Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

H.235.5

(09/2005)

SÉRIE H: SYSTÈMES AUDIOVISUELS ET MULTIMÉDIAS

Infrastructure des services audiovisuels – Aspects système

Cadre de sécurité H.323: cadre de l'authentification sécurisée pendant l'échange de messages RAS au moyen de secrets partagés faibles

Recommandation UIT-T H.235.5

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.235.5

Cadre de sécurité H.323: cadre de l'authentification sécurisée pendant l'échange de messages RAS au moyen de secrets partagés faibles

Résumé

La présente Recommandation décrit le cadre de l'authentification mutuelle entre participants au cours de l'échange de messages RAS H.225.0. Les méthodes fondées sur la "preuve de possession" décrites permettent une utilisation sûre des secrets partagés tels que les mots de passe qui, s'ils étaient utilisés en tant que tels, ne garantiraient pas une sécurité suffisante.

Sont également décrites les extensions de ce cadre visant à permettre la négociation simultanée des paramètres de sécurité de la couche de transport pour la protection d'un canal de signalisation d'appel ultérieur.

Dans les anciennes versions de la sous-série H.235, ce profil était défini dans l'Annexe H/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.235.5 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, mots de passe, sécurité.

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

Introduction

Dans de nombreuses applications, un point d'extrémité (ou son utilisateur) et son portier ne peuvent échanger qu'un "petit" secret tel qu'un mot de passe ou un "numéro d'identification personnel" (PIN, *personal identification number*). Ce type de secret (ci-après dénommé "mot de passe") et toute clé de chiffrement calculée à partir de celui-ci sont faibles d'un point de vue cryptographique. Les mécanismes d'authentification de type épreuve/réponse, tels que décrits dans le § 10, prévoient des échantillons de textes en clair et de textes chiffrés correspondants, et sont par conséquent exposés à des attaques de type "force brute" de la part d'un observateur de la transaction en question lorsque les authentifications sont effectuées au moyen de simples mots de passe. L'observateur peut ainsi récupérer le mot de passe ou le numéro PIN et se faire ensuite passer pour le point d'extrémité afin d'obtenir un service.

Un ensemble de protocoles classés sous la rubrique générique de l'échange de clés chiffrées utilise un secret partagé pour "occulter" un échange de clés Diffie-Hellman de telle façon que l'attaquant doit résoudre une série de problèmes de logarithme discret pour valider une attaque de type force brute par rapport au secret partagé. Selon le protocole d'échange de clés chiffrées (EKE, *encrypted key exchange*) de Bellovin et Merritt [B&M], le secret partagé est utilisé pour chiffrer les clés publiques Diffie-Hellman conformément à un algorithme symétrique. Selon la méthode d'échange de clés par élévation à la puissance à partir d'un mot de passe simple (SPEKE, *simple password exponential key exchange*) de Jablon [Jab], le secret partagé est utilisé pour choisir un générateur différent du groupe Diffie-Hellman. Ces protocoles combinent la sécurité d'un échange de clés Diffie-Hellman fort avec l'utilisation du secret partagé de telle manière qu'un attaquant ne puisse pas obtenir un texte en clair connu en vue d'une utilisation dans le cadre d'une attaque de type force brute à l'encontre du secret sans résoudre le problème du logarithme discret Diffie-Hellman. Un des avantages de ces protocoles tient à ce qu'ils multiplient les forces du problème Diffie-Hellman par la force du chiffrement de clé secrète (ou inversement). Un des éventuels inconvénients est qu'ils font généralement l'objet d'une protection par brevet.

Recommandation UIT-T H.235.5

Cadre de sécurité H.323: cadre de l'authentification sécurisée pendant l'échange de messages RAS au moyen de secrets partagés faibles

1 Domaine d'application

La présente Recommandation peut être utilisée pour tout portier ou point d'extrémité utilisant le protocole RAS H.225.0.

2 Références

2.1 Références normatives

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

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

– Recommandation UIT-T H.235.0 (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.245 (2005), *Protocole de commande pour communications multimédias*.

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

– Federal Information Processing Standard FIPS PUB 180-2, *Secure Hash Standard*, U.S. Department of Commerce, Technology Administration, National Institute of Standards and Technology, 1er Août 2002.

– NIST Special Publication 800-38A 2001, *Recommendation for Block Cipher Modes of Operation – Methods and Techniques*. http://www.csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf.

2.2 Références informatives

[AES] IETF RFC 3268 (2002), *Advanced Encryption Standard (AES) Ciphersuites for Transport Layer Security*.

[B&M] BELLOVIN (S.), MERRITT (M.): U.S. Patent 5,241,599, August 31, 1993, originally assigned to AT&T Bell Laboratories, now assigned to Lucent Technologies.

[Jab] JABLON (D.): Strong Password-Only Authenticated Key Exchange, *Computer Communication Review*, ACM SIGCOMM, Vol. 26, No. 5, pp. 5-26, October 1996.

[NIST SP 800-57] NIST Draft Special Publication 800-57 (2005), *Recommendation for Key Management, Part 1: General Guideline*. http://www.csrc.nist.gov/publications/drafts/draft-800-57-Part1-April2005.pdf

[RFC2104] IETF RFC 2104 (1997), *HMAC: Keyed-Hashing for Message Authentication*.

[RFC2412] IETF RFC 2412 (1998), *The OAKLEY Key Determination Protocol*.

[RFC2246] IETF RFC 2246 (1999), *The TLS Protocol Version 1.0*.

[RFC3546] IETF RFC 3546 (2003), *Transport Layer Security (TLS) Extensions*.

3 Termes et définitions

Aucun.

4 Abréviations

La présente Recommandation utilise les abréviations suivantes:

ACF confirmation d'admission (*admission confirm*)

AES norme de chiffrement perfectionnée (*advanced encryption standard*)

ARJ rejet d'admission (*admission reject*)

ARQ demande d'admission (*admission request*)

CBC chiffrement par chaînage de blocs (*cipher block chaining*)

CTR mode compteur (*counter mode*) (voir NIST SP 800-38A)

DH Diffie-Hellman

EKE échange de clés chiffrées (*encrypted key exchange*)

GCF confirmation de portier (*gatekeeper confirm*)

GK portier (*gatekeeper*)

GRJ rejet de portier (*gatekeeper reject*)

GRQ demande de portier (*gatekeeper request*)

HMAC code d'authentification de message "d'après les signaux parasites" (*hashed message authentication code*)

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

ID identificateur

LCF confirmation d'emplacement (*location confirm*)

LRJ rejet d'emplacement (*location reject*)

LRQ demande de localisation (*location request*)

MIM intercepteur (*man-in-the-middle*)

OID identificateur d'objet (*object identifier*)

PIN numéro d'identification personnel (*personal identification number*)

PRF fonction pseudo-aléatoire (*pseudo-random function*)

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

RCF confirmation d'enregistrement (*registration confirm*)

RFC demande de commentaires (*request for comments*)

RRJ rejet d'enregistrement (*registration reject*)

RRQ demande d'enregistrement (*registration request*)

SHA1 algorithme de hachage sécurisé n° 1 (*secure hash algorithm 1*)

SPEKE échange de clés par élévation à la puissance à partir d'un mot de passe simple (*simple password exponential key exchange*)

TLS sécurité de la couche Transport (*transport layer security*)

UDP protocole datagramme d'utilisateur (*user datagram protocol*)

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.

Pour les autres conventions, on se reportera au § 5/H.235.0.

6 Cadre de base

6.1 Capacités de négociation améliorées dans la Rec. UIT-T H.235.0

La Rec. UIT-T H.235.0 permet de prendre en charge le présent cadre de sécurité grâce à l'inclusion de l'élément générique suivant dans le champ **ClearToken**:

- **profileInfo** est une séquence d'éléments propres à un profil donné, chaque élément étant identifié par sa propre valeur entière telle que définie par le profil dont l'identificateur d'objet (OID) est acheminé dans l'élément **ClearToken.tokenOID**.

Dans les descriptions qui suivent, plusieurs éléments sont acheminés dans la séquence **profileInfo**. Pour faciliter la discussion, on donnera à chacun de ces éléments un nom plutôt qu'une valeur identifiante.

6.2 Utilisation entre le point d'extrémité et le portier

Le cadre de base, dans lequel le demandeur est un point d'extrémité souhaitant s'enregistrer auprès d'un portier, et dans lequel le répondant est ce portier, est simple. Dans ce qui suit, on part implicitement du principe que chaque champ **ClearToken** mentionné est identifié au moyen de l'élément **tokenOID** du profil d'authentification. Le champ **ClearToken** est censé être étendu. Les éléments **random** et/ou **random2** peuvent être utilisés par un profil de deux façons différentes: ils peuvent être inclus dans le calcul de la clé d'authentification et/ou dans un champ **ClearToken** de profil dans chaque message RAS ultérieur (par exemple, RRQ/RCF) afin d'éviter les réexécutions. L'échange pour l'enregistrement d'un point d'extrémité se fait de la façon suivante:

1) le point d'extrémité annonce son intention de participer à un ou plusieurs mécanismes d'authentification ou de négociation de clé en incluant le ou les identificateurs d'objet appropriés pour le ou les profils souhaités dans des éléments **authenticationMechanism.keyExch** de l'élément **authenticationCapability** du message **G**atekeeper**ReQ**uest. On suppose que chaque identificateur OID spécifique définit entièrement une procédure d'authentification en termes de système à clé publique (par

exemple, Diffie-Hellman ou courbe elliptique) et de groupe concerné (par exemple, un des groupes OAKLEY décrits dans la norme RFC 2412), d'algorithme de chiffrement symétrique (par exemple, AES-128-CBC avec extraction cryptographique), de fonction de calcul de clé (par exemple, au moyen de la fonction pseudo-aléatoire décrite au § 10/H.235.0), et de code d'authentification de message (par exemple, HMAC-SHA1-96 [RCF2104]) ainsi que la séquence dans laquelle ils sont utilisés. Le point d'extrémité inclut aussi un ou plusieurs champs **ClearToken** de profil dans le message GRQ, chacun acheminant l'identificateur OID pour le profil concerné, ainsi que les données de clé publique (chiffrées) nécessaires de la façon suivante:

a) **tokenOID** achemine l'identificateur OID du profil, tel qu'il figure dans l'élément **authenticationCapability** du message GRQ encapsulant;

b) **timeStamp** peut être utilisé pour garantir l'actualité et éviter les réexécutions;

c) **password** ne doit pas être utilisé pour le mot de passe réel;

d) **dhkey** achemine les paramètres de clé Diffie-Hellman, s'il est utilisé. L'élément **halfkey** inclus est chiffré comme spécifié par le profil choisi;

e) **challenge** n'est pas requis;

f) **random** est fourni par le demandeur et est utilisé pour éviter les attaques par réexécution;

g) **certificate** peut être utilisé si l'échange de certificats fait partie du profil;

h) **generalID** peut être utilisé s'il est requis par le profil;

i) **eckasdhkey** achemine les paramètres de clé de type courbe elliptique, s'il est utilisé dans le profil. L'élément **public-key** inclus devrait être chiffré comme spécifié par le profil;

j) **sendersID** peut être utilisé comme spécifié par le profil;

k) un élément **profileInfo**, **initVect**, peut être fourni avec les données (chiffrées) de clé publique (**dhkey** ou **eckasdhkey**) si le profil exige un vecteur d'initialisation pour le déchiffrement;

l) s'il souhaite utiliser des données de clé obtenues d'un échange antérieur, le demandeur doit inclure un élément **profileInfo**, appelé **sessionID**, contenant l'identificateur attribué au cours de l'échange antérieur. Dans ce cas, **dhkey**, **eckasdhkey** et/ou **initVect** ne devraient pas être inclus;

m) s'il souhaite établir une session TLS pour une connexion de signalisation d'appel, le demandeur peut inclure un ou plusieurs éléments **profileInfo** contenant des suites cryptographiques TLS. Le message ne doit contenir qu'une suite cryptographique (celle négociée préalablement) si l'élément **sessionID** est présent;

n) s'il souhaite établir une session TLS pour la signalisation d'appel, le demandeur peut inclure un élément **profileInfo** contenant une liste de méthodes de compression; une seule méthode de compression (celle négociée préalablement) doit être incluse si l'élément **sessionID** est présent;

o) d'autres éléments **profileInfo** peuvent être utilisés pour tout paramètre additionnel requis pour les procédures relevant du profil.

2) Lorsqu'il reçoit le message GRQ, le portier choisit un profil **AuthenticationMechanism** parmi la liste proposée, génère une clé privée appropriée, calcule la clé publique correspondante, génère si nécessaire un vecteur d'initialisation pour le chiffrement symétrique au moyen du mot de passe, chiffre la clé publique, génère un identificateur de session unique, et génère enfin une quantité aléatoire, tous ces éléments étant codés dans un champ **ClearToken**. En fonction du profil, l'utilisation des éléments de ClearToken est la suivante:

a) **tokenOID** achemine l'identificateur OID du profil, tel qu'il a été choisi à partir de l'élément **authenticationMethod** du message GCF encapsulant;

b) **timeStamp** peut être utilisé pour garantir l'actualité et éviter les réexécutions;

c) **password** ne doit pas être utilisé pour le mot de passe réel;

d) **dhkey** achemine les paramètres de clé Diffie-Hellman, s'il est utilisé. L'élément **halfkey** inclus est chiffré comme spécifié par le profil choisi;

e) **challenge** est utilisé pour acheminer un vecteur d'initialisation, s'il est requis pour le chiffrement de clé comme spécifié par le profil, ou il peut être utilisé pour acheminer une chaîne aléatoire que le point d'extrémité doit renvoyer pour éviter les attaques par réexécution;

f) **random** peut contenir la valeur unique imprévisible fournie par le demandeur pour éviter les attaques par réexécution;

g) **certificate** peut être utilisé si l'échange de certificats fait partie du profil;

h) **generalID** peut être utilisé s'il est requis par le profil;

i) **eckasdhkey** achemine les paramètres de clé de type courbe elliptique, s'il est utilisé par le profil. L'élément **public-key** inclus devrait être chiffré comme spécifié par le profil;

j) **sendersID** peut être utilisé comme spécifié par le profil;

k) **random** (ou un élément **profileInfo** additionnel, appelé **random2**, si le profil exige que les deux nombres aléatoires restent lors de l'échange de message) devrait contenir une valeur unique imprévisible fournie par le répondant pour éviter les attaques par réexécution;

l) **initVect** est fourni avec les données de clé publique (chiffrées) (**dhkey** ou **eckasdhkey**) si le profil exige un vecteur de signalisation pour le déchiffrement;

m) **sessionID** est un identificateur unique (pour le portier) servant à identifier cette session d'enregistrement. Dans certains profils, il peut aussi servir d'identificateur de session TLS pour l'établissement rapide d'un canal de signalisation d'appel protégé par TLS;

n) **profileInfo** peut être utilisé pour tout paramètre additionnel requis pour les procédures relevant du profil.

Le portier calcule ensuite le secret partagé ou la clé maîtresse au moyen de sa clé privée et de la clé publique (déchiffrée) à partir du message GCF, et déduit à partir de la clé maîtresse les clés de chiffrement, les clés d'authentification ou les autres données nécessaires, conformément au profil. Le champ **ClearToken** décrit ci-dessus est placé dans le message **G**atekeeper**C**on**F**irm. On doit contrôler l'intégrité du message GCF et l'authentifier au moyen de la clé d'authentification calculée, puis l'envoyer au point d'extrémité. Le résultat de l'authentification/du contrôle d'intégrité peut être renvoyé de différentes façons, comme spécifié par le profil: au moyen d'un élément **profileInfo** propre au profil, ou au moyen d'une des procédures spécifiées dans la Rec. UIT-T H.235.1.

3) Le point d'extrémité examine l'élément **authenticationMechanism.keyExch** sélectionné à partir du message GCF et extrait les paramètres du champ **ClearToken** identifié par l'identificateur **tokenOID** correspondant. Le point d'extrémité choisit ensuite sa clé privée, calcule la clé publique correspondante et choisit tout autre paramètre requis par le profil. Il calcule ensuite le secret partagé ou la clé maîtresse au moyen de sa clé privée et de la clé publique (déchiffrée) à partir du message GCF, et en déduit les clés de chiffrement, les clés d'authentification ou les autres données nécessaires, conformément au profil. Le point d'extrémité doit ensuite vérifier l'intégrité du message GCF. Si le résultat de cette vérification est incorrect, le point d'extrémité doit éliminer le message GCF ainsi que toutes les données de clé qui en résultent, et continuer à attendre un message GRQ valable. La

reprise RAS standard entraînera la retransmission du message GRQ et, probablement, la réception d'un message GCF intact. Si plusieurs retransmissions ne permettent pas d'obtenir une réponse correcte, le point d'extrémité devrait renoncer à s'enregistrer et informer son utilisateur du problème. Il est à noter que chaque message GRQ envoyé donne à un usurpateur de passerelle une chance supplémentaire d'essayer de deviner le mot de passe de l'utilisateur et de voir son hypothèse validée par l'acceptation du message GRQ. Si le résultat du contrôle d'intégrité du message GCF est correct, le point d'extrémité a validé le portier et peut s'enregistrer et, au cours du processus, s'authentifier auprès du portier.

4) Le point d'extrémité remplit ensuite un champ **ClearToken** avec l'identificateur **tokenOID** de profil selon une méthode analogue à celle appliquée par le portier, telle que décrite ci-dessus. Tout champ du jeton en clair du message GCF, qui est considéré comme une épreuve par le profil devrait figurer dans le champ **ClearToken**. Le champ **ClearToken** doit comprendre les éléments **random** et **random2** du message GCF reçu (voir ci-dessus) si ces éléments sont spécifiés par le profil dans le but d'éviter les réexécutions. Le champ **ClearToken** est ensuite placé dans un message **R**egistration **ReQ**uest à renvoyer au portier. Le point d'extrémité devrait ensuite authentifier le message RRQ entier et l'envoyer au portier. A partir de là, le point d'extrémité ne devrait ni accepter ni envoyer de messages RAS qui ne soient pas authentifiés par le profil convenu au moyen de la clé d'authentification calculée à partir des données de clé partagée.

5) Le portier reçoit le message RRQ et doit utiliser les données de clé partagée pour vérifier l'intégrité du message RRQ par rapport à l'élément inclus d'authentification et de contrôle de l'intégrité. Si le résultat du contrôle d'intégrité est incorrect, le portier doit ignorer le message RRQ reçu et attendre un message RRQ valable. Si aucun message de ce type n'arrive, le point d'extrémité abandonnera finalement toute tentative d'enregistrement et repartira à la recherche d'un portier. Si le contrôle d'intégrité est positif, le portier préparera un message de confirmation d'enregistrement à renvoyer au point d'extrémité. Selon le profil utilisé, ce message RCF peut contenir un champ **ClearToken** qui inclut les éléments **random**, **random2**, et/ou **challenge** du champ **ClearToken** de profil d'authentification fourni dans le message RRQ. Le message RCF ainsi que tous les messages RAS ultérieurs doivent contenir un élément valable d'authentification et de contrôle de l'intégrité calculé au moyen de la clé et de l'algorithme d'authentification négociés.

6) Lorsqu'il reçoit le message RCF, le point d'extrémité vérifie son intégrité au moyen de l'élément d'authentification et de contrôle de l'intégrité, qui est inclus. Si le résultat de cette vérification est incorrect, le message RCF doit être éliminé; si aucun message RCF valable n'est reçu, même après retransmission du message RRQ, la session doit être abandonnée et le point d'extrémité doit repartir à la recherche d'un nouveau portier. Si le résultat du contrôle d'intégrité du message RCF est correct, l'identificateur de session et la suite cryptographique choisie peuvent, s'ils sont présents, être extraits du champ **ClearToken** du message en vue d'une utilisation ultérieure lors de l'établissement d'un canal de signalisation d'appel sécurisé.

6.3 Utilisation de profils entre portiers

Pratiquement la même procédure peut être utilisée entre portiers dans un échange de messages LRQ/LCF. Dans ce cas, aucune sélection explicite de profil n'est possible; le portier d'origine doit proposer un ou plusieurs profils en incluant le ou les champs **ClearToken** appropriés tels que décrits ci-dessus pour le message GRQ. Le portier répondant peut choisir un profil proposé et devrait renvoyer le champ **ClearToken** correspondant tel que décrit ci-dessus pour le message GCF. Il est à noter que dans ce cas, le portier appelant ne s'authentifie pas auprès du portier répondant tant qu'il n'établit pas un canal de signalisation d'appel vers ce portier.

Cette procédure peut être employée dans un mode multidiffusion si un groupe de portiers partage un seul secret à utiliser à cette fin. Le message LRQ multidiffusé sera fondé sur ce secret; les portiers

qui répondent au moyen d'un message LCF utiliseront cette clé pour décoder la clé publique Diffie-Hellman offerte et choisiront chacun leur propre **nonce** et clé privée Diffie-Hellman pour leur réponse. Les clés de session résultantes seront propres à la paire finale de portiers.

6.4 Chiffrement et authentification des canaux de signalisation

Si le routage par portier est pris en charge par le portier, les données de clé maîtresse nouvellement négociées ainsi que les paramètres cryptographiques identifiés peuvent être utilisés pour authentifier et sécuriser le canal de signalisation d'appel (par exemple, en établissant une session TLS pour la signalisation d'appel). Si une session TLS doit être utilisée, le portier doit inclure dans le champ **ClearToken** de profil renvoyé les éléments **cipherSuite** et **compress** sélectionnés.

7 Profil de sécurité spécifique (SP1)

Le présent paragraphe décrit un profil de sécurité standard qui devrait offrir un secret partagé correspondant à un nombre aléatoire de 80 bits (voir [NIST SP 800-57]). Ce profil a la forme suivante:

- l'identificateur d'objet pour ce profil (appelé "SP1") sera {itu-t (0) recommendation (0) h (8) 235 version (0) 3 60};
- négociation de clé maîtresse, K_m: échange de clés Diffie-Hellman au moyen du groupe 2 bien établi OAKLEY [RFC 2412], suivi de la réduction de hachage SHA1 [FIPS PUB 180-1] du secret Diffie-Hellman: K_m = SHA1 (secret partagé Diffie-Hellman);
- algorithme de chiffrement symétrique: doit être l'algorithme AES-128 en mode de compteur segmenté avec un discriminateur de participant de 2 octets, D, un vecteur d'initialisation de 12 octets, IV, et un champ de compteur de 2 octets, C, tel que ce compteur soit égal à D || IV || C, et C = 0 initialement. Voir [NIST800-38A] pour une description du mode CTR. Le discriminateur de participant, D, est mis à 0x3636 lorsque le vecteur IV est généré par le participant qui a envoyé le message GRQ/RRQ ou LRQ, et est mis à 0x5c5c lorsque le vecteur IV est généré par le participant qui a répondu au moyen d'un message GCF/RCF ou LCF. Chaque participant doit s'assurer que chaque vecteur IV qu'il génère est unique; pour cela, il peut utiliser sa propre méthode;
- chiffrement de clé Diffie-Hellman: doit utiliser le mode de compteur segmenté AES-128 pour chiffrer la clé publique Diffie-Hellman (représentée par une chaîne d'octets selon l'ordre des octets dans le réseau); le vecteur d'initialisation doit être acheminé dans le champ **ClearToken.initVect** et la clé de 16 octets, K_p, doit être construite sous la forme des 128 bits de plus fort poids de la valeur de hachage SHA1 du mot de passe de l'utilisateur: K_p = Trunc(SHA1(mot de passe de l'utilisateur), 16), où Trunc(x,y) tronque la chaîne d'octets x à y octets. Il est à noter que cette clé est généralement considérée comme étant une clé faible;
- prévention des réexécutions: chaque participant doit indiquer un nombre "aléatoire" de 32 bits (qui peut contenir un champ de compteur garantissant l'unicité); les nombres aléatoires sont utilisés explicitement pour le calcul des clés; par conséquent, ils n'ont besoin d'être transmis qu'une seule fois chacun;
- calcul de la clé d'authentification, K_a: au moyen de la fonction pseudo-aléatoire (PRF, *pseudo-random function*), décrite au § 10/H.235.0, dénommée PRF (*in_key*, *label*, *outkey_len*) avec *in_key* = K_m, et *label* = "auth_key" || R_e || R_g, où R_e est un **nonce** obtenu à partir d'un élément **ProfileElement** du message GRQ, R_g est un **nonce** obtenu à partir d'un élément **ProfileElement** du message GCF et *outkey_len* = 128;
- fonction d'authentification et d'intégrité de message: au moyen d'un champ **ClearToken** dont l'identificateur **tokenOID** est mis à "SP1" et un élément **ProfileElement.octets** mis à la valeur de hachage HMAC-SHA1-96 calculée sur la totalité du message comme décrit

dans la Rec. UIT-T H.225.0; cette procédure doit s'appliquer à tous les messages RAS et de signalisation d'appel (sauf le message GRQ ou LRQ, qui ne contient pas d'identificateur **sessionID**);

- clé de chiffrement d'élément, K_e: certains éléments de messages de signalisation d'appel (ou éléments tunnélisés dans ceux-ci) peuvent être chiffrés conformément à l'algorithme AES-128 en mode de compteur segmenté au moyen de la clé K_e = PRF(K_m, "encrypt_key" || R_e || R_g, 128). Par exemple, cette clé peut servir à chiffrer des clés de session de média en vue de leur distribution dans des éléments **h235Key** tels qu'utilisés dans les procédures de connexion rapide et/ou H.245. Dans ce cas, "SP1" est utilisé comme identificateur OID d'algorithme de chiffrement.

Ce profil utilise les éléments **ProfileElement** définis dans le Tableau 1. Ces éléments sont acheminés dans la séquence d'éléments **ClearToken.profileInfo** telle que définie dans la Rec. UIT-T H.235.0.

Tableau 1/H.235.5 – Eléments de profil

Nom de l'élément (utilisé dans le texte)	Valeur de l'identificateur ElementID	Choix de l'élément (longueur)	Description de l'élément
initVect	1	Octets (12)	Vecteur d'initialisation pour chiffrement EKE
nonce	2	Octets (nombre quelconque)	Valeur unique et imprévisible
cipherSuite	3	Octets (2)	Suite cryptographique TLS
compression	4	Octets (1)	Algorithme de compression TLS
sessionID	5	Octets (1..)	Elément unique pouvant correspondre à un identificateur de session TLS
integrityCheck	6	Octets (12)	Valeur de contrôle calculée au moyen d'une clé

La séquence d'enregistrement doit avoir la forme suivante:

– le point d'extrémité doit envoyer le message GRQ avec l'élément **authenticationCapability** qui comporte un élément **AuthenticationMechanism.keyExch** contenant l'identificateur OID "SP1" et un champ **ClearToken** correspondant avec **tokenID** = "SP1", **dhkey** contenant une clé publique de 1024 bits chiffrée à partir du champ **initVect** en tant que vecteur IV et de la clé calculée à partir du mot de passe de l'utilisateur, et **nonce** = nombre aléatoire de 32 bits choisi par le point d'extrémité;

– le portier doit répondre au moyen d'un message GCF avec l'élément **authenticationMode** égal à un élément **AuthenticationMechanism.keyExch** contenant l'identificateur OID "SP1" et un champ **ClearToken** avec **tokenID** = "SP1", **dhkey** contenant une clé publique non chiffrée de 1024 bits, **nonce** = nombre aléatoire de 32 bits choisi par le portier et **integrityCheck** contenant la valeur de hachage d'authentification calculée au moyen de la clé d'authentification calculée, K_a. Il est à noter qu'il n'est pas nécessaire pour le portier de chiffrer sa demi-clé Diffie-Hellman dans le message GCF dans ce profil car il s'agit du premier participant à s'authentifier en montrant sa capacité à authentifier le message GCF au moyen de la clé d'authentification calculée. Ce mode permet au portier de réutiliser ses clés Diffie-Hellman avec plusieurs points d'extrémité. Voir le § 10.5;

– le point d'extrémité doit répondre au moyen d'un message RRQ contenant la valeur d'authentification et de contrôle d'intégrité dans un élément **ProfileElement** avec le champ **elementID** mis à **integrityCheck** et le champ **element** mis à la valeur calculée à partir de la clé d'authentification déduite, K_a;

– on doit authentifier et contrôler l'intégrité des messages RAS ultérieurs, y compris le message RCF, au moyen de la même procédure et de la même clé. Les messages de signalisation d'appel H.225.0 (et les messages H.245 tunnélisés, s'ils sont présents) doivent être authentifiés au moyen d'un champ **ClearToken** avec le champ **tokenOID** mis à "SP1" et contenant un élément **ProfileElement profileInfo** avec le champ **elementID** mis à **integrityCheck** et le champ **element** mis à la valeur calculée.

– la clé de chiffrement, K_e, ainsi que l'algorithme de chiffrement AES-128 en mode de compteur segmenté, peuvent être utilisés par le portier et par le point d'extrémité pour chiffrer certaines informations qui sont transportées selon le protocole RAS, le protocole de signalisation d'appel et/ou le protocole H.245. Par exemple, le portier peut distribuer des clés de chiffrement de média protégées par la clé K_e et l'algorithme de chiffrement de profil;

– s'il doit se réenregistrer tout en gardant l'identificateur de session et le secret maître originaux, le point d'extrémité devrait tenter de le faire en incluant explicitement l'identificateur de session dans le message GRQ (et en n'incluant pas de demi-clé Diffie-Hellman);

– ce profil doit pouvoir être utilisé entre les portiers (voir le § 6.3).

8 Profil de sécurité amélioré (SP2)

Le présent paragraphe définit un nouveau profil de sécurité fondé sur le profil d'origine, SP1. Il est désigné de façon informelle par SP2 et de façon formelle par l'identificateur OID {itu-t (0) recommendation (0) h (8) 235 version (0) 4 62}. Ce profil est identique au profil SP1, aux exceptions près spécifiées dans les paragraphes qui suivent. Les améliorations spécifiques par rapport au profil SP1 sont les suivantes:

– améliorations concernant l'attribution de numéros de séquence aux messages de signalisation d'appel pour contrer les attaques par réexécution;

– salage de la clé de chiffrement générée à partir du mot de passe au moyen du pseudonyme du point d'extrémité pour contrer les attaques de type dictionnaire;

– la taille de nonce est augmentée et peut désormais varier;

– une clé de salage est calculée pour être utilisée avec le vecteur d'initialisation du chiffrement;

– un transport plus efficace du **ClearToken** de profil est assuré au moyen de **genericData**.

Le profil SP2 utilise les éléments de profil du Tableau 1 ainsi que les éléments de profils additionnels décrits dans le Tableau 2:

Tableau 2/H.235.5 – Eléments de profil additionnels pour le profil SP2

Nom de l'élément (utilisé dans le texte)	Valeur de l'identificateur ElementID	Choix de l'élément (longueur)	Description de l'élément
seqNumber	7	Octets (4)	Numéro de séquence à 32 bits selon l'ordre des octets dans le réseau
connectID	8	Octets (2)	Identificateur de connexion de signalisation. (facultatif, valeur par défaut = 0)
endpointID	9	Octets (nombre variable)	AliasAddress codé en ASN.1 associé au point d'extrémité et à son mot de passe. (facultatif)

8.1 Numéro de séquence des messages de signalisation d'appel

Les messages de signalisation d'appel H.225.0 ne contiennent pas de numéro de séquence car ils sont transportés sur une connexion fiable (TCP) qui est chargée de la mise en séquence. Néanmoins, l'absence d'identificateur de message unique au niveau application expose la signalisation d'appel à des attaques par réexécution ou par réflexion. On peut surmonter ce problème en ajoutant un numéro de séquence et, éventuellement, un identificateur de connexion à chaque message de signalisation d'appel. Il est à noter que cette technique n'empêche pas complètement les attaques par réexécution ou par réflexion, mais elle réduit considérablement les chances de succès de l'attaquant.

Les numéros de séquence doivent être uniques dans chaque sens pour éviter les réflexions. Pour cela, on peut imposer, dans certaines limites, que l'émetteur du message GRQ (point d'extrémité) ou LRQ (portier) parte de 0 (zéro) pour le numéro de séquence des messages de signalisation d'appel émis et à 2^{31} pour le numéro de séquence des messages reçus, le portier destinataire ayant un comportement correspondant. Ainsi, le temps nécessaire à la survenue éventuelle d'un chevauchement est très long (presque 600 heures au débit peu courant de un message par milliseconde.) Pour des appels successifs utilisant le même identificateur SessionID, il convient d'utiliser le numéro de séquence inutilisé suivant dans chaque sens. (Afin de tenir compte des messages qui seraient éventuellement perdus sur des connexions défaillantes, le récepteur devrait accepter les messages se trouvant dans une petite fenêtre (par exemple 5 à 10) après le dernier numéro de séquence reçu et continuer à partir de là.) Les dispositifs qui prennent en charge plusieurs connexions de signalisation d'appel simultanées avec le même identificateur de session, peuvent utiliser un identificateur **connectID** facultatif pour déterminer des espaces de numéro de séquence distincts pour les différents appels. S'il n'est pas spécifié, l'identificateur **connectID** est supposé être égal à 0 (zéro).

8.2 Génération d'une clé de chiffrement faible à partir du mot de passe

Pour éviter les attaques de type dictionnaire dans lesquelles un numéro PIN est deviné, utilisé pour chiffrer une clé publique D-H puis appliqué successivement à tous les pseudonymes de point d'extrémité connus, il est souhaitable de "saler" la clé de chiffrement avec le pseudonyme proprement dit. En particulier, la clé fondée sur le mot de passe, K_p, doit être calculée à partir de la concaténation du mot de passe et de l'identificateur **endpointID** fourni:

K_p = Trunc(SHA1(mot de passe de l'utilisateur || **endpointID**), 16)

Le champ **AliasAddress** de l'identificateur **endpointID** contiendra généralement l'un des pseudonymes inclus dans l'élément **endpointAlias** du message GRQ, mais ce n'est pas nécessaire. Par exemple, l'identificateur **endpointID** peut identifier une passerelle prenant en charge de nombreux points d'extrémité dont les pseudonymes sont énumérés dans **endpointType**.

8.3 Taille de nonce

Selon le profil de sécurité 1, chaque participant est tenu de fournir un nonce à 4 octets (32 bits) dans le cadre du protocole de négociation de clé. Lorsque le nonce est fourni pendant la négociation de clé initiale, une taille de 32 bits est peut-être suffisante pour garantir que le nonce est tout nouveau dans les cas où le portier qui répond réutilise la même clé publique Diffie-Hellman mais le demandeur génère une nouvelle clé. Toutefois, lorsque de nouvelles clés de session sont négociées à partir d'une clé maîtresse négociée précédemment, une longueur totale de 64 bits ne permettra peut-être pas d'établir une différence suffisante entre chaque ensemble de clés calculées. Il est proposé que la taille de nonce soit variable, comprise entre 4 octets au minimum et 16 octets au maximum.

8.4 Salage du vecteur d'initialisation

Comme mesure supplémentaire d'obscurcissement, une clé de salage de session à 112 bits, K_s, est calculée comme suit à partir de la clé maîtresse négociée:

$K_s = PRF(K_m, \text{"satting\_key"} \| R_e \| R_g, 112)$

Le compteur AES-128-CM initial pour le chiffrement et le déchiffrement est alors construit comme suit:

Compteur = (K_s ^ (D || IV)) || C, où C est le champ du compteur à 16 bits, initialement à zéro.

8.5 Codage du champ ClearToken

Le profil de sécurité 1 utilise une séquence de champs **clearToken** pour acheminer les paramètres du profil. Chaque message H.225.0 contient une séquence de champs **ClearToken**, à l'exception du choix **empty** de **h323-message-body**; tous les messages acheminent un élément **genericData**. Les procédures du profil SP1 permettent d'avoir une structure relativement régulière du champ **ClearToken**, qui peut ainsi être codé par avance et acheminé en tant que paramètre **raw** avec **id.standard** mis à 1 dans un élément **GenericData** identifié par l'identificateur OID SP2. Cette forme permet d'identifier le champ **clearToken** au moyen de l'identificateur OID "néant" {0,0} et surtout, elle facilite la localisation du jeton ainsi que de la valeur de contrôle qu'il contient, car la forme codée du champ **ClearToken** seul devient disponible dans le cadre du processus normal de codage et de décodage. Ainsi, il est plus rapide de localiser l'élément integrityCheck dans le champ ClearToken codé que dans la totalité du message codé.

9 Extensions du cadre (à titre indicatif)

Les éléments qui suivent peuvent être incorporés dans un profil de sécurité défini dans le présent cadre.

9.1 Utilisation de la clé maîtresse pour sécuriser le canal de signalisation d'appel via le protocole TLS

Les données de clé, qui sont négociées au cours de l'échange RAS, peuvent aussi servir à calculer des clés de session afin de protéger le canal de signalisation d'appel via le protocole de transport TLS ([RFC 2246], [RFC 3546]). En effet, la négociation RAS remplace le protocole initial de prise de contact TLS. Cela n'est évidemment valable que si la signalisation d'appel est routée par le portier. Cela est particulièrement utile pour l'authentification et la signalisation entre portiers au moyen de l'échange de messages LRQ/LCF. Dans ce cas, il n'existe pas de troisième message RAS par lequel le portier appelant peut s'authentifier auprès du portier appelé à partir des données de clé négociées, mais l'appelant peut être implicitement authentifié par sa capacité à établir le canal de signalisation d'appel au moyen des paramètres corrects de session TLS. La Figure 1 illustre le flux d'informations en jeu: le protocole RAS est utilisé pour négocier la clé maîtresse de session;

l'identificateur de session et le secret maître-test correspondant sont distribués au logiciel TLS, l'identificateur de session étant utilisé par la couche de signalisation d'appel pour établir le canal de signalisation d'appel via le protocole TLS. Le moyen par lequel le transfert du secret est réalisé dépend de l'implémentation; il n'entre donc pas dans le cadre de la présente Recommandation. Il convient de noter que la présente Recommandation spécifie le port 1300 comme le port d'écoute TLS par défaut pour la signalisation d'appel. Le point d'extrémité doit cependant utiliser une des adresses de transport de signalisation d'appel fournies par le portier.

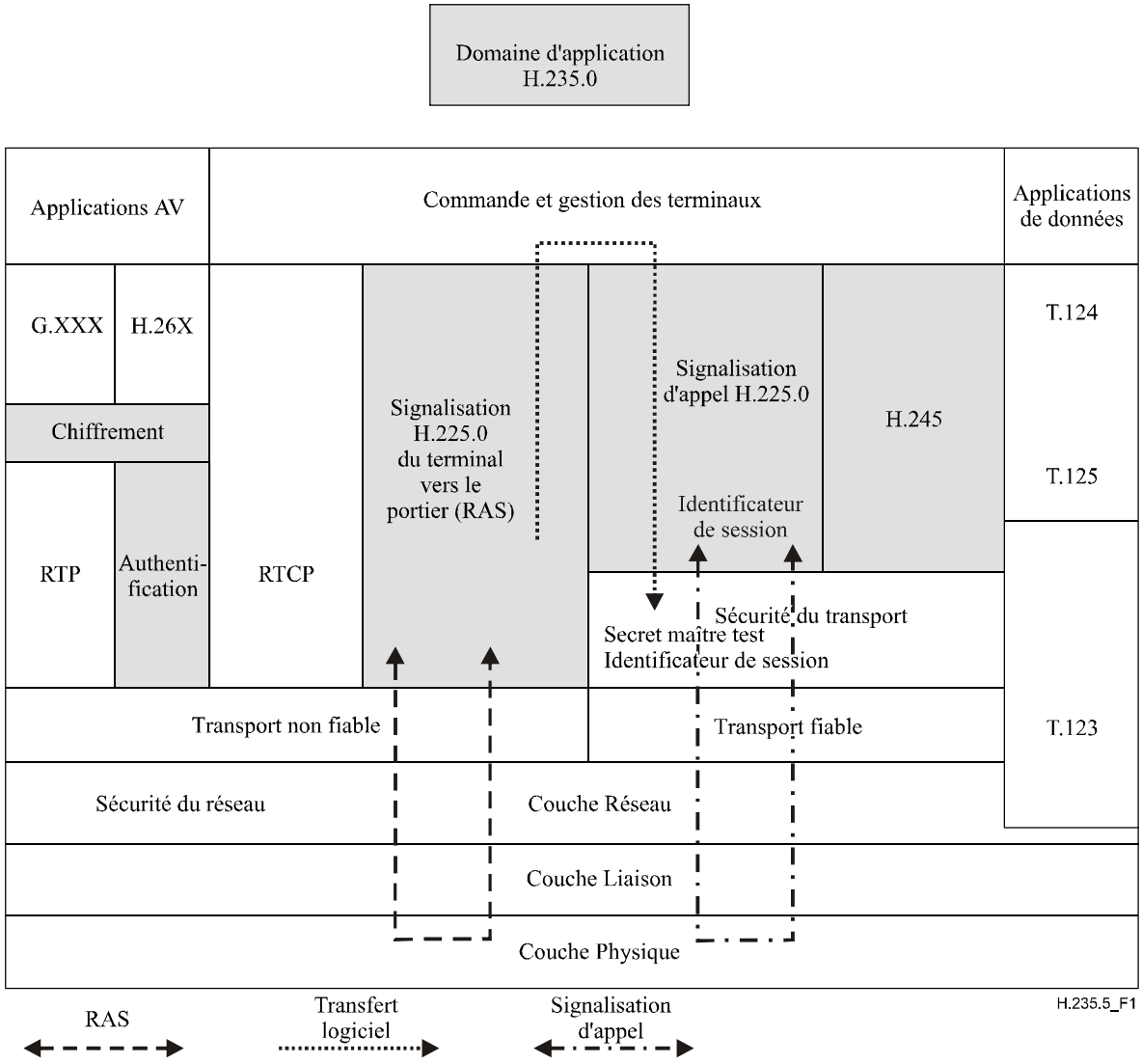

Figure 1/H.235.5 – Flux d'information pour le profil de sécurité et le protocole TLS

Le paragraphe qui suit décrit les différentes étapes du cadre de base de la Figure 1.

9.1.1 Enregistrement du point d'extrémité

Un point d'extrémité peut évaluer la capacité d'un portier à prendre en charge la signalisation d'appel protégée par le protocole TLS en incluant un ou plusieurs éléments **cipherSuite** et un ou plusieurs éléments **compression** dans le champ **ClearToken** de profil du message GRQ envoyé à l'étape 1 ci-dessus. S'il souhaite utiliser une session préalablement négociée, le point d'extrémité doit aussi inclure **sessionID** dans **ClearToken** (et ne doit indiquer que la suite cryptographique et la méthode de compression correspondant spécifiquement à la session requise). Si la négociation est fondée sur une session TLS existante, aucune donnée cryptographique n'est requise dans le champ **ClearToken** de profil à part **nonce**.

Si une session demandée n'existe pas, le portier doit choisir un autre profil d'authentification (si cela est possible) ou renvoyer un message GRJ avec **GatekeeperRejectReason.resourceUnavailable**. Si la session demandée existe, les données de clé maîtresse sont obtenues à partir de la session TLS, et utilisées (avec l'élément **random** du message GRQ et l'élément **random2** généré par le portier) pour calculer la clé d'authentification en vue de l'échange de messages RAS. L'identificateur **sessionID**, la suite cryptographique **cipherSuite**, la méthode **compress** et le **nonce** du portier doivent être renvoyés dans le champ **ClearToken** de profil d'un message GCF.

S'il peut prendre en charge la négociation de session TLS, le portier doit calculer les données de clé maîtresse comme spécifié dans le profil, attribuer un nouvel identificateur de session et le renvoyer dans le **ClearToken** de profil, dans l'identificateur **sessionID**. Le champ **ClearToken** de profil doit aussi contenir les paramètres de sécurité requis à l'étape 2 ci-dessus, ainsi qu'une seule suite cryptographique **cipherSuite** choisie, une seule méthode **compress** choisie et l'identificateur **sessionID** non égal à zéro. Il est à noter que la méthode d'échange de clé de la suite cryptographique choisie n'a pas d'importance. Si le portier convient d'une protection TLS pour la signalisation d'appel, toutes les adresses de transport de signalisation d'appel échangées dans les messages ultérieurs RRQ/RCF ou ARQ/ACF doivent être protégées par le protocole TLS.

Si la négociation TLS et/ou le routage par portier ne sont pas pris en charge par le portier, aucun paramètre TLS ne doit être renvoyé; cependant, les procédures d'authentification définies à l'étape 3 ci-dessus peuvent être maintenues. Le point d'extrémité doit déterminer s'il est prêt à procéder sans protection TLS de la signalisation d'appel; il peut choisir de procéder ainsi tout en continuant à utiliser le profil d'authentification. Une fois que la séquence d'enregistrement a été exécutée, la session TLS peut être utilisée pour effectuer un établissement rapide d'une ou de plusieurs connexions de signalisation d'appel en direction du portier, sans avoir à renégocier de données de clé au moyen de méthodes applicables aux clés publiques.

Les sessions TLS ont une durée de vie limitée. Par conséquent, il peut être nécessaire pour un point d'extrémité de renégocier les paramètres de session et d'obtenir un nouvel identificateur de session. Cela peut être réalisé en procédant à l'échange des éléments nécessaires de **ClearToken** décrits ci-dessus dans une séquence d'enregistrement simplifiée ("de maintien en vie", "keepalive"). Cette séquence ne doit pas avoir d'incidence sur la clé d'authentification RAS.

9.2 Utilisation de certificats pour l'authentification du portier

S'il est difficile d'échanger des chaînes de certificats valables en mode RAS (en raison de la taille limitée des paquets UDP), il est possible qu'un serveur s'authentifie auprès du point d'extrémité si ce dernier peut obtenir une copie fiable de la clé publique du serveur par d'autres moyens. Le serveur peut inclure simplement, dans le message GCF, un élément **CryptoH323Token.cryptoGKCert** dont l'identificateur **ClearToken.tokenOID** est mis à l'identificateur OID du profil de sécurité choisi.

9.3 Utilisation d'autres mécanismes de sécurité de signalisation

Les paramètres négociés dans le cadre d'un profil de sécurité défini dans la présente Recommandation peuvent être employés dans des mécanismes de sécurité au niveau transport et/ou application tels que déterminés par le profil donné. La séquence **profileInfo** qui a été ajoutée au champ **ClearToken** H.235 a été prévue, si nécessaire, pour cette utilisation.

10 Menaces (à titre indicatif)

10.1 Attaques passives

Le système décrit ci-dessus n'est actuellement pas vulnérable aux attaques passives, sous réserve que la négociation Diffie-Hellman ne soit pas elle-même vulnérable à ces attaques.

10.2 Attaques de type déni de service

Ce système est exposé aux attaques actives de type déni de service, dans lesquelles un tiers répond au message GRQ initial au moyen d'un message GRJ parasite. Ce type d'attaque peut ne pas être détecté: si le portier qui rejette la demande est légitime et connaît le secret partagé (par exemple, le portier est le portier du point d'extrémité et l'élément **rejectReason** a la valeur **resourceUnavailable**), le portier pourrait alors mener à bien la négociation de la clé et authentifier le message GRJ en renvoyant, dans ce message, les mêmes éléments que ceux décrits pour le message GCF (à ceci près que l'identificateur OID renvoyé dans l'élément **authenticationMode** du message GCF serait renvoyé dans un élément **ClearToken.profileInfo** du message GRJ). Cela dépend de la définition de chaque profil.

S'il n'est pas authentifié, le message GRJ pourrait provenir d'un attaquant. Avant de donner suite au message GRJ (par exemple, en cherchant un portier de remplacement), le point d'extrémité devrait attendre la réception éventuelle d'un autre message GRJ ou d'un message GCF authentifié provenant du portier correct. Sinon, il devrait essayer chaque portier proposé dans un quelconque élément **altGKInfo** reçu dans tous les messages GRJ (l'un d'eux est en principe légitime). Dans tous les cas, seul le portier correct (qui connaît le secret partagé) peut renvoyer un message GCF authentifié.

10.3 Attaques par intercepteur

Il est tentant d'envisager comme mode d'échange l'échange de clés Diffie-Hellman non chiffrées, avec utilisation du mot de passe ou du numéro PIN pour calculer des clés de session à partir du secret Diffie-Hellman. Cependant, ce mode d'échange est vulnérable aux attaques par intercepteur qui peuvent être utilisées pour trouver le "petit" secret partagé par la force brute au moyen de la valeur de contrôle d'intégrité fournie par le portier légitime dans le message GCF.

Tout intercepteur peut évidemment manipuler tout message RAS authentifié pour faire en sorte qu'il soit éliminé en raison d'un échec du contrôle d'intégrité. Si tous les messages peuvent être manipulés, cela peut conduire à un déni de service.

10.4 Prévoir les attaques

Un attaquant peut se faire passer soit pour un point d'extrémité légitime, soit pour un portier légitime, soit pour les deux (intercepteur) et tenter de deviner le secret partagé de façon empirique. Par exemple, l'attaquant (qui est censé connaître les données du profil d'authentification mais pas le secret partagé) peut essayer de deviner un secret partagé et tenter de s'enregistrer en envoyant un message GRQ à partir de cette hypothèse. En général, le portier répondra à cette tentative au moyen d'un message GCF contenant la clé publique du portier (chiffrée au moyen du véritable secret partagé) et une valeur ICV calculée au moyen de la clé déduite qui dépend de la manière dont le portier a déchiffré la clé publique chiffrée par l'attaquant. Ce dernier peut utiliser cette information pour vérifier son hypothèse du secret partagé. Si cette hypothèse confirme la valeur ICV du message GCF, il est probable qu'elle corresponde au véritable secret partagé; cela peut être confirmé par la séquence d'enregistrement. Si l'hypothèse ne peut pas être utilisée pour reproduire la valeur ICV du message GCF, l'attaquant doit émettre une autre hypothèse et faire une autre tentative. Avec un petit espace de clé pour le secret partagé, le nombre d'hypothèses pour une recherche "force brute" peut ne pas être prohibitif. Cette attaque nécessite la participation active du portier (ou du point d'extrémité si l'attaquant se fait passer pour le portier). La méthode traditionnelle utilisée pour contrecarrer une telle attaque consiste à limiter le nombre de tentatives infructueuses et, lorsqu'un seuil est atteint, à considérer toutes les tentatives ultérieures comme non valables (au moins pendant une période donnée) et à déclencher une alarme; cependant, ces procédures dépendent de l'implémentation.

10.5 Demi-clé non chiffrée par le portier

Comme il est mentionné plus haut, l'échange EKE peut rester sûr, sous certaines conditions, si le portier répondant ne chiffre pas sa demi-clé Diffie-Hellman. En particulier, le portier doit être le premier participant à prouver qu'il connaît le secret partagé (numéro PIN) au moyen de la valeur ICV. Si tel n'est pas le cas, le portier (ou un intrus se faisant passer pour lui) pourrait simplement essayer tous les numéros PIN possibles pour déchiffrer la demi-clé Diffie-Hellman du point d'extrémité, calculer le secret partagé Diffie-Hellman qui en résulte, calculer la clé d'authentification et la tester par rapport à la valeur ICV fournie par le point d'extrémité. Cela n'est pas possible si le point d'extrémité peut d'abord contrôler la valeur ICV fournie par le portier et refuser de poursuivre l'enregistrement si la valeur ICV n'est pas celle qui est attendue.

L'utilisation d'une demi-clé non chiffrée est un avantage pour le portier dans la mesure où celui-ci peut réutiliser sa clé privée correspondante avec plusieurs points d'extrémité. Cela serait impossible si la même clé était distribuée de façon chiffrée au moyen de plusieurs secrets partagés ou numéros PIN. Un tiers observateur pourrait collecter des exemples de la demi-clé chiffrée au moyen de deux numéros PIN différents, puis chercher les combinaisons possibles de deux numéros PIN pour voir quelle paire a permis d'obtenir la même demi-clé une fois déchiffrée. S'il existe disons 10^8 numéros PIN possibles, il n'existe alors que 10^{16} combinaisons possibles à essayer. Cela équivaut à chercher un nombre aléatoire de 54 bits, ce qui n'est pas du tout infaisable. Même si l'on obtient plusieurs solutions possibles, on peut trouver rapidement la solution correcte grâce à une troisième observation.

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