Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

H.235

Corrigendum 1

(01/2005)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Infrastructure des services audiovisuels – Aspects système

Sécurité et chiffrement pour les terminaux multimédias de la série H (terminaux H.323 et autres terminaux de type H.245)

Corrigendum 1

Recommandation UIT-T H.235 (2003) – Corrigendum 1

Union internationale des télécommunications

RECOMMANDATIONS UIT-T DE LA SÉRIE H

SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

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

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

Recommandation UIT-T H.235

Sécurité et chiffrement pour les terminaux multimédias de la série H (terminaux H.323 et autres terminaux de type H.245)

Corrigendum 1

Résumé

La version 3 de la Rec. UIT-T H.235, qui remplace la deuxième, définit une procédure applicable aux signaux DTMF chiffrés, des identificateurs d'objet pour l'algorithme de chiffrement AES des charges utiles de médias, le mode de chiffrement amélioré OFB des flux (mode EOFB) pour le chiffrement des flux de médias; elle décrit également une option d'authentification seulement dans l'Annexe D applicable au franchissement des dispositifs NAT/pare-feu, une procédure de distribution des clés sur le canal RAS, des procédures de transport de clé de session mieux sécurisé et des procédures de distribution et de mise à jour de clés de session plus fiables, des procédures permettant de sécuriser des flux de charge utile multiples, une meilleure prise en charge de la sécurité pour les appels acheminés directement (nouvelle Annexe I), des moyens plus souples de signalement des erreurs, des précisions et des améliorations d'efficacité pour la sécurité à démarrage rapide et pour la signalisation Diffie-Hellman avec des paramètres Diffie-Hellman plus longs et introduit des modifications tirées du guide à l'usage des responsables de l'implémentation de la Rec. UIT-T H.323.

L'Amendement 1 à la version 3 de la Rec. UIT-T H.235 complétait cette dernière par une nouvelle Annexe H et par de nouvelles fonctionnalités dans l'Annexe I. Les modifications qui sont apportées à la notation ASN.1 pour tenir compte de la nouvelle Annexe H visent à prendre en charge de nouvelles fonctions identifiées par la séquence **profileInfo** du champ ClearToken. L'Amendement 1 visait également à apporter quelques corrections et mises à jour du texte de la version 3 de la Rec. UIT-T H.235.

Le Corrigendum 1 aligne la définition donnée pour la fonction pseudo aléatoire au § B.7 de l'Annexe B sur la définition donnée pour cette même fonction dans la norme RFC 3830; il apporte des modifications de forme aux Figures F.2 et F.3 ainsi qu'un certain nombre de corrections en différents points de l'Annexe I.

Source

Le Corrigendum 1 de la Recommandation UIT-T H.235 (2003) a été approuvé le 8 janvier 2005 par la Commission d'études 16 (2005-2008) de l'UIT-T selon la procédure définie dans la Recommandation UIT-T A.8.

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 avait 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

Sécurité et chiffrement pour les terminaux multimédias de la série H (terminaux H.323 et autres terminaux de type H.245)

Corrigendum 1

...

2 Références normatives

…

– IETF RFC 3830 (2004), MIKEY: Multimedia Internet KEYing.

...

Annexe B

Points spécifiques de la Rec. UIT-T H.323

...

B.7 Fonction pseudo aléatoire (PRF)

Dans ce paragraphe, on définit une fonction pseudo aléatoire (PRF, *pseudo-random function*) dans le but de déduire des clés dynamiques à partir d'éléments de clé statique et d'une valeur aléatoire.

NOTE – Cette fonction PRF est identique à la fonction PRF MIKEY (voir [MIKEY]/RFC ~~xxxx~~3830 section 4.1.2).

La méthode de calcul de la clé fait appel aux paramètres d'entrée suivants:

- *inkey*: clé d'entrée en direction de la fonction de dérivation;
- *inkey_len*: longueur en bits de la clé d'entrée;
- *label*: étiquette spécifiée dépendant du type de clé à obtenir et de la valeur aléatoire **challenge**.
- *outkey_len*: longueur souhaitée en bits de la clé de sortie.

La fonction pseudo-aléatoire dispose des sorties suivantes:

- *outkey*: clé de sortie de la longueur désirée.

~~Soit HMAC (voir [RFC 2104]), une fonction d'authentification de message fondée sur l'algorithme SHA1 (voir [ISO/CEI 10118-3]). De manière analogue à la [RFC 2246], définissons:~~Cette fonction PRF doit utiliser la fonction PRF définie à la section 4.1.2 de la norme RFC 3830.

~~*P (s, label, m) =* HMAC (*s, A_1 || label*) ||~~

~~HMAC (*s, A_2 || label*) || ...~~

~~HMAC (*s, A_m || label*)~~

où:

A_0 = étiquette,

A_i = HMAC (s, A_{i-1}).

L'algorithme SHA1 [ISO/CEI 10118-3] est l'algorithme par défaut, mais on peut utiliser le HMAC faisant appel à d'autres fonctions de hachage; ce point appelle un complément d'étude.

La procédure suivante décrit une fonction pseudo aléatoire appelée *PRF(inkey, label)*, utilisée pour calculer la clé de sortie, outkey:

- soit n = *inkey_len/512*, arrondi à l'entier le plus proche;
- scinder *inkey* en *n* blocs, *inkey* = s_1 || ... || s_n, où tous les s_i, sauf éventuellement s_n, ont une longueur de 512 bits chacun;
- soit *m* = *outkey_len/160*, arrondi à l'entier le plus proche.

La clé de sortie, *outkey*, est obtenue comme étant les bits de plus fort poids de *outkey_len* de

PRF(inkey, label) = P(s_1, label, m) XOR P(s_2, label, m) XOR ... XOR P(s_n, label, m).

...

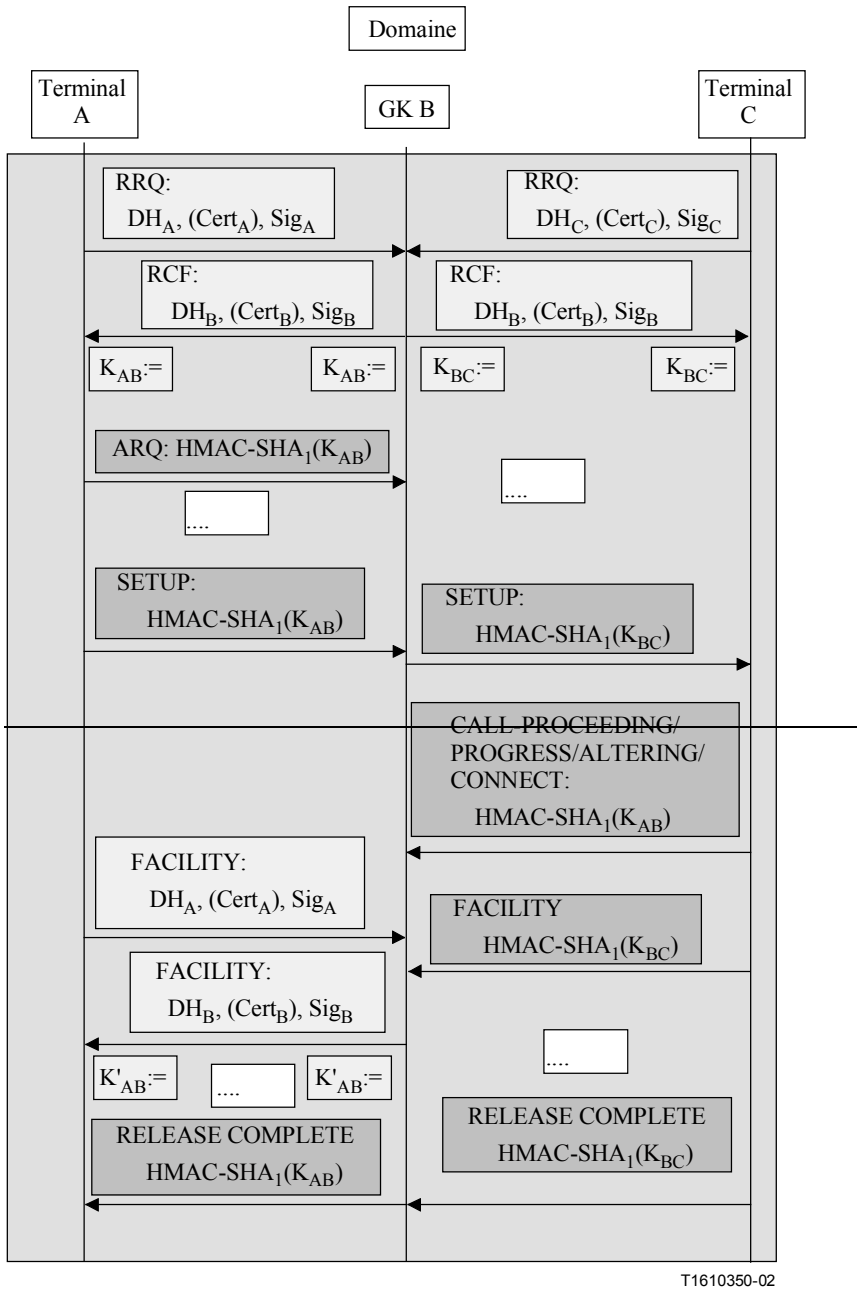
Annexe F

Profil hybride de sécurité

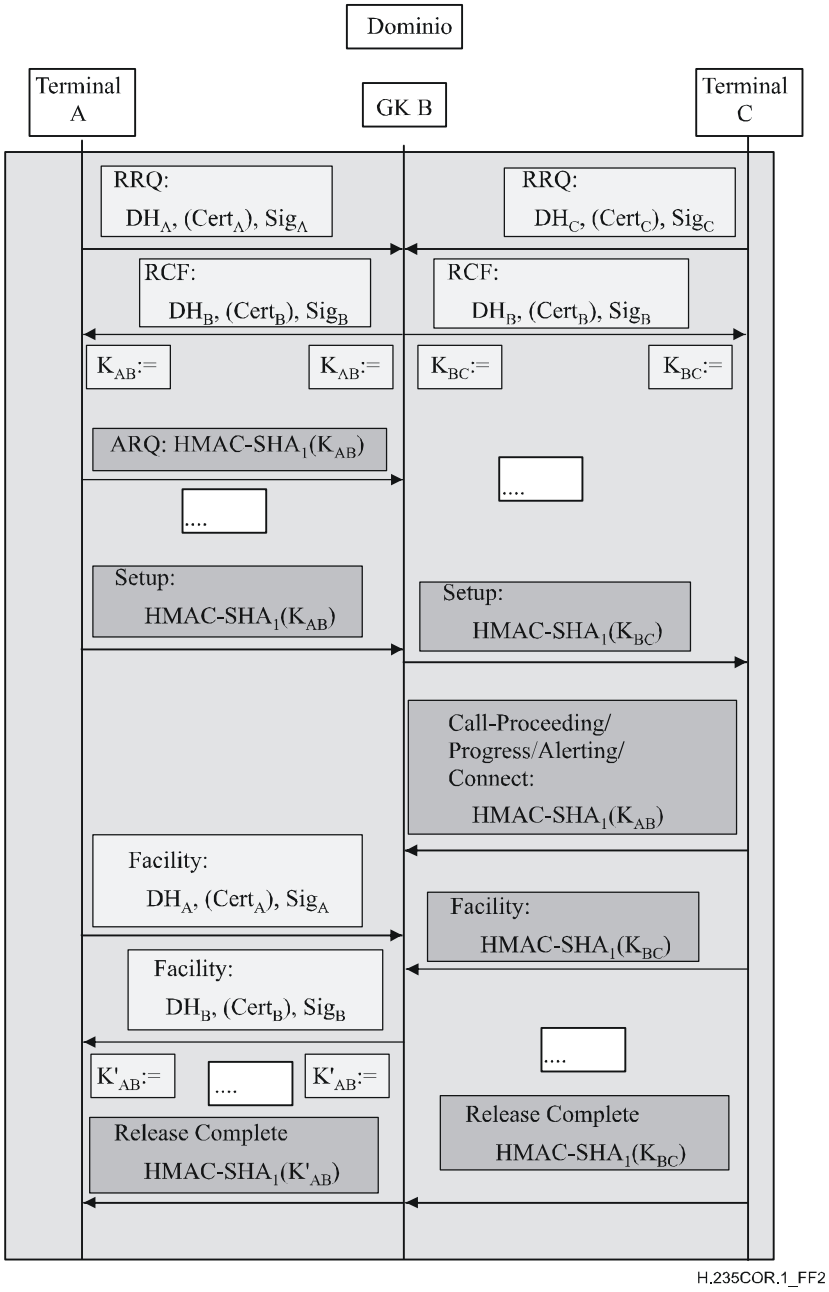
...

F.10 Exemples avec organigrammes

...

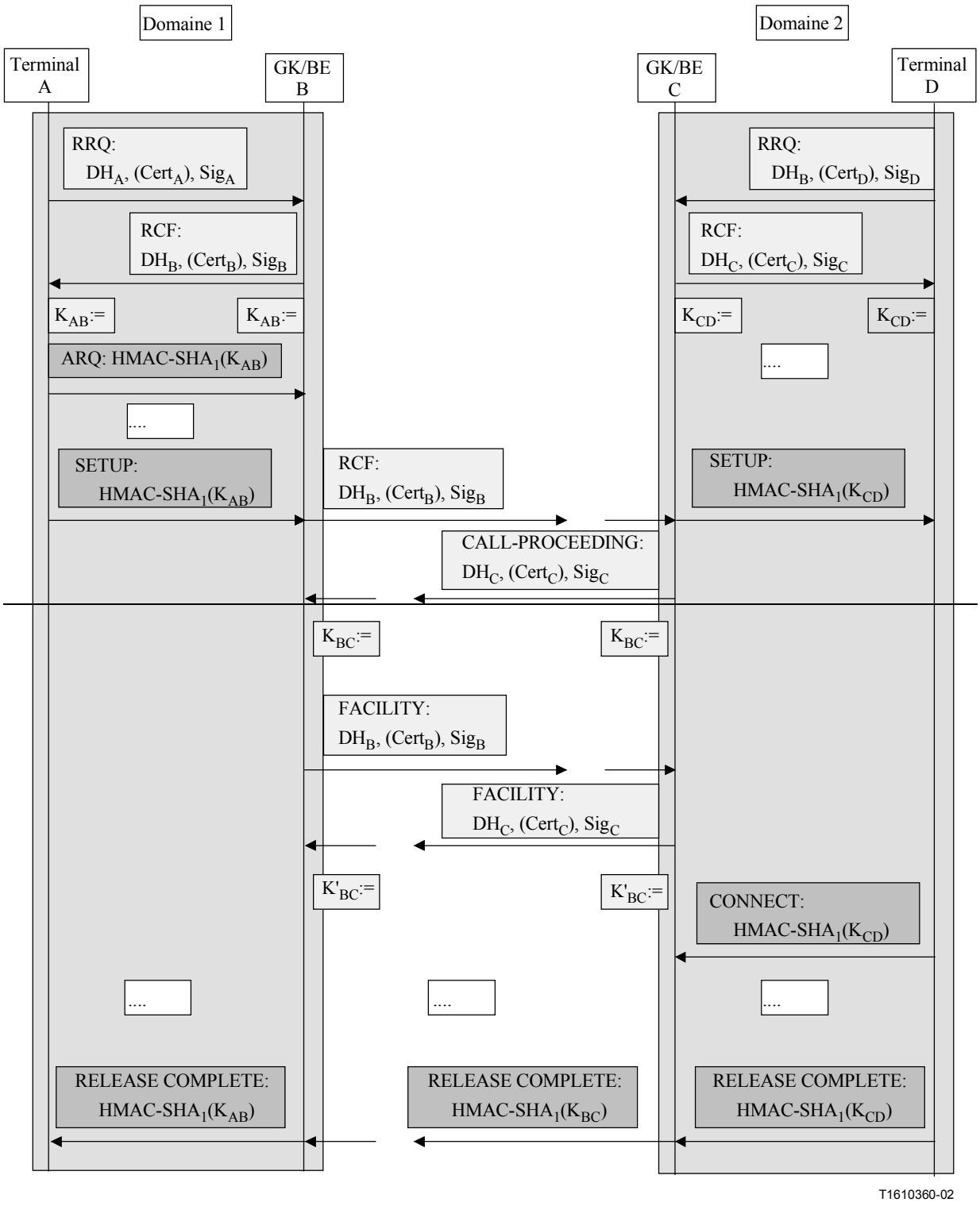


Cert	Certificat d'utilisateur	K, K'	Clé de liaison symétrique
DH_A	Jeton g^a mod p Diffie-Hellman	Sig	Signature numérique
DH_B	Jeton g^b mod p Diffie-Hellman		
EP	Point d'extrémité (Terminal)		
GK	Portier		



Cert	certificat d'utilisateur	K, K'	Clé de liaison symétrique
DH_A	jeton g^a mod p Diffie-Hellman	Sig	signature numérique
DH_B	jeton g^b mod p Diffie-Hellman		
EP	point d'extrémité (Terminal)		
GK	portier		

Figure F.2/H.235 – Flux de messages dans un domaine administratif


Domaine 1
Domaine 2
Terminal A
GK/BE B
GK/BE C
Terminal D
RRQ: DHA, (CertA), SigA
RRQ: DHB, (CertD), SigD
RCF: DHB, (CertB), SigB
RCF: DHC, (CertC), SigC
KAB:=
KAB:=
KCD:=
KCD:=
ARQ: HMAC-SHA1(KAB)
....
....
SETUP: HMAC-SHA1(KAB)
RCF: DHB, (CertB), SigB
SETUP: HMAC-SHA1(KCD)
CALL-PROCEEDING: DHC, (CertC), SigC
KBC:=
KBC:=
FACILITY: DHB, (CertB), SigB
FACILITY: DHC, (CertC), SigC
K'BC:=
K'BC:=
CONNECT: HMAC-SHA1(KCD)
....
....
....
RELEASE COMPLETE: HMAC-SHA1(KAB)
RELEASE COMPLETE: HMAC-SHA1(KBC)
RELEASE COMPLETE: HMAC-SHA1(KCD)
T1610360-02

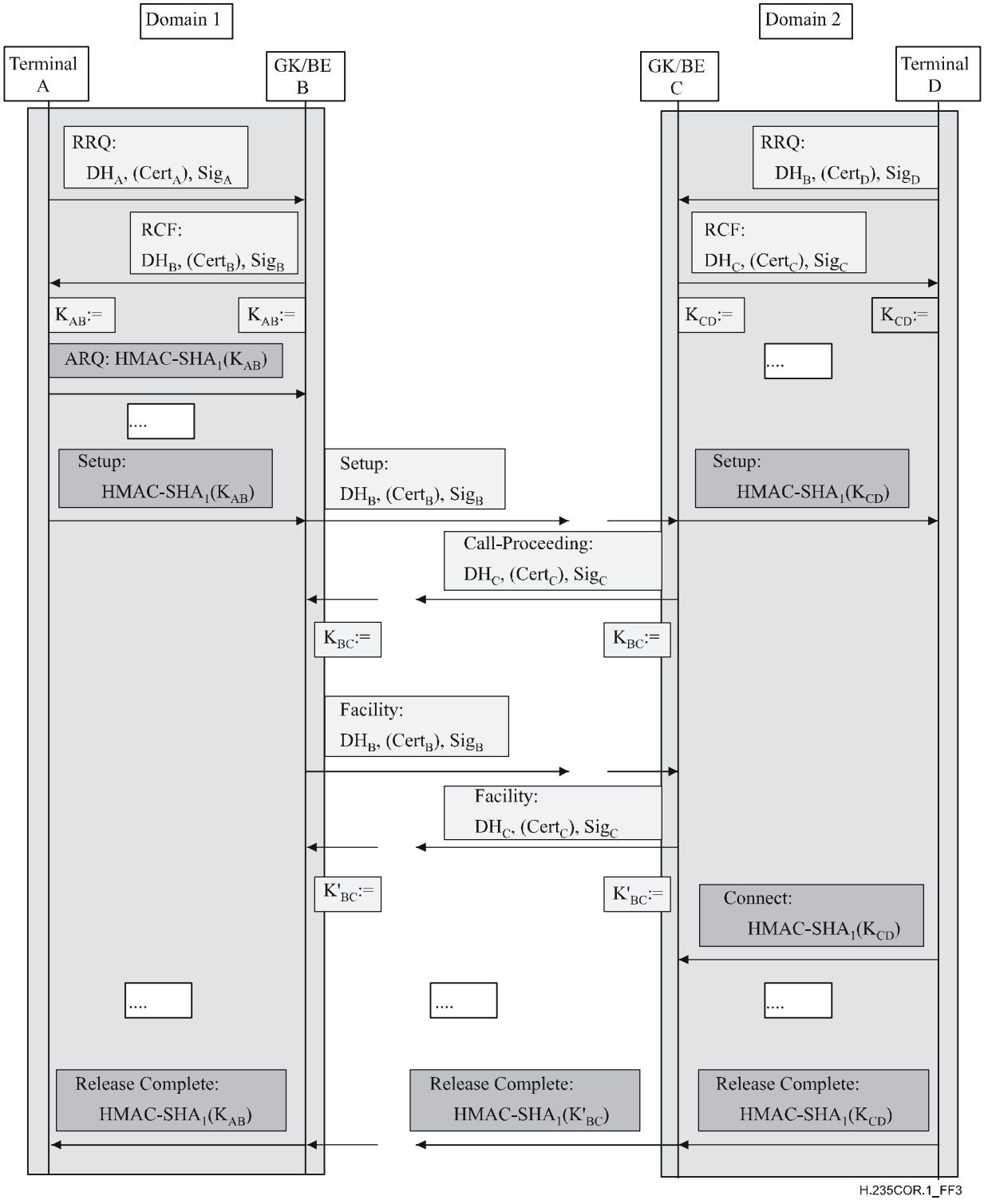


Figure F.3/H.235 – Flux de messages dans un domaine à plusieurs administrations

…

Annexe I

Prise en charge des appels à acheminement direct

...

I.5 Symboles et abréviations

La présente annexe utilise les abréviations suivantes:

$ENC_{K;\ S,\ IV}(M)$	chiffrement EOFB de *M* au moyen de la clé secrète *K* et de la clé secrète de salage *S* et du vecteur initial *IV (EOFB encryption of M using secret key K and secret salting key S and initial vector IV)*
CT	clearToken
DRC	appel à acheminement direct *(direct-routed call)*
EPID	identificateur de point d'extrémité *(endpoint identifier)*
GKID	identificateur de portier (*gatekeeper identifier*)
K_{AG}	Secret partagé (Annexe D, Annexe F) entre EP A et GK G (*shared secret (Annex D, Annex F) between EP A and GK G*)
K_{BH}	secret partagé (Annexe D, Annexe F) entre EP B et GK G (*shared secret (Annex D, Annex F) between EP B and GK H)*
K_{GH}	secret partagé (Annexe D, Annexe F) entre GK G et GK H (*shared secret*)
KS_{AG}	clé de salage partagée secrète entre EP A et GK G (*secret, shared salting key between EP A and GK G*)
KS_{BH}	clé de salage partagée secrète entre EP B et GK H (*secret, shared salting key between EP B and GK H*)
KS_{GH}	clé de salage partagée secrète entre GK G et GK H (*secret, shared salting key between GK G and GK H*)
EK_{AG}	clé de chiffrement partagée entre EP A et GK G (*the encryption key shared between EP A and GK G*)
EK_{BH}	clé de chiffrement partagée entre EP B et GK H (*the encryption key shared between EP B and GK H*)
EK_{GH}	clé de chiffrement partagée entre GK G et GK H (*the encryption key shared between GK G and GK H*)
K_{AB}	clé de chiffrement partagée entre EP A et EP B (*the encryption key shared between EP A and EP B*)
PRF	fonction pseudo-aléatoire (*pseudo-random function*)

...

I.9 Procédure DRC

Les points d'extrémité en mesure de prendre en charge ce profil de sécurité doivent l'indiquer pendant l'envoi des messages **GRQ** et/ou **RRQ** en incluant un ClearToken distinct dans lequel **tokenOID** est mis à "I0"; les autres champs dans ce ClearToken ne doivent pas être utilisés. Les portiers disposant des capacités spécifiées dans l'Annexe I souhaitant offrir cette fonctionnalité

doivent répondre respectivement par un message **GCF** ou **RCF** avec un ClearToken distinct inclus dans un **tokenOID** mis à "I0", tous les autres champs du ClearToken étant inutilisés.

Avant qu'un point d'extrémité A commence à envoyer directement des messages de signalisation d'appel à un autre point d'extrémité B, le point d'extrémité A ou B doit demander son admission au portier G ou H au moyen d'un message **ARQ**. Le point d'extrémité A doit inclure dans le message **ARQ** un ClearToken distinct avec **tokenOID** mis à "I0", tous les autres champs de ClearToken étant inutilisés.

Cette procédure s'applique aussi bien à un seul portier commun à plusieurs points d'extrémité qu'à plusieurs portiers enchaînés. Dans le cas de plusieurs portiers, le portier G – zone de laquelle l'appel provient – devrait localiser le portier H au moyen du mécanisme **LRQ** (multidiffusion) tel que décrit dans le § 8.1.6/H.323 "signalisation facultative par l'extrémité appelée". La communication entre deux portiers doit être sécurisée conformément à l'Annexe D. Pour cela, on part du principe qu'un secret partagé commun K_{GH} est disponible. Etant donné que le message **LRQ** parmi les portiers est généralement un message multidiffusion, 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 portier.

NOTE – Cette hypothèse limite l'échelonnabilité dans le cas général et ne permet pas l'authentification de sources. 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 multidiffusion 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 à l'extrémité distante, le message **LRQ** doit alors acheminer un jeton ClearToken distinct dont l'identificateur **tokenOID** est mis à "I0"; tous les autres champs de ce ClearToken ne devraient pas être utilisés. En mode multidiffusion, l'identificateur **generalID** dans le jeton ~~Crypto~~ClearToken du message **LRQ** ne doit pas être utilisé. La communication entre portiers H.501 et/ou H.510 fera l'objet d'un complément d'étude.

EK_{BH} désigne la clé de chiffrement et KS_{BH} désigne la clé de salage qui ~~est~~sont partagée<u>s</u> 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 doit générer une constante Challenge-B aléatoire, 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, telle que définie dans le § I.10 où la constante Challenge-B remplace la constante **challenge** et où CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→keyDerivationOID** ~~dans **V3KeySyncMaterial**~~ doit définir "Annex I-HMAC-SHA1-PRF"; voir le § I.12.

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 doit générer une constante Challenge-G aléatoire, 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, telle que définie dans le § 11 où la constante Challenge-G remplace la constante **challenge**. CT_{HG} **→challenge** doit contenir la constante challenge-G. L'identificateur (ID) du point d'extrémité B doit être défini dans CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→generalID**.

Le portier H doit transmettre la clé EK_{BH} chiffrée au portier G. Le mode de chiffrement OFB amélioré (EOFB) (voir le § B.2.5) doit être utilisé avec le secret, la clé de salage KS_{GH} propre au point d'extrémité. Les algorithmes de chiffrement applicables sont les suivants (voir le § D.11):

- DES (56 bits) dans le mode EOFB au moyen de l'identificateur OID "Y1": optionnel;
- 3DES (168 bits) dans le mode EOFB externe au moyen de l'identificateur OID "Z1": optionnel;

- AES (128 bits) dans le mode EOFB au moyen de l'identificateur OID "Z2": par défaut et recommandé;
- compatible RC2 (56 bits) dans le mode EOFB au moyen de l'identificateur OID "X1": optionnel.

Pour le mode de chiffrement EOFB, le portier H doit générer une valeur initiale aléatoire IV. Pour les OID "X1", OID "Y1"et OID "Z1", le vecteur IV occupe 64 bits et doit être acheminé dans ~~le champ~~CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→params→iv8** ~~de l'élément **params** dans **V3KeySyncMaterial**~~; en revanche, le vecteur IV occupe 128 bits pour l'OID "Z2" et doit être acheminé dans ~~le champ~~ CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→params→iv16** ~~de **params** dans **V3KeySyncMaterial**~~.

Le portier H doit inclure $ENC_{EK_{GH},\ KS_{GH},\ IV}(EK_{BH})$ dans le ClearToken CT_{HG} dont l'identificateur **tokenOID** est mis à "I3". Le texte chiffré obtenu $ENC_{EK_{GH},\ KS_{GH},\ IV}(EK_{BH})$ doit être acheminé dans CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→encryptedSessionKey** ~~la structure de données **h235key** comme faisant partie de **secureSharedSecret** où il doit être inséré dans le **encryptedSessionKey** de la structure de données **secureSharedSecret**~~. L'algorithme de chiffrement doit être indiqué dans CT_{HG}**→h235Key→V3KeySyncMaterial→algorithmOID** ("X1", "Y1", "Z1" ou "Z2") ~~dans **V3KeySyncMaterial**~~. La composante Challenge-B doit être placée dans CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→clearSaltingKey**. CT_{HG}**→generalID** doit être mis à la valeur de l'identificateur du portier G alors que CT_{HG}**→sendersID** doit être mis à la valeur de l'identificateur du portier H. La réponse **LCF** doit définir le ClearToken CT_{HG}.

Le portier G constatant que les points d'extrémité A et B sont compatibles avec la présente annexe doit générer les éléments de clé et les ClearToken comme spécifié ci-dessous.

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

Deux ClearToken doivent être inclus, un CT_A pour l'appelant A et un autre CT_B pour l'appelé B. Chaque **ClearToken** doit contenir un identificateur OID ("I1" ou "I2") à l'intérieur de **tokenOID** qui indique si le jeton est destiné à l'appelant (OID "I1" pour CT_A) ou à l'appelé (OID "I2" pour CT_B).

Le **ClearToken** tel que défini dans la présente annexe, peut être utilisé en association avec d'autres profils de sécurité tels que ceux décrits dans les Annexes D ou F qui mettent en œuvre des **ClearTokens** également. En pareil cas, un ClearToken conforme à l'Annexe I doit utiliser d'autres champs de **ClearToken** également. Par exemple, afin d'utiliser l'Annexe I en association avec l'Annexe D, les champs **timeStamp**, **random**, **generalID**, **sendersID** et **dhkey** doivent être présents et être utilisés tels que décrits par les profils de sécurité définis dans l'Annexe D.

L'identificateur de portier (GKID) du portier G doit être inséré dans CT_A**→sendersID** et dans CT_B**→sendersID** tandis que CT_A**→generalID** ~~ne~~ doit ~~pas~~ contenir ~~d~~l'identificateur ~~des~~du point~~s~~ d'extrémité A ~~(CT_A) ou~~ et CT_B**→generalID** l'identificateur du point d'extrémité B ~~(CT_B)~~.

~~EK désigne la clé de chiffrement qui est partagée entre un point d'extrémité et son portier.~~ Le portier G doit générer les données de clés de salage KS_{HG} et les données de clé de chiffrement EK_{HG} à partir de K_{HG} au moyen de la procédure de calcul de clé fondée sur la fonction PRF, telle que définie dans le § 11, la constante **challenge** étant remplacée par CT_{HG} **→challenge**.

Les clés de chiffrement EK_{AG} et EK_{BH} pour la clé chiffrée de bout en bout K_{AB} doivent être calculées à partir du secret partagé entre le portier et les points d'extrémité (EK_{AG} ou EK_{BH}) en utilisant la procédure de calcul de clé **fondée** sur la fonction PRF comme défini au § I.10 où CT_A**→h235Key→V3KeySyncMaterial→secureSharedSecret→keyDerivationOID** et

CT_B**→h235Key→V3KeySyncMaterial→secureSharedSecret→keyDerivationOID** ~~dans **V3KeySyncMaterial**~~ doivent contenir "Annex I-HMAC-SHA1-PRF", voir § I.12 et où CT_A**→challenge** doit contenir la constante Challenge-A.

~~Le portier G doit générer un secret de session commun partagé K_{AB}, qui est partagé entre le point d'extrémité A et le point B.~~ Le portier G doit reproduire la constante Challenge-B figurant dans CT_{HG}**→h235Key→V3KeySyncMaterial→secureSharedSecret→clearSaltingKey** dans CT_B**→challenge**.

Ce secret de session K_{AB} doit être chiffré par EK_{AG} (pour l'identificateur CT destiné au point d'extrémité A) ou par EK_{BH} (pour l'identificateur CT destiné au point d'extrémité B) au moyen de l'algorithme de chiffrement.

Le mode de chiffrement OFB amélioré (EOFB) (voir § B.2.5) doit être utilisé avec le secret, la clé de salage KS_{AG} ou KS_{BH} propre au point d'extrémité. Les algorithmes de chiffrement applicables sont les suivants (voir § D.11):

- DES (56 bits) dans le mode EOFB utilisant l'OID "Y1": optionnel;
- 3DES (168 bits) dans le mode EOFB externe utilisant l'OID "Z1": optionnel;
- AES (128 bits) dans le mode EOFB utilisant l'OID "Z2": par défaut et recommandé;
- compatible RC2 (56 bits) dans le mode EOFB utilisant l'OID "X1": optionnel.

Pour le mode de chiffrement EOFB, le portier G doit générer une valeur initiale aléatoire IV. Pour les OID "X1", OID "Y1" et OID "Z1", le vecteur IV occupe 64 bits et doit être acheminé dans ~~le champ~~ CT_A**→h235Key→V3KeySyncMaterial→secureSharedSecret→params→iv8** ~~de l'élément **params**~~ et dans ~~**V3KeySyncMaterial**~~ CT_B**→h235Key→V3KeySyncMaterial→secureSharedSecret→params→iv8**; en revanche le vecteur IV occupe 128 bits pour l'OID "Z2" et doit être acheminé dans ~~le champ~~ CT_A**→h235Key→V3KeySyncMaterial→secureSharedSecret→params→iv16** ~~de params~~ et dans ~~**V3KeySyncMaterial**~~ CT_B**→h235Key→V3KeySyncMaterial→secureSharedSecret→params→iv16**.

Le~~s~~ texte~~s respectifs~~ chiffré~~s~~ obtenu~~s~~ $ENC_{EK_{AG},\ KS_{AG},\ IV}(K_{AB})$ doit être acheminé dans CT_A**→h235Key→V3KeySyncMaterial→secureSharedSecret→encryptedSessionKey** et le texte chiffré obtenu $ENC_{EK_{BG},\ KS_{BG},\ IV}(K_{AB})$ doit être acheminé dans CT_B**→h235Key→V3KeySyncMaterial→secureSharedSecret→** ~~doivent être alors acheminés dans la structure de données **h235key** comme faisant partie de **secureShareSecret** où ils doivent être insérés dans le **encryptedSessionKey** de la structure de données **secureSharedSecret**~~. L'algorithme de chiffrement doit être indiqué dans CT_A**→h235Key→V3KeySyncMaterial→secureSharedSecret→algorithmOID** et dans CT_B**→h235Key→V3KeySyncMaterial→secureSharedSecret→algorithmOID** ("X1", "Y1", "Z1" ou "Z2") ~~dans **V3KeySyncMaterial**~~.

Pour le ClearToken destiné au point d'extrémité A, l'identificateur de point d'extrémité du point d'extrémité B ($EPID_B$) doit être inséré dans CT_A**→h235Key→V3KeySyncMaterial→secureSharedSecret→generalID** ~~de **V3KeySyncMaterial**~~. De même que pour le ClearToken destiné au point d'extrémité B, ~~le point d~~l'identificateur de point d'extrémité du point d'extrémité A ($EPID_A$) doit être inséré dans ~~l'élément~~ CT_B**→h235Key→V3KeySyncMaterial→secureSharedSecret→generalID** ~~de **V3KeySyncMaterial**~~.

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

Le portier G doit inclure à la fois les identificateurs CT_A et CT_B de ClearToken dans le message **ACF** en direction du point d'extrémité A.

Le point d'extrémité A doit identifier le CT_A par inspection de l'identificateur **tokenOID** "I1" dans ClearToken.

Le point d'extrémité A doit vérifier que l'identificateur CT_A est tout nouveau au moyen du **timestamp**. Des contrôles de sécurité plus poussés doivent être effectués pour vérifier le **generalID** et le **sendersID** de ClearToken et le **generalID** dans **V3KeySyncMaterial**. Si après vérification, l'identificateur CT_A reçu est tout nouveau, le point d'extrémité A doit récupérer le vecteur IV et calculer EK_{AG} et KS_{AG} comme décrit ci-dessus pour le portier G. Le point d'extrémité A doit décrypter l'information **encryptedSessionKey** qui se trouve dans **V3KeySyncMaterial** du CT_A pour obtenir la clé ~~E~~K_{AB}.

Si après vérification, il s'avère que l'identificateur 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 l'identificateur CT_B et doit être sécurisé (authentifié et/ou protégé dans son intégrité) au moyen des profils décrits dans les Annexes D ou F en se servant de la clé K_{AB} comme secret partagé. A cette fin, l'élément **generalID** du jeton ClearToken haché de l'Annexe D, (non pas CT_B!) ne doit pas être utilisé à moins que le point d'extrémité A ait déjà un $EPID_B$ disponible (par exemple, par configuration ou par mémorisation d'une communication antérieure. S'il utilise une valeur $EPID_B$ pour l'identificateur **generalID** dans le message SETUP, le point d'extrémité A doit accepter la valeur de l'identificateur **sendersID** dans le message de signalisation d'appel renvoyé en tant que l'$EPID_B$ Vrai.

Le point d'extrémité B doit identifier CT_B par inspection de l'identificateur **tokenID** "I2" dans ClearToken.

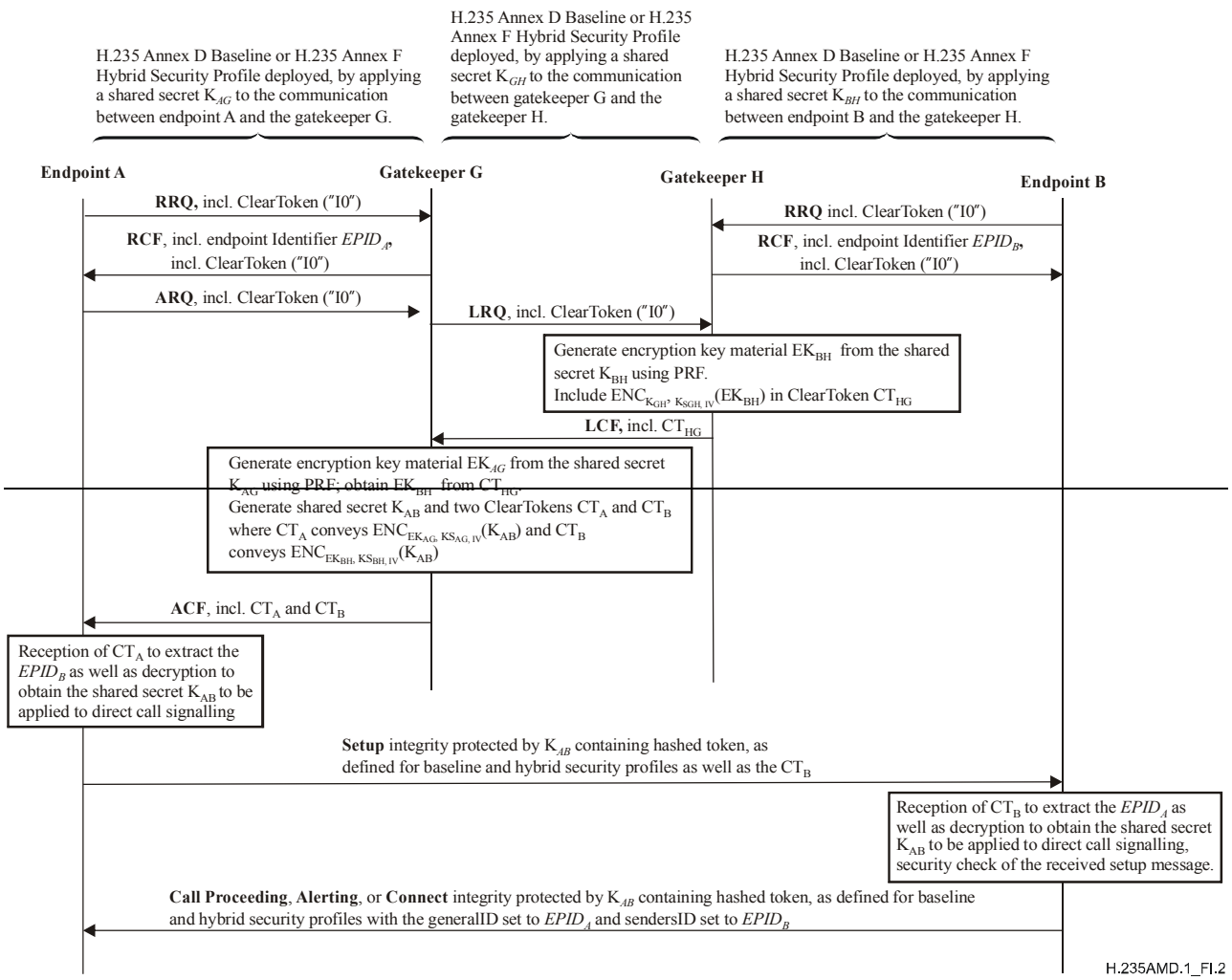
Le point d'extrémité B doit vérifier que l'identificateur CT_B est tout nouveau en utilisant l'horodate **timeStamp**. Des vérifications plus poussées doivent être effectuées sur **sendersID** de ClearToken et sur **generalID** dans **V3KeySyncMaterial**. Si l'identificateur CT_B reçu est après vérification effectivement tout nouveau, le point d'extrémité B doit récupérer le vecteur IV et calculer EK_{BH}~~G~~ et KS_{BH}~~G~~ tel que décrit ci-dessus pour le portier. Le point d'extrémité B doit décrypter l'information **encryptedSessionKey** se trouvant dans **V3KeySyncMaterial** de l'identificateur CT_B pour obtenir le secret partagé K_{AB}.

Si après vérification, il s'avère que l'identificateur CT_B est tout nouveau, le point d'extrémité B est en utilise 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 l'identificateur n'est pas tout nouveau ou si la vérification de sécurité du message SETUP révèle un problème, le point d'extrémité B doit répondre par un message RELEASE-COMPLETE, l'élément **ReleaseCompleteReason** étant mis à une erreur de sécurité définie dans le § B.2.2.

Lorsque la sécurité de media est appliquée (voir § D.7), le point d'extrémité A et le point d'extrémité B doivent procéder à l'échange des demi-clés de Diffie-Hellman conformément au § D.7.1 et établir une clé maître dynamique de session à partir de laquelle des clés propres au média peuvent être déduites.

Le point d'extrémité B doit inclure generalID mis à $EPID_A$ et sendersID mis à $EPID_B$ pour la protection de tout message de signalisation d'appel H.225.0 destiné à EP A (par exemple, Call Proceeding, Alerting ou Connect).

La Figure I.2 illustre le flux de communications de base.


H.235 Annex D Baseline or H.235 Annex F Hybrid Security Profile deployed, by applying a shared secret K_{AG} to the communication between endpoint A and the gatekeeper G.
H.235 Annex D Baseline or H.235 Annex F Hybrid Security Profile deployed, by applying a shared secret K_{GH} to the communication between gatekeeper G and the gatekeeper H.
H.235 Annex D Baseline or H.235 Annex F Hybrid Security Profile deployed, by applying a shared secret K_{BH} to the communication between endpoint B and the gatekeeper H.
Endpoint A
Gatekeeper G
Gatekeeper H
Endpoint B
RRQ, incl. ClearToken (″I0″)
RRQ incl. ClearToken (″I0″)
RCF, incl. endpoint Identifier $EPID_A$, incl. ClearToken (″I0″)
RCF, incl. endpoint Identifier $EPID_B$, incl. ClearToken (″I0″)
ARQ, incl. ClearToken (″I0″)
LRQ, incl. ClearToken (″I0″)
Generate encryption key material EK_{BH} from the shared secret K_{BH} using PRF.
Include $ENC_{K_{GH}, KS_{GH, IV}}(EK_{BH})$ in ClearToken CT_{HG}
LCF, incl. CT_{HG}
Generate encryption key material EK_{AG} from the shared secret K_{AG} using PRF; obtain EK_{BH} from CT_{HG}.
Generate shared secret K_{AB} and two ClearTokens CT_A and CT_B where CT_A conveys $ENC_{EK_{AG}, KS_{AG, IV}}(K_{AB})$ and CT_B conveys $ENC_{EK_{BH}, KS_{BH, IV}}(K_{AB})$
ACF, incl. CT_A and CT_B
Reception of CT_A to extract the $EPID_B$ as well as decryption to obtain the shared secret K_{AB} to be applied to direct call signalling
Setup integrity protected by K_{AB} containing hashed token, as defined for baseline and hybrid security profiles as well as the CT_B
Reception of CT_B to extract the $EPID_A$ as well as decryption to obtain the shared secret K_{AB} to be applied to direct call signalling, security check of the received setup message.
Call Proceeding, Alerting, or Connect integrity protected by K_{AB} containing hashed token, as defined for baseline and hybrid security profiles with the generalID set to $EPID_A$ and sendersID set to $EPID_B$
H.235AMD.1_FI.2

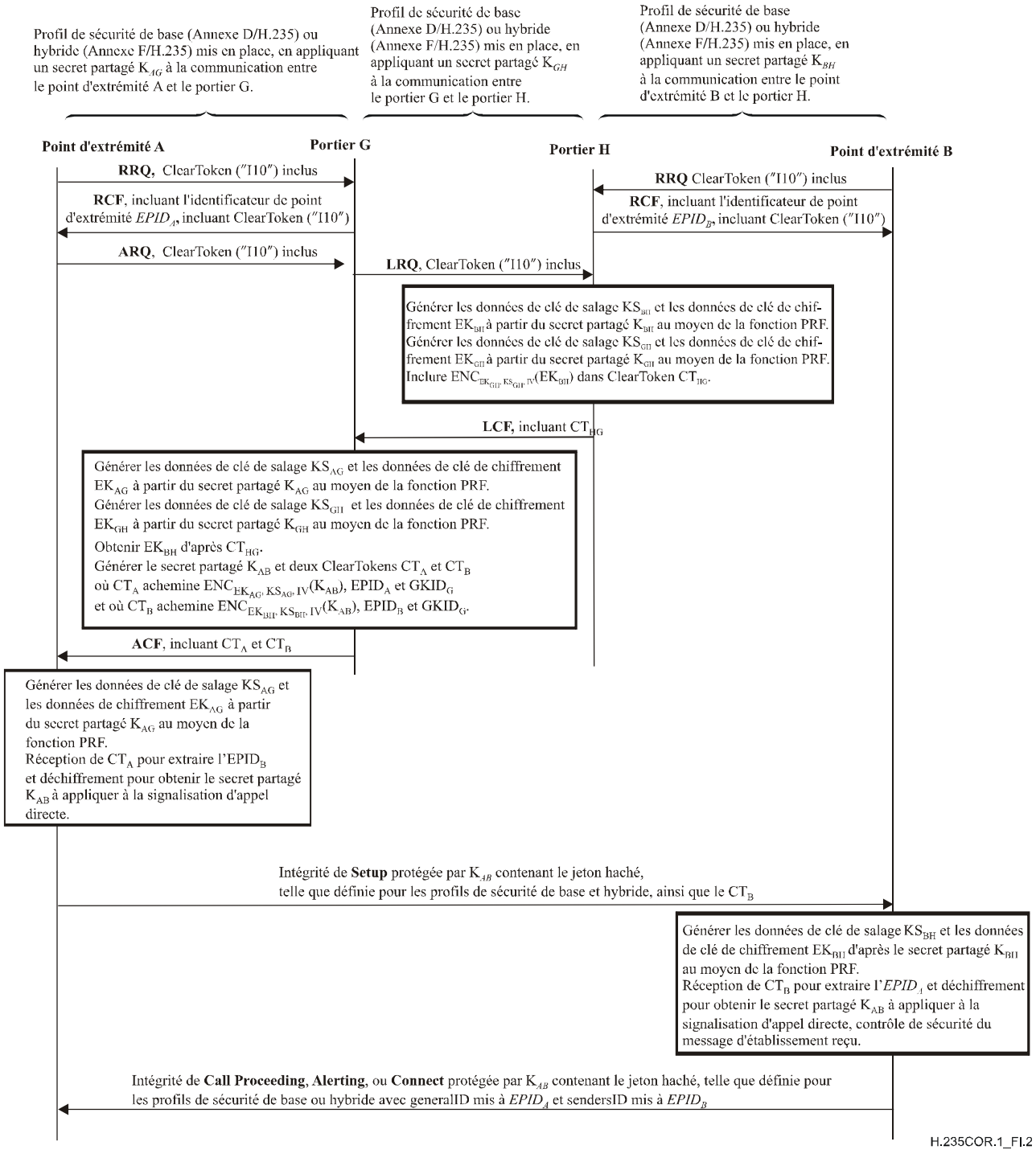


Figure I.2/H.235 – Flux de communications de base

I.10 Procédure d'obtention de la clé au moyen de la fonction PRF

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

~~La clé de chiffrement EK_{AG} doit être calculée au moyen de la fonction PRF (voir § B.7), le paramètre *inkey* étant mis à K_{AG} et *label* mis à la constante 0x2AD01C64 || **challenge**.~~

~~De même, la clé de chiffrement EK_{BG} doit être calculée en utilisant cette fonction PRF, le paramètre *inkey* étant mis à K_{BH} et *label* à la constante 0x1B5C7973 || **challenge**. Dans les deux cas, le paramètre *outkey_len* doit être mis à la longueur requise de la clé de chiffrement pour l'algorithme de chiffrement choisi.~~

~~En utilisant cette même fonction, un secret, une clé de salage partagée doivent être générés par le portier et par chaque point d'extrémité. La clé de salage, lorsqu'elle est utilisée dans le mode de chiffrement EOFB, empêche les attaques de texte clair connues du CT_B par le point d'extrémité A dans lequel le point d'extrémité A pourrait dans les autres cas tenter de découvrir K_{BH}.~~

~~KS_{AG} désigne la clé de salage secrète partagée qui est partagée entre le point d'extrémité A et le portier G. KS_{AG} doit être calculé en utilisant la fonction PRF, le paramètre *inkey* étant mis à K_{AG} et le paramètre *label* à la constante 0x150533E1 || **challenge**. KS_{BH} doit être calculé en utilisant la fonction PRF, le paramètre *inkey* étant mis à K_{BH} et le paramètre *label* à la constante 0x39A2C14B || **challenge**.~~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_{HG}).

Afin d'obtenir les données de clé voulues (EK_{AG}, par exemple), la fonction PRF (voir § B.7) doit être utilisée avec les paramètres du Tableau I.0, 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**), 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 – Les entiers constants à 32 bits (c'est-à-dire 0x2AD01C64, etc.) sont extrait des chiffres décimaux de *e* (à savoir: 2,7182 ...), chaque constante comportant neuf chiffres décimaux (les premiers neuf chiffres décimaux 718281828 = 0x2AD01C64). Les chaînes de neuf chiffres décimaux ne sont pas choisies aléatoirement, mais sont des "fragments" consécutifs des chiffres décimaux de *e*.~~

NOTE – Pour EK_{AG}, KS_{AG}, EK_{BH} et KS_{BH}, les constantes entières à 32 bits (c'est-à-dire 0x2AD01C64, etc.) sont extraites des décimales de *e* (à savoir: 2,7182 ...) et correspondent respectivement au premier, deuxième, quatrième et septième bloc de 9 décimales. Pour EK_{GH} et KS_{GH}, les constantes entières à 32 bits sont extraites des décimales de π (3,1415 ...) et correspondent respectivement aux 10 premières décimales et aux 8 décimales suivantes de π.

Tableau I.0/H.235 – 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_{GH}	0x54655307 || **Challenge-G**
KS_{GH}	K_{GH}	0x35855C60 || **Challenge-G**

I.11 Procédure de calcul de la clé en utilisant la Norme FIPS-140

Le présent paragraphe peut décrire une procédure qui indique comment obtenir les éléments 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.

I.12 Liste des identificateurs d'objet

Tableau I.1/H.235 – Identificateurs d'objet utilisés dans l'Annexe I/H.235

Référence de l'identificateur d'objet	Valeur de l'identificateur d'objet	Description
"I0"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 48}	Utilisé dans la procédure DRC utilisant les messages GRQ/RRQ et GCF/RCF et ARQ pour laisser le point d'extrémité/portier indiquer la prise en charge de l'Annexe I.
"I1"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 49}	Utilisé dans la procédure DRC pour le ClearToken tokenOID indiquant que le ClearToken $\underline{CT_A}$ détient une clé de bout en bout pour l'appelant.
"I2"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 50}	Utilisé dans la procédure DRC pour le ClearToken tokenOID indiquant que le ClearToken $\underline{CT_B}$ détient une clé de bout en bout pour l'appelé.
"I3"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 52}	Utilisé dans la procédure DRC pour le ClearToken tokenOID entre portiers indiquant que le ClearToken $\underline{CT_{HG}}$ détient une clé de chiffrement pour le portier d'origine.
"Annex I-HMAC-SHA1-PRF"	{itu-t (0) recommendation (0) h (8) 235 version (0) 3 51}	Utilisé dans la procédure DRC pour le keyDerivationOID dans V3KeySyncMaterial pour indiquer que la méthode appliquée pour l'obtention de la clé figurant dans le I.10 utilisant la fonction pseudo-aléatoire HMAC-SHA1.

...

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, 2005