_B→**h235Key→secureSharedSecret→algorithmOID** ("X1", "Y1", "Z1" ou "Z2"). Pour les algorithmes de chiffrement EOFB, **encryptedSaltingKey** n'est pas utilisé.

Le portier H envoie le message **LCF** au portier G. Dans le cas d'un nuage de portiers, le message **LCF** est relayé par les différents portiers. Le long du trajet des portiers, chaque portier reçoit le message **LCF** de son voisin immédiat en amont, vérifie qu'il contient le jeton CT_{HG} et le retransmet à son voisin immédiat en aval.

Si le portier H ne prend pas en charge l'algorithme DH ou si la politique de sécurité ne permet pas d'utiliser la procédure DRC3, un repli se produit vers la procédure DRC2, auquel cas la phase **LCF** et toutes les phases suivantes sont identiques à celles de la procédure DRC2.

11.5 Phase ACF

Après avoir reçu le message **LCF**, le portier G, constatant que le champ **tokenOID** du ClearToken distinct est mis à "I33", obtient les paramètres DH de l'appelé et crée un ClearToken désigné par CT_A avec le champ **tokenOID** mis à "I11" comme spécifié ci-dessous.

Tout d'abord, le portier G produit un élément Challenge-A aléatoire, qui est mis dans CT_A→**challenge**, et CT_A→**h235Key→secureSharedSecret→keyDerivationOID** contient

"AnnexI-HMAC-SHA1-PRF". Le portier G utilise ensuite la clé partagée K_{AG} et l'élément Challenge-A pour calculer les données de clé EK_{AG} et la clé de salage KS_{AG} au moyen de la procédure de calcul de clé fondée sur la fonction PRF.

Le portier G utilise ensuite les paramètres DH de l'appelant qu'il a obtenus dans la phase **LRQ** et les paramètres DH de l'appelé pour calculer la clé de session $K_{AG} = g^{xy} \bmod p$.

Le portier G copie ensuite le ClearToken CT_B dont le champ **tokenOID** est mis à "I12", du message **LCF** dans le message **ACF**.

Enfin, le portier G chiffre la clé K_{AB} au moyen des clés EK_{AG} et KS_{AG} sous la forme $ENC_{EK_{AG}, KS_{AG}, IV}(K_{AB})$, place la clé K_{AB} chiffrée dans CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** et copie le jeton CT_B du message **LCF** dans le message **ACF**.

Le mode de chiffrement OFB amélioré (EOFB) (voir § 8.4/H.235.6) est utilisé avec la clé de salage secrète KS_{AG} propre au point d'extrémité.

Les algorithmes de chiffrement applicables sont les suivants (voir le Tableau 6/H.235.6):

– DES (56 bits) en mode EOFB avec l'identificateur OID "Y1": facultatif;

– 3DES (168 bits) en mode EOFB externe avec l'identificateur OID "Z1": facultatif;

– AES (128 bits) en mode EOFB avec l'identificateur OID "Z2": algorithme par défaut et recommandé;

– Compatible RC2 (56 bits) en mode EOFB avec l'identificateur OID "X1": facultatif.

Pour le mode de chiffrement EOFB, le portier G génère une valeur initiale aléatoire IV. Pour les identificateurs OID "X1", "Y1" et "Z1", le vecteur IV occupe 64 bits et est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv8**; pour l'identificateur OID "Z2", le vecteur IV occupe 128 bits et est acheminé dans CT_A→**h235Key**→**secureSharedSecret**→**params**→**iv16**. L'algorithme de chiffrement est indiqué dans CT_A→**h235Key**→**secureSharedSecret**→**algorithmOID** ("X1", "Y1", "Z1" ou "Z2").

Si le champ **tokenOID** du ClearToken (du message **LCF**) vaut "I23", cela signifie qu'un repli s'est produit vers la procédure DRC2 et le portier G est libre d'accepter ou non la politique de sécurité du portier H. S'il accepte, la phase **ACF** et la phase SETUP qui suit sont identiques à celles de la procédure DRC2. Dans le cas contraire, le portier G répond par un message de rejet correspondant indiquant une défaillance de sécurité en mettant le motif de rejet à securityDenial.

Le portier G envoie le message **ACF** au point d'extrémité A.

11.6 Phase SETUP

Le point d'extrémité A identifie le jeton CT_A en inspectant CT_A →**tokenOID** "I11". Il vérifie que le CT_A obtenu est tout nouveau en contrôlant l'horodate. D'autres contrôles de sécurité sont opérés pour vérifier les champs **generalID** et **sendersID** du ClearToken et le champ **generalID** de **secureSharedSecret**. Si après vérification, il s'avère que le CT_A reçu est tout nouveau, le point d'extrémité A récupère le vecteur IV et calcule les clés EK_{AG} et KS_{AG} comme décrit ci-dessus pour le portier G en utilisant CT_A→**challenge** comme élément Challenge-A. Le point d'extrémité A déchiffre CT_A→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** pour obtenir K_{AG}.

Si après vérification, il s'avère que le CT_A reçu est tout nouveau, le point d'extrémité A est en mesure d'envoyer au point d'extrémité B un message SETUP contenant le CT_B. Le message SETUP est sécurisé (il est authentifié et/ou son intégrité est protégée) conformément au profil H.235.1 ou au profil H.235.3 par l'application du secret partagé K_{AB}. Pour cela, le champ **generalID** du ClearToken haché H.235.1 (pas le CT_B!) n'est pas utilisé sauf si le point d'extrémité A dispose déjà d'un identificateur $EPID_B$ (par exemple par configuration ou mémorisé à partir d'une ancienne communication). Si le point d'extrémité A utilise une valeur $EPID_B$ pour le champ **generalID** du

message SETUP, il doit accepter la valeur du champ **sendersID** du message de signalisation d'appel renvoyé en tant qu'identificateur $EPID_B$ vrai.

Le point d'extrémité B identifie le jeton CT_B en inspectant l'identificateur **tokenOID** "I12" dans le ClearToken.

Le point d'extrémité B vérifie que le CT_B obtenu est tout nouveau en contrôlant l'horodate **timestamp**. D'autres contrôles de sécurité sont opérés pour vérifier le champ **sendersID** du ClearToken et le champ **generalID** de **secureSharedSecret**. Si après vérification, il s'avère que le CT_B reçu est tout nouveau, le point d'extrémité B récupère le vecteur IV et calcule les clés EK_{BH} et KS_{BH} en utilisant CT_B→**challenge** comme élément Challenge-B. Le point d'extrémité B déchiffre CT_B→**h235Key**→**secureSharedSecret**→**encryptedSessionKey** pour obtenir K_{AB}.

Si après vérification, il s'avère que le CT_B est tout nouveau, le point d'extrémité B est en mesure de poursuivre la signalisation d'appel en répondant par un message CALL-PROCEEDING, ALERTING ou CONNECT etc., selon le cas. S'il s'avère que le CT_B n'est pas tout nouveau ou si le contrôle de sécurité du message SETUP révèle un problème, le point d'extrémité B répond par un message RELEASE-COMPLETE avec le motif **ReleaseCompleteReason** mis à une erreur de sécurité définie au § 11.1/H.235.0.

Lorsque la sécurité de média doit être mise en œuvre (voir le § 6.1/H.235.6), les points d'extrémité A et B s'échangent des demi-clés Diffie-Hellman conformément au § 8.5/H.235.6 et établissent une clé maître dynamique fondée sur la session à partir de laquelle des clés de session propres au média peuvent être déduites.

Le point d'extrémité B inclut le champ **generalID** mis à l'identificateur $EPID_A$ et le champ **sendersID** mis à l'identificateur $EPID_B$ pour la protection de tout message de signalisation d'appel H.225.0 destiné au point d'extrémité A (par exemple Call Proceeding, Alerting ou Connect).

La Figure 4 montre le flux de communication de base:

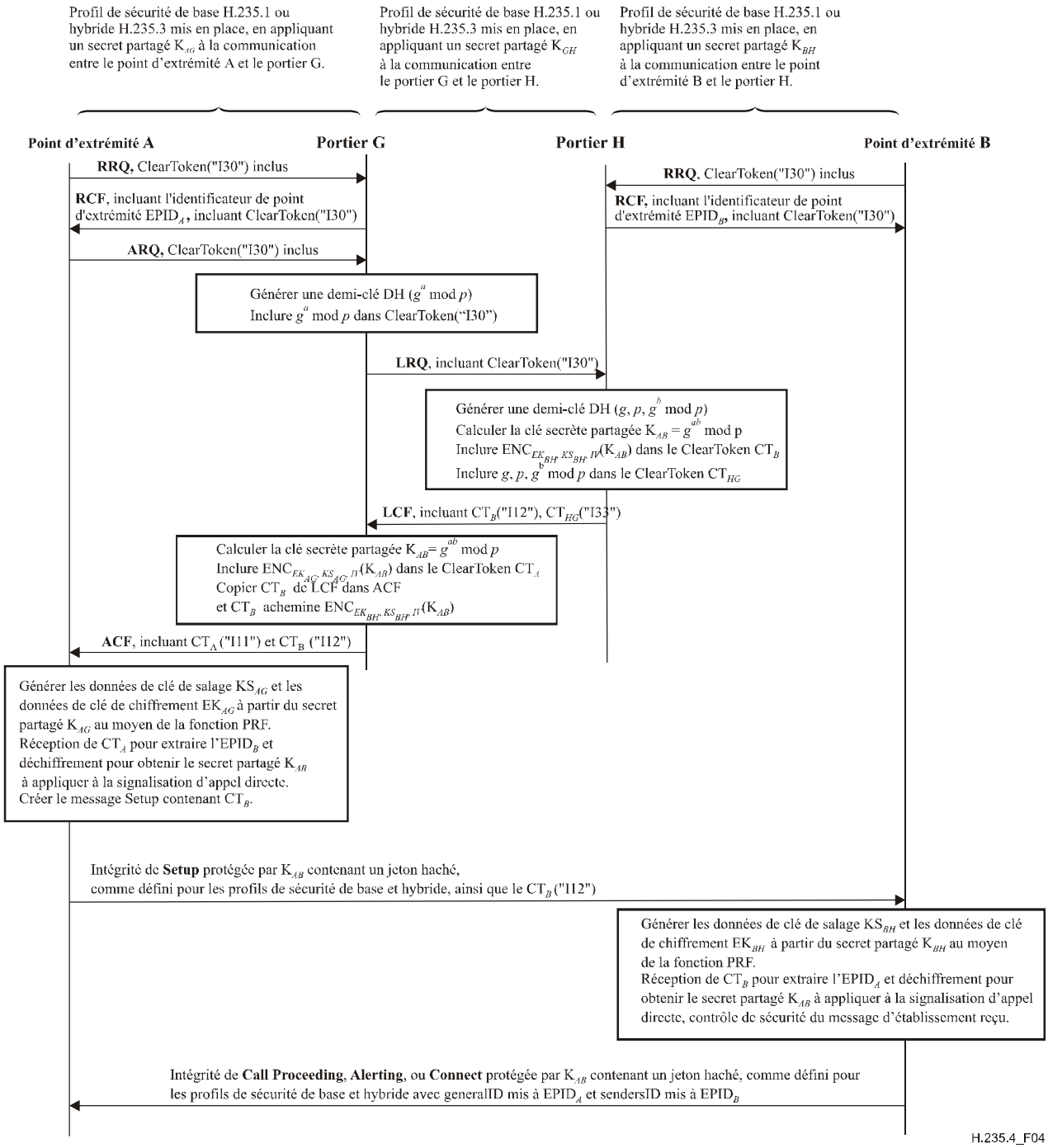

Figure 4/H.235.4 – Flux de communication dans la procédure DRC3

12 Procédure de calcul de la clé au moyen de la fonction PRF

Le présent paragraphe décrit une procédure qui indique comment calculer les données de clé à partir du secret partagé et d'autres paramètres.

La procédure définie dans le présent paragraphe permet de calculer une clé de chiffrement et une clé de salage à partir d'une clé partagée. Cette procédure est uniforme, quel que soit le secret partagé (K_{AG}, K_{BH} ou K_{GH}).

Afin d'obtenir les données de clé voulues (EK_{AG}, par exemple), la fonction PRF (voir § 10/H.235.0) doit être utilisée avec les paramètres du Tableau 1, dont le paramètre *inkey* mis à la clé partagée correspondante (K_{AG}, par exemple), le paramètre *label* devant être mis à la constante correspondante (0x2AD01C64 || **challenge-A**, par exemple), où le symbole || indique qu'il y a concaténation. Le paramètre *outkey_len* doit être mis à la longueur requise pour les données de clé voulues, qui dépend de l'algorithme de chiffrement choisi.

NOTE – Pour EK_{AG}, KS_{AG}, EK_{BH} et KS_{BH}, les entiers constants à 32 bits (c'est-à-dire 0x2AD01C64, etc.) correspondent à des chiffres décimaux de *e* (à savoir: 2,71828...) et pour EK_{GH} et KS_{GH}, ils correspondent à des chiffres décimaux de π (c'est-à-dire 3,14159...). Pour EK_{AG}, EK_{BH}, KS_{AG} et KS_{BH}, les entiers à 32 bits proviennent de blocs de 9 chiffres décimaux, respectivement les premier, deuxième, quatrième et septième blocs. Pour EK_{GH} et KS_{GH}, les entiers à 32 bits proviennent respectivement des 10 premiers chiffres décimaux de π et des 8 chiffres décimaux suivants de π.

Tableau 1/H.235.4 – Calcul des clés de chiffrement et de salage à partir d'un secret partagé

Clé voulue	Paramètre inkey de la fonction PRF	Constante || challenge
EK_{AG}	K_{AG}	0x2AD01C64 || **Challenge-A**
KS_{AG}	K_{AG}	0x150533E1 || **Challenge-A**
EK_{BH}	K_{BH}	0x1B5C7973 || **Challenge-B**
KS_{BH}	K_{BH}	0x39A2C14B || **Challenge-B**
EK_{GH}	K_{HG}	0x54655307 || **Challenge-G**
KS_{GH}	K_{HG}	0x35855C60 || **Challenge-G**

13 Procédure de calcul de la clé fondée sur la Norme FIPS-140

Le présent paragraphe pourra décrire une procédure qui indique comment calculer les données de clé à partir d'un secret partagé et d'autres paramètres au moyen d'un module de chiffrement conforme à la Norme FIPS-140. Ce sujet appelle un complément d'étude.

14 Liste des identificateurs d'objet

Tableau 2/H.235.4 – Identificateurs d'objet utilisés dans la Rec. UIT-T H.235.4

Référence de l'identificateur d'objet	Valeur de l'identificateur d'objet	Description
"I10"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 48}	Utilisé dans la procédure DRC1 pendant l'échange de messages GRQ/RRQ et GCF/RCF et ARQ pour permettre au point d'extrémité/portier d'indiquer la prise en charge de la procédure DRC1.
"I11"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 49}	Utilisé dans les procédures DRC1, DRC2 et DRC3 pour le tokenOID de ClearToken, indiquant que le ClearToken CT_A contient une clé de bout en bout pour l'appelant.
"I12"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 50}	Utilisé dans les procédures DRC1, DRC2 et DRC3 pour le tokenOID de ClearToken, indiquant que le ClearToken CT_B contient une clé de bout en bout pour l'appelé.
"I13"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 52}	Utilisé dans la procédure DRC1 pour le tokenOID de ClearToken entre portiers, indiquant que le ClearToken CT_{HG} contient une clé de chiffrement pour le portier d'origine.
"I20"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 53}	Utilisé dans la procédure DRC2 pendant l'échange de messages GRQ/RRQ et GCF/RCF et ARQ pour permettre au point d'extrémité/portier d'indiquer la prise en charge de la procédure DRC2.
"I23"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 56}	Utilisé dans la procédure DRC2 pour le tokenOID de ClearToken entre portiers, indiquant que le ClearToken CT_{HG} contient une clé de chiffrement pour le portier d'origine.
"I30"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 34}	A utiliser dans un ClearToken distinct dans les messages GRQ/RRQ, GCF/RCF, ARQ pour indiquer la prise en charge de la procédure DRC3. A utiliser dans un ClearToken distinct dans un message LRQ pour indiquer l'acheminement des paramètres DH de l'appelant.
"I33"	{itu-t (0) recommendation (0) h (8) 235 version (0) 4 37}	A utiliser dans un ClearToken distinct dans un message LCF pour indiquer l'acheminement des paramètres DH de l'appelé.
"Annex I-HMAC-SHA1-PRF"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 51}	Utilisé dans les procédures DRC1, DRC2 et DRC3 pour le keyDerivationOID dans V3KeySyncMaterial pour indiquer que la méthode de calcul de la clé fondée sur la fonction pseudo-aléatoire HMAC-SHA1 définie au § 12 est appliquée.

SÉRIES DES RECOMMANDATIONS UIT-T

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

Imprimé en Suisse
Genève, 2006