Union internationale des télécommunications

UIT-T

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

J.1015

(04/2020)

SÉRIE J: RÉSEAUX CÂBLÉS ET TRANSMISSION DES SIGNAUX RADIOPHONIQUES, TÉLÉVISUELS ET AUTRES SIGNAUX MULTIMÉDIAS

Accès conditionnel et protection – Solutions d'accès conditionnel et de gestion des droits numériques intégrées interchangeables

Interface commune intégrée pour les solutions CA/DRM interchangeables: Système de sécurité évoluée – Bloc d'échelle de clés

Recommandation UIT-T J.1015

Recommandation UIT-T J.1015

Interface commune intégrée pour les solutions CA/DRM interchangeables: Système de sécurité évoluée – Bloc d'échelle de clés

Résumé

La Recommandation UIT-T J.1015, qui fait partie d'une publication en plusieurs parties sur l'interface commune intégrée pour les solutions de type accès conditionnel/gestion des droits numériques (CA/DRM) interchangeables, porte sur le bloc d'échelle de clés du système de sécurité évoluée.

Cette Recommandation UIT-T, qui est une transposition de la norme [b-ETSI GS ECI 001-5-2] de l'ETSI, est le fruit d'une collaboration entre la CE 9 de l'UIT-T et l'ETSI ISG ECI.

Historique

Edition	Recommandation	Approbation	Commission d'études	ID unique*
1.0	UIT-T J.1015	23-04-2020	9	11.1002/1000/13576

Mots clés

Accès conditionnel, CA, gestion des droits numériques, DRM, échange.

* Pour accéder à la Recommandation, reporter cet URL http://handle.itu.int/ dans votre navigateur Web, suivi de l'identifiant unique, par exemple http://handle.itu.int/11.1002/1000/11830-en.

AVANT-PROPOS

L'Union internationale des télécommunications (UIT) est une institution spécialisée des Nations Unies dans le domaine des télécommunications et des technologies de l'information et de la communication (ICT). Le Secteur de la normalisation des télécommunications (UIT-T) 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 on 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.

À la date d'approbation de la présente Recommandation, l'UIT n'avait pas été avisée de l'existence d'une propriété intellectuelle protégée par des brevets à acquérir pour mettre en œuvre la présente Recommandation. Toutefois, comme il ne s'agit peut-être pas de renseignements les plus récents, il est vivement recommandé aux développeurs de consulter la base de données des brevets du TSB sous http://www.itu.int/ITU-T/ipr/.

TABLE DES MATIÈRES

Introduction

La présente Recommandation UIT-T[1], qui est une transposition de la norme [b-ETSI GS ECI 001-5-2] de l'ETSI, est le fruit d'une collaboration entre la CE 9 de l'UIT-T et l'ETSI ISG ECI.

L'objectif de la présente Recommandation est de faciliter l'interopérabilité et la concurrence en matière de services de communications électroniques et, en particulier, sur le marché de la radiodiffusion et des appareils audiovisuels. Toutefois, d'autres technologies sont disponibles et peuvent également être appropriées et efficaces en fonction de la situation dans les États Membres.

Un fournisseur de contenus chiffre ses contenus numériques et emploie un **Système de protection des contenus**[2] pour bloquer tout accès non autorisé à ceux-ci. Un consommateur utilise un **Récepteur de contenus** pour accéder aux contenus protégés. À cette fin, le **Récepteur de contenus** comprend un microprocesseur qui effectue une ou plusieurs opérations de déchiffrement de contenus. Un protocole de création de clés de chiffrement est employé pour sécuriser le transport des clés de déchiffrement de contenus entre le **Système de protection des contenus** et le microprocesseur. Les étapes du protocole appliqué au sein du microprocesseur sont appelées "échelle de clés" dans la présente Recommandation, qui contient les spécifications de l'échelle de clés correspondant au protocole de création de clés décrit dans le document [b-ROEL].

L'échelle de clés et le protocole peuvent aussi servir à sécuriser le transport de clés de chiffrement de contenus vers le microprocesseur. Ces clés sont nécessaires lorsque le microprocesseur doit rechiffrer des contenus, le microprocesseur pouvant effectuer à cette fin une ou plusieurs opérations de chiffrement de contenus. L'enregistrement de vidéos personnelles et l'exportation de contenus protégés vers un **Système de protection des contenus** différent sont des exemples classiques de cas dans lesquels les contenus doivent être rechiffrés. Dans l'ensemble de la présente Recommandation, les clés de chiffrement et de déchiffrement de contenus sont appelées **Mots de contrôle** (CW).

La présente Recommandation contient en outre les spécifications d'un mécanisme d'authentification. Celui-ci est étroitement lié à l'échelle de clés et peut servir à authentifier différentes entités, en particulier le microprocesseur.

L'échelle de clés et le mécanisme d'authentification décrits ci-après sont indépendants du **Système de protection des contenus** et du **Fournisseur de contenus**. Ce dernier peut donc utiliser n'importe quel **Système de protection des contenus** conforme aux spécifications; de même, le consommateur peut utiliser le **Récepteur de contenus** pour accéder aux contenus de n'importe quel fournisseur utilisant un **Système de protection de contenus** conforme.

Les mécanismes spécifiés dans la présente Recommandation prévoient qu'une **Autorité de certification** gère un certificat de clé publique pour chaque microprocesseur. Cette **Autorité de certification** a notamment pour rôle d'envoyer les informations sur les certifications et sur leur révocation aux **Fournisseurs de contenus** qui souhaitent employer l'échelle de clés et/ou le mécanisme d'authentification. Les **Fournisseurs de contenus** utilisent alors ces informations pour alimenter leur **Système de protection de contenus** conforme; comme nous allons le voir de manière détaillée dans le § 7, la clé publique du certificat d'un microprocesseur permet au **Système de protection de contenus** de produire les messages pertinents à passer en entrée à l'échelle de clés et au mécanisme d'authentification du microprocesseur.

---

1 Plusieurs domaines nécessitant des développements supplémentaires ont été identifiés dans l'Appendice I.

2 Dans le texte de la présente Recommandation, on utilise des caractères gras pour les termes dont la définition est propre au contexte de l'interface commune intégrée et peut différer de l'usage courant.

Recommandation UIT-T J.1015

Interface commune intégrée pour les solutions CA/DRM interchangeables: Système de sécurité évoluée – Bloc d'échelle de clés

1 Domaine d'application

La présente Recommandation contient les spécifications d'un bloc d'échelle de clés destiné au microprocesseur d'un **Récepteur de contenus**. Ce bloc d'échelle de clés se compose d'une échelle de clés, qui permet de sécuriser le transport de **Mots de contrôle** (CW) vers le microprocesseur, et d'un mécanisme d'authentification. La présente Recommandation contient en outre des spécifications sur la personnalisation de certains éléments d'un microprocesseur conforme.

La présente Recommandation est destinée aux fabricants de microprocesseurs.

2 Références

Les Recommandations UIT-T et autres références suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions de la présente Recommandation. Au moment de la publication, les éditions indiquées étaient en vigueur. Les Recommandations et autres références étant sujettes à révision, les utilisateurs de la présente Recommandation sont invités à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et autres références énumérées ci-dessous. Une liste des Recommandations UIT-T en vigueur est publiée périodiquement. 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.

[ISO/CEI 9797-1] ISO/CEI 9797-1:2011, *Technologies de l'information – Techniques de sécurité – Codes d'authentification de message (MAC) – Partie 1: Mécanismes utilisant un chiffrement par blocs*.

[IETF RFC 8017] IETF RFC 8017 (2016), *PKCS #1: RSA cryptography specifications version 2.2*.

[NIST FIPS 180-4] NIST FIPS PUB 180-4 (2015), *Secure Hash Standard (SHS)*.

[NIST FIPS 197] NIST FIPS PUB 197 (2001), *Specification for the Advanced Encryption Standard (AES)*.

[NIST SP 800-107] NIST SP 800-107 Revision 1 (2012), *Recommendation for applications using approved hash algorithms*.

3 Définitions

3.1 Termes définis ailleurs

Aucun.

3.2 Termes définis dans la présente Recommandation

La présente Recommandation définit les termes suivants:

3.2.1 Autorité de certification: partie chargée de la gestion des certificats de clé publique dans un **Écosystème** d'interface commune intégrée (**ECI**). Toutes les autres parties du système font confiance à l'autorité de certification pour effectuer les opérations associées aux certificats.

3.2.2 ID du microprocesseur: dans un **Écosystème ECI**, numéro non secret employé comme identifiant d'un microprocesseur.

3.2.3 Système de protection des contenus: dans un **Écosystème ECI**, système employant des techniques de chiffrement pour gérer l'accès à des contenus et des services. L'expression "système de protection des services" est aussi souvent employée dans le même sens. En général, il s'agit soit de systèmes d'accès conditionnel (CAS), soit de systèmes de gestion des droits numériques (DRM).

3.2.4 Fournisseur de contenus: dans un **Écosystème ECI**, partie qui distribue des contenus numériques à un **Récepteur de contenus**.

3.2.5 Récepteur de contenus: dans un **Écosystème ECI**, dispositif employé pour accéder à des contenus numériques. Un **Récepteur de contenus** contient un microprocesseur doté d'un **Désembrouilleur de contenus**.

3.2.6 Désembrouilleur de contenus: dans un **Écosystème ECI**, élément du microprocesseur capable de déchiffrer des contenus. Un désembrouilleur de contenus peut également être en mesure de chiffrer des contenus (aux fins du rechiffrement de contenus). Dans la présente Recommandation, le chiffrement/déchiffrement de contenus utilise un **Système de chiffrement symétrique**. Pour les contenus MPEG-2, le chiffrement et le déchiffrement de contenus sont également appelés respectivement embrouillage et désembrouillage.

3.2.7 Mot de contrôle: dans un **Écosystème ECI**, clé secrète utilisée pour chiffrer et déchiffrer des contenus. Dans les systèmes de gestion des droits numériques, un mot de contrôle est généralement appelé clé de contenu.

3.2.8 Fonction de hachage cryptographique: dans un **Écosystème ECI**, fonction de chiffrement sans clé qui reçoit en entrée des données de taille arbitraire, appelées message, et qui produit un bloc de données de sortie de taille fixe, appelé condensé du message. Dans la présente Recommandation, on suppose que la **fonction de hachage cryptographique** se comporte comme une fonction aléatoire et qu'elle est résistante à la seconde préimage.

3.2.9 Système de signature numérique: dans un **Écosystème ECI**, système de chiffrement asymétrique par clé qui est utilisé pour protéger l'authenticité des données. Un **Système de signature numérique** se compose d'un algorithme de création de clé, d'une opération de création de signature et d'une opération de vérification de signature. Les clés sont créées sous forme de paires (clé secrète/privée, clé publique). Les données sont signées à l'aide d'une clé secrète/privée et la clé publique correspondante sert à vérifier la signature. Le **Système de signature numérique** spécifié dans la présente Recommandation est utilisé pour protéger l'authenticité des messages comme défini dans le document [b-ROEL]; en particulier, le système n'est pas utilisé pour assurer la non-répudiation ou l'authentification de la source dans la présente Recommandation.

3.2.10 Écosystème ECI: opération commerciale fondée sur une autorité de confiance et sur plusieurs plates-formes et équipements de locaux d'abonnés conformes à l'interface **ECI** qui sont déployés sur le terrain.

3.2.11 Algorithme des codes d'authentification de message: dans un **Écosystème ECI**, algorithme de chiffrement symétrique par clé qui est utilisé pour protéger l'authenticité des données. Un **Algorithme des codes d'authentification de message** reçoit en entrée un message et une clé secrète et produit un bloc de données de sortie appelé MAC. L'**Algorithme des codes d'authentification de message** spécifié dans la présente Recommandation est utilisé pour lier sur le plan cryptographique un cryptogramme et ses données associées; en particulier, l'algorithme n'est pas utilisé pour assurer l'authentification de la source dans la présente Recommandation.

3.2.12 Système de chiffrement par clé publique: dans un **Écosystème ECI**, système de chiffrement asymétrique par clé qui est utilisé pour protéger la confidentialité des données. Un **Système de chiffrement par clé publique** se compose d'un algorithme de création de clé, d'une opération de chiffrement et d'une opération de déchiffrement. Les clés sont créées sous forme de paires (clé publique, clé secrète/privée). Les données sont chiffrées à l'aide d'une clé publique et sont récupérées à partir du cryptogramme à l'aide de la clé secrète/privée correspondante.

3.2.13 Système de chiffrement symétrique: dans un **Écosystème ECI**, système de chiffrement symétrique par clé qui est utilisé pour protéger la confidentialité des données. Un **Système de chiffrement symétrique** se compose d'un algorithme de création de clé, d'une opération de chiffrement et d'une opération de déchiffrement. Les opérations de chiffrement et de déchiffrement d'un **Système de chiffrement symétrique** utilisent la même clé secrète en entrée.

4 Abréviations et acronymes

La présente Recommandation utilise les abréviations et acronymes suivants:

AD1 données associées 1 (*associated data 1*)

AD2 données associées 2 (*associated data 2*)

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

AK clé d'authentification (*authentication key*)

CA accès conditionnel (*conditional access*)

CID identifiant de société (*company identifier*)

CISSA algorithme d'embrouillage commun de la TVIP orienté vers les logiciels (*common IPTV software-oriented scrambling algorithm*)

CPK clé publique de microprocesseur (*chipset public key*)

CPU unité centrale de traitement (*central processing unit*)

CSA algorithme d'embrouillage commun (*common scrambling algorithm*)

CSK clé secrète/privée de microprocesseur (*chipset secret/private key*)

CW mot de contrôle (*control word*)

DRM gestion des droits numériques (*digital rights management*)

DVB radiodiffusion vidéonumérique (*digital video broadcasting*)

ECB répertoire de codes électroniques (*electronic code book*)

ECI interface commune intégrée (*embedded common interface*)

ECP protection de contenu améliorée (*enhanced content protection*)

ID identifiant/identificateur (*identifier*)

LK clé de liaison (*link key*)

MAC code d'authentification de message (*message authentication code*)

MK clé MAC (*MAC key*)

OUI identifiant unique d'une organisation (*organizationally unique identifier*)

RSA Rivest Shamir Adleman

SHA algorithme de hachage sûr (*secure hash algorithm*)

SIM message d'entrée signé (*signed input message*)

SPK clé publique de l'expéditeur (*sender public key*)

SSK clé secrète/privée de l'expéditeur (*sender secret/private key*)

T étiquette (*tag*)

URI informations sur les règles d'utilisation (*usage rules information*)

5 Conventions

Dans le texte de la présente Recommandation, on utilise des caractères gras pour les termes dont la définition est propre au contexte de l'interface commune intégrée et peut différer de l'usage courant.

6 Identifiant et paire de clés principales du microprocesseur

Le présent paragraphe contient des spécifications sur certains éléments de personnalisation d'un microprocesseur conforme. Chaque microprocesseur est associé à une chaîne de bits permettant de l'identifier, appelée **ID du microprocesseur**, et à une paire de clés principales du microprocesseur.

Un **ID du microprocesseur** de 64 bits, unique à l'échelle mondiale, doit être attribué à chaque microprocesseur conforme. Si les bits de cet identifiant sont numérotés de 0 à 63 de gauche à droite, et si le ième bit ($0 \leq i \leq 63$) est désigné par b_i, alors ce sont les bits (b_0, b_1, b_2, b_3) qui doivent contenir l'identifiant de l'autorité d'enregistrement. Chaque valeur de l'identifiant doit être associée à au plus une autorité d'enregistrement. L'**ID du microprocesseur** doit aussi contenir l'identifiant du fabricant du microprocesseur. Dans l'identifiant d'un microprocesseur conforme, la valeur de l'identifiant de l'autorité d'enregistrement et celle de l'identifiant du fabricant du microprocesseur doivent permettre d'identifier strictement le fabricant ayant produit le microprocesseur. En outre, l'identifiant de l'autorité d'enregistrement figurant dans les bits (b_0, b_1, b_2, b_3) doit permettre de gérer l'attribution des identifiants de fabricant de microprocesseur susceptibles d'être utilisés conjointement avec cette valeur.

Si (b_0, b_1, b_2, b_3) = (0, 0, 0, 0), l'autorité d'enregistrement est celle de l'IEEE (https://standards.ieee.org/faqs/regauth.html), et l'on emploie l'OUI de 24 bits ou le CID de 24 bits pour déterminer l'identité du fabricant du microprocesseur. De plus, si (b_0, b_1, b_2, b_3) = (0, 0, 0, 0), alors l'OUI ou le CID doivent se trouver dans les bits (b_4, b_5, *…*, b_{27}), b_4 étant le bit ayant la valeur la plus élevée de l'octet 0 de l'OUI ou du CID (voir aussi [b-IEEE-OUI]) et b_{27} étant le bit ayant la valeur la moins élevée de l'octet 0 de l'OUI ou du CID.

Toutes les autres valeurs de l'identifiant de l'autorité d'enregistrement sont réservées à un futur emploi.

La paire de clés principales du microprocesseur est associée à un **Système de chiffrement par clé publique**. Elle se compose d'une clé secrète/privée (CSK) et d'une clé publique du microprocesseur (CPK). Comme indiqué plus en détail dans les § 7 et 8, la paire (CSK, CPK) est la paire de clés principales aussi bien pour l'échelle de clés que pour le mécanisme d'authentification. Le **Système de chiffrement par clé publique** et les représentations des clés CSK et CPK sont spécifiés dans le § 10.2.

Il est préférable que le microprocesseur conforme crée sa propre paire de clés (CSK, CPK) pour éviter la divulgation de la valeur de la clé CSK à une tierce partie dans le système. Dans ce cas, seul l'**ID du microprocesseur** est envoyé au microprocesseur au cours de la personnalisation de celui-ci, l'authenticité de l'**ID du microprocesseur** devant être protégée au cours de ce processus. Si le microprocesseur ne crée pas sa propre paire de clés, l'authenticité du triplet (**ID du microprocesseur**, CSK, CPK) et la confidentialité de la clé CSK doivent être protégées pendant que le triplet est envoyé au microprocesseur.

Le nombre aléatoire employé dans l'algorithme de création de la paire de clés (CSK, CPK) doit avoir une entropie d'au moins 128 bits.

Un microprocesseur conforme doit stocker en permanence le triplet (**ID du microprocesseur**, CSK, CPK) et doit disposer de procédures permettant de protéger la confidentialité de la clé CSK et l'intégrité du triplet stocké.

L'unité centrale de traitement (CPU) du **Récepteur de contenus** doit avoir un accès en lecture à l'**ID du microprocesseur** et à la clé CPK stockée. Cet accès permet à une **Autorité de certification** d'utiliser les informations envoyées pour créer un certificat de clé publique destiné au

microprocesseur. Par ailleurs, l'**ID du microprocesseur** envoyé permet à un **Fournisseur de contenus** de vérifier l'identité du microprocesseur ainsi que son certificat.

L'**Autorité de certification** doit conserver la paire (**ID du microprocesseur**, CPK) pour chacun des microprocesseurs qu'elle gère. Il est possible que plusieurs **Autorités de certification** interviennent dans le système. L'authenticité de la paire (**ID du microprocesseur**, CPK) doit être protégée pendant que cette paire est envoyée à l'**Autorité** ou aux **Autorités de certification**.

7 Échelle de clés

7.1 Aperçu général

Le présent paragraphe contient une description de l'architecture fonctionnelle de l'échelle de clés. Dans l'ensemble de la présente Recommandation, le bloc chargé, au sein du microprocesseur, d'exécuter cette échelle est appelé bloc d'échelle de clés. On trouvera à la Figure 7-1 une illustration de l'échelle de clés. Comme nous l'avons indiqué dans le § 6, et comme le montre la figure ci-dessus, le microprocesseur est personnalisé par l'ajout d'un **ID du microprocesseur** et d'une clé CSK.

Le bloc d'échelle de clés produit notamment en sortie un **Mot de contrôle** désigné par CW, qui sert à déchiffrer ou à chiffrer des contenus. Il produit aussi une chaîne de bits qui se compose des informations sur les règles d'utilisation d'un mot de contrôle (CW-URI) (on trouvera dans le § 7.3.1 les spécifications de la chaîne CW-URI et les règles d'utilisation connexes). Le mot CW et la chaîne CW-URI sont passés en entrée au **Désembrouilleur de contenus** (qui n'apparaît pas dans la Figure 7-1).

- Le bloc d'échelle de clés et le **Désembrouilleur de contenus** doivent s'exécuter dans un seul et même microprocesseur au silicium.
- Si le **Désembrouilleur de contenus** dispose d'une interface avec un processeur du **Récepteur de contenus** qui permet à ce processeur de passer des CW non chiffrés vers le **Désembrouilleur** (c'est-à-dire en contournant le bloc d'échelle de clés), il doit être possible de désactiver cette fonctionnalité de manière permanente.
- Si l'échelle de clés calcule un CW, seul le bloc d'échelle de clés et le **Désembrouilleur de contenus** peuvent y avoir accès.
- L'authenticité de la paire (CW, CW-URI) et la confidentialité du CW doivent être protégées pendant que ces éléments sont envoyés du bloc d'échelle de clés au **Désembrouilleur de contenus**.

Le bloc d'échelle de clés doit disposer d'une interface avec un processeur du **Récepteur de contenus**. Il peut s'agir par exemple d'un processeur de sécurité, ou de l'unité centrale de traitement du **Récepteur de contenus**. Comme nous l'avons indiqué dans le § 6 et comme le montre la Figure 7-1, ce processeur a accès en lecture à l'**ID du microprocesseur**, ce qui permet à un **Fournisseur de contenus** de reconnaître le microprocesseur et d'obtenir le certificat de clé publique correspondant, qui contient la clé CPK de l'**Autorité de certification**. La valeur de la clé CPK doit être connue pour pouvoir calculer l'un des messages d'entrée de l'échelle de clés, comme décrit dans le § 7.2.

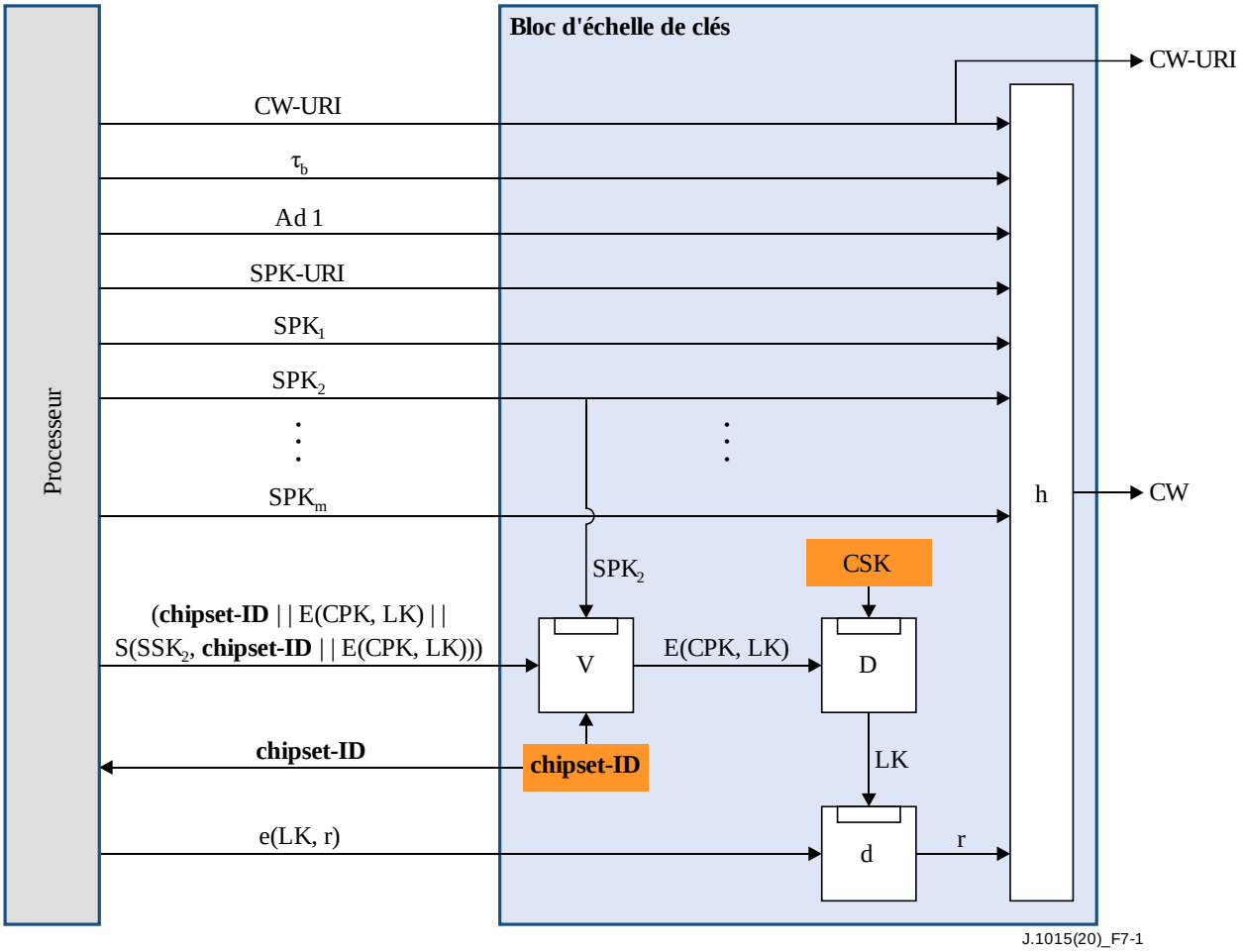

Figure 7-1 – Échelle de clés

Nous avons dit dans le § 6 que la paire (CSK, CPK) était associée à un **Système de chiffrement par clé publique**. Les opérations de chiffrement et déchiffrement correspondantes sont respectivement désignées par les lettres E et D. Ces lettres représentent des opérations de chiffrement par clé prenant chacune deux informations en entrée: une clé et un message. Dans la présente Recommandation, nous prenons pour hypothèse que la clé est entrée en premier dans une opération ou un algorithme de chiffrement par clé.

Exemple: Le chiffrement d'un message *M* par l'opération E et la clé CPK s'écrit de la manière suivante: E(CPK, *M*).

Le mécanisme décrit dans la présente Recommandation utilise aussi un **Système de signature numérique**; on désigne respectivement par S et V les opérations de création et de vérification de la signature. Toute paire de clés de ce système est associée à un expéditeur et se compose d'une clé secrète/privée de l'expéditeur (SSK) et d'une clé publique de l'expéditeur (SPK). Dans la présente Recommandation, nous prenons pour hypothèse que l'expéditeur est un **Système de protection de contenus**. Comme le montre la Figure 7-1, un certain nombre de clés SPK différentes, notées SPK_1, SPK_2, ..., SPK_m ($m \geq 1$) sont entrées dans le bloc d'échelle de clés. Celui-ci prend en charge des échelles de clés de toute valeur m telle que $1 \leq m \leq 16$. Dans la pratique, chaque paire de clés (SSK_i, SPK_i) telle que $1 \leq i \leq m$ est généralement associée à un seul **Système de protection des contenus**; néanmoins, elle peut aussi être partagée entre plusieurs système.

La chaîne de bits CW-URI est associée aux clés SPK_1, SPK_2, ..., SPK_m. Cette entrée du bloc d'échelle de clés fournit les informations sur les règles d'utilisation de SPK_1, SPK_2, ..., SPK_m. La chaîne SPK-URI et les règles d'utilisation connexes sont spécifiées dans le § 7.3.2. Comme le montre la Figure 7-1, on emploie l'une des clés SPK et l'opération de vérification V pour vérifier la signature

du message en entrée (**chipset-ID** || E(CPK, LK) || S(SSK_i, **chipset-ID** || E(CPK, LK))). Ce message d'entrée signé (SIM) est également désigné par le mot SIM_{KL} dans le texte ci-après. La Figure 7-1 repose sur l'hypothèse que i = 2 et que la chaîne SPK-URI et les règles d'utilisation autorisent l'emploi de la clé SPK_2 pour vérifier la signature.

L'échelle de clés dispose aussi d'un **Système de chiffrement symétrique**. Les opérations de chiffrement et de déchiffrement de ce système sont respectivement désignées par e et d. L'échelle de clés utilise une clé de liaison (LK) à titre de clé dans ce système, ainsi qu'un nombre aléatoire *r* (ou toute autre clé LK du type décrit dans le § 7.4) à titre de message. Le nombre aléatoire *r* est représenté par une chaîne de bits dont la longueur est 128 bits.

Enfin, le bloc d'échelle de clés exécute une fonction h fondée sur une **Fonction de hachage cryptographique** (on trouvera les spécifications de h dans le § 10.4).

Comme le montre la Figure 7-1, les deux autres entrées du bloc d'échelle de clés sont τ_b et données associées 1 (AD1). Ces deux entrées sont représentées par des chaînes de bits:

- τ_b a une longueur de 8 bits. Son emploi est décrit dans le § 7.5.
- AD1 a une longueur de 256 bits. Les spécifications de ses contenus n'entrent pas dans le cadre de la présente Recommandation, dans laquelle nous prenons pour hypothèse que le bloc d'échelle de clés utilise AD1 uniquement à titre d'entrée de h. Il peut passer AD1 (ou une partie d'AD1) au moment où il passe la chaîne CW-URI et le mot CW au **Désembrouilleur de contenus**.

Dans la présente Recommandation, nous prenons pour hypothèse que les clés symétriques, les cryptogrammes et les signatures sont représentés par des chaînes de bits. Leur longueur est définie dans le § 10, qui définit également la représentation des clés asymétriques.

7.2 Calculs de l'échelle de clés

L'expéditeur associé à la paire de clés (SSK_i, SPK_i) pour certains *i* tels que $1 \leq i \leq m$ crée le message d'entrée signé SIM_{KL}, qui peut par exemple être le message suivant:

(**chipset-ID** || E(CPK, LK) || S(SSK_i, **chipset-ID** || E(CPK, LK)))

en suivant les étapes ci-dessous:

Calcul de SIM_{KL} (expéditeur)

1) Créer une clé LK.
2) Calculer le cryptogramme E(CPK, LK).
3) Concaténer **chipset-ID** et E(CPK, LK); la chaîne de bits obtenue est désignée par (**chipset-ID** || E(CPK, LK)).
4) Signer la chaîne de bits (**chipset-ID** || E(CPK, LK)) au moyen de la clé SSK_i; la signature est désignée par S(SSK_i, **chipset-ID** || E(CPK, LK))).
5) Ajouter cette signature en suffixe à la chaîne de bits (**chipset-ID** || E(CPK, LK)).

Après avoir reçu le message SIM_{KL} et la clé publique de l'expéditeur SPK_i, le bloc d'échelle de clés passe aux étapes suivantes pour récupérer LK (dans la Figure 7-1, nous avons pris pour hypothèse que les trois premières étapes étaient effectuées par l'opération de vérification V).

Calcul de LK (bloc d'échelle de clés)

1) Vérifier si l'**ID du microprocesseur** reçu est égal à celui qui est stocké. Si ces deux valeurs sont différentes, le bloc d'échelle de clés arrête les calculs.
2) Vérifier si la chaîne SPK-URI et les règles d'utilisation définies dans le § 7.3.2 autorisent l'opération V à employer la clé SPK_i pour vérifier la signature. Si tel n'est pas le cas, le bloc d'échelle de clés arrête les calculs.

3) Utiliser le message SIMKL et la clé SPK_i pour vérifier la signature. Si la signature n'est pas valable, le bloc d'échelle de clés arrête les calculs.

4) Calculer LK = D(CSK, E(CPK, LK)).

Le bloc d'échelle de clés utilise ensuite LK pour traiter le message d'entrée e(LK, *r*) (voir également la Figure 7-1). L'expéditeur peut créer ce message selon les étapes suivantes:

Calcul de e(LK, *r*) (expéditeur)

1) Créer une chaîne de bits aléatoire *r*.

2) Calculer e(LK, *r*).

Après avoir reçu e(LK, *r*) et avoir calculé LK, le bloc d'échelle de clés calcule *r* selon les étapes suivantes (voir aussi la Figure 7-1):

Calcul de *r* (bloc d'échelle de clés)

1) *r* = d(LK, e(LK, *r*)).

Le bloc d'échelle de clés utilise ensuite la fonction h pour calculer le CW. Comme le montre la Figure 7-1, les entrées de h sont CW-URI, τ_b, AD1, SPK-URI, SPK_1, SPK_2, ..., SPK_m, et *r*. L'exécution de l'échelle de clés doit garantir que la clé publique utilisée pour vérifier l'authenticité de SIM_{KL} (ou plus précisément, la SIM_{KL} contenant la clé LK associée au nombre aléatoire *r*) fait partie des entrées de la clé SPK envoyées vers h lorsque CW est déduit de *r*. Puis le bloc d'échelle de clés doit passer la chaîne CW-URI et le mot CW au **Désembrouilleur de contenus**.

7.3 Informations sur les règles d'utilisation

7.3.1 Informations sur les règles d'utilisation d'un mot de contrôle

La chaîne CW-URI doit avoir une longueur de 64 bits, qui sont numérotés de 0 à 63 de gauche à droite. La valeur de CW-URI définit l'utilisation autorisée de CW (voir aussi le Tableau 7-1) selon la règle d'utilisation suivante: si la valeur d'un bit est 1, l'utilisation spécifiée est autorisée, sinon elle ne l'est pas. Le **Désembrouilleur de contenus** doit respecter cette règle d'utilisation. Il doit ignorer:

i) la valeur des bits qui sont réservés à un futur emploi; et

ii) la valeur des bits correspondant à une mise en œuvre que le **Désembrouilleur de contenus** ne prend pas en charge.

Tableau 7-1 – Définition des informations sur les règles d'utilisation d'un mot de contrôle

Numéro de bit	Description
0	Chiffrement
1	Déchiffrement
2 … 7	Réservé à un futur emploi
8	Radiodiffusion vidéonumérique (DVB), version 2 de l'algorithme d'embrouillage commun (CSA2) [b-ETSI TS 100 289]
9	DVB CSA3 [b-ETSI TS 100 289]
10 … 15	Réservé à un futur emploi
16	DVB, version 1 de l'algorithme d'embrouillage commun de la TVIP orienté vers les logiciels (CISSA) [b-ETSI TS 103 127]
17 … 23	Réservé à un futur emploi

Tableau 7-1 – Définition des informations sur les règles d'utilisation d'un mot de contrôle

Numéro de bit	Description
24	Algorithme d'embrouillage/désembrouillage commun de l'Advanced Television Systems Committee [b-ATSC A/70-1]
25 … 31	Réservé à un futur emploi
32	Cryptage commun des fichiers – système de protection cenc [b-ISO/CEI 23001-7]
33	Cryptage commun des fichiers – système de protection cbc1 [b-ISO/CEI 23001-7]
34	Cryptage commun des fichiers – système de protection cens [b-ISO/CEI 23001-7]
35	Cryptage commun des fichiers – système de protection cbcs [b-ISO/CEI 23001-7]
36 … 39	Réservé à un futur emploi
40	ChinaDRM – mode enchaînement de blocs de chiffrement [b-GY/T 277-2014]
41	ChinaDRM – mode compteur [b-GY/T 277-2014]
42 … 63	Réservé à un futur emploi

7.3.2 Informations sur les règles d'utilisation d'une clé publique de l'expéditeur

La chaîne SPK-URI doit avoir une longueur de 64 bits, qui sont numérotés de 0 à 63 de gauche à droite. Cette chaîne est définie dans le Tableau 7-2. En particulier, si SPK_1, SPK_2, ..., et SPK_m sont passés en entrée dans l'échelle de clés en même temps que SPK-URI (comme décrit dans le Tableau 7-2), les bits 0, 1, …, m-1 et les bits 16, 17, .., 16+m-1 de SPK-URI sont employés pour définir deux sous-ensembles de {SPK_1, SPK_2, ..., SPK_m}, SPK_i (i = 1, 2, …, m) étant un élément du premier sous-ensemble si et seulement si la valeur du bit i-1 est égale à 1, et SPK_i étant un élément du second sous-ensemble si et seulement si la valeur du bit i+15 est égale à 1. Dans le texte ci-après, ces deux sous-ensembles sont respectivement désignés par S_1 et S_2. En suivant cette notation, l'échelle de clés doit appliquer la règle d'utilisation suivante:

Règle d'utilisation 1 de la clé SPK: dans l'échelle de clés, l'opération de vérification V est autorisée à utiliser la clé SPK_i pour vérifier la signature de SIM_{KL} si et seulement si $SPK_i \in S_1$.

La règle d'utilisation 1 doit être respectée par le bloc d'échelle de clés.

L'ensemble S_2 est utilisé si les contenus doivent être rechiffrés. Dans ce cas, deux ensembles de valeurs sont nécessaires en entrée, comme le montre la Figure 7-1: l'un est associé au mot CW servant au déchiffrement des contenus, et l'autre est associé au mot CW servant au chiffrement. La règle d'utilisation permettant de mettre ces deux ensembles en relation est alors la suivante:

Règle d'utilisation 2 de la clé SPK: dans l'échelle de clés, l'opération de vérification V est autorisée à utiliser la clé SPK_i pour vérifier la signature de SIM_{KL} dans le mot CW si et seulement si $SPK_i \in S_2$ et qu'elle est associée au mot CW servant à déchiffrer les contenus.

Si le **Désembrouilleur de contenus** prend en charge le rechiffrement de contenus, le bloc d'échelle de clés doit appliquer la règle d'utilisation 2 de la clé SPK. Si le bloc n'applique pas cette règle, c'est un autre élément du microprocesseur qui devra le faire, et l'exécution devra garantir que la valeur de la chaîne SPK-URI passée en entrée de la fonction h est égale à la valeur de la chaîne SPK-URI utilisée pour appliquer la règle d'utilisation 2 de la clé SPK.

Tableau 7-2 – Définition des informations sur les règles d'utilisation d'une clé publique de l'expéditeur

Numéro de bit	Description
0	spk1_in_set1 *(SPK 1 dans l'ensemble 1)*
1	spk2_in_set1
2	spk3_in_set1
…	…
15	spk16_in_set1
16	spk1_in_set2
17	spk2_in_set2
18	spk3_in_set2
…	…
31	spk16_in_set2
32 … 63	Réservé à un futur emploi

7.4 Couches de clés supplémentaires

7.4.1 Aperçu général

La spécification indiquée dans les § 7.1 et 7.2 repose sur l'hypothèse qu'une couche de clés donnée de l'échelle de clés est associée à certaines clés LK. L'échelle de clés doit aussi prendre en charge des couches de clés supplémentaires pour proposer des clés LK, comme le montre la Figure 7-2 (dans laquelle apparaissent uniquement les éléments de l'échelle de clés pertinents au regard du présent paragraphe).

Comme dans la Figure 7-2, soit t le nombre de clés LK dans l'échelle de clés. Le bloc d'échelle de clés doit prendre en charge des échelles de clés pour toutes les valeurs de t telles que $1 \leq t \leq 24$. À noter que $t = 1$ dans le système décrit dans les § 7.1 et 7.2.

7.4.2 Calculs de l'échelle de clés

L'expéditeur peut créer les messages d'entrée de t destinés au bloc d'échelle de clés selon les étapes suivantes:

Calcul de $e(LK_1, LK_2)$, $e(LK_2, LK_3)$, …, $e(LK_{t-1}, LK_t)$, et $e(LK_t, r)$ (expéditeur)

1) Créer des clés LK_i pour $i = 1, 2, \ldots, t$ et une chaîne de bits aléatoires r.
2) Calculer $e(LK_{i-1}, LK_i)$ pour i = 2, 3, ..., t.
3) Calculer $e(LK_t, r)$.

Après avoir reçu $e(LK_1, LK_2)$, $e(LK_2, LK_3)$, …, $e(LK_{t-1}, LK_t)$, et $e(LK_t, r)$, le bloc d'échelle de clés doit calculer r selon les étapes suivantes (voir aussi la Figure 7-2):

Calcul de r (bloc d'échelle de clés)

1) Calculer $LK_i = d(LK_{i-1}, e(LK_{i-1}, LK_i))$ pour i = 2, 3, ..., t.
2) Calculer $r = d(LK_t, e(LK_t, r))$.

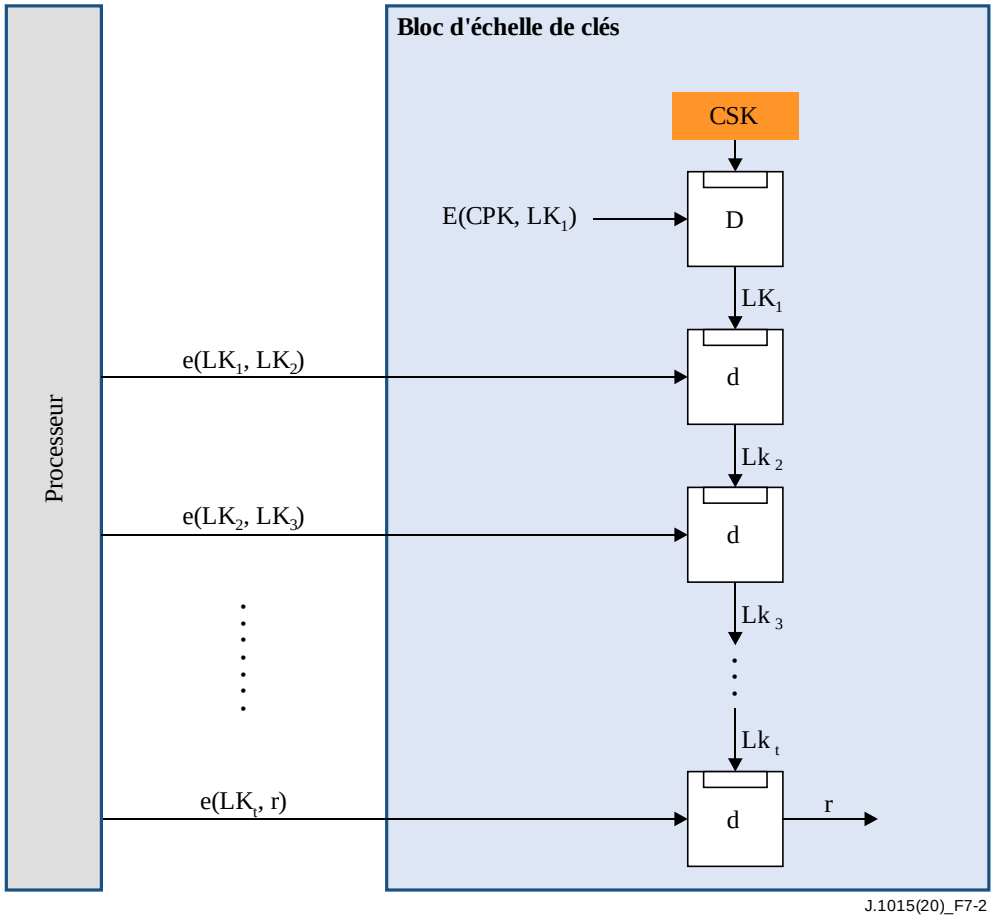

Figure 7-2 – Couches de clés supplémentaires

7.5 Données associées 2

L'expéditeur peut aussi choisir d'envoyer à l'échelle de clés des données associées 2 (AD2), en même temps que le cryptogramme e(LK_t, r). La valeur de la chaîne de bits τ_b (présentée dans le § 7.1) signale la présence de données AD2. Comme nous l'avons indiqué dans le § 7.1, τ_b a une longueur de 8 bits. Dans le texte ci-après, τ désigne la représentation de τ_b par des entiers, le bit le moins élevé de τ correspondant au bit le plus à droite de τ_b. L'échelle de clés doit prendre en charge toutes les valeurs de $\tau \in \{0, 64, 96, 128\}$. Si $\tau = 0$ il n'y a pas de données AD2; on emploie alors le système présenté dans les § 7.1 à 7.4. Dans la description ci-après, nous prenons pour hypothèse que $\tau \neq 0$, ce qui signifie que des données AD2 sont présentes. Le bloc d'échelle de clés doit vérifier si la valeur reçue de $\tau \in \{64, 96, 128\}$; dans le cas contraire, le bloc d'échelle de clé arrête les calculs.

Dans la présente Recommandation, nous prenons aussi pour hypothèse que les données AD2 sont représentées par une chaîne de bits. Si Len désigne la longueur d'AD2 en bits, le bloc d'échelle de clés doit prendre en charge toutes les valeurs de Len telles que $1 \leq \text{Len} \leq 256$.

Soit Len_b la représentation binaire de l'entier Len – 1 dans laquelle le bit le plus à droite de Len_b correspond au bit le moins élevé de Len – 1; Len_b a une longueur de 8 bits.

La chaîne de bits AD2 doit être liée par un chiffrement au cryptogramme e(LK_t, r) correspondant pendant son envoi vers l'échelle de clés. À cette fin, l'expéditeur et le bloc d'échelle de clés doivent utiliser un **Algorithme des codes d'authentification de message** appelé mac. Les entrées de cet algorithme sont la clé MCA secrète (MK), le message Len_b || e(LK_t, r) || AD2, et τ. La sortie de l'algorithme mac est une étiquette (T) de longueur τ bits. L'algorithme MAC est spécifié dans le § 10.5.

Soit 00…0 une chaîne de 128 bits composée uniquement de zéros; l'expéditeur peut suivre les étapes suivantes pour créer le message d'entrée Len_b || e(LK_t, *r*) || AD2 || T destiné à l'échelle de clés (comme illustré dans la Figure 7-3):

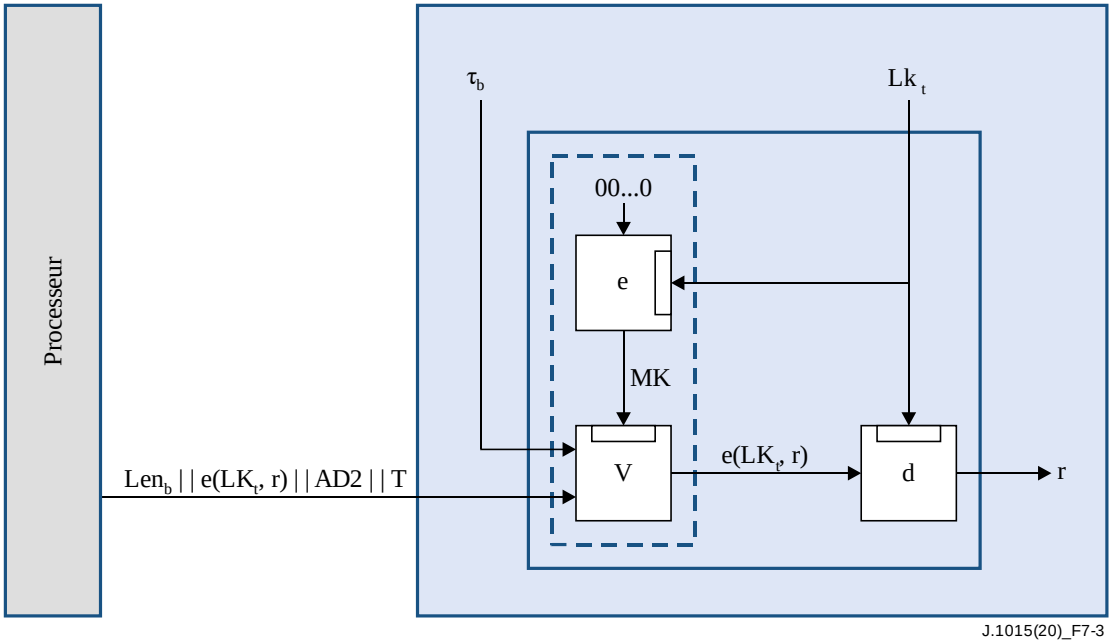

Figure 7-3 – Données associées 2

Calcul de Len_b || e(LK_t, *r*) || AD2 || T (expéditeur)

1) Calculer MK = e(LK_t, 00…0).
2) Ajouter le préfixe Len_b à la chaîne de bits e(LK_t, *r*) || AD2.
3) Calculer T = mac(MK, Len_b || e(LK_t, *r*) || AD2, τ).
4) Ajouter le suffixe T à la chaîne de bits Len_b || e(LK_t, *r*) || AD2.

Le message Len_b || e(LK_t, *r*) || AD2 || T remplace le message e(LK_t, *r*) qui est employé si $\tau = 0$. Après avoir reçu Len_b || e(LK_t, *r*) || AD2 || T, le bloc d'échelle de clés calcule e(LK_t, *r*) selon les étapes suivantes (les étapes 2-4 sont désignées par V dans la Figure 7-3):

Calcul de e(LK_t, *r*) (bloc d'échelle de clés)

1) Calculer MK = e(LK_t, 00…0).
2) Calculer T = mac(MK, Len_b || e(LK_t, *r*) || AD2, τ).
3) Vérifier si la valeur T reçue est égale à la valeur T calculée. Si tel n'est pas le cas, le bloc d'échelle de clés arrête les calculs.
4) Extraire e(LK_t, *r*) du message reçu.

Les spécifications des contenus de la chaîne AD2 n'entrent pas dans le cadre de la présente Recommandation, dans laquelle nous prenons pour hypothèse que le bloc d'échelle de clés utilise les données AD2 uniquement en tant qu'entrée de l'algorithme MAC. Il peut passer AD2 (ou une partie d'AD2) au moment où il passe CW-URI et CW au **Désembrouilleur de contenus**.

Si $\tau \neq 0$, l'exécution de l'échelle de clés doit garantir que la valeur de τ_b passée en même temps que Len_b || e(LK_t, *r*) || AD2 en entrée de l'algorithme MAC soit aussi passée en entrée de h quand CW est déduit de *r*.

Seul le bloc d'échelle de clés doit avoir accès aux éléments CSK, LK_i ($1 \leq i \leq t$), MK et *r* lorsqu'ils ne sont pas chiffrés.

8 Mécanisme d'authentification

8.1 Aperçu général

Le présent paragraphe traite de l'architecture fonctionnelle du mécanisme d'authentification dont dispose le bloc d'échelle de clés. Ce mécanisme, qui est illustré dans la Figure 8-1, est étroitement lié à l'échelle de clés décrite dans le § 7. Il utilise en particulier les mêmes opérations de vérification V et D, la même fonction h, le même **ID du microprocesseur** et la même clé CSK. En revanche, il ne fait pas appel à des clés LK, et ni ce mécanisme ni la fonction h n'utilisent en entrée les éléments CW-URI, SPK-URI et τ_b. En outre, la sortie de la fonction h n'est pas un CW destiné au déchiffrement ou au chiffrement de contenus (au demeurant, le mécanisme d'authentification ne s'interface pas directement avec le **Désembrouilleur de contenus**), mais une clé d'authentification (AK), qui fait partie du **Système de chiffrement symétrique** également employé par l'échelle de clés. Comme le montre la Figure 8-1, seule son opération de déchiffrement d est employée dans l'exécution du mécanisme d'authentification.

Le message d'entrée (**chipset-ID** || E(CPK, *r*) || S(SSK$_i$, **chipset-ID** || E(CPK, *r*))) est également appelé SIM$_{AUTH}$ dans la suite du présent document.

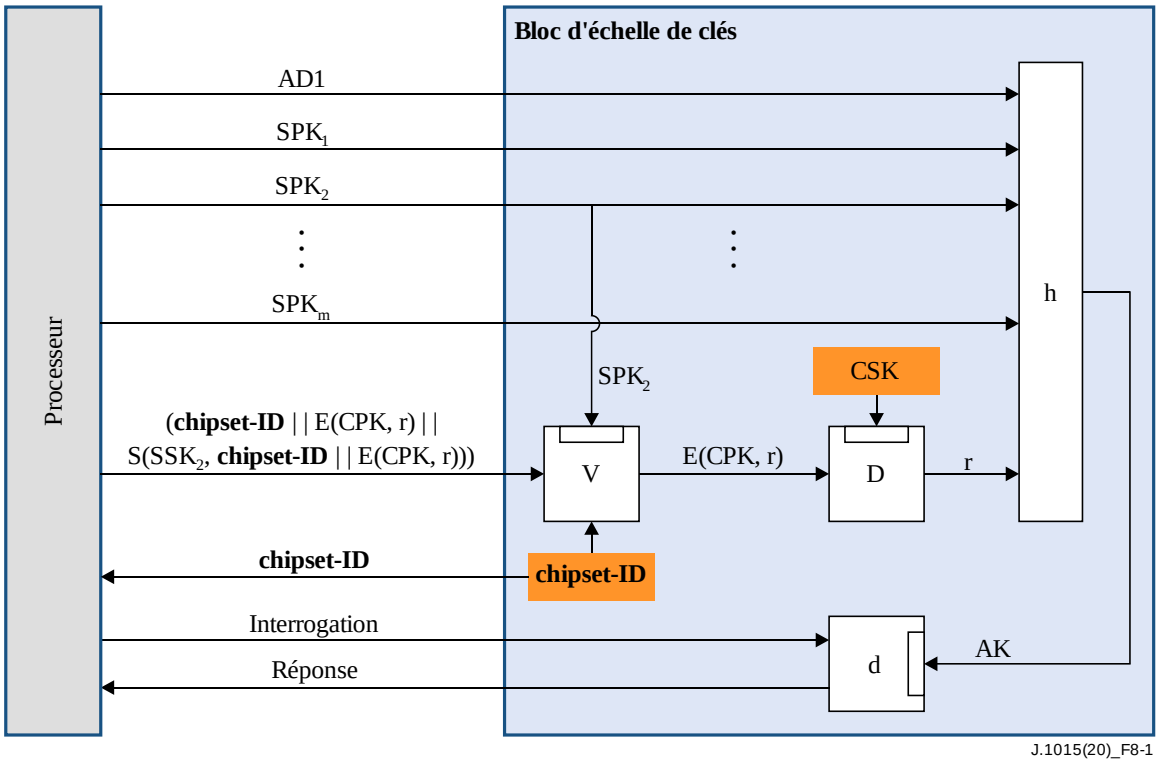

Figure 8-1 – Mécanisme d'authentification

8.2 Calculs du mécanisme d'authentification

L'expéditeur associé à la paire de clés (SSK$_i$, SPK$_i$) pour certains *i* tels que $1 \leq i \leq m$ crée le message SIM$_{AUTH}$, qui peut par exemple être le message suivant:

(**chipset-ID** || E(CPK, *r*) || S(SSK$_i$, **chipset-ID** || E(CPK, *r*))),

en suivant les étapes ci-dessous:

Calcul de SIM$_{AUTH}$ (expéditeur)

1) Créer une chaîne de bits aléatoires *r*.

2) Calculer le cryptogramme E(CPK, *r*).

3) Concaténer **chipset-ID** et E(CPK, *r*); la chaîne de bits obtenue est désignée par (**chipset-ID** || E(CPK, *r*)).

4) Signer le message (**chipset-ID** || E(CPK, *r*)) au moyen de la clé SSK_i; la signature est désignée par S(SSK_i, **chipset-ID** || E(CPK, *r*)).

5) Ajouter cette signature en suffixe à la chaîne de bits (**chipset-ID** || E(CPK, *r*)).

Après avoir reçu le message SIM_{AUTH} et la clé SPK_i, le bloc d'échelle de clés passe aux étapes suivantes pour récupérer *r* (dans la Figure 8-1, les deux premières étapes sont désignées par V).

Calcul de *r* (bloc d'échelle de clés)

1) Vérifier si l'**ID du microprocesseur** reçu est égal à celui qui est stocké. Si ces deux valeurs sont différentes, le bloc d'échelle de clés arrête les calculs.

2) Utiliser le message SIM_{AUTH} reçu et la clé SPK_i pour vérifier la signature. Si la signature n'est pas valable, le bloc d'échelle de clés arrête les calculs.

3) Calculer *r* = D(CSK, E(CPK, *r*)).

Si le mécanisme d'authentification est employé, la procédure de vérification V est autorisée à utiliser n'importe quelle clé SPK_i telle que $1 \leq i \leq m$ pour vérifier la signature SIM_{AUTH} (dans la Figure 8-1, $i = 2$ par hypothèse).

On passe ensuite AD1, SPK_1, SPK_2, ..., SPK_m et *r* en entrée à la fonction h, qui produit en sortie la clé AK. L'exécution du mécanisme d'authentification doit garantir que la clé publique employée pour vérifier l'authenticité de SIM_{AUTH} (ou plus précisément, la SIM_{AUTH} contenant le nombre aléatoire *r*) fait partie des entrées de la clé SPK envoyées vers h lorsque la clé AK est déduite de *r*.

L'expéditeur peut ensuite créer un message d'entrée appelé une interrogation. Une fois que le mécanisme d'authentification a reçu l'interrogation du processeur, il calcule la réponse selon l'étape suivante:

Calcul de la réponse (bloc d'échelle de clés)

1) Calculer réponse = d(AK, interrogation).

Comme le montre la Figure 8-1, le mécanisme d'authentification doit envoyer une réponse au processeur.

Seul le bloc d'échelle de clés peut accéder à une clé AK non chiffrée.

9 Primitives de conversion de données

9.1 BS2OSP

La primitive BS2OSP de conversion d'une chaîne de bits en chaîne d'octets sert à définir le **Système de chiffrement par clé publique** décrit dans le § 10.2 et le **Système de signature numérique** décrit dans le § 10.3.

Fonction:	BS2OSP(x)	
Entrée:	x	une chaîne de bits de longueur $8j$ ($j \geq 1$).
Sortie:	X	une chaîne d'octets de longueur *j*.

Étapes:

1) Soit x_i un bit tel que $0 \leq i \leq 8j\text{-}1$, et $x = x_0\, x_1 \ldots x_{8j-1}$.

2) Soit X_i l'octet défini par $X_i = x_{8i+1} \ldots x_{8i+7}$ avec $0 \leq i \leq j\text{-}1$.

3) La chaîne d'octets en sortie $X = X_0\, X_1 \ldots X_{j-1}$.

9.2 OS2BSP

La primitive OS2BSP de conversion d'une chaîne d'octets en chaîne de bits sert à définir la primitive I2BSP décrite dans le § 9.3 et le **Système de chiffrement par clé publique** décrit dans le § 10.2.

Fonction:	OS2BSP(X)	
Entrée:	X	une chaîne d'octets de longueur j ($j \geq 1$).
Sortie:	x	une chaîne de bits de longueur $8j$.

Étapes:

1) Soit X_i un octet tel que $0 \leq i \leq j\text{-}1$, et soit $X = X_0\, X_1 \ldots X_{j-1}$.
2) Soit x_i un bit tel que $0 \leq i \leq 8j\text{-}1$, ces bits étant définis par $X_i = x_{8i+1} \ldots x_{8i+7}$ avec $0 \leq i \leq j\text{-}1$.
3) La chaîne de bits en sortie $x = x_0\, x_1 \ldots x_{8j-1}$.

9.3 I2BSP

La primitive I2BSP de conversion d'un entier en chaîne de bits sert à définir la fonction h décrite dans le § 10.4.

Fonction:	I2BSP(x)	
Entrée:	x	un entier de 2 048 bits.
Sortie:	une chaîne de bits de longueur 2 048.	

Étape: Si la primitive I2OSP(x, xLen) de conversion d'un entier en chaîne d'octets est définie de la manière spécifiée dans la spécification [IETF RFC 8017], alors I2BSP(x) = OS2BSP(I2OSP(x, 256)).

10 Opérations de chiffrement

10.1 Système de chiffrement symétrique

Le **Système de chiffrement symétrique** employé par le bloc d'échelle de clés doit être le système AES-128 (norme de chiffrement évoluée-128) [NIST FIPS 197] exploité en mode ECB (répertoire de codes électroniques).

10.2 Système de chiffrement par clé publique

Le **Système de chiffrement par clé publique** doit être le système RSAES-PKCS1-v1_5 décrit dans la spécification [IETF RFC 8017]. Ce système se compose d'une opération de chiffrement et d'une opération de déchiffrement. Comme nous l'avons mentionné dans les § 7 et 8, seule l'opération de déchiffrement est effectuée dans le bloc d'échelle de clés; l'opération de chiffrement, pour sa part, est effectuée dans le système de l'expéditeur.

La valeur de la clé CPK doit être égale à la valeur de la clé publique RSA (Rivest Shamir Adleman) du destinataire, désignée par (n, e) dans la spécification [IETF RFC 8017]. Dans la présente Recommandation, la longueur de n est de 2 048 bits; en conséquence, la longueur de n, désignée par k dans la spécification [IETF RFC 8017], est de 256 octets. La valeur de la clé CSK doit être égale à la valeur de la clé privée RSA du destinataire, désignée par K dans la spécification [IETF RFC 8017].

Le message à chiffrer est désigné par M dans la spécification [IETF RFC 8017]; il est représenté par une chaîne d'octets. Dans la présente Recommandation, M est défini par M = BS2OSP(LK) dans le contexte de l'échelle de clés, et M = BS2OSP(r) dans le contexte du mécanisme d'authentification. M a donc une longueur de 16 octets dans la présente Recommandation.

La clé CPK et le message M sont passés en entrée dans l'opération de chiffrement:

RSAES-PKCS1-v1_5-Encrypt(CPK, *M*).

Le résultat de cette opération est un cryptogramme appelé *C* et représenté par une chaîne d'octets dans la spécification [IETF RFC 8017]. Dans la présente Recommandation, *C* a une longueur de 256 octets. Sa valeur est égale à BS2OSP(E(CPK, LK)) dans le contexte de l'échelle de clés et à BS2OSP(E(CPK, *r*)) dans le contexte du mécanisme d'authentification.

La clé CSK et le message *C* sont passés en entrée dans l'opération de déchiffrement:

RSAES-PKCS-v1_5-Decrypt(CSK, *C*).

Cette opération est effectuée dans le bloc d'échelle de clés. Elle produit le message *M*, qui est représenté par une chaîne d'octets. Le bloc d'échelle de clés calcule ensuite LK = OS2BSP(*M*) dans le contexte de l'échelle de clés et *r* = OS2BSP(*M*) dans le contexte du mécanisme d'authentification.

10.3 Système de signature numérique

Le **Système de signature numérique** doit être le système RSASSA-PKCS1-v1_5 décrit dans la spécification [IETF RFC 8017]. Ce système se compose d'une opération de création de signature et d'une opération de vérification de signature. Comme nous l'avons mentionné dans les § 7 et 8, seule l'opération de vérification de signature est effectuée dans le bloc d'échelle de clés; l'opération de création de signature, pour sa part, est effectuée dans le système de l'expéditeur.

La valeur de la clé publique RSA du signataire est désignée par (*n*, *e*) et elle est représentée par une paire d'entiers dans la spécification [IETF RFC 8017]. Dans la présente Recommandation, la longueur du modulo de *n* dans la clé publique RSA du signataire est de 2 048 bits; en conséquence, la longueur de *n*, désignée par *k* dans la spécification [IETF RFC 8017], est de 256 octets. En outre, la valeur de l'exposant public *e* est égale à $2^{16} + 1$ dans la présente Recommandation. Un microprocesseur conforme doit stocker cette valeur de manière permanente et disposer de mesures permettant de protéger son intégrité. La valeur de la clé SPK doit être égale au modulo de *n*, ce qui implique que la longueur de la clé SPK doit être de 2 048 bits et que le SPK est représenté par un entier. La valeur de la clé SSK doit être égale à la valeur de la clé privée RSA du signataire, désignée par *K* dans la spécification [IETF RFC 8017].

Le message à signer est désigné par *M* et il est représenté par une chaîne d'octets dans la spécification [IETF RFC 8017]. Dans la présente Recommandation, *M* est défini par:

M = BS2OSP(**chipset-ID** || E(CPK, LK))

dans le contexte de l'échelle de clés, et

M = BS2OSP(**chipset-ID** || E(CPK, *r*))

dans le contexte du mécanisme d'authentification. *M* a donc une longueur de 8 + 256 = 264 octets dans les deux contextes.

La clé SSK et le message *M* sont passés en entrée dans l'opération de création de la signature:

RSASSA-PKCS1-v1_5-Sign(SSK, *M*).

Le résultat de cette opération est une signature appelée *S* et représentée par une chaîne d'octets dans la spécification [IETF RFC 8017]. Dans la présente Recommandation, *S* a une longueur de 256 octets. Sa valeur est égale à BS2OSP(S(SSK_i, **chipset-ID** || E(CPK, LK)))) dans le contexte de l'échelle de clés et à BS2OSP(S(SSK_i, **chipset-ID** || E(CPK, *r*)))) dans le contexte du mécanisme d'authentification.

Le couple (SPK, $2^{16} + 1$), le message *M* et la signature *S* sont passés en entrée dans l'opération de vérification de la signature:

RSAES-PKCS-v1_5-Verify((SPK, $2^{16} + 1$), *M*, *S*).

Cette opération est effectuée dans le bloc d'échelle de clés. Elle produit soit une "signature valable", soit une "signature non valable".

10.4 Fonction h

Le présent paragraphe contient la spécification de la fonction h. Nous avons vu dans les § 7 et 8 que les entrées de cette fonction étaient les suivantes:

- CW-URI, τ_b, AD1, SPK-URI, SPK_1, SPK_2, ..., SPK_m, et r dans le contexte de l'échelle de clés.
- AD1, SPK_1, SPK_2, ..., SPK_m, et r dans le contexte du mécanisme d'authentification.

Si le bloc d'échelle de clés ne reçoit pas l'un de ces deux ensembles d'entrées, ou si la longueur d'une entrée n'est pas conforme aux spécifications de la présente Recommandation, le bloc d'échelle de clés arrête les calculs. Dans tous les autres cas, la fonction h applique d'abord la primitive de conversion de données I2BSP à chacune des clés SPK reçues en entrée. Puis elle concatène les chaînes de bits représentant ses entrées pour produire le message M de la manière suivante dans le contexte de l'échelle de clés:

$$M = r \,||\, \text{CW-URI} \,||\, \tau_b \,||\, \text{AD1} \,||\, \text{SPK-URI} \,||\, \text{I2BSP}(SPK_1) \,||\, \text{I2BSP}(SPK_2) \,||\, ... \,||\, \text{I2BSP}(SPK_m),$$

et dans le contexte du mécanisme d'authentification:

$$M = r \,||\, \text{AD1} \,||\, \text{I2BSP}(SPK_1) \,||\, \text{I2BSP}(SPK_2) \,||\, ... \,||\, \text{I2BSP}(SPK_m).$$

Il convient de se souvenir que les éléments r, CW-URI, τ_b, AD1, SPK-URI et $\text{I2BSP}(SPK_i)$ ont respectivement une longueur de 128 bits, 64 bits, 8 bits, 256 bits, 64 bits et 2 048 bits. La longueur de M, désignée par l dans la norme [NIST FIPS 180-4], est donc égale à 520 + 2 048 m, où m est le nombre d'éléments $\text{I2BSP}(SPK_m)$ dans le bloc de messages M, dans le contexte de l'échelle de clés et à 384 + 2 048 m dans le contexte du mécanisme d'authentification.

La fonction h applique ensuite l'algorithme de hachage sécurisé-256(M) [SHA-256(M)] défini dans la norme [NIST FIPS 180-4].

Dans le contexte du mécanisme d'authentification, la fonction h tronque ce condensé de message en 256 bits pour produire un message en 128 bits qu'il envoie en sortie. Dans le contexte de l'échelle de clés, le bloc d'échelle de clés passe le résultat en 256 bits de l'algorithme SHA-256 au **Désembrouilleur de contenus**. Si la longueur du mot CW est de N bits, le **Désembrouilleur de contenus** tronque ce résultat à N bits. Dans les deux cas, il faut employer la méthode de troncature définie dans la norme [NIST SP 800-107].

10.5 Algorithme des codes d'authentification de message

Le présent paragraphe contient la définition de l'algorithme MAC présenté dans le § 7.5. Cet **Algorithme des codes d'authentification de message** est défini dans la norme [ISO/CEI 9797-1] et se caractérise notamment par les éléments suivants:

- Le chiffrement par blocs doit être conforme à la norme AES-128.
- Le remplissage du message doit s'effectuer selon la méthode de remplissage 3. À cette fin, il faut calculer la longueur de bits du message non rempli $e(LK_t, r) \,||\, A$, qui est égale à Len + 129.
- Il faut utiliser l'algorithme MAC 1 pour calculer le code d'authentification de message (MAC) à partir du message et de la clé secrète.
- Le code MAC doit avoir une longueur de τ bits.

Appendice I

Domaines nécessitant des développements supplémentaires

(Cet appendice ne fait pas partie intégrante de la présente Recommandation.)

Il a été établi que la présente Recommandation nécessite des développements et une validation supplémentaires afin de répondre aux exigences énoncées dans la Recommandation [b-UIT-T J.1010], et que la Recommandation [b-UIT-T J.1010] doit être mise à jour pour refléter les exigences de la spécification Enhanced Content Protection (ECP) de MovieLabs [b-ECP]. Les Recommandations [b-UIT-T J.1011], [b-UIT-T J.1012], [b-UIT-T J.1013], [b-UIT-T J.1014], UIT-T J.1015 et [b-UIT-T J.1015.1] devront à l'avenir être mises à jour pour refléter ces mises à jour de la Recommandation [b-UIT-T J.1010].

Plusieurs États Membres de l'UIT ainsi que des parties prenantes de divers secteurs – notamment des fabricants d'appareils et de composants électroniques, des propriétaires et des titulaires de licences de contenus protégés par le droit d'auteur, des fournisseurs de services over-the-top (OTT) et de télévision linéaire, et des fournisseurs de solutions de système d'accès conditionnel (CAS) et de gestion des droits numériques (DRM) – basées dans le monde entier se sont déclarés préoccupés par le fait que l'interface commune intégrée (ECI) ne répond pas pleinement aux exigences de la spécification ECP, ni aux exigences plus larges du secteur en matière de protection des contenus.

Plus précisément, leurs préoccupations ont été exprimées dans des contributions soumises à la réunion de la Commission d'étude 9 (CE 9) de l'UIT-T tenue du 16 au 23 avril 2020. Dans leurs contributions, Israël, l'Australie, Samsung (Membre de secteur de l'UIT-T) ainsi que Sky Group et MovieLabs (Associés de la CE 9) ont proposé d'apporter plusieurs modifications aux Recommandations relatives à l'interface ECI, mais aucun accord n'a été trouvé à leur sujet. Ces points sont répertoriés dans le document [b-CE 9 Rapport 17 Ann.1].

Les propositions visent à:

1) simplifier le système ECI en réduisant son champ d'application;
2) supprimer la gestion DRM;
3) supprimer le rechiffrement de contenu;
4) supprimer la gestion des logiciels;
5) ajouter des API pour les opérations de stockage et de chiffrement sécurisées;
6) autoriser des échelles de clés propres aux fournisseurs;
7) utiliser les exigences TEE J.1207;
8) inclure l'implémentation TEE pour les machines virtuelles;
9) augmenter la puissance des algorithmes de chiffrement, par exemple en utilisant SHA-384;
10) utiliser des certificats standard, comme UIT-T X.509;
11) revoir les communications entre clients;
12) mener des échanges supplémentaires avec l'ETSI;
13) effectuer une évaluation par les pairs supplémentaire;
14) envisager des alternatives au modèle de l'autorité de confiance;
15) définir plus précisément les aspects techniques des règles de conformité et de robustesse de l'interface ECI;
16) ajouter des exigences en matière de diversité, par exemple la randomisation de l'espace d'adresses;
17) ajouter des exigences relatives à la vérification de l'intégrité de l'exécution.

Ces propositions reflètent le fait que la protection des contenus et les menaces de compromission de contenu sont en constante évolution. L'interface ECI a été conçue initialement près de dix ans avant l'approbation de la présente Recommandation UIT-T. Des systèmes comme l'interface ECI doivent être évalués régulièrement en fonction à la fois des techniques d'attaque et des exigences de protection du secteur les plus récentes.

D'autres mécanismes existent pour assurer l'interopérabilité. En particulier, s'agissant de la gestion DRM, la plupart des services vidéo sur l'Internet ont déployé d'autres solutions pour assurer l'interopérabilité et répondre à leurs besoins.

Il est important d'apporter davantage de clarté, car de nombreux États Membres considèrent les normes de l'UIT comme des guides importants pour le développement de leurs marchés et de leurs secteurs. La liste de préoccupations vise à faire en sorte que la mise en œuvre de l'interface ICE sur les marchés nationaux puisse reposer sur une compréhension parfaite des conséquences de la présente Recommandation de l'UIT-T et que les questions soient prises en compte au moment d'examiner une législation, une réglementation ou des besoins du marché exigeant que les équipements de télévision numérique grand public soient interopérables. Elle vise également à faire en sorte que les fabricants d'équipements techniques, qui peuvent préférer utiliser un ensemble unique d'exigences ou d'autres normes pour concevoir les produits, puissent prendre en compte ces questions lors du développement de produits pour des marchés différents.

Bibliographie

[b-UIT-T J.1010] Recommandation UIT-T J.1010 (2016), *Interface commune intégrée pour les solutions CA/DRM interchangeables; Cas d'utilisation et exigences*.

[b-UIT-T J.1011] Recommandation UIT-T J.1011 (2016), *Interface commune intégrée pour les solutions CA/DRM interchangeables; Architecture, définitions et vue d'ensemble*.

[b-UIT-T J.1012] Recommandation UIT-T J.1012 (2020), *Interface commune intégrée pour les solutions CA/DRM interchangeables; Conteneur CA/DRM, chargeur, interfaces et révocation*.

[b-UIT-T J.1013] Recommandation UIT-T J.1013 (2020), *Interface commune intégrée pour les solutions CA/DRM interchangeables; Machine virtuelle*.

[b-UIT-T J.1014] Recommandation UIT-T J.1014 (2020), *Interface commune intégrée pour les solutions CA/DRM interchangeables; Sécurité évoluée – Fonctionnalités propres aux interfaces ECI*.

[b-UIT-T J.1015.1] Recommandation UIT-T J.1015.1 (2020), *Interface commune intégrée (ECI) pour les solutions CA/DRM interchangeables; Système de sécurité évoluée – Bloc d'échelle de clés: authentification des informations sur les règles d'utilisation des mots de contrôle et des données associées 1*.

[b-CE 9 Rapport 17 Ann.1] Rapport de la réunion de la CE 9 de l'UIT-T, SG9-R17-Annexe 1 (2020), Annexe 1 au Rapport 17 de la réunion de la CE 9 organisée de manière entièrement virtuelle du 16 au 23 avril 2020. https://www.itu.int/md/T17-SG09-R-0017/en

[b-ISO/CEI 23001-7] ISO/CEI 23001-7:2016: *Technologies de l'information – Technologies des systèmes MPEG – Partie 7: Cryptage commun des fichiers au format de fichier de médias de la base ISO*.

[b-ATSC A/70-1] ATSC Standard A/70 Part 1:2010, *Conditional access system for terrestrial broadcast*. https://www.atsc.org/standard/a70-part-12010-conditional-access-system-for-terrestrial-broadcast/

[b-ETSI GS ECI 001-5-2] ETSI GS ECI 001-5-2 V1.1.1 (2017-07), *Embedded Common Interface (ECI) for exchangeable CA/DRM solutions; Part 5: The Advanced Security System; Sub-part 2: Key Ladder Block*. https://www.etsi.org/deliver/etsi_gs/ECI/001_099/0010502/01.01.01_60/gs_ECI0010502v010101p.pdf

[b-ETSI TS 100 289] ETSI TS 100 289 V1.1.1 (2011), *Digital video broadcasting (DVB); Support for use of the DVB scrambling algorithm version 3 within digital broadcasting systems*. https://www.etsi.org/deliver/etsi_ts/100200_100299/100289/01.01.01_60/ts_100289v010101p.pdf

[b-ETSI TS 103 127] ETSI TS 103 127 V1.1.1 (2013), *Digital video broadcasting (DVB); Content scrambling algorithms for DVB-IPTV services using MPEG2 transport streams*. https://www.etsi.org/deliver/etsi_ts/103100_103199/103127/01.01.01_60/ts_103127v010101p.pdf

[b-GY/T 277-2014] ChinaDRM Lab, GY/T 277-2014: 互联网电视数字版权管理技术规范 [Spécification technique sur la gestion des droits numériques pour la télévision Internet].

[b-IEEE-OUI] IEEE Standards Association (2017), *Guidelines for use of extended unique identifier (EUI), organizationally unique identifier (OUI) and company ID (CID)*.
https://standards.ieee.org/develop/regauth/tut/eui.pdf

[b-ROEL] Roelse, P. (2014). *A new key establishment protocol and its application in pay-TV systems*.
https://arxiv.org/pdf/1308.4371.pdf

[b-ECP] MovieLabs Specification for Enhanced Content Protection – Version 1.2.
https://movielabs.com/ngvideo/MovieLabs_ECP_Spec_v1.2.pdf

SÉRIES DES RECOMMANDATIONS UIT-T

Série A	Organisation du travail de l'UIT-T
Série D	Principes de tarification et de comptabilité et questions de politique générale et d'économie relatives aux télécommunications internationales/TIC
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	Environnement et TIC, changement climatique, déchets d'équipements électriques et électroniques, efficacité énergétique; 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 et mesures et tests associés
Série R	Transmission télégraphique
Série S	Équipements 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, réseaux de prochaine génération, Internet des objets et villes intelligentes
Série Z	Langages et aspects généraux logiciels des systèmes de télécommunication

Imprimé en Suisse
Genève, 2020