UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T H.234

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

(11/94)

TRANSMISSION DE SIGNAUX NON TÉLÉPHONIQUES

SYSTÈME DE GESTION DE CLÉS DE CHIFFREMENT ET D'AUTHENTIFICATION POUR LES SERVICES AUDIOVISUELS

Recommandation UIT-T H.234

(Antérieurement «Recommandation du CCITT»)

AVANT-PROPOS

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

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

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution n° 1 de la CMNT (Helsinki, 1er-12 mars 1993).

La Recommandation UIT-T H.234, que l'on doit à la Commission d'études 15 (1993-1996) de l'UIT-T, a été approuvée le 1er novembre 1994 selon la procédure définie dans la Résolution n° 1 de la CMNT.

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 de télécommunications.

TABLE DES MATIÈRES

Page

RÉSUMÉ

On trouvera dans ce document la description de trois méthodes de gestion des clés de chiffrement:

- les méthodes ISO 8732;
- Diffie-Hellman; et
- RSA.

Elles s'appliquent au chiffrement des signaux audiovisuels transmis numériquement dans la structure de trame H.221. Les messages de gestion définis dans ce texte sont transmis dans le canal du signal de commande de chiffrement (ECS) (*encryption control signal*) H.221 dont la structure et l'utilisation sont spécifiés dans la Recommandation H.233.

Recommandation H.234

SYSTÈME DE GESTION DE CLÉS DE CHIFFREMENT ET D'AUTHENTIFICATION POUR LES SERVICES AUDIOVISUELS

(Genève, 1994)

1 Champ d'application

Un système de chiffrement comprend deux parties, le mécanisme de confidentialité ou processus de chiffrement des données, et un sous-système de gestion de clés. Le présent document décrit les méthodes d'authentification et de gestion des clés pour un système de chiffrement destiné à être utilisé dans les services audiovisuels à bande étroite conformes aux Recommandations UIT-T H.221, H.230 et H.242. La spécification de la confidentialité étant indépendante, elle est traitée à part dans la Recommandation H.233.

La confidentialité est assurée à l'aide de *clés* de chiffrement. Ces clés sont chargées dans le mécanisme de chiffrement du système de confidentialité et régissent la manière dont les données transmises sont chiffrées et déchiffrées. Si un tiers accède aux clés utilisées, le système de chiffrement n'est plus sûr.

La maintenance des clés par les utilisateurs est donc un élément important de tout système de confidentialité. Trois méthodes pratiques de gestion des clés sont spécifiées dans le présent document. Dans les cas où la gestion automatique des clés n'est pas possible, une autre solution non spécifiée – gestion manuelle des clés, par exemple – peut être utilisée.

La première de ces méthodes est désignée sous l'appellation ISO 8732. Elle repose sur des clés mises en place manuellement dans des systèmes où elles bénéficient d'une protection physique de haut niveau, ensuite s'effectue un échange cryptographique de clé sous contrôle de ces clés installées manuellement. L'algorithme utilisé pour cet échange de clés est identique à celui utilisé pour chiffrer la communication elle-même. La sécurité des clés dans cet échange est fonction de la sécurité des clés installées manuellement.

Les clés échangées cryptographiquement peuvent être utilisées pour une seule session, ou pour plusieurs sessions sur une période donnée (un mois, par exemple). ISO 8732 contient non seulement des protocoles pour l'échange cryptographique d'informations entre les deux terminaux, mais aussi des protocoles physiques destinés à assurer la sécurité de l'installation manuelle des clés.

Il existe deux environnements distincts: un environnement point à point (à deux couches), dans lequel les deux terminaux partagent une clé commune, et un environnement à trois couches, dans lequel les deux terminaux qui souhaitent entrer en communication ne partagent pas une clé commune mais utilisent les équipements d'un tiers avec lequel chacun de ces deux terminaux partagent une clé commune. Les interfaces avec ce tiers n'entrent pas dans le cadre de la présente Recommandation, bien qu'il faille établir une distinction entre ces deux environnements.

A noter que l'échange de clés spécifié au 2.3.2 fait fonctionnellement double emploi avec la Norme X9.17, en ce sens que les clés échangées par cette norme sont suffisamment solides pour servir de clés de session. Toutefois, pour suivre la forme de la présente Recommandation, ces clés auront la fonction de la *clé* décrite au 2.3.2.

La deuxième de ces méthodes est une méthode simple mais sûre connue sous le nom de «Diffie-Hellman élargie», dans laquelle la génération et l'échange des clés se fait automatiquement par le système lui-même (lequel échange de clés est lui-même chiffré). Cette méthode ne nécessite aucune intervention des utilisateurs avant la fin de l'échange des clés; ceux-ci sont ensuite engagés à confirmer *verbalement* un code de contrôle fourni par le terminal. Cette méthode est tout à fait indiquée pour empêcher des personnes extérieures d'écouter une communication audiovisuelle acheminée sur une voie de transmission par satellite, par exemple. Pour pénétrer le système, l'intrus devrait intercepter la totalité de la communication bidirectionnelle avant que le chiffrement ne soit activé, et échanger les clés avec les deux correspondants, en se faisant passer auprès de chacun d'entre eux pour l'autre. Cette méthode n'assure pas d'authentification.

La troisième méthode est plus complexe, assurant un degré plus élevé de secret ainsi que l'*authentification* des entités assurant les services audiovisuels (terminaux, MCU, etc.). La «méthode RSA», très voisine de la méthode de clés publiques spécifiée dans la Recommandation X.509, utilise l'algorithme RSA. Cette méthode nécessite la création d'une agence de sécurité, accessible à l'ensemble de la population des entités à interconnecter: la certification est de fait «hors ligne», et repose sur l'intégrité de l'agence. Ce mécanisme d'authentification, qui permet aux correspondants participant à une communication conférence de s'identifier entre eux sans erreur possible, peut être mis en œuvre tant pour des communications multipoint que pour des communications point à point.

A chacune de ces méthodes doit être associée un canal clair exempt d'erreur. A noter qu'aucune d'entre elles n'assure le contrôle d'accès, l'intégrité des données et la non-répudiation.

Le présent document fait état d'une quatrième méthode appelée «échange manuel de clés».

L'échange manuel de clés est défini comme l'introduction par les opérateurs de clés de chiffrement de clés directement dans les terminaux, sans échanges de messages H.234. La même clé est introduite dans les deux terminaux. La longueur des clés dépend de l'algorithme de chiffrement. L'ordre d'introduction des bits pour les clés est le suivant: bit de plus fort poids (msb) en premier et bit de plus faible poids (lsb) en dernier. Le mécanisme d'introduction des clés dans le terminal dépend de fait du terminal et ne relève pas de la présente Recommandation.

Exemples:

- utilisation d'un clavier téléphonique pour introduire: (msb) 00111010...01110100 (lsb)
- téléchargement de la même clé à partir d'un ordinateur
- utilisation d'un clavier pour introduire l'équivalent de caractères hexadécimaux: (msb) 3A...74 (lsb)

L'introduction manuelle peut intervenir avant l'établissement de la communication ou pendant celle-ci. Dans ce dernier cas, les correspondants peuvent décider de lancer le chiffrement en cours de conférence, d'introduire une clé à l'aide de l'interface fournie par le terminal, puis de démarrer le chiffrement par l'intermédiaire de l'interface utilisateur du terminal. C'est lorsque le chiffrement est demandé par l'intermédiaire de l'interface utilisateur qu'il est procédé à l'envoi du code BAS «chiffrement en service» à l'ouverture du canal ECS, au choix des algorithmes de chiffrement, à l'adoption du mode manuel de gestion des clés et à l'échange des clés de session.

Pour qu'un système de chiffrement puisse être considéré comme étant privé, tous les correspondants qui participent à la conférence doivent savoir quelles sont les personnes/les installations qui ont accès aux données chiffrées, qu'il s'agisse d'autres correspondants participant à la même conférence ou d'équipements du type MCU ou dispositifs de conversion. Cela nécessite une période de préparation initiale avant que la Conférence ne commence de manière que les entités puissent s'authentifier. Ainsi, toutes les entités qui ont accès aux données chiffrées sont identifiées sans erreur possible par toutes les autres entités avant que la conférence ne commence. Le cadre d'authentification permet en outre de fournir à tout exploitant de réseau divers renseignements – sur la facturation d'une communication MCU, par exemple.

Si l'équipement MCU (du type dit «à plusieurs niveaux de sécurité») donne accès à des données en clair, il doit être intégré au dispositif d'authentification. En outre, les utilisateurs doivent être informés de la présence d'un MCU à plusieurs niveaux de sécurité dans le réseau.

L'article 2 traite des aspects communs à toutes les méthodes, les articles 3, 4 et 5 traitant respectivement des méthodes ISO 8732, Diffie-Hellman et RSA.

Définitions

AVSE: Entité de services audiovisuels (terminaux, MCU, etc.).

clé: Clé de chiffrement de clés.

2 Système de messages et échange de clés

2.1 Canal de message

Le système décrit ci-dessous comporte plusieurs messages définis acheminés en séquence entre les deux extrémités de la liaison. Le canal exempt d'erreurs à utiliser à cet effet est décrit (blocs d'échange de sessions) dans la Recommandation H.233.

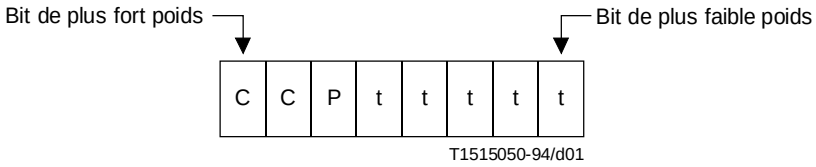
2.2 Formats des messages

Les messages utilisés par le système de chiffrement pour la répartition et l'authentification des clés ont un format de type ILC (identificateur, longueur, contenu) avec entrelacement, comme indiqué dans la Recommandation X.209. Le codage de la longueur peut être de forme courte ou de forme longue. La forme indéfinie spécifiée dans la Recommandation X.209 ne sera pas utilisée.

Un bref rappel de quelques-unes des définitions de la Recommandation X.209 utilisées dans le cadre de la présente Recommandation est présenté ci-dessous.

2.2.1 Identificateur

Un identificateur est un octet dont la structure est la suivante:



Les deux bits CC, «Classe d'étiquette», définissent le type d'identificateur qui aura la valeur 10 (en fonction du contexte) pour les identificateurs définis dans la présente Recommandation.

Le bit primitive/constructeur (P) indique si le contenu est une primitive ou s'il est composé d'éléments entrelacés.

L'étiquette à 5 bits (ttttt) définit l'identificateur de manière spécifique (en fonction de sa classe).

Les identificateurs qui figurent dans la présente Recommandation se présentent donc tous sous la forme d'un octet du type: 1 0 P t_1 t_2 t_3 t_4 t_5.

2.2.2 Longueur

La longueur du contenu, exprimée en nombre d'octets, est elle-même variable.

La forme courte, qui est d'un octet, est à utiliser de préférence à la forme longue lorsque L est inférieur à 128. Le bit 8 a la valeur 0 et les bits 7 à 1 codent L sous forme de nombre binaire sans signe dont le bit de plus fort poids et le bit de plus faible poids sont respectivement le bit 7 et le bit 1.

La forme longue, qui varie de 2 à 127 octets, est utilisée lorsque L est supérieur ou égal à 128 et inférieur à 2 à la puissance 1008. Le bit 8 du premier octet a la valeur 1. Les bits 7 à 1 du premier octet affectent une valeur de codage inférieure d'une unité à la longueur en octets sous la forme d'un nombre binaire sans signe dont le bit de plus poids fort et le bit de plus faible poids sont respectivement le bit 7 et le bit 1. L lui-même est codé sous la forme d'un nombre binaire sans signe dont le bit de plus fort poids et le bit de plus faible poids sont respectivement le bit 8 du deuxième octet et le bit 1 du dernier octet. Ce nombre binaire doit être codé en un nombre aussi faible que possible d'octets, sans octet de gauche contenant la valeur 0.

2.2.3 Chaîne binaire

Une chaîne binaire en forme de primitive compte huit bits par octet, précédés d'un octet qui code le nombre de bits inutilisés du dernier octet du contenu – de zéro à sept – sous la forme d'un nombre binaire sans signe dont le bit de plus fort poids et le bit de plus faible poids sont respectivement le bit 8 et le bit 1.

2.3 Démarrage du système de chiffrement

Le démarrage du système fait intervenir trois messages (P0, P1, P2) décrits en détail ci-dessous. Le système de chiffrement est lancé par l'envoi d'un message (en provenance de l'une ou l'autre extrémité) de type (P0). Le message (P0) inclut des bits qui décrivent les mécanismes – ISO 8732 et/ou Diffie-Hellman et/ou RSA – que l'expéditeur peut admettre. Le destinataire d'un tel message détermine le mécanisme à utiliser et y répond par un message de type (P0) ou de type (P1) selon le résultat.

Si l'expéditeur et le destinataire envoient le message (P0) en même temps, le choix demeure possible par comparaison des champs de bits échangés:

- si les deux extrémités acceptent le même mécanisme, le mécanisme utilisé est retenu; si plusieurs mécanismes peuvent être acceptés, on retient, par ordre de préférence, les mécanismes ISO 8732, puis Diffie-Hellman, puis RSA/X.509 et, enfin, l'option non spécifiée que la présente Recommandation qualifie de «manuelle»;
- s'il n'existe pas de fonction commune, le chiffrement de la liaison est impossible.

2.3.1 Messages de lancement

Nom du message:	Demande de système de chiffrement (P0).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000000
Signification:	L'expéditeur de ce message souhaite utiliser un système de chiffrement. Ce message peut être utilisé pour tenter de lancer le chiffrement ou pour répondre à un autre message P0.
Contenu:	Un octet de primitive tel que représenté ci-dessous. Le champ de bits à l'intérieur du contenu indique le type de mécanisme qui peut être utilisé. (msb) 0000XDRM (lsb). X est mis à '1' si le mécanisme ISO 8732 est accepté, ou à '0' dans l'hypothèse inverse. D est mis à '1' si le mécanisme Diffie-Hellman est accepté, ou à '0' si ce mécanisme n'est pas accepté. R est mis à '1' si le mécanisme RSA est accepté, ou à '0' si ce mécanisme n'est pas accepté. M est mis à '1' s'il existe un système de gestion de clés non spécifié de type à introduction manuelle de clés, ou à '0' dans l'hypothèse inverse.
Dans la «notation de syntaxe abstraite» ASN.1 de la Recommandation X.209:	RequestEncryptionSystem ::= [0] IMPLICIT OCTET STRING
	La longueur du contenu de ce message est toujours d'un octet.

Nom du message:	Chiffrement impossible (P1).
Signification:	Envoyé dans réponse à (P0). L'expéditeur de ce message n'utilisera pas un système de chiffrement.
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000001
Contenu:	Ce message n'a pas de contenu.

Nom du message:	Echec du lancement du système de chiffrement (P2).
Signification:	L'expéditeur de ce message n'a pas réussi à lancer son système de chiffrement. Cet échec peut être dû à une défaillance au stade de l'échange des clés mais, pour des raisons de sécurité, la cause de l'échec n'est pas indiquée dans le message.
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000010
Contenu:	Ce message n'a pas de contenu.

2.3.2 Echange des clés de session

Les clés de session utilisées pour le chiffrement de l'information proviennent de l'échange des clés de session. Le message qui contient les clés de session est composé comme indiqué ici et chiffré à l'aide d'une clé de chiffrement de clés (désignée dans la présente Recommandation par la dénomination abrégée *clé*) résultant de l'authentification ou du protocole d'échange de *clé*. A noter la distinction entre ces deux types de clés: les clés de session sont utilisées pour le chiffrement/déchiffrement du signal audiovisuel dans la structure de trame décrite dans la Recommandation H.221, alors que la *clé* n'est utilisée que pour le chiffrement et le déchiffrement lors de l'échange des clés de session.

Le mécanisme de chiffrement exige des clés d'une longueur de N bits. Une *clé* commune est établie par les deux correspondants, elle aussi d'une longueur N bits; dans le cas du mécanisme RSA une #clé# d'authentification supplémentaire est utilisée pour obtenir la *clé*.

La *clé* commune est utilisée pour le chiffrement des clés de N-bits, comme indiqué dans le présent paragraphe (voir la Figure 1). La méthode de chiffrement utilisée doit être la même que celle choisie pour le chiffrement du signal audiovisuel, ce dont attestera la transmission du message P9 défini à cette fin dans la Recommandation H.233.

Le message d'échange des clés de session comprend un identificateur de message à 8 bits, un vecteur d'initialisation avec correction d'erreur et une valeur aléatoire de 4N-bits. Chaque extrémité envoie cette valeur dont elle extrait le jeu de 4 clés de session. Chaque clé a une longueur de N bits, la valeur de N dépendant de l'algorithme de chiffrement à utiliser (par exemple, dans le cas du chiffrement-B (B-crypt), N = 56).

Les numéros aléatoires émis et reçus sont traités comme quatre blocs de N-bits, comme indiqué ci-après:

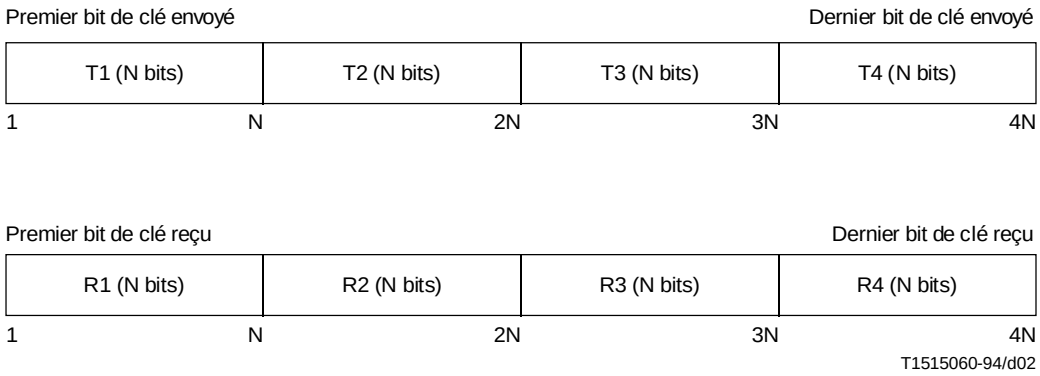


FIGURE 1/H.234

Ordre des bits d'échange de clés

Chacune des quatre clés est composée d'un OU exclusif du point de vue des bits d'un bloc émis et d'un bloc reçu, préservant l'ordre des bits: en d'autres termes, le bit de plus fort poids de la clé (c'est-à-dire le bit de plus fort poids du premier octet ou mot des données de clés chargées dans le dispositif de chiffrement) est composé de l'OU exclusif du point de vue des bits des deux premiers bits des blocs. D'après l'ordre des bits de la Figure 1, les quatre clés sont obtenues comme suit:

«Envoi clé de chiffrement # 1»	composée du bloc R3 de l'OU exclusif du bloc T1
«Envoi clé de chiffrement # 2»	composée du bloc R4 de l'OU exclusif du bloc T2
«Réception clé de chiffrement # 1»	composée du bloc R1 de l'OU exclusif du bloc T3
«Réception clé de chiffrement # 2»	composée du bloc R2 de l'OU exclusif du bloc T4

La clé de chiffrement # 1 doit être utilisée pour le chiffrement du contenu du signal de verrouillage de trame «chiffrement en service» spécifié au A.3/H.221. Lorsque MLP est EN SERVICE dans le cadre d'une commande BAS du Tableau A.1/H.221 ou du Tableau A.2/H.221, le chiffrement du canal MLP doit être effectué selon les normes spécifiées dans les Recommandations UIT-T de la série T.120, à l'aide de la même clé # 1 ou de la clé de remplacement # 2.

L'algorithme choisi peut nécessiter la parité des clés - ce dont il sera décidé à l'échelon local indépendamment de la transmission.

Seul le jeu des quatre clés de N bits de longueur (4N bits) fait l'objet d'un contrôle. Si le mode «OU exclusif» donne un résultat nul pour toutes les valeurs de ce jeu (c'est-à-dire si toutes les valeurs des clés de N bits de longueur sont nulles), il n'est pas procédé au chargement des clés et le système de secret n'est pas lancé.

Message d'échange des clés de session (P6)

Ce message est constitué de l'identificateur du message, d'un vecteur d'initialisation par défaut de 96 bits y compris les bits de correction d'erreur et d'un numéro aléatoire 4N bits.

Nom du message:	Présentation de l'information de clé de session (P6).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100110
Signification:	L'expéditeur de ce message procède à l'échange des informations de clé de session.
Contenu:	Un constructeur contenant le vecteur d'initialisation (non chiffré) utilisé pour le chiffrement des données de clé de session, et l'information de clé de session chiffrée dans le format indiqué.
Dans la «notation de syntaxe abstraite» de la Recommandation X.209:	SessionKeyInformation ::= [6] IMPLICIT SEQUENCE { vecteur d'initialisation [0] IMPLICIT BIT STRING, information de clé de session [1] IMPLICIT BIT STRING }

3 Gestion des clés ISO 8732

3.1 Introduction

La norme de la Référence 1 prévoit un processus uniforme pour la protection et l'échange de clés cryptographiques aux fins d'authentification et de chiffrement. Cette norme définit la gestion manuelle et automatique des équipements contenant des clés, ce qui recouvre:

- le contrôle de ces équipements pendant leur durée de vie pour empêcher la divulgation, la modification ou le remplacement non autorisés de données;
- la répartition de ces équipements pour permettre l'échange d'informations entre équipements ou installations cryptographiques;
- la nécessité d'assurer l'intégrité de ces équipements aux différentes phases de leur durée de vie, dont les suivantes: production, distribution, entreposage, introduction, utilisation et destruction;
- la reprise en cas de défaillance du processus de gestion des clés ou de doute quant à l'intégrité des équipements.

L'algorithme utilisé pour le chiffrement des clés distribuées cryptographiquement est normalement identique à celui utilisé pour chiffrer la communication elle-même; il peut être négocié par des échanges du message P8. En cas d'utilisation d'un algorithme autre que DES (Norme de chiffrage de données) (*data encryption standard*), le système de gestion des clés n'est pas rigoureusement conforme à la Référence 1; il s'en écarte uniquement sur ce point.

3.2 Architecture de gestion de clés

Une liste des conditions applicables à deux correspondants en communication figure dans la Référence 1. Il existe une architecture à deux couches et une architecture à trois couches. L'une et l'autre peuvent être utilisées pour l'échange de clés.

3.3 Environnements de gestion de clés

Il existe trois environnements pour la répartition de clés:

- l'environnement point à point;
- le centre de répartition de clés (CKD) (*key distribution centre*); et
- le centre de transposition de clés (CKT) (*key translation centre*) .

Des précisions sur ces environnements sont données dans la Référence 1.

L'environnement point à point est un environnement à deux couches, dans lequel les deux terminaux partagent une même clé. Cette clé commune est supposée avoir été répartie manuellement à l'aide des protocoles de sécurité et de la protection physique définis dans ISO 8732. L'échange de clés automatique spécifié dans ISO 8732 permet qu'une *clé* commune soit générée par un terminal, transmise à l'autre terminal en toute sécurité et que cette clé soit utilisée pour la création des clés de session spécifiées au 2.3.2.

Sans entrer dans les détails des distinctions entre un centre de répartition de clés (CKD) et un centre de transposition de clés (CKT), la présente Recommandation précise toutefois que la clé utilisée en partage par chacun des terminaux avec un même correspondant ou centre (CKD ou CKT) tiers doit être une clé de longueur double. La manière dont un terminal – le terminal A, par exemple – est relié au centre n'est pas non plus spécifiée dans la présente Recommandation, mais à la fin de l'échange avec le centre, le terminal A est en possession non seulement d'une *clé* en clair, mais aussi d'une *clé* chiffrée selon la clé de longueur double (voir ISO 8732 pour la spécification de l'algorithme) du terminal B. Il envoie cette clé chiffrée dans le bloc SE par l'intermédiaire du canal ECS au terminal B, dans lequel elle est ensuite convertie en une *clé* en clair, après quoi le protocole d'échange pour la session peut commencer.

3.4 Echanges de messages de service cryptographiques

ISO 8732 recommande l'utilisation de textes pour l'échange de messages (Référence 1). Elle indique l'ordre et les circonstances dans lesquels les messages sont envoyés. Le message suivant (P11) déclenche le mécanisme d'envoi d'un message de service cryptographique (CSM) ISO 8732. Chaque octet représente un caractère du texte.

L'ordre des bits est tel que le bit de plus fort poids est transmis en premier.

Nom du message:	Message de service cryptographique (P11).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10101011
Signification:	L'expéditeur de ce message envoie un seul message de service cryptographique.
Contenu:	Une chaîne de texte de primitive.
Dans la «notation de syntaxe abstraite n° 1» de la Recommandation X.209:	CryptographicServiceMessage ::= [11] IMPLICIT VisibleString

On admet que l'interface utilisateur du terminal comporte des protocoles permettant d'identifier par leur nom les clés et autres identificateurs appropriés implicites dans le protocole ISO 8732. Par exemple, dans un réseau privé, chaque groupe de deux correspondants en communication dans un environnement à deux couches peut disposer en partage d'une clé portant un nom, intégrée dans l'unité cryptographique du système, clé dont le mécanisme chargé de placer l'appel peut divulguer automatiquement l'identité au sous-système cryptographique.

La Référence 1 spécifie les messages de service pour les conditions d'erreur et les réponses sur erreur. Si deux terminaux, tous deux conformes à ISO 8732, essaient de communiquer entre eux dans des circonstances où ni l'un ni l'autre ne correspond en fait à aucun des trois environnements, les protocoles (qui exigent généralement des identificateurs ou des noms connus pour les clés, les compteurs, les centres...) s'interrompront et la tentative de service cryptographique prendra fin, avec notification aux exploitants des terminaux. Pour établir une communication qui nécessite une opération de chiffrement, les utilisateurs des deux terminaux se replieront individuellement sur un autre mécanisme d'échange de gestion de clés, ou s'implanteront dans l'un des trois environnements (très probablement par l'intermédiaire d'un tiers ou d'un centre commun).

3.5 Exemple d'échange de messages ISO 8732

Examinons à titre d'exemple la Figure 2 qui représente le flux de messages normal. Le premier message envoyé est le message RSI (Service demande) (*request service*). Le paragraphe 8.4 de la Référence 1 décrit le format du message CSM [message de service cryptographique] (*cryptographic service message*), qui est le suivant:

CSM(MCL/ ...)

et dont tous les caractères sont du type ASCII, les parenthèses indiquant le début et la fin du message, la barre oblique (/) servant à séparer les étiquettes de champ du contenu des champs.

Dans le cas considéré ici, le contenu des champs MCL étant le message RSI, le texte effectivement envoyé est le suivant:

CSM(MCL/RSI ...)

L'ordre des champs pour le message RSI est indiqué dans le Tableau III de ISO 8732. Cet ordre est le suivant: MCL RCV ORG SVR EDC (facultatif). Dans l'exemple considéré ici, le champ EDC facultatif est omis.

Le Tableau II définit chaque champ plus en détail. Le message envoyé serait donc le suivant:

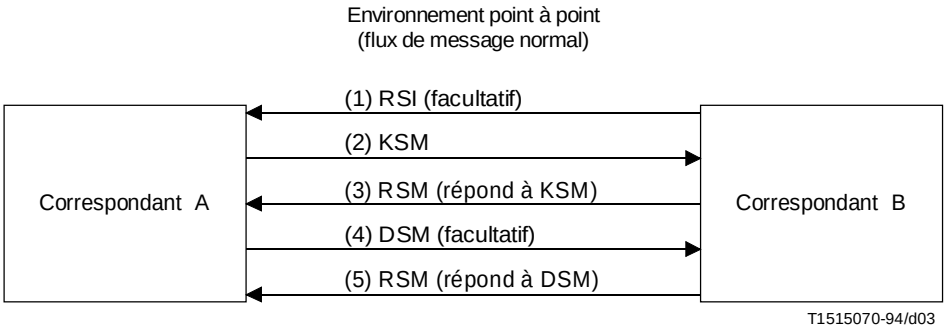
CSM(MCL/RSI"RCV/A"ORG/B"SVR/KK.KD.IV)

où:

"	Espace intercalaire servant de séparateur de champ
A	Destinataire
B	Expéditeur
.	Séparateur de champ secondaire
SVR	Demande de service
KK	Demande de *clé*
KD	Demande de deux clés de données
IV	Demande d'IV
MCL	Classe de message
RCV	Destinataire
ORG	Emetteur

Le paragraphe 9.7 de ISO 8732 décrit le message RSI plus en détail.

Le deuxième message serait pour KSM [message de service de clé] (*key service message*), le troisième pour RSM [message de service de réponse] (*response service message*), le quatrième pour DSM [message de service déconnecté] (*disconnected service message*) et le cinquième à nouveau pour RSM.



NOTE – Le processus de déconnexion (DSM) peut être lancé par le correspondant A ou par le correspondant B; le cas représenté ici est celui du lancement par le correspondant A.

FIGURE 2/H.234

4 Distribution de clés Diffie-Hellman élargie

4.1 Introduction

L'échange reprend mais sous une forme élargie la distribution Diffie-Hellman de manière à exploiter les propriétés de la liaison audiovisuelle pour assurer un élément de protection contre la mise en dérivation de lignes actives. L'échange conduit donc à l'utilisation d'une valeur secrète commune tant pour le contrôle de la ligne que pour l'échange des clés de session.

L'opération se déroule comme suit (voir l'Appendice I, [1]):

1) le protocole de répartition de *clé* procède à l'échange des données conformément au protocole décrit ici;
2) les données de (1) sont utilisées pour l'échange des clés de session qui serviront au chiffrement de la liaison;
3) les données de (1) sont utilisées pour contrôler la liaison.

4.2 Le protocole de base

Ce protocole consiste en un échange initial de données, suivi d'un échange bidirectionnel de résultats intermédiaires dont sont tirées les données partagées.

4.2.1 Méthode d'échange de *clé*

La méthode utilisée est une version double de la méthode Diffie-Hellman de base. Le double échange est utilisé de manière que la *clé* qui en résulte ne soit pas entièrement fonction d'un nombre premier et d'une racine primitive choisis dans un seul terminal.

Considérons deux entités de services audiovisuels (AVSE) A et B.

A envoie à B: le nombre entier p_A,

la racine primitive probabiliste a_A,

la valeur $c_1 = \{a_A{}^{a1} \text{mod } p_A\}$ où a_1 est un nombre aléatoire connu uniquement de A.

B envoie à A: le nombre entier p_B,

la racine primitive probabiliste a_B,

la valeur $c_2 = \{a_B{}^{b1} \text{mod } p_B\}$ où b_1 est un nombre aléatoire connu uniquement de B.

A envoie à B: la valeur $c_3 = \{a_B{}^{a2} \text{mod } p_B\}$ où a_2 est un nombre aléatoire connu uniquement de A.

B envoie à A: la valeur $c_4 = \{a_A{}^{b2} \text{mod } p_A\}$ où b_2 est un nombre aléatoire connu uniquement de B.

Calculons une paire de résultats r_1 et r_2 pour A, puis pour B.

L'AVSE A forme: $r_1 = c_4{}^{a1} \text{mod } p_A$ et $r_2 = c_2{}^{a2} \text{mod } p_B$

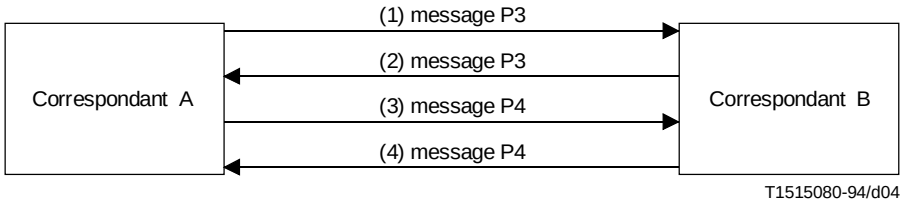
L'AVSE B forme: $r_1 = c_1{}^{b2} \text{mod } p_A$ et $r_2 = c_3{}^{b1} \text{mod } p_B$

A et B sont maintenant en possession des mêmes valeurs de résultats $r_1 = a_A{}^{a1.b2} \text{mod } p_A$ et $r_2 = a_B{}^{a2.b1} \text{mod } p_B$.

Le résultat final R_{12} est obtenu par un 'OU-Exclusif' du point de vue des bits de r_1 avec r_2. Si r_1 et r_2 n'ont pas la même longueur et si l'on désigne par L la plus petite de ces deux longueurs, l'opération OU-exclusif est la suivante:

{(L bits de plus faible poids de r_1).OU-Excl.(L bits de plus faible poids de r_2)}

L'échange Diffie-Hellman double est donc représenté comme indiqué à la Figure 3.



(1) p_A, a_A, ($a_A{}^{a1} \text{mod } p_A$) par message {P3}

(2) p_B, a_B, ($a_B{}^{b1} \text{mod } p_B$) par message {P3}

(3) $a_B{}^{a2} \text{mod } p_B$) par message {P4}

(4) $a_A{}^{b2} \text{mod } p_A$) par message {P4}

FIGURE 3/H.234

Echange Diffie-Hellman double

4.2.2 Détermination de la *clé*

Comme indiqué ci-dessus, A et B forment $r_1 = (a_A{}^{a1.b2} \text{mod } p_A)$ et $r_2 = (a_B{}^{a2.b1} \text{mod } p_B)$, R_{12} étant ensuite formé par «OU-Exclusif» du point de vue des bits de ces valeurs. A et B contrôlent la valeur du résultat et si tous les bits ont pour valeur 0, le message «Echec du lancement du système de chiffrement» (P2) est envoyé à l'autre entité.

R_{12} est une valeur de K-bit disponible à chaque extrémité de la liaison. Cette valeur est utilisée pour déterminer le code de contrôle et la *clé* qui est utilisée pour le chiffrement des clés de session. Pour un mécanisme de confidentialité de N-bits, et avec un code de contrôle de M-bits, les M bits de plus faible poids forment le code de contrôle et les N bits qui suivent forment la *clé*, comme le montre la Figure 4. La valeur de M est de 64 bits. La valeur de N, qui correspond à la longueur de la *clé*, est déterminée par l'algorithme de chiffrement à utiliser.

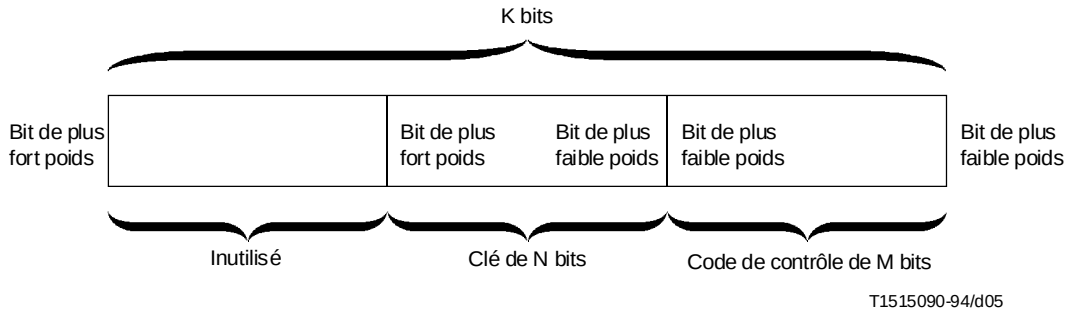


FIGURE 4/H.234

Interprétation des résultats de la répartition des clés

A noter que K doit être d'une longueur supérieure à M + N bits. Pour un algorithme de chiffrement de 64 bits et un code de contrôle de 64 bits, la longueur de K doit être supérieure à 128 bits. Dans la pratique, K dépassera sensiblement cette longueur.

4.3 Messages Diffie-Hellman

Le présent paragraphe décrit le contenu des messages nécessaires au lancement du système de chiffrement et à l'échange de *clé* Diffie-Hellman.

4.3.1 Information d'échange de *clé*

Nom du message:	Présentation de l'information d'échange de *clé* (P3).
Signification:	L'expéditeur de ce message envoie l'information d'échange de *clé* qui y est incluse dans le cadre d'un échange Diffie-Hellman double.
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100011
Contenu:	Un type constructeur comportant les différentes primitives indiquées ci-dessous, à savoir: racine primitive, nombre entier et résultat intermédiaire. A noter que le terme racine primitive n'a aucun rapport avec le terme primitive utilisé dans les définitions de message.
Dans la notation de l'ASN.1:	KeyExchangeInformation ::= [3] IMPLICIT SEQUENCE { racine primitive [0] IMPLICIT BIT STRING, nombre entier [1] IMPLICIT BIT STRING, résultat intermédiaire [2] IMPLICIT BIT STRING } Le contenu de la racine primitive (Primitive Root) est une chaîne binaire de type primitive. Le contenu du nombre entier (Prime) est une chaîne binaire de type primitive. Le contenu du résultat intermédiaire (Intermediate Result) est une chaîne binaire de type primitive, contenant un des résultats intermédiaires de l'échange Diffie-Hellman.

4.3.2 Paramètres d'échange de *clé* intermédiaire

Nom du message:	Information d'échange de *clé* intermédiaire (P4).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10000100
Signification:	L'expéditeur de ce message envoie l'information d'échange de *clé* qui y est incluse dans le cadre d'un échange Diffie-Hellman.
Contenu:	Chaîne binaire de primitive contenant le résultat intermédiaire.
Dans la notation ASN.1:	IntermediateKeyExchangeInformation ::= [4] IMPLICIT BIT STRING
	La chaîne binaire pour le résultat intermédiaire contient l'un des résultats intermédiaires pour l'échange Diffie-Hellman. Les messages P3 et P4 forment un échange D-H double dont la dernière *clé* D-H est déterminée par les deux extrémités d'une liaison.

4.3.3 Information de code de contrôle en provenance de MCU

Nom du message:	Présentation de l'information de code de contrôle en provenance de MCU (P5).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100101
Signification:	Un MCU envoie l'information de code de contrôle incluse dans le message, obtenue par suite des échanges Diffie-Hellman.
Contenu:	Un constructeur pour l'identificateur de la liaison et le code de contrôle.
Dans la notation ASN.1:	Information de code de contrôle de liaison ::= [5] IMPLICIT SEQUENCE { Identificateur de liaison [0] IMPLICIT BIT STRING, Code de contrôle [1] IMPLICIT BIT STRING }

Un MCU enverra un message (P5) pour chacune des liaisons pour lesquelles l'échange de *clé* Diffie-Hellman est terminé.

A noter que l'identificateur de la liaison est utilisé pour identifier la liaison de MCU à laquelle le code de contrôle se rapporte. Une connaissance de la configuration de MCU est nécessaire pour interpréter cet identificateur. (Voir aussi la Note relative au 4.4 ci-dessous.)

4.4 Extension pour contrôles de ligne

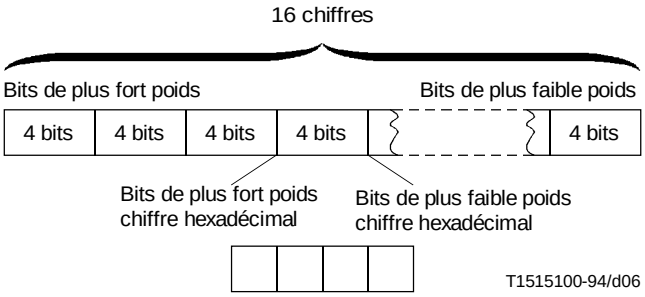
Le code de contrôle à 64 bits obtenu au 4.2 doit être présenté par le terminal en totalité ou en partie sous la forme d'un nombre hexadécimal de 16 chiffres, l'ordre des bits étant celui représenté à la Figure 5 et la terminologie utilisée étant celle de la Figure 4.

La valeur est présentée à chaque utilisateur comme indiqué, c'est-à-dire que le chiffre de gauche est déterminé à partir de l'extrémité bit de plus fort poids du code de contrôle. Il n'est pas nécessaire de présenter tous les chiffres; les quatre chiffres de gauche suffiront probablement, le risque correspondant de non détection d'un problème de ligne n'étant que d'une probabilité sur 2^{16}. La valeur présentée est transmise verbalement, sur le canal audiovisuel, par l'un des deux utilisateurs à l'autre utilisateur; ce dernier doit s'assurer que cette valeur correspond à celle affichée sur son terminal.

NOTE – Il est proposé que le contrôle verbal puisse être effectué avant le chiffrement effectif des signaux audio; en outre, ce processus et le processus proposé comme variante pour le cas du fonctionnement multipoint décrit au 4.3.3 devront coïncider dans le temps.

5 Exploitation avec mécanisme RSA

NOTE – Le terme «clé» au sens du présent article est à prendre partout au sens de #clé# défini au 2.3.2.



NOTE – Chaque bloc de quatre bits du code de contrôle forme un chiffre hexadécimal qui sera présenté sur l'écran de l'utilisateur.

FIGURE 5/H.234

Ordre des bits pour les contrôles de ligne

5.1 Introduction

5.1.1 Considérations générales

Le présent paragraphe décrit un cadre d'authentification utilisant le mécanisme RSA pour des services audiovisuels comportant des connexions point à point et multipoint.

Les procédures et fonctions d'authentification décrites sont fondées sur la Recommandation UIT-T X.509. Dans la présente Recommandation, l'authentification est établie à l'aide d'un ou de plusieurs niveaux de ce que l'on appelle les autorités de certification. Une autorité de certification (CA) délivre des certificats de manière autonome à des entités ou à d'autres CA, certificats que ces entités ou CA peuvent utiliser pour s'authentifier auprès d'autres entités ou CA. Dans le cas de services audiovisuels, les entités peuvent être les terminaux d'utilisateur ou des MCU à plusieurs niveaux de sécurité.

Le cadre d'authentification spécifique décrit ici utilise deux niveaux de CA. Au niveau inférieur, chaque domaine de réseau – un pays ou une société, par exemple – aura sa propre CA. Pour permettre l'authentification des services audiovisuels entre les différents domaines, ces CA auront une CA commune d'un niveau supérieur chargée de les authentifier. Cette CA commune doit garantir un même niveau de sécurité aux utilisateurs.

Lorsque cela n'est pas possible, il existe une autre solution, bien que plus complexe, comme indiqué brièvement au 5.5.

Les CA du niveau du domaine du réseau doivent offrir plusieurs niveaux de sécurité pour ne pas reproduire les noms d'identification sur les certificats. On présume que l'authentification proprement dite doit être établie dans un environnement n'offrant pas plusieurs niveaux de sécurité. En outre, sitôt authentifiée, l'entité bénéficie de plusieurs niveaux de sécurité (jusqu'à la fin de la communication).

5.1.2 Notation

CA	Autorité de certification (*certification authority*)
CCA	Autorité de certification de pays (*country certification authority*)
GCA	Autorité de certification générale (*general certification authority*)
h[*]	Résultat de la fonction h appliquée à *
X<<Y>>	Le certificat de l'entité Y est généré par l'entité X
Xp	Clé publique RSA de l'entité X
Xs	Clé secrète RSA de l'entité X
Xp[*]	Chiffrement/déchiffrement de [*] avec la clé Xp. Dans le cas du mécanisme RSA, cette opération est effectuée par élévation à une puissance.
Xs[*]	Chiffrement/déchiffrement de [*] avec la clé Xs. Dans le cas du mécanisme RSA, cette opération est effectuée par élévation à une puissance.

5.2 Mise en place du système

Le système spécifié ici comporte une hiérarchie à trois niveaux. Au niveau inférieur se trouvent les AVSE. Chacune de celles-ci n'est rattachée qu'à une seule CA de niveau intermédiaire lorsqu'elle communique avec une autre AVSE. Les CA de ce niveau intermédiaire font office d'autorités de certification pour un groupe d'entités (relevant toutes en principe du même pays ou du même domaine du réseau). Ces CA, que l'on appellera CCA (autorités de certification de pays), délivrent des certificats aux entités auxquelles elles sont rattachées. Au niveau supérieur se trouve une seule CA appelée la GCA (autorité de certification générale). La GCA délivre des certificats à toutes les CCA. La Figure 6 donne une représentation visuelle de cette hiérarchie.

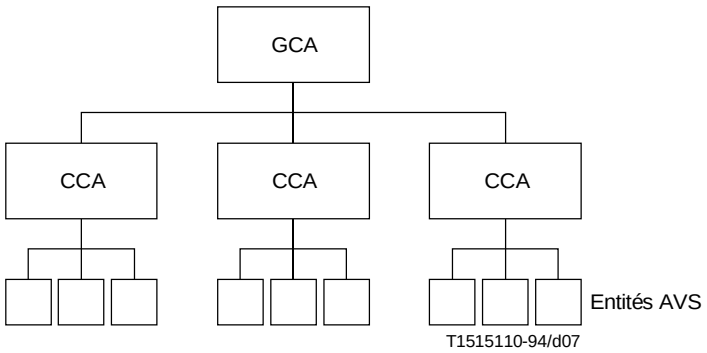


FIGURE 6/H.234

Hiérarchie des autorités de certification

Le cadre d'authentification utilise l'algorithme cryptographique RSA. Il s'agit d'un algorithme dit à clé publique dans lequel les clés de chiffrement et de déchiffrement diffèrent. L'une de ces clés peut être rendue publique, l'autre demeurant secrète. Ces clés sont respectivement appelées la *clé publique* et la *clé secrète.*

L'authentification utilise également une fonction de hachage h(*), qui établit une correspondance entre une séquence de caractères de longueur arbitraire et une séquence de caractères de longueur limitée, qui ne dépasse pas la longueur du module RSA utilisé. La fonction h(*) n'est pas spécifiée dans la présente Recommandation, mais doit l'être par l'autorité de certification. Un exemple d'une telle fonction de hachage accessible au public est donnée dans l'Appendice I [3].

5.3 Génération et répartition des clés d'authentification

Une clé d'authentification comporte deux clés, l'une secrète et l'autre publique, formant une paire pour l'algorithme RSA. Chaque autorité CA et chaque entité AVS a sa propre paire de clés d'authentification.

L'autorité GCA génère sa propre clé d'authentification, constituée d'une clé secrète GCAs et d'une clé publique GCAp.

Chaque autorité CCA génère sa propre clé d'authentification, constituée d'une clé secrète CCAs et d'une clé publique CCAp. L'autorité CCA communique la clé CCAp à l'autorité GCA, laquelle certifie cette clé.

La clé d'authentification d'une AVSE U, constituée d'une clé secrète Us et d'une clé publique Up, est générée par son autorité CCA. Les clés Up et Us sont communiquées à l'AVSE. L'autorité CCA certifie la clé Up.

La génération de la clé d'authentification GCA et la génération et la répartition des clés d'authentification CCA doivent faire l'objet d'un consensus international.

NOTE – L'interface physique entre les autorités de certification et les entités AVS n'entre pas dans le cadre de la présente Recommandation.

5.4 Certification

L'autorité GCA certifie une clé publique CCAp en calculant un certificat, appelé GCA<<CCA>>, comportant l'information suivante:

GCA<<CCA>>: GCA,CCA,CCAp,T1,GCAs[h(GCA,CCA,CCAp,T1)]

où:

GCA est l'identité de l'autorité GCA

CCA est l'identité de l'autorité CCA

CCAp est la clé publique CCA

T1 est la date de début et de fin de validité du certificat

GCAs[*] est le chiffrement de * avec la clé GCAs

NOTE – L'identité de l'autorité GCA est incluse ici conformément à la Recommandation X.509, bien que dans le système décrit l'identité déterminée pour cette autorité GCA constitue un cas particulier.

L'autorité CCA certifie une clé publique Xp d'une AVSE X en calculant un certificat, appelé CCA<<X>> comportant l'information suivante:

CCA<<X>>: CCA,X,Xp,T2,CCAs[h(CCA,X,XpT2)]

où:

GCA est l'identité de l'autorité GCA

X est l'identité de l'entité X

Xp est la clé publique X

T2 est la date de début et de fin de validité du certificat

CCAs[*] est le chiffrement de * avec la clé CCAs

La clé GCAp, les informations GCA<<CCA>> et CCA <<X>> ainsi que la clé Xs sont communiquées à l'entité X, sous la forme par exemple d'une carte à mémoire ou d'un module intégré. En outre, l'entité X doit avoir une copie imprimée de la clé GCAp qui pourra lui servir de référence en cas de doute quant à l'intégrité de la clé GCAp.

Contrôle des certificats

On peut contrôler GCA<<CCA>> en calculant h (GCA,CCA,CCAp,T1) à l'aide de GCAp et en comparant le résultat obtenu avec GCAp[GCAs[h(GCA,CCA,CCAp,T1)]]; les résultats de ces opérations doivent être identiques. On peut contrôler CCA<<X>> en calculant h(CCA,X,Xp,T2) à l'aide de CCAp et en comparant le résultat obtenu avec CCAp[CCAs[h(CCA,X,Xp,T2)]]; les résultats de ces opérations doivent être identiques.

Le système décrit au 5.4 est résumé sur la Figure 7 pour les AVSE X et Y avec respectivement comme autorités de certification CA1 et CA2.

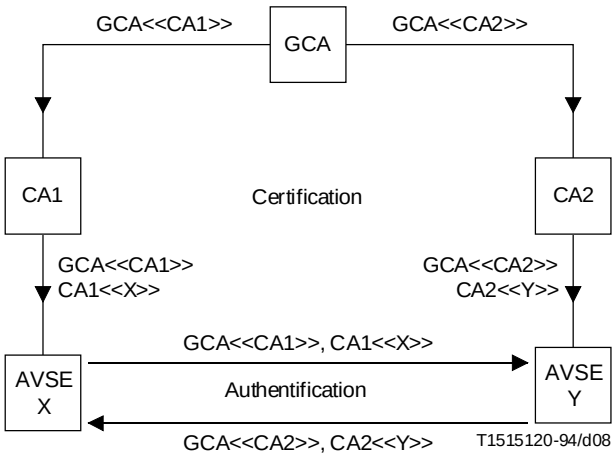


FIGURE 7/H.234

Résumé de la procédure de certification

5.5 Autre solution pour la certification sans autorité GCA

Si deux exploitants/sociétés d'exploitation de réseau souhaitent que leurs AVSE s'authentifient l'une l'autre, leurs autorités de certification CA1 et CA2 doivent se certifier mutuellement en échangeant les certificats CA1<<CA2>> et CA2<<CA1>>. Ce système sera d'un fonctionnement complexe, du fait que les AVSE X et Y pourront avoir non seulement à introduire un annuaire externe pour obtenir CA1<<CA2>> ou CA2<<CA1>> mais aussi à échanger préalablement les identités de leurs autorités de certification. Pour de plus amples précisions, voir la Figure 8.

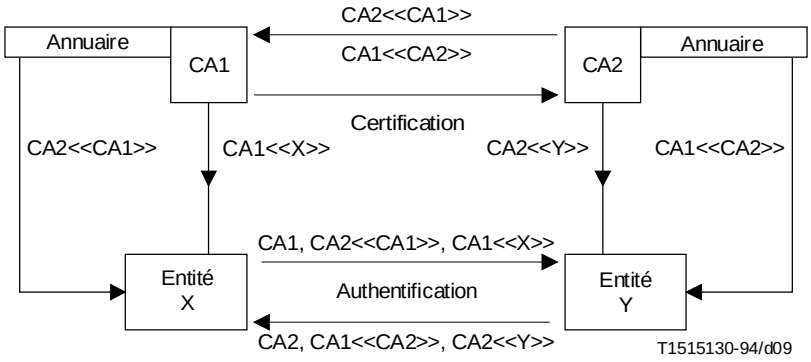


FIGURE 8/H.234

Certification sans autorité de certification supérieure

5.6 Authentification des entités

La procédure d'authentification est expliquée en détail ci-dessous; elle s'applique sur toutes les communications possibles, c'est-à-dire de MCU à MCU, de terminal à MCU, de MCU à terminal et de terminal à terminal.

La procédure d'authentification entre deux entités au moment de l'établissement de la communication fait intervenir quatre messages:

- RSA.P1 – Lancement de l'authentification;
- RSA.P2 – Réponse d'authentification;
- RSA.P3 – Fin d'authentification;
- RSA.P4 – Authentification infructueuse.

Les messages RSA.P1 et RSA.P3 sont envoyés par l'entité appelante, désignée par X; le message RSA.P2 est envoyé par l'entité appelée, désignée par Y. Les clés CCAs des entités X et Y sont respectivement désignées par CX et CY.

Le contenu du message RSA.P1 est:

GCA<<CX>>,CX<<X>>, RX,Y,Xs[h(RX,Y)]

où RX est un nombre aléatoire généré par X.

Y dans ces conditions,

1) obtient Xp à partir du message RSA.P1 et contrôle Xp à l'aide des certificats, avec la clé GCAp comme niveau de sécurité;
2) contrôle l'intégrité du message en calculant h(RX,Y) et en le comparant avec Xp[Xs[h(RX,Y)]]; les résultats doivent être identiques;
3) contrôle les dates d'expiration des certificats;
4) contrôle l'intégrité de X.

Le contenu du message RSA.P2 est:

GCA<<CY>>, CY<<Y>>,RY,X,RX,Xp[KY],Ys[h(RY,X,RX,KY)]

où RY est un nombre aléatoire et KY les données de clé (voir 2.3.2), tous deux générés par Y.

X dans ces conditions:

1) obtient Yp à partir du message RSA.P2 et contrôle Yp à l'aide des certificats, avec la clé GCAp comme niveau de sécurité;

2) déchiffre Xp[KY], obtenant ainsi KY;

3) contrôle l'intégrité du message en calculant h(RY,X,RX,KY) et en le comparant avec Yp[Ys[h(RY,X,RX,KY)]]; les résultats doivent être identiques;

4) contrôle les dates d'expiration des certificats;

5) s'assure que la valeur de RX est la même que celle envoyée dans le message RSA.P1;

6) contrôle l'intégrité de Y.

Le contenu du message RSA.P3 est:

RY,Y,Yp[KX],Xs[h(RY,Y,KX)]

où KX correspond aux données de clé générées par X.

Y dans ces conditions:

1) déchiffre Yp[KX], obtenant ainsi KX;

2) contrôle l'intégrité du message en calculant h(RY,Y,KX) et en le comparant avec Xp[Xs[h(RY,Y,KX)]]; les résultats doivent être identiques;

3) s'assure que la valeur de RY est la même que celle envoyée dans le message RSA.P2;

4) contrôle l'intégrité de X.

Si l'un quelconque des contrôles portant sur les messages RSA.P1, RSA.P2 ou RSA.P3 échoue, il convient d'interrompre l'établissement de la communication par l'envoi d'un message RSA.P4 – Authentification infructueuse. Le message RSA.P4 peut être envoyé aussi bien par l'entité X que par l'entité Y, et après le message RSA.P1, le message RSA.P2 ou le message RSA.P3. L'envoi du message RSA.P4 doit mettre fin à la procédure d'établissement de la communication.

NOTES

1 Il est possible d'accélérer les calculs RSA par le choix de paramètres publics spécifiques.

2 Ce système diffère de la spécification X.509 initiale en ce que KX est envoyé dans le message RSA.P3 et non pas dans le message RSA.P1. Cela présente comme avantage que l'entité X n'a pas à consulter un annuaire pour obtenir Yp. Pour l'entité X comme pour l'entité Y, la clé GCAp est le seul niveau de sécurité: tant que cette clé offre plusieurs niveaux de sécurité, de même que l'information secrète d'une entité qui est ainsi protégée contre le vol, les entités X et Y n'ont pas besoin d'accéder aux annuaires. En outre, dans le message RSA.P3, l'identité de l'entité Y est ajoutée pour des raisons de sécurité et dans les messages RSA.P2 et RSA.P3, la signature est apposée respectivement sur clés non chiffrées KY et KX.

5.6.1 Transmission simultanée de messages RSA.P1

Si l'entité X envoie à une entité Y un message de lancement:

RSA.P1(X–>Y): GCA<<CX>>,CX<<X>>,RX,Y,Xs[h/RX,Y)]

et si, avant réception du message RSA.P2 (Y–>X), l'entité Y envoie à l'entité X un message de lancement:

RSA.P1(Y–>X):GCA<<CY>>,CY<<Y>>,RY,X,Ys[h(RY,X)]

alors, les entités X et Y remédieront à cette situation en *comparant RX et RY*.

Si RX>RY, il ne doit pas être tenu compte du message RSA.P1 (Y–>X) et l'entité Y doit répondre par un message RSA.P2.

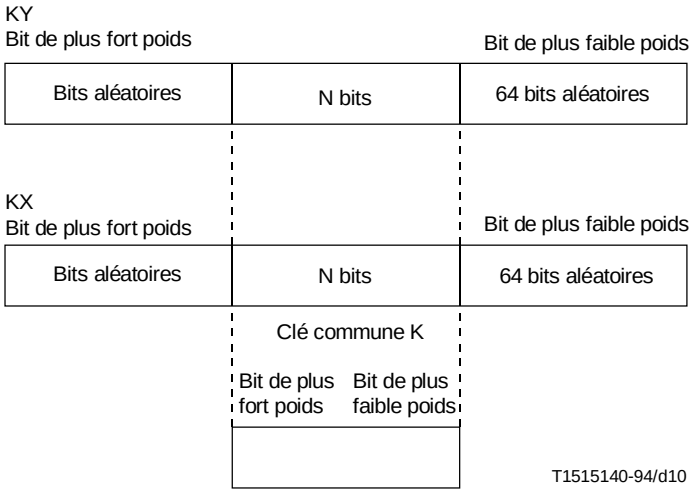
Si RY>RX, il ne doit pas être tenu compte du message RSA.P1 (X–>Y) et l'entité X doit répondre par un message RSA.P2.

Si, par coïncidence, RX = RY, il convient de ne pas tenir compte des deux messages RSA.P1 et de mettre fin à la procédure d'authentification par l'envoi d'un message RSA.P4 (authentification infructueuse).

5.7 Génération d'une clé de chiffrement des clés de session

Les données de clé KY et KX transmises dans les messages RSA.P2 et RSA.P3 seront utilisées pour créer une *clé* K commune qui servira au chiffrement des messages d'échange de clés de session, comme indiqué au 2.3.2. (Un jeu de 4 clés de session est tiré de ces messages.) Si l'on désigne par N la longueur de K, on obtient alors K en prenant la somme modulo 2 des bits 64 à 64 + N – 1 de KX et les bits 64 à 64 + N – 1 de KY (le bit 0 désignant ici le bit de plus faible poids de KX et KY). Le bit 64 de KX et le bit 64 de KY génèrent ensemble le bit 0 de K. La valeur de N, qui correspond à la longueur de la *clé*, est déterminée par l'algorithme de chiffrement à utiliser.

Les bits inutilisés de KX et KY (indice 0 à 63 et 64 + N et supérieurs) doivent être complétés par des informations aléatoires. La génération de la clé commune K à partir de KX et de KY est représentée sous forme de diagramme sur la Figure 9.



NOTE – Les blocs de N bits de KX et KY sont ajoutés modulo 2 pour former la clé commune K.

FIGURE 9/H.234

Génération d'une *clé* commune

5.8 Messages RSA

Le présent paragraphe donne le détail des messages nécessaires dans le système d'authentification utilisant le mécanisme RSA, décrit au 5.6. Les descriptions sont fondées sur la Recommandation UIT-T X.209. Quelques-unes des définitions de la Recommandation X.209 utilisées dans le présent paragraphe ont été brièvement rappelées au 2.2.

5.8.1 Lancement de l'authentification

Nom du message:	Lancement de l'authentification (RSA.P1).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10100111
Signification:	L'expéditeur de ce message souhaite engager une procédure d'authentification avec le destinataire prévu et envoie l'information nécessaire au lancement de la procédure.
Contenu:	Un constructeur, comprenant deux constructeurs pour les certificats GCA<<CX>> et CX<<X>> et trois primitives pour un numéro aléatoire RX, une identité Y et une information de hachage chiffrée Xs[h(RX,Y)].
En notation ASN.1:	RSA.P1::= [7] IMPLICIT SEQUENCE { Certificat de l'autorité GCA pour l'autorité CCA [0] IMPLICIT GCA-Certificate, Certificat de l'autorité CCA pour l'entité [1] IMPLICIT CCA-Certificate, Numéro aléatoire de l'entité appelante [2] IMPLICIT BIT STRING, Identité de l'entité appelée [3] IMPLICIT BIT STRING, Information de hachage incluse dans la clé de secret appelante [4] IMPLICIT BIT STRING }

Le contenu du numéro aléatoire de l'entité appelante est une chaîne binaire de base.

Le contenu de l'identité de l'entité appelée est une chaîne binaire de base.

Le contenu de l'information de hachage incluse dans la clé de secret appelante est une chaîne binaire de base.

Contenu du certificat de l'autorité GCA pour l'autorité CCA: un constructeur comprenant cinq chaînes de base pour une identité d'autorité GCA, une identité d'autorité CCA, une clé publique CCAp, une date limite de validité T1 et une information de hachage chiffrée GCAs[h(GCA,CCA,CCAp,T1)].

En notation ASN.1:

Certificat de l'autorité GCA::= SEQUENCE{Identité de l'autorité GCA [0] IMPLICIT BIT STRING,

Identité de l'autorité CCA [1] IMPLICIT BIT STRING,

Clé publique CCA [2] IMPLICIT BIT STRING,

Date limite de validité du certificat [3] IMPLICIT BIT STRING,

Information de hachage incluse dans la clé secrète de l'autorité GCA [4] IMPLICIT BIT STRING}

Le contenu de l'identité de l'autorité GCA est une chaîne binaire de base.

Le contenu de l'identité de l'autorité CCA est une chaîne binaire de base.

Le contenu de la clé publique de l'autorité CCA est une chaîne binaire de base.

Le contenu de la date limite de validité du certificat est une chaîne binaire de base.

Le contenu de l'information de hachage incluse dans la clé de secret de l'autorité GCA est une chaîne binaire de base.

Contenu du certificat de l'autorité CCA pour l'entité: un constructeur comprenant cinq chaînes de base pour une identité d'autorité CCA, une identité d'entité X, une clé publique Xp, une date limite de validité T2 et une information de hachage chiffrée CCAs[h(CCA,X,Xp,T2)].

En notation ASN.1:

Certificat de l'autorité CCA::= SEQUENCE{ Identité de l'autorité CCA [0] IMPLICIT BIT STRING,

Identité de l'entité [1] IMPLICIT BIT STRING,

Clé publique de l'entité [2] IMPLICIT BIT STRING,

Date limite de validité du certificat [3] IMPLICIT BIT STRING,

Information de hachage incluse dans la clé secrète de l'autorité CCA [4] IMPLICIT BIT STRING}

Le contenu de l'identité de l'autorité CCA est une chaîne binaire de base.

Le contenu de l'identité de l'entité est une chaîne binaire de base.

Le contenu de la clé publique de l'entité est une chaîne binaire de base.

Le contenu de la date limite de validité du certificat est une chaîne binaire de base.

Le contenu de l'information de hachage incluse dans la clé de secret de l'autorité CCA est une chaîne binaire de base.

5.8.2 Réponse d'authentification

Nom du message:	Réponse d'authentification (RSA.P2).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10101000
Signification:	L'expéditeur de ce message répond au lancement d'une authentification et envoie l'information nécessaire à la procédure d'authentification.
Contenu:	Un constructeur, comprenant deux constructeurs pour les certificats GCA<<CY>> et CY<<Y>> et cinq primitives pour un numéro aléatoire RY, une identité d'entité X, un numéro aléatoire RX, une information de clé chiffrée Xp[KY] et une information de hachage chiffrée Ys[h(RY,X,RX,KY)].
En notation ASN.1:	RSA.P2 ::= [8] IMPLICIT SEQUENCE { Certificat de l'autorité GCA pour l'autorité CCA [0] IMPLICIT GCA-Certificate, Certificat de l'autorité CCA pour l'entité [1] IMPLICIT CCA-Certificate, Numéro aléatoire de l'entité appelée [2] IMPLICIT BIT STRING, Identité de l'entité appelante [3] IMPLICIT BIT STRING, Numéro aléatoire de l'entité appelante [4] IMPLICIT BIT STRING, Information de clé incluse dans la clé publique appelante [5] IMPLICIT BIT STRING, Information de hachage incluse dans la clé secrète appelée [6] IMPLICIT BIT STRING }

Le contenu du numéro aléatoire de l'entité appelée est une chaîne binaire de base.

Le contenu de l'identité de l'entité appelante est une chaîne binaire de base.

Le contenu du numéro aléatoire de l'entité appelante est une chaîne binaire de base.

Le contenu de l'information de clé incluse dans la clé publique appelante est une chaîne binaire de base.

Le contenu de l'information de hachage incluse dans la clé secrète appelée est une chaîne binaire de base.

Les contenus du certificat de l'autorité GCA pour l'autorité CCA et du certificat de l'autorité CCA pour l'entité sont analogues à ceux décrits au 5.8.1.

5.8.3 Authentification complète

Nom du message:	Authentification complète (RSA.P3).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10101001
Signification:	L'expéditeur de ce message, en sa qualité de demandeur de la procédure d'authentification, envoie l'information nécessaire pour mener à bien la procédure d'authentification.
Contenu:	Un constructeur, comprenant quatre primitives pour un numéro aléatoire RY, une identité d'entité Y, une information de clé chiffrée Yp[KX] et une information de hachage chiffrée Xs[h(RY,Y,KX)].
En notation ASN.1:	RSA.P3 ::= [9] IMPLICIT SEQUENCE { Numéro aléatoire de l'entité appelée [0] IMPLICIT BIT STRING, Identité de l'entité appelée [1] IMPLICIT BIT STRING, Information de clé incluse dans la clé publique appelée [2] IMPLICIT BIT STRING, Information de hachage incluse dans la clé secrète appelante [3] IMPLICIT BIT STRING }

Le contenu du numéro aléatoire de l'entité appelée est une chaîne binaire de base.

Le contenu de l'identité de l'entité appelante est une chaîne binaire de base.

Le contenu de l'information de clé incluse dans la clé publique appelée est une chaîne binaire de base.

Le contenu de l'information de hachage incluse dans la clé secrète appelante est une chaîne binaire de base.

5.8.4 Authentification infructueuse

Nom du message:	Authentification infructueuse (RSA.P4).
Identificateur du message:	1 0 P t_1 t_2 t_3 t_4 t_5 = 10001010
Signification:	L'expéditeur de ce message indique que quelque chose n'a pas fonctionné convenablement pendant la procédure d'authentification et que celle-ci va être interrompue. L'envoi ou la réception de ce message doit déclencher l'interruption de la procédure d'établissement de la communication.
Contenu:	Ce message n'a pas de contenu.

6 Exploitation avec MCU

Dans le cas d'un MCU «à plusieurs niveaux de sécurité» (aux entrées duquel est effectué le déchiffrement de tous les signaux, d'où la nécessité d'installer le MCU dans un endroit où il sera en sécurité), les communications entre chaque terminal audiovisuel et le MCU doivent être chiffrées comme indiqué dans la présente Recommandation pour une liaison point à point. Bien évidemment, cette méthode n'est pas applicable aux raccordements de terminaux téléphoniques pour conférence par l'intermédiaire du réseau téléphonique analogique.

La présente Recommandation ne contient aucune disposition relative à l'exploitation d'un MCU sans cette méthode de déchiffrement.

7 Références normatives

- ISO 8732, *Banque – Gestion de clés*.
- Recommandation UIT-T X.209, *Spécification des règles de codage de base pour la notation de syntaxe abstraite numéro un (ASN.1)*.
- Recommandation UIT-T H.233, *Système de confidentialité pour les services audiovisuels*.

– Recommandation UIT-T H.221, *Structure de trame d'un canal à débit variable de 64 à 1920 kbit/s pour les téléservices audiovisuels.*

– Recommandation UIT-T H.230, *Signaux de contrôle et d'indication synchrones de la trame pour les systèmes audiovisuels*.

– Recommandation UIT-T H.242, *Procédures permettant d'établir des communications entre des terminaux audiovisuels à l'aide de canaux numériques dont le débit peut aller jusqu'à 2 Mbit/s.*

– Recommandation UIT-T X.509, *L'annuaire – Cadre d'authentification*.

Appendice I

Bibliographie

[1] DIFFIE (W.), HELLMAN (M.): New directions in cryptography, *IEEE Transactions IT-22*, 6, pages 644 à 654, (novembre 1976).

[2] RIVEST (R.L.), SHAMIR (A.) ADLEMAN (L.): A Method for Obtaining Digital Signatures and Public-Key Cryptosystems, *Communications of the ACM*, 21, 2, 120-126, (février 1978) .

[3] The MD4 Message Digest Algorithms, *RSA Data Security Inc*., Redwood City Californie 94065.