UNION INTERNATIONALE DES TÉLÉCOMMUNICATIONS

UIT-T X.830

SECTEUR DE LA NORMALISATION
DES TÉLÉCOMMUNICATIONS
DE L'UIT

(04/95)

RÉSEAUX DE COMMUNICATION DE DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS

SÉCURITÉ

TECHNOLOGIES DE L'INFORMATION – INTERCONNEXION DES SYSTÈMES OUVERTS – SÉCURITÉ GÉNÉRIQUE DES COUCHES SUPÉRIEURES: APERÇU GÉNÉRAL, MODÈLES ET NOTATION

Recommandation UIT-T X.830

(Antérieurement «Recommandation du CCITT»)

AVANT-PROPOS

L'UIT (Union internationale des télécommunications) est une institution spécialisée des Nations Unies dans le domaine des télécommunications. L'UIT-T (Secteur de la normalisation des télécommunications) est un organe permanent de l'UIT. Au sein de l'UIT-T, qui est l'entité qui établit les normes mondiales (Recommandations) sur les télécommunications, participent quelque 179 pays membres, 84 exploitations de télécommunications reconnues, 145 organisations scientifiques et industrielles et 38 organisations internationales.

L'approbation des Recommandations par les Membres de l'UIT-T s'effectue selon la procédure définie dans la Résolution nº 1 de la Conférence mondiale de normalisation des télécommunications (CMNT), (Helsinki, 1993). De plus, la CMNT, qui se réunit tous les quatre ans, approuve les Recommandations qui lui sont soumises et établit le programme d'études pour la période suivante.

Dans certains secteurs de la technologie 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. Le texte de la Recommandation X.830 de l'UIT-T a été approuvé le 10 avril 1995. Son texte est publié, sous forme identique, comme Norme internationale ISO/CEI 11586-1.

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.

RECOMMANDATIONS UIT-T DE LA SÉRIE X

RÉSEAUX DE COMMUNICATION DE DONNÉES ET COMMUNICATION ENTRE SYSTÈMES OUVERTS

(Février 1994)

ORGANISATION DES RECOMMANDATIONS DE LA SÉRIE X

Domaine	Recommandations
RÉSEAUX PUBLICS POUR DONNÉES	
Services et services complémentaires	X.1-X.19
Interfaces	X.20-X.49
Transmission, signalisation et commutation	X.50-X.89
Aspects réseau	X.90-X.149
Maintenance	X.150-X.179
Dispositions administratives	X.180-X.199
INTERCONNEXION DES SYSTÈMES OUVERTS	
Modèle et notation	X.200-X.209
Définition des services	X.210-X.219
Spécifications des protocoles en mode connexion	X.220-X.229
Spécifications des protocoles en mode sans connexion	X.230-X.239
Formulaires PICS	X.240-X.259
Identification des protocoles	X.260-X.269
Protocoles de sécurité	X.270-X.279
Objets gérés de couche	X.280-X.289
Test de conformité	X.290-X.299
INTERFONCTIONNEMENT DES RÉSEAUX	
Considérations générales	X.300-X.349
Systèmes mobiles de transmission de données	X.350-X.369
Gestion	X.370-X.399
SYSTÈMES DE MESSAGERIE	X.400-X.499
ANNUAIRE	X.500-X.599
RÉSEAUTAGE OSI ET ASPECTS DES SYSTÈMES	
Réseautage	X.600-X.649
Dénomination, adressage et enregistrement	X.650-X.679
Notation de syntaxe abstraite numéro un (ASN.1)	X.680-X.699
GESTION OSI	X.700-X.799
SÉCURITÉ	X.800-X.849
APPLICATIONS OSI	
Engagement, concomitance et rétablissement	X.850-X.859
Traitement des transactions	X.860-X.879
Opérations distantes	X.880-X.899
TRAITEMENT OUVERT RÉPARTI	X.900-X.999

TABLE DES MATIÈRES

Résumé

La présente Recommandation fait partie d'une série de Recommandations comprenant un ensemble de moyens destinés à la réalisation des protocoles des couches supérieures de l'OSI qui prennent pour charge les services de sécurité. Elle définit:

a) des modèles généraux de fonctions de protocole d'échanges de sécurité et des transformations de sécurité;

b) une série d'outils de notation pour spécifier les besoins de protection sélective des champs dans une spécification de syntaxe abstraite, les échanges de sécurité et les transformations de sécurité;

c) une série de lignes directrices informatives sur l'application des moyens de sécurité génériques des couches supérieures traités dans la présente série de Recommandations.

Introduction

La présente Recommandation | Norme internationale appartient à une série de Recommandations | Normes internationales qui fournissent un ensemble de moyens destinés à la réalisation des protocoles des couches supérieures pour prendre en charge les services de sécurité. La structure de cette série est la suivante:

– Partie 1: aperçu général, modèles et notation

– Partie 2: définition du service «Elément de service d'échange de sécurité»

– Partie 3: spécification du protocole «Elément de service d'échange de sécurité»

– Partie 4: spécification de la syntaxe de protection du transfert

– Partie 5: formulaire PICS pour l'élément de service d'échange de sécurité

– Partie 6: formulaire PICS pour la syntaxe de protection du transfert

La présente Recommandation | Norme internationale constitue la Partie 1 de cette série.

Des lignes directrices informatives sur l'utilisation des moyens décrits dans la présente série sont données à l'Annexe G.

Il est important de noter que ces moyens de sécurité génériques n'assurent pas eux-mêmes des services de sécurité mais qu'ils sont des outils pour la réalisation des protocoles liés à la sécurité. Par ailleurs, ces moyens ne donnent pas des solutions uniques à toutes les conditions requises de sécurité dans les communications. Sans doute faudra-t-il encore ajouter aux spécifications des normes d'application des éléments de sécurité fonctionnant conjointement avec les services de sécurité génériques assurés par les moyens de sécurité génériques des couches supérieures.

NORME INTERNATIONALE

RECOMMANDATION UIT-T

TECHNOLOGIES DE L'INFORMATION – INTERCONNEXION DES SYSTÈMES OUVERTS – SÉCURITÉ GÉNÉRIQUE DES COUCHES SUPÉRIEURES: APERÇU GÉNÉRAL, MODÈLES ET NOTATION

1 Domaine d'application

1.1 La présente série de Recommandations | Normes internationales définit une série de moyens génériques utilisés dans l'établissement de services de sécurité dans les applications de l'OSI. Elles comprennent:

a) une série d'outils de notation permettant de spécifier les besoins de protection sélective des champs dans une spécification de syntaxe abstraite et permettant la spécification d'échanges de sécurité et de transformations de sécurité;

b) une définition du service, la spécification du protocole et le formulaire PICS pour l'élément de service d'application (ASE) qui contribueront à assurer les services de sécurité dans la couche application de l'OSI;

c) une spécification et un formulaire PICS pour une syntaxe de transfert de sécurité, associés à la couche présentation pour les services de sécurité dans la couche application.

1.2 La présente Recommandation | Norme internationale:

a) définit des modèles généraux des fonctions de protocole d'échange de sécurité et de transformation de sécurité, fondés sur les concepts décrits dans le modèle de sécurité des couches supérieures de l'OSI (Rec. UIT-T X.803 | ISO/CEI 10745);

b) définit une série d'outils de notation facilitant la spécification des besoins de protection sélective des champs dans une spécification de syntaxe abstraite, ainsi que la spécification des échanges de sécurité et des transformations de sécurité;

c) définit une série de lignes directrices informatives sur l'application des moyens de sécurité génériques des couches supérieures traités dans la présente série de Recommandations | Normes internationales.

1.3 La présente Recommandation | Norme internationale:

a) ne définit pas de série complète de moyens de sécurité des couches supérieures pouvant être requis par d'autres Recommandations | Normes internationales;

b) ne définit pas de série complète de moyens de sécurité pour des applications spécifiques;

c) ne définit pas le mécanisme employé pour prendre en charge les services de sécurité.

1.4 Le modèle de l'échange de sécurité et la notation qui l'accompagne sont tous deux destinés à servir de base à la définition de l'élément de service d'échange de sécurité dans les parties futures de la présente série de Recommandations | Normes internationales et à être utilisés par tout autre élément ASE pouvant introduire des échanges de sécurité dans sa propre spécification.

2 Références normatives

Les Recommandations et Normes internationales suivantes contiennent des dispositions qui, par suite de la référence qui y est faite, constituent des dispositions valables pour la présente Recommandation | Norme internationale. Au moment de la publication, les éditions indiquées étaient en vigueur. Toutes Recommandations et Normes sont sujettes à révision et

les parties prenantes aux accords fondés sur la présente Recommandation | Norme internationale sont invitées à rechercher la possibilité d'appliquer les éditions les plus récentes des Recommandations et Normes indiquées ci-après. Les membres de la CEI et de l'ISO possèdent le registre des Normes internationales en vigueur. Le Bureau de la normalisation des télécommunications de l'UIT tient à jour une liste des Recommandations de l'UIT-T en vigueur.

2.1 Recommandations | Normes internationales identiques

- Recommandation UIT-T X.200 (1994) | ISO/CEI 7498-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de référence de base: Le modèle de référence de base.*
- Recommandation UIT-T X.207 (1993) | ISO/CEI 9545:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Structure de la Couche Application.*
- Recommandation UIT-T X.214 (1993) | ISO/CEI 8072:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service de Transport.*
- Recommandation UIT-T X.216 (1994) | ISO/CEI 8822:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Définition du service de Présentation.*
- Recommandation UIT-T X.217 (1995) | ISO/CEI 8649:...[1)], *Technologies de l'information – Interconnexion des systèmes ouverts – Définition de service applicable à l'élément de service de contrôle d'association*.
- Recommandation UIT-T X.226 (1994) | ISO/CEI 8823-1:1994, *Technologies de l'information – Interconnexion des systèmes ouverts – Protocole de présentation en mode connexion: Spécification du protocole.*
- Recommandation UIT-T X.509 (1993) | ISO/CEI 9594-8:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – L'Annuaire: Cadre d'authentification*.
- Recommandation UIT-T X.511 (1993) | ISO/CEI 9594-3:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – L'Annuaire: Définition du service abstrait.*
- Recommandation X.660 du CCITT (1992) | ISO/CEI 9834-1:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Procédures pour le fonctionnement des autorités d'enregistrement OSI – Procédures générales*.
- Recommandation UIT-T X.680 (1994) | ISO/CEI 8824-1:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: Spécification de la notation de base.*
- Recommandation UIT-T X.681 (1994) | ISO/CEI 8824-2:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: Spécification des objets informationnels*.
- Recommandation UIT-T X.682 (1994) | ISO/CEI 8824-3:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: Spécification des contraintes.*
- Recommandation UIT-T X.683 (1994) | ISO/CEI 8824-4:1995, *Technologies de l'information – Notation de syntaxe abstraite numéro un: Paramétrage des spécifications de la notation de syntaxe abstraite numéro un.*
- Recommandation UIT-T X.690 (1994) | ISO/CEI 8825-1:1995, *Technologies de l'information – Règles de codage de la notation de syntaxe abstraite numéro un: Spécification des règles de codage de base, des règles de codage canoniques et des règles de codage distinctives*.
- Recommandation UIT-T X.803 (1994) | ISO/CEI 10745:1995, *Technologies de l'information – Interconnexion des systèmes ouverts – Modèle de sécurité pour les couches supérieures*.
- Recommandation UIT-T X.811 (1995) | ISO/CEI 10181-2:1996, *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour systèmes ouverts: Cadre d'authentification.*
- Recommandation UIT-T X.812[1)] | ISO/CEI 10181-3:...[1)], *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: contrôle d'accès*.

---

1) Actuellement à l'état de projet.

2.2 Paires de Recommandations | Normes internationales équivalentes par leur contenu technique

– Recommandation X.800 du CCITT (1991), *Architecture de sécurité pour l'interconnexion en systèmes ouverts d'applications du CCITT*.

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

3 Définitions

3.1 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. UIT-T X.200 | ISO/CEI 7498-1:

– syntaxe de transfert.

3.2 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. X.800 du CCITT | ISO 7498-2:

– contrôle d'accès;
– confidentialité;
– authentification de l'origine des données;
– déchiffrement;
– signature numérique;
– chiffrement;
– intégrité;
– clé;
– gestion des clés;
– protection sélective des champs.

3.3 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. UIT-T X.216 | ISO/CEI 8822:

– syntaxe abstraite;
– contexte de présentation;
– valeur de données de présentation.

3.4 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. UIT-T X.207 | ISO/CEI 9545:

– association d'application;
– contexte d'application;
– élément de service d'application (ASE);
– association d'objets de service Application (association ASO).

3.5 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. UIT-T X.811 | ISO/CEI 10181-2:

– échange d'authentification;
– déclarant;
– authentification d'entité;
– vérificateur.

3.6 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. UIT-T X.812 | ISO/CEI 10181-3:

– certificat de contrôle d'accès.

3.7 Les termes suivants sont utilisés tels qu'ils sont définis dans la Rec. UIT-T X.803 | ISO/CEI 10745:

– association de sécurité;
– fonction de communication de la sécurité (SCF);
– échange de sécurité;
– item d'échange de sécurité;

– fonction d'échange de sécurité;

– transformation de sécurité;

– objet de sécurité de système (SSO).

3.8 Pour les besoins de la présente Recommandation | Norme internationale, les définitions suivantes s'appliquent:

3.8.1 association de sécurité liée à un contexte de présentation: association de sécurité qui est établie en même temps qu'un contexte de présentation de protection et qui s'applique à toutes les valeurs de données de présentation envoyées dans un seul sens, selon ce contexte de présentation de protection; les attributs de l'association de sécurité sont spécifiés de manière explicite dans le codage de la première valeur de données de présentation selon le contexte de présentation de protection.

3.8.2 association de sécurité liée à un item unique: association de sécurité s'appliquant à une valeur de données de présentation unique protégée indépendamment, et qui n'est pas associée à un contexte de présentation; les attributs de l'association de sécurité sont spécifiés de façon explicite dans le codage de la valeur de données de présentation.

3.8.3 association de sécurité établie extérieurement: association de sécurité qui est établie indépendamment des instances de son utilisation et qui a un identificateur globalement unique qui permet de s'y référer au moment de son utilisation.

3.8.4 règles de codage initiales: règles de codage ASN.1 utilisées pour produire une chaîne de bits non protégée à partir d'une valeur d'un type ASN.1 quand cette valeur doit être protégée au moyen d'une transformation de sécurité.

3.8.5 contexte de protection de présentation: contexte de présentation qui associe une syntaxe de protection de transfert à une syntaxe abstraite.

3.8.6 syntaxe de protection de transfert: syntaxe de transfert qui utilise une transformation de sécurité.

3.8.7 mappage de protection: spécification qui projette un besoin de protection, identifié par un nom dans une spécification de syntaxe abstraite, sur une transformation de sécurité spécifique qui doit être utilisée pour répondre à ce besoin.

4 Abréviations

ACSE Elément de service de contrôle d'association (*association control service element*)

ASE Elément de service d'application (*application-service-element*)

ASO Objet de service d'applications (*application-service-object*)

GULS Sécurité générique des couches supérieures (*generic upper layers security*)

OSI Interconnexion des systèmes ouverts (*open systems interconnection*)

PDU Unité de données de protocole (*protocol-data-unit*)

PDV Valeur de données de présentation (*presentation data value*)

PICS Déclaration de conformité d'une instance de protocole (*protocol implementation conformance statement*)

SCF Fonction de communication de sécurité (*security communication function*)

SEI Item d'échange de sécurité (*security exchange item*)

SESE Elément de service d'échange de sécurité (*security exchange service element*)

SSO Objet de sécurité de système (*system security object*)

5 Aperçu général

Les Normes de sécurité générique des couches supérieures (GULS) (*generic upper layers security*) définissent une série d'outils de construction de protocoles et de composantes de protocole pour assurer la sécurité de la transmission des informations. Ces moyens sont destinés à l'établissement de services de sécurité dans les couches supérieures de l'OSI (la couche Application, avec l'aide parfois de la couche Présentation).

NOTE – Les services de sécurité pour les applications de l'OSI peuvent être assurés au moyen de mécanismes de sécurité dans les couches supérieures ou dans les couches inférieures. Dans ce dernier cas, la protection est obtenue en spécifiant une qualité de service appropriée pour la protection (telle qu'elle est définie dans la Rec. UIT-T X.214 | ISO/CEI 8072) pour l'élément ACSE à l'établissement d'une association d'application. Cette demande de qualité de service de la protection est transmise en transparence au service de transport via les couches de Présentation et de Session. La fourniture de services de sécurité dans les couches inférieures ne relève pas de la présente Recommandation | Norme internationale.

Les moyens mis à disposition par les Normes GULS sont:

- un moyen général pour réaliser les composantes de protocole de couche Application en vue de prendre en charge l'échange d'informations liées à la sécurité entre deux instances d'entité d'application en communication (le concept d'échange de sécurité, qui est assuré par l'élément SESE); ces moyens sont décrits à l'article 6;

- une approche générale de l'utilisation des moyens de la couche Présentation pour effectuer des transformations liées à la sécurité sur des items d'information afin de protéger ces derniers (la syntaxe générique de protection de transfert); ces moyens sont décrits à l'article 7;

- des outils de notation de syntaxe abstraite pour aider le concepteur de protocoles d'application à spécifier que la protection de sécurité doit s'appliquer à des champs sélectionnés de ce protocole (un type paramétré PROTECTED, et la variante PROTECTED-Q de ce type); ces moyens sont décrits à l'article 8.

Les échanges de sécurité sont utilisés pour des besoins tels que l'authentification des entités et la gestion des clés. Les transformations de sécurité (ainsi que la syntaxe générique de protection de transfert et/ou le type paramétré PROTECTED ou ses variantes) sont utilisées pour des besoins tels que l'intégrité, la confidentialité, l'authentification de l'origine des données et/ou la non-répudiation.

Le modèle de sécurité des couches supérieures (Rec. UIT-T X.803 | ISO/CEI 10745) contient un modèle architectural pour les spécifications GULS qui décrit les rôles et les fonctions d'échange de sécurité et les transformations de sécurité.

Les fonctions d'échange de sécurité donnent les moyens nécessaires pour communiquer les renseignements de sécurité entre les invocations d'entité d'application dans le cadre du fonctionnement d'un mécanisme de sécurité, c'est-à-dire qu'elles produisent et traitent les informations de contrôle de protocole d'application dans un but lié à la sécurité. Les échanges de sécurité peuvent être spécifiés au moyen de la notation de la présente Recommandation | Norme internationale, pour être ensuite importés dans toute spécification de syntaxe abstraite. L'élément de service d'échange de sécurité (SESE) est un élément de service d'application (ASE) défini dans la Rec. UIT-T X.831 | ISO/CEI 11586-2 et dans la Rec. UIT-T X.832 | ISO/CEI 11586-3. L'élément SESE fournit un moyen pour acheminer les échanges de sécurité qui répond à l'objectif de rendre les éléments ASE spécifiques à l'application indépendants du mécanisme de sécurité utilisé. Toutefois, certains aspects de la spécification d'une application qui englobe directement des dispositions de sécurité seront dépendants du mécanisme.

Les transformations de sécurité peuvent être prises en charge par la syntaxe générique de protection de transfert décrite dans la Rec. UIT-T X.833 | ISO/CEI 11586-4.

6 Echanges de sécurité

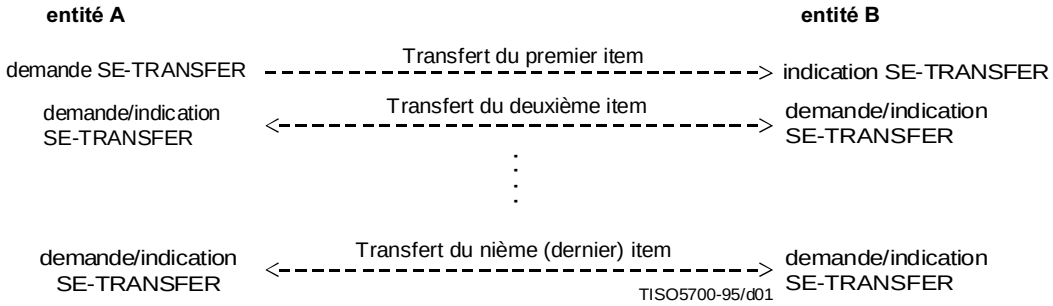
6.1 Modèle d'échange de sécurité

La présente Recommandation | Norme internationale précise le modèle procédural de l'échange de sécurité tel qu'il est présenté dans la Rec. UIT-T X.803 | ISO/CEI 10745.

Un échange de sécurité survient entre deux entités A et B. Il est constitué du transfert d'un item d'échange de sécurité (SEI) de A à B, éventuellement suivi d'un ou de plusieurs transferts d'items SEI dans les deux sens entre A et B. Le

nombre de transferts dépend de l'échange de sécurité en question. Chaque item SEI peut comprendre une structure de données arbitrairement complexe pouvant être représentée par un type ASN.1 quelconque. Il peut inclure des composantes qui sont individuellement protégées au moyen de la notation PROTECTED décrite à l'article 8.

Le diagramme de la Figure 1 illustre la suite de transferts d'items SEI d'un échange de sécurité à n allers-retours et le service SESE correspondant (SE-TRANSFER) défini dans la Rec. UIT-T X.831 | ISO/CEI 11586-2.



NOTE – La double flèche indique que le transfert peut être fait par A ou par B.

Figure 1 – Modèle d'échange de sécurité

Deux classes d'échanges sont définies:

- *à l'alternat* – Les transferts successifs d'éléments sont effectués dans un sens puis dans l'autre, et un seul transfert est actif à un moment donné;
- *libre* – Le sens de transfert n'est soumis à aucune contrainte, et des transferts dans les deux sens peuvent être actifs simultanément.

Quand un échange de sécurité est en cours, d'autres transferts d'information peuvent survenir et d'autres échanges de sécurité peuvent être en cours sur la même association d'application. Toutefois, les règles du contexte d'application limitent en général de telles activités qui se chevauchent. Les valeurs de données de présentation acheminant des items SEI peuvent être concaténées ou entrelacées avec d'autres valeurs de données de présentation, voire y être intégrées.

6.2 Notation pour spécifier les échanges de sécurité

La spécification d'un échange de sécurité inclut la spécification des types d'item SEI pouvant être échangés, la déclaration d'une éventuelle contrainte de classement s'appliquant au transfert de ces items SEI, la déclaration des conditions d'erreur pouvant résulter du transfert de chaque item SEI ainsi qu'une déclaration concernant la sémantique associée (ou une référence à celle-ci).

La définition d'un échange de sécurité englobe:

a) l'assignation d'un identificateur d'objet global ou d'une valeur entière locale à l'échange de sécurité afin que son utilisation puisse être identifiée sans équivoque dans le protocole;

b) la spécification de la syntaxe abstraite des items SEI et les notifications d'erreur transférées dans cet échange de sécurité.

Pour permettre de spécifier cette information sous une forme utilisable par le protocole SESE, trois définitions de classe d'objets informationnels ASN.1 (voir dans la Rec. UIT-T X.681 | ISO/CEI 8824-2) sont prévues:

- la classe d'objets informationnels SECURITY-EXCHANGE est utilisée pour spécifier un échange de sécurité particulier; un objet informationnel de cette classe contient un ou plusieurs objets informationnels SEC-EXCHG-ITEM;

– la classe d'objets informationnels SEC-EXCHG-ITEM est utilisée pour définir un item SEI; un objet informationnel de cette classe peut contenir un ou plusieurs objets informationnels ERROR;

– la classe d'objets informationnels SE-ERROR est utilisée pour définir une situation d'erreur qui peut résulter du transfert d'un item SEI.

NOTE – L'Annexe G contient les lignes directrices montrant comment ces classes d'objets informationnels sont utilisées dans la définition d'un contexte d'application complet.

```
SECURITY-EXCHANGE ::= CLASS
-- Cette définition de classe d'objet d'information doit être utilisée pour
-- spécifier une instance particulière d'un échange de sécurité.
{
  &SE-Items         SEC-EXCHG-ITEM,
  -- Ceci est un ensemble d'objet d'information ASN.1
  -- comprenant une série d'items d'échange de sécurité
  &sE-Identifier    Identifier          UNIQUE
  -- Un identificateur local ou global pour l'échange de sécurité en question
}
WITH SYNTAX
-- La syntaxe suivante est utilisée pour spécifier un échange de sécurité donné.
{
  SE-ITEMS          &SE-Items
  IDENTIFIER        &sE-Identifier
}
Identifier ::= CHOICE
{
  local             INTEGER,
  global            OBJECT IDENTIFIER
}

SEC-EXCHG-ITEM ::= CLASS
{
  &ItemType,
      -- Type ASN.1 de cet item d'échange
  &itemId           INTEGER,
      -- Identificateur de cet item, par exemple 1, 2, 3, ..
  &Errors           SE-ERROR            OPTIONAL
      -- Liste facultative des erreurs pouvant résulter du transfert de cet item
}
WITH SYNTAX
{
  ITEM-TYPE         &ItemType
  ITEM-ID           &itemId
  [ERRORS           &Errors]
}

SE-ERROR ::= CLASS
{
  &ParameterType    OPTIONAL,
  -- Type ASN.1 d'un paramètre pour accompagner le signalement de
  -- la situation d'erreur à l'expéditeur de l'item SEI
  &errorCode        Identifier          UNIQUE
  -- Un identificateur utilisé pour signaler la situation
  -- d'erreur à l'expéditeur de l'item SEI
}
WITH SYNTAX
{
  [PARAMETER        &ParameterType]
  ERROR-CODE        &errorCode
}
```

Les exemples de l'utilisation de cette notation sont donnés à l'Annexe C.

7 Transformation de sécurité

7.1 Modèle de transformation de sécurité

Une transformation de sécurité est une fonction de sécurité (ou une combinaison de ces fonctions) appliquée aux données de l'utilisateur afin de protéger celles-ci au cours de leur communication ou de leur enregistrement. Une transformation de sécurité fait intervenir un processus de codage, appliqué avant la communication ou l'enregistrement, et un décodage qui peut être appliqué (mais ne doit pas toujours l'être) après réception ou récupération des données. Des exemples de transformations de sécurité sont:

a) l'application d'un processus de chiffrement au codage des données et un processus de déchiffrement correspondant au décodage;

b) la production d'un cachet ou d'une signature et son rattachement aux données au moment du codage, suivi d'un contrôle et d'un retrait du cachet ou de la signature au moment du décodage;

c) la combinaison des fonctions a) et b) en une seule transformation de sécurité.

Les transformations de sécurité qui sont définies au moyen de la notation du 7.2 conviennent pour les applications OSI (conjointement avec la syntaxe générique de protection de transfert définie dans la Rec. UIT-T X.833 | ISO/CEI 11586-4) ou pour d'autres fins, y compris la protection directe au cours de l'enregistrement local et des communications non-OSI.

NOTE – Le paragraphe 7.1.5 décrit l'utilisation des transformations de sécurité sur une connexion de présentation OSI. Le paragraphe 7.1.6 décrit leur utilisation indépendamment du protocole de présentation OSI.

Les transformations de sécurité peuvent être le principal moyen de fournir un service de sécurité (tel que la confidentialité, l'intégrité, l'authentification de l'origine des données) ou peuvent contribuer à la fourniture d'un service de sécurité (tel que l'authentification de l'entité, le contrôle d'accès, la non-répudiation).

La Figure 2 illustre les étapes successives de la protection d'une donnée élémentaire en vue de son transfert ou de son stockage.

Au niveau d'un système de codage, le processus consistant à obtenir une représentation transformée (protégée) d'une donnée élémentaire non protégée est le suivant:

a) si la donnée élémentaire non protégée est une valeur d'un type ASN.1, telle que spécifiée dans une définition de syntaxe abstraite, codage au moyen des règles de codage initiales et représentation sous forme de chaîne binaire;

b) application du processus de codage de la transformation de sécurité à la représentation sous forme de chaîne binaire de la donnée élémentaire non protégée, éventuellement en utilisant des renseignements supplémentaires obtenus sur place, pour obtenir une donnée élémentaire transformée qui est une valeur du type ASN.1 XformedDataType (le type précis est spécifié dans le cadre de la définition de la transformation de sécurité);

c) codage de la valeur ASN.1 résultant de b) (éventuellement dans le cadre du processus de codage d'une valeur ASN.1 enveloppante, telle que la structure de syntaxe de protection de transfert définie dans la Rec. UIT-T X.833 | ISO/CEI 11586-4).

Au niveau d'un système de décodage, le processus consistant à récupérer la donnée élémentaire non protégée et/ou vérifier un compromis de sécurité consiste:

d) à décoder la donnée élémentaire transformée reçue ou récupérée, qui est valeur du type ASN.1 XformedDataType (ce processus de décodage peut faire partie du décodage d'une valeur ASN.1 enveloppante, telle que la structure de syntaxe de protection de transfert définie dans la Rec. UIT-T X.833 | ISO/CEI 11586-4);

e) à appliquer le processus de décodage de la transformation de sécurité à la valeur reçue ou récupérée, éventuellement en utilisant des renseignements additionnels obtenus sur place, et produire une restitution conformément à ce processus de décodage (selon la transformation en question, les restitutions peuvent inclure une copie récupérée de la donnée élémentaire non protégée, une indication de réussite ou de défaillance de la vérification de la signature ou du cachet et/ou une copie d'une signature pour son enregistrement sur place en vue d'un usage ultérieur);

f) si la restitution de l'étape e) est une copie récupérée de la donnée élémentaire non protégée, et si cette donnée est une valeur d'un type ASN.1 tel que spécifié dans la définition de la syntaxe abstraite, à décoder cette donnée élémentaire au moyen des mêmes règles de codage initiales que celles de l'étape a).

L'établissement des règles de codage initiales des étapes a) et f) est décrit au 7.1.4. On notera qu'en général les transformations de sécurité peuvent agir sur des données élémentaires autres que des valeurs de types ASN.1 (telles des chaînes binaires arbitraires), raison pour laquelle ce processus de codage n'est pas toujours requis.

L'établissement des règles de codage utilisées aux étapes c) et d) dépend du contexte de stockage ou de communication, et il est indépendant de la transformation de sécurité particulière utilisée.

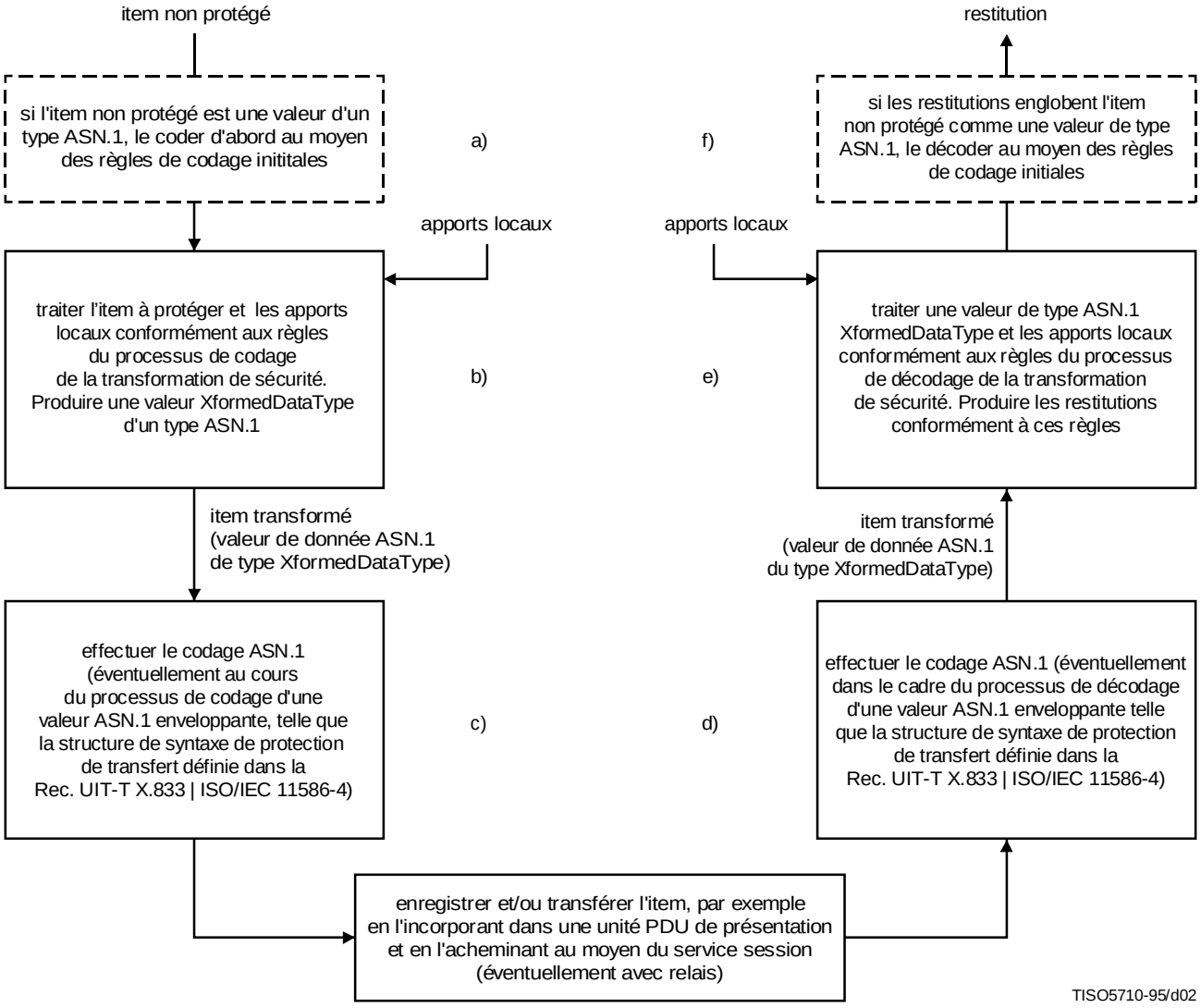


Figure 2 – Stockage et transfert protégés d'un item

7.1.1 Emplacement architectural des transformations de sécurité dans les couches supérieures de l'OSI

Une transformation de sécurité agit dans le contexte d'une association de sécurité entre deux ou plusieurs systèmes. Chaque système comporte un objet de sécurité du système (SSO) prenant en charge une telle association de sécurité. Les objets SSO effectuent des processus de codage et de décodage de la transformation de sécurité (tels que le chiffrement, la production/vérification de la signature numérique) et enregistrent l'information d'état de sécurité nécessaire (clés,

algorithmes, paramètres, état de chaînage). Le comportement interne de tels objets SSO est régi par des spécifications particulières de la transformation de sécurité en même temps que par des spécifications d'appui telles que les algorithmes (qui ne relèvent pas de la présente Recommandation | Norme internationale). En termes de la Figure 2, les fonctions indiquées dans les cases b) et e) sont modélisées dans des objets SSO.

On trouve aussi des fonctions de communication de sécurité (SCF) dans les entités de présentation des systèmes de codage et de décodage. Ces fonctions SCF prennent en charge les besoins de communication des objets SSO. En termes de la Figure 2, les fonctions indiquées dans les cases a), c), d) et f) sont modélisées dans des fonctions SCF. Les définitions du comportement des fonctions SCF sont contenues dans l'article 8 de la présente Recommandation | Norme internationale et dans la Rec. UIT-T X.833 | ISO/CEI 11586-4.

7.1.2 Associations de sécurité

Une transformation de sécurité peut être appliquée de façon itérative à une suite de valeurs de données logiquement ordonnées telles que des valeurs de données de présentation transférées de manière séquentielle, dans un seul sens, entre deux systèmes. La même protection est appliquée à chacune des valeurs de données. L'application d'une transformation de sécurité à une telle suite est régie par une association de sécurité. Il peut exister simultanément plusieurs associations de sécurité entre deux systèmes, produisant généralement des types de protection différents.

La présente Recommandation | Norme internationale concerne les aspects d'une association de sécurité qui s'appliquent aux communications des couches supérieures ou au stockage de l'information. Du point de vue des couches supérieures de l'OSI, une association de sécurité est une forme d'association ASO.

La présente Recommandation | Norme internationale reconnaît trois types d'association de sécurité:

a) *association de sécurité établie extérieurement* – Association de sécurité qui est établie indépendamment des instances de son utilisation, qui a un identificateur globalement unique qui permet de s'y référer quand il y a lieu de l'utiliser. Les moyens d'établir une telle association de sécurité ne sont pas spécifiés dans la présente Recommandation | Norme internationale, et la durée de vie de l'association n'est pas limitée aux dispositions contenues dans la présente Recommandation | Norme internationale. L'identificateur d'une association de sécurité établie extérieurement comprend une valeur entière ainsi que l'identité du système attribuant cette valeur entière; (Cette dernière identité pouvant être connue explicitement, par exemple l'expéditeur ou le destinataire des données, elle n'a pas toujours besoin d'être véhiculée dans le protocole.)

b) *association de sécurité liée à un item unique* – Association de sécurité s'appliquant à une valeur de données de présentation unique, bénéficiant d'une protection indépendante, qui n'est pas associée à un contexte de présentation; les attributs de l'association de sécurité sont indiqués de manière explicite et associés au codage de la valeur de données de présentation. La durée de vie d'une association de sécurité liée à un item unique est limitée à la durée de vie de l'une des valeurs de données de présentation;

c) *association de sécurité liée à un contexte de présentation* – Association de sécurité qui est établie en même temps qu'un contexte de présentation de protection, et qui s'applique à toutes les valeurs de données de présentation envoyées dans un seul sens de ce contexte; les attributs de l'association de sécurité sont associés de façon explicite au codage de la première valeur de données de présentation selon le contexte de présentation de protection. Ce type d'association de sécurité ne peut s'appliquer que si la protection est assurée par l'utilisation conjointe du service de présentation OSI et du protocole spécifiés respectivement dans la Rec. UIT-T X.216 | ISO/CEI 8822 et dans la Rec. UIT-T X.226 | ISO/CEI 8823-1. La durée de vie de l'association de sécurité est la même que celle du contexte de présentation de protection correspondant.

Le déroulement d'une transformation de sécurité peut être régi par l'information locale de l'état de sécurité et/ou par les paramètres qui sont transférés ou stockés avec les valeurs de données codées. L'information locale sur la sécurité peut être maintenue d'une application de transformation de sécurité à la suivante, dans la même association de sécurité. A titre d'exemple, les transformations assurant l'intégrité de la séquence de valeurs de données de présentation dans une association de sécurité, l'information d'état telle qu'un numéro de séquence d'intégrité ou une valeur de chaînage cryptographique sera retenue d'une application de la transformation à la suivante. Les valeurs des paramètres statiques (voir le 7.1.3) sont également conservées tout au long de l'association de sécurité.

7.1.3 Paramètres d'une transformation de sécurité

Quand on utilise une transformation de sécurité, il faut éventuellement acheminer les valeurs des paramètres entre des fonctions de codage et de décodage en même temps que les valeurs de données transformées. Les paramètres sont de deux types:

a) *paramètres statiques* – Ces paramètres maintiennent des valeurs constantes tout au long d'une association de sécurité; ils sont spécifiés par le codeur de données quand la transformation de sécurité est appliquée pour la première fois dans une association de sécurité ou plus tôt;

b) *paramètres dynamiques* – Les valeurs de ces paramètres peuvent changer dynamiquement pendant l'emploi de la transformation dans une association de sécurité; le codeur des données signale de telles modifications dans le flux de données.

Exemples de paramètres statiques:

– le ou les identificateurs du ou des algorithmes utilisés dans une transformation de sécurité;

– si nécessaire, le mode de fonctionnement d'un algorithme;

– la ou les clés et le ou les identificateurs de la ou des clés qu'il faut utiliser avec le ou les algorithmes ci-dessus;

– si nécessaire, la ou les valeurs du ou des vecteurs d'initialisation.

Un exemple de paramètre dynamique est une clé qui est changée après une certaine période d'utilisation.

Les valeurs des paramètres peuvent être codées sans protection ou peuvent elles-mêmes nécessiter une protection. Les paramètres non protégés sont acheminés dans des champs explicites de la syntaxe de protection de transfert qui prend en charge la transformation de sécurité. Les paramètres protégés sont considérés comme des apports au processus de codage de la transformation de sécurité, tout comme la valeur à protéger. Les règles de transformation de sécurité doivent stipuler comment ces paramètres doivent être représentés, comment la représentation est combinée avec la valeur de syntaxe abstraite codée et comment le résultat est traité pour produire une valeur de donnée ASN.1 pour transfert ou stockage.

NOTE – A titre d'exemple de l'acheminement d'un paramètre protégé, se référer à la définition de la transformation de sécurité GULS SIGNED au D.4.

Les données des paramètres (telles que les clés) requises par les transformations de sécurité peuvent aussi être obtenues par d'autres moyens, notamment:

– des échanges de protocole de couche Application antérieurs (tels qu'un échange de sécurité à dérivation de clé acheminé par l'élément SESE);

– des moyens locaux (tels que l'insertion manuelle de clés).

7.1.4 Etablissement des règles de codage initiales

Les règles s'appliquant au processus de codage initial (et de décodage final) (modélisé dans les cases a) et f) de la Figure 2) sont déterminées de la manière suivante:

a) une transformation de sécurité peut pourvoir à l'acheminement d'une indication des règles de codage initiales en tant que paramètre statique (protégé ou non) de la transformation de sécurité;

b) par défaut, chaque spécification de transformation de sécurité identifie les règles de codage initiales par défaut.

NOTE – Quand on utilise des signatures numériques pour la non-répudiation, l'item transformé (c'est-à-dire les données signées) doit éventuellement être stocké dans un système récepteur et/ou être relié à une autre entité. Dans de telles circonstances, la connaissance des règles de codage initiales utilisées dans le calcul de la signature peut être préservée. Dans le cas d'une signature numérique, il est recommandé d'utiliser les règles de codage par défaut identifiées dans la spécification de la transformation de sécurité qui sera utilisée. De cette manière, les connaissances requises peuvent être préservées par stockage ou par relais de l'identificateur de transformation de sécurité avec la signature.

7.1.5 Utilisation des transformations de sécurité sur une connexion de présentation OSI

La couche Présentation OSI associe une syntaxe de transfert à chaque syntaxe abstraite utilisée. Quand on utilise une transformation de sécurité, la syntaxe de transfert est considérée comme une syntaxe de protection de transfert.

En vertu de la Rec. X.208 du CCITT | ISO/CEI 8824, les valeurs de données de présentation doivent être transférées:

a) dans un contexte de présentation négocié; ou

b) hors d'un contexte de présentation (en tant qu'option quand on utilise la notation ASN.1 EXTERNAL ou EMBEDDED PDV).

Dans les deux cas, une valeur de données de présentation à protéger est représentée selon une syntaxe de transfert de protection. Une syntaxe de transfert de protection définie conformément à la Rec. UIT-T X.833 | ISO/CEI 11586-4 s'applique à la communication des paramètres de transformation de sécurité statiques et dynamiques.

Le cas a) ci-dessus fait intervenir un contexte de protection de présentation. Toutes les valeurs des données de présentation transférées dans un sens dans un tel contexte de présentation sont protégées au moyen de la même transformation de sécurité et sont régies par la seule association de sécurité. Quand un contexte de protection de présentation est établi (au moyen des procédures pour l'établissement d'un contexte de présentation spécifié dans la Rec. UIT-T X.216 | ISO/CEI 8822 et dans la Rec. UIT-T X.226 | ISO 8823-1), la première valeur de données de présentation dans chaque sens, selon ce contexte de présentation:

a) doit faire référence à une association de sécurité établie extérieurement; ou

b) doit définir une nouvelle association de sécurité liée à un contexte de présentation.

Quand une valeur de données de présentation n'est pas codée selon un contexte de présentation, la valeur de présentation:

a) doit faire une référence explicite à une association de sécurité établie extérieurement; ou

b) doit définir une nouvelle association de sécurité liée à un élément unique.

Sur une même connexion de présentation OSI, différentes associations de sécurité s'appliquent à chaque sens de transfert. Ces associations de sécurité peuvent utiliser la même transformation de sécurité, mais ne sont pas tenues de le faire.

NOTE – Les restrictions ci-dessus (à savoir l'obligation, lors de l'utilisation du protocole de présentation OSI, d'appliquer des associations de sécurité différentes à chaque sens de transfert) garantissent qu'aucune variable d'état cryptographique n'est commune aux deux sens de transfert. L'existence de variables communes aux deux sens de transfert, entraînerait la nécessité d'utiliser des éléments de protocole de gestion d'état complexes dans la couche Présentation pour gérer de tels événements lors de la resynchronisation de la couche Session. En pratique, il est peu probable que des associations de sécurité définies séparément pour les deux sens de transfert aient des attributs communs, dérivés d'une association de sécurité d'un niveau plus général.

7.1.6 Utilisation des transformations de sécurité indépendamment du protocole de présentation OSI

Les transformations de sécurité peuvent être utilisées indépendamment du protocole de présentation OSI, par exemple pour la protection au cours du stockage. Les concepts et les procédures décrits aux 7.1.2 à 7.1.5 s'appliquent avec les restrictions ci-après.

Toutes les valeurs de données de présentation protégées sont représentées à l'extérieur des contextes de présentation.

Une association de sécurité liée à un item unique ou établie extérieurement peut être employée. Les associations de sécurité liées à un contexte de présentation ne peuvent pas être appliquées. Quand l'information est protégée non pas en vue d'un échange, mais seulement de son utilisation par l'expéditeur, les transformations de sécurité peuvent également être employées sans association de sécurité.

Si l'on utilise une association de sécurité établie extérieurement, la durée de vie de cette association doit couvrir la durée du stockage des données protégées.

7.2 Notation pour spécifier les transformations de sécurité

La spécification des transformations de sécurité englobe la spécification de données élémentaires qui doivent être reconnues par la structure de syntaxe de protection de transfert. La définition suivante de la classe d'objets informationnels ASN.1 (voir la Rec. UIT-T X.681 | ISO/CEI 8824-2) est donnée à cet effet:

```
SECURITY-TRANSFORMATION ::= CLASS
-- Cette définition de classe d'objet d'information doit être utilisée pour
-- spécifier une instance particulière d'une transformation de sécurité.
{
  &sT-Identifier      OBJECT IDENTIFIER UNIQUE,
  -- Identificateur à utiliser pour signaler l'application
  -- de la transformation de sécurité en question
```

```
    &initialEncodingRules OBJECT IDENTIFIER
        DEFAULT {joint-iso-ccitt asn1 (1) ber-derived (2)
        canonical-encoding (0)},
    -- Règles de codage initial par défaut, produisant une chaîne
    -- binaire avant l'application du processus de codage d'une
    -- transformation de sécurité
    &StaticUnprotectedParm OPTIONAL,
    -- Type ASN.1 pour l'acheminement de paramètres statiques non protégés
    &DynamicUnprotectedParm OPTIONAL,
    -- Type ASN.1 pour l'acheminement de paramètres dynamiques non protégés
    &XformedDataType,
    -- Type ASN.1 de la valeur ASN.1 produite par le processus de
    -- codage de la transformation de sécurité
    &QualifierType  OPTIONAL
    -- &QualifierType spécifie le type ASN.1 du paramètre qualificateur
    -- utilisé avec la notation PROTECTED-Q.
}
WITH SYNTAX
-- La syntaxe suivante sert à spécifier une transformation
-- de sécurité donnée.
{
    IDENTIFIER                      &sT-Identifier
    [ INITIAL-ENCODING-RULES        &initialEncodingRules ]
    [ STATIC-UNPROT-PARM            &StaticUnprotectedParm ]
    [ DYNAMIC-UNPROT-PARM           &DynamicUnprotectedParm ]
    XFORMED-DATA-TYPE               &XformedDataType
    [ QUALIFIER-TYPE                &QualifierType ]
}
```

Des exemples d'utilisation de cette notation sont donnés à l'Annexe D.

La spécification de la transformation de sécurité doit aussi donner les précisions suivantes (bien qu'aucune notation formelle prenant en charge une telle spécification ne soit donnée dans la présente Recommandation | Norme internationale):

- *processus de codage* – Une description du processus de transformation qui est appliqué, du côté codage, à l'item non protégé et aux paramètres protégés transférés afin de produire la valeur transformée résultante (qui est une valeur ASN.1 du type &XformedDataType);
- *apports locaux au processus de codage* – La liste des apports au processus de codage dont l'origine est locale;
- *processus de décodage* – Une description du processus de transformation appliqué, à l'extrémité décodage, à la valeur transformée reçue ou récupérée (qui est du type &XformedDataType) afin de produire la chaîne binaire non protégée résultante (s'il y en a une) et les valeurs des paramètres protégés transférés;
- *apports locaux au processus de décodage* – La liste des apports au processus de décodage dont l'origine est locale;
- *restitutions du processus de décodage* – Une liste des restitutions de processus de décodage (peut ou ne peut pas inclure une valeur récupérée de l'item non protégé);
- *paramètres* – Une description de la signification sémantique de tous les paramètres, des valeurs par défaut pour les paramètres et des circonstances dans lesquelles les changements de paramètre dynamique devraient survenir;
- *qualificateurs de transformation* – Une description des règles s'appliquant aux qualificateurs (s'il y en a) de transformation, spécifiés par l'invocateur, qui s'appliquent à cette transformation;
- *erreurs* – Une description des situations d'erreur qui peuvent survenir au cours du processus de décodage.

8 Notation de syntaxe abstraite pour la protection sélective des champs

La notation de syntaxe abstraite suivante sert à spécifier les besoins de protection abstraite pour un type de donnée ASN.1 sélectionné. La protection acquise est mise en correspondance avec une transformation de sécurité d'une série qui fournit (à un niveau abstrait) la forme de protection requise. Certaines transformations de sécurité acceptent des qualificateurs d'entrée pour gérer le fonctionnement de la protection requise, par exemple, l'identificateur pour l'association de sécurité pour laquelle la protection doit être appliquée. Dans ces cas, une extension de la notation de base a été définie afin de permettre à l'utilisateur de la notation de spécifier les qualificateurs.

Le présent article spécifie:

a) la notation de syntaxe abstraite protégée de base pour spécifier les besoins de protection abstraite pour un champ sélectionné dans une spécification de syntaxe abstraite;

b) la notation de syntaxe abstraite protégée qualifiée pour spécifier les besoins de protection abstraite, ainsi qu'un qualificateur associé, pour un champ sélectionné dans une spécification de syntaxe abstraite;

c) la notation de mappage protecteur pour spécifier les mises en correspondance éventuelles avec une ou plusieurs transformations de sécurité qui assurent la protection requise.

8.1 Notation de base

Pour aider le rédacteur d'une syntaxe abstraite à indiquer les besoins de protection sélective des champs, le type paramétré ASN.1 suivant a été défini (voir la Rec. UIT-T X.683 | ISO/CEI 8824-4):

```
PROTECTED {BaseType, PROTECTION-MAPPING: protectionReqd}  ::=
CHOICE
{
  dirEncrypt BIT STRING (CONSTRAINED BY {BaseType
        -- dirEncrypt ne doit être utilisé qu'avec
        -- dirEncryptedTransformation,
        -- et produit le même codage que
        -- le type ENCRYPTED X.509/9594-8-- }),
  dirSign   SEQUENCE
       {
         baseType BaseType  OPTIONAL,
          -- Doit être présent pour dirSignedTransformation
          -- et doit être omis pour
          -- dirSignatureTransformation
         algorithmId AlgorithmIdentifier,
         encipheredHash BIT STRING (CONSTRAINED BY
              {BaseType -- contient le codage compacté et chiffré
                        -- d'une valeur de BaseType --})
       }
        -- dirSign ne doit être utilisé qu'avec
        -- dirSignedTransformation ou
        -- dirSignatureTransformation seulement, et produit le
        -- même codage que
        -- le type SIGNED ou SIGNATURE X.509/9594-8 correspondant--,
  noTransform [0] BaseType,
        -- noTransform n'appelle pas de transformation de sécurité.
        -- Selon la politique de sécurité, noTransform peut être utilisé
        -- si une protection adéquate est assurée par les couches basses
        -- et si toute application au travers de laquelle les données peuvent
        -- être relayées garantit le maintien de la protection requise.
        -- Cette variante ne peut être utilisée que si protectionReqd.&bypass
        -- Permitted a la valeur TRUE.
```

```
    direct    [1] SyntaxStructure
                {{protectionReqd.&SecurityTransformation}},
              -- direct produit une valeur de syntaxe de transfert de protection
              -- qui est codée en utilisant les mêmes règles que l'ASN.1
              -- enveloppant. (Le type SyntaxStructure est importé de la
              -- Rec. UIT-T X.833 | ISO/CEI 11586-4)
    embedded  [2] EMBEDDED PDV (WITH COMPONENTS {
                identification (WITH COMPONENTS {
                    presentation-context-id,
                    context-negotiation (WITH COMPONENTS {
                        transfer-syntax (CONSTRAINED BY
                            {OBJECT IDENTIFIER :
                            protectionReqd.&protTransferSyntax})}),
                    transfer-syntax (CONSTRAINED BY
                            {OBJECT IDENTIFIER :
                            protectionReqd.&protTransferSyntax})}),
              data-value (WITH COMPONENTS {notation (BaseType)})
               -- La valeur de donnée est une valeur de type BaseType
              })
}
-- BaseType est le type à protéger, et protectionReqd est un objet ASN.1
-- de la classe PROTECTION-MAPPING. L'emploi de PROTECTED
-- nécessite que l'on importe dans le module d'utilisateur le
-- type paramétré PROTECTED avec la définition nécessaire de l'objet
-- PROTECTION-MAPPING.
```

La classe d'objet PROTECTION-MAPPING et sa signification sont examinées au 8.3. La série d'objets admissibles pour «protectionReqd» variera selon les spécifications de syntaxe abstraite, qui dépend de la gamme de transformations distinctes requise. Les mises en correspondance des objets PROTECTION-MAPPING avec les transformations sont contenues dans une série de définitions d'objet PROTECTION-MAPPING. Cette série de définitions peut être spécifiée dans un module ASN.1 séparé de la spécification de syntaxe abstraite (indépendant du mécanisme) et de la définition de la transformation (indépendant de l'application).

Les diverses variantes spécifiées dans le CHOIX sont prévues pour être utilisées dans les circonstances suivantes:

- *dirEncrypt et dirSign* – Ces variantes produisent le type &XformedDataType de la transformation de sécurité utilisée. Ces variantes sont prévues afin de disposer d'un moyen par lequel la notation PROTECTED puisse produire des codages binaires identiques à ceux des types paramétrés ENCRYPTED, SIGNED et SIGNATURE définis dans la Rec. UIT-T X.509 | ISO/CEI 9594-8;

- *noTransform* – Cette variante n'utilise pas de transformation de sécurité. Elle est permise si le mappage de protection en vigueur (voir 8.3 et 8.4) indique &bypassPermitted = TRUE. L'item est codé dans sa forme non protégée. Selon la police de sécurité, noTransform peut être utilisé si une protection adéquate est assurée par les couches basses et s'il est certain que des relais d'application par lesquels les données peuvent transiter maintiennent la protection requise;

- *direct* – Cette variante importe directement une valeur de syntaxe de transfert de protection, telle que définie dans la Rec. UIT-T X.833 | ISO/CEI 11586-4, dans la spécification ASN.1 enveloppante. Elle permet l'utilisation d'une association de sécurité établie extérieurement ou d'une association de sécurité liée à un élément unique. Elle ne permet pas l'utilisation d'un contexte de présentation négocié. Les règles de codage utilisées pour le codage de la structure de la syntaxe de transfert protectrice [modélisée dans les pavés c) et d) de la Figure 2 du 7.1] doivent être les mêmes que celles utilisées pour le type ASN.1 enveloppant la notation PROTECTED;

- *embedded* – Cette variante assure la plus grande souplesse, permet d'associer la protection à un contexte de présentation négocié et permet d'utiliser une syntaxe de transfert de protection différente de celle définie dans la Rec. UIT-T X.833 | ISO/CEI 11586-4.

NOTE – Il est recommandé de déterminer comme suit l'utilisation de ces variantes:

a) utiliser la variante «direct» si aucun des points b), c) et d) ne s'applique;

b) lorsque la compatibilité binaire avec les types paramétrés ENCRYPTED, SIGNED ou SIGNATURE est requise pour des raisons de compatibilité amont, utiliser la variante «dirEncrypt» ou «dirSign» (selon celle qui est applicable);

c) lorsque le mappage de protection en vigueur &bypassPermitted = TRUE et que la politique de sécurité le permet, utiliser la variante «noTransform»;

d) lorsqu'il est nécessaire d'associer la protection à un contexte de présentation négociée, utiliser la variante «embedded».

Une situation d'erreur peut survenir dans le système de décodage lors du traitement d'une valeur dans un champ protégé. On peut taiter une telle situation d'erreur au moyen de la notation de traitement des erreurs ASN.1 définie dans la Rec. UIT-T X.682 | ISO/CEI 8824-3.

Des exemples de l'utilisation de cette notation figurent au I.1.

8.2 Notation avec qualificateur de transformation

En tant qu'alternative de la notation PROTECTED décrite au 8.1, la notation PROTECTED-Q permet à son utilisateur de fournir en outre un paramètre qualificateur. Ces paramètres qualificateurs sont utilisés pour l'une des raisons suivantes ou pour les deux:

a) pour identifier une association de sécurité spécifique établie extérieurement;

b) pour fournir un ou plusieurs paramètres qui seront utilisés par la transformation de sécurité, tels qu'un algorithme, un mode de fonctionnement et/ou des identificateurs de clé.

NOTE – Certains identificateurs d'algorithme peuvent faire intervenir un mode de fonctionnement particulier. Dans d'autres cas, le mode de fonctionnement peut être spécifié comme un paramètre additionnel.

On peut spécifier plusieurs qualificateurs en utilisant le type ASN.1 SEQUENCE ou SET approprié. Dans le système de codage, le qualificateur est utilisé par les fonctions du système local pour déterminer l'objet SSO approprié et/ou pour acheminer un paramètre jusqu'à cet objet SSO. Un qualificateur acheminé jusqu'à un objet SSO doit être compatible avec la transformation de sécurité qui est en cours d'utilisation, comme indiqué dans la spécification de la transformation de sécurité. Quand le mappage protecteur spécifié permet un choix de transformations de sécurité, celle qui est sélectionnée, quel que soit le cas, doit avoir un type &QualifierType compatible avec la valeur spécifiée par l'utilisateur de la notation PROTECTED-Q. La valeur du qualificateur peut être acheminée (mais elle ne l'est pas nécessairement) au système de décodage dans la syntaxe de protection de transfert (telle que l'identificateur de l'association de sécurité établie extérieurement, ou en tant que paramètre de la transformation de sécurité).

Le type paramétré ASN.1 suivant (voir la Rec. UIT-T X.683 | ISO/CEI 8824-4) a été défini:

```
PROTECTED-Q {BaseType, PROTECTION-MAPPING: protectionReqd,
  PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier} ::=
    PROTECTED {BaseType, protectionReqd} (CONSTRAINED BY
    {PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier
        -- La valeur du qualificateur doit être mise
        -- à la disposition de la transformation de sécurité utilisée
      })
  -- BaseType est le type à protéger, et protectionReqd est un objet de la classe PROTECTION-MAPPING.
  -- L'utilisation de PROTECTED nécessite l'importation, dans le module
  -- de l'utilisateur, du type paramétré PROTECTED, avec la définition nécessaire
  -- de l'objet PROTECTION-MAPPING.
```

Des exemples d'utilisation de cette notation figurent aux I.2 et I.3.

8.3 Mise en correspondance des besoins de protection et des transformations de sécurité

Un mappage de protection établit une relation entre un besoin de protection, identifié par son nom dans une spécification de syntaxe abstraite, et une transformation spécifique qu'il y a lieu d'utiliser pour répondre à ce besoin. Ce concept est introduit pour permettre de spécifier de tels mappages indépendamment de la spécification de la syntaxe abstraite principale, qui peut alors être indépendante du mécanisme. Pour la protection nommée dans une syntaxe abstraite, la transformation effectivement utilisée doit être différente selon le contexte d'application.

Un mappage de protection peut forcer à choisir une transformation de sécurité de la manière suivante:

- en donnant une liste de transformations de sécurité, dans laquelle sera choisie, au moment opportun, la transformation nécessaire sur la base de la politique de sécurité locale et d'autres considérations du système local;
- en énonçant des règles de sélection spécialisées.

Voici des exemples de mappage de protection, qui sont intégralement définis à l'Annexe E:

- *confidentialité* – Données protégées quant à la confidentialité, par chiffrement/déchiffrement, mais auxquelles le chiffrement/déchiffrement peut ne pas être appliqué, si la politique de sécurité le prescrit;
- *chiffré* – Chiffrement/déchiffrement par un type d'algorithme non spécifié;
- *signé* – Production/vérification d'une signature numérique, associée aux données signées;
- *signature* – Production/vérification d'une signature numérique pour le transfert, indépendamment des données signées.

D'autres mappages sont possibles, par exemple la projection spécifique sur des transformations pour chiffrement par clé publique, chiffrement symétrique, cachetage, hachage ou chiffrement unidirectionnel.

8.4 Notation pour spécifier des mappages de protection

La définition de classe d'objet d'information ASN.1 suivante (voir la Rec. UIT-T X.681 | ISO/CEI 8824-2) est proposée pour définir les mappages spécifiques:

```
PROTECTION-MAPPING ::=  CLASS
{
    &SecurityTransformation SECURITY-TRANSFORMATION,
    -- &SecurityTransformation spécifie une série d'objets ASN.1 de la classe SECURITY-TRANSFORMATION.
    -- L'utilisation du mappage de protection en question sous-entend l'utilisation de l'une des transformations
    -- qui ont été spécifiées, dont le choix est laissé au système de codage. Les règles de sélection entre ces
    -- transformations de sécurité peuvent être précisées dans des observations.
    &protTransferSyntax OBJECT IDENTIFIER
            DEFAULT {joint-iso-ccitt genericULS (20)
            generalTransferSyntax (2)},
    -- Identifie la syntaxe de protection de transfert à utiliser
    -- dans un codage EMDEDDED PDV pour l'option  "embedded".
    &bypassPermitted BOOLEAN DEFAULT FALSE
    -- Indique s'il est permis de passer outre à la protection
}
WITH SYNTAX
{
    SECURITY-TRANSFORMATION          &SecurityTransformation
    [ PROTECTING-TRANSFER-SYNTAX     &protTransferSyntax ]
    [ BYPASS-PERMITTED               &bypassPermitted ]
}
```

9 Conformité

Un système réputé conforme à la présente Recommandation | Norme internationale l'est également par rapport aux échanges de sécurité GULS ou aux transformations de sécurité spécifiées dans les Annexes C et D.

a) Par l'utilisation d'un échange de sécurité quelconque spécifié dans l'Annexe C, tel qu'identifié par l'identificateur d'objet ASN.1 pour le module «GulsSecurityExchanges» donné à l'Annexe C, le système prendra en charge l'ASN.1 applicable et toute stipulation associée de l'Annexe C.

b) Lors de l'utilisation d'une des transformations de sécurité spécifiée à l'Annexe D, telle qu'identifiée par l'identificateur d'objet ASN.1 pour le module «GulsSecurityTransformations» donné à l'Annexe D, le système prendra en charge l'ASN.1 applicable et toute stipulation associée de l'Annexe D.

Les besoins de conformité statiques et dynamiques particuliers sont énoncés dans les paragraphes applicables des Annexes C et D.

La mise en œuvre des structures définies dans les Annexes C, D et E relève du choix de l'utilisateur de la présente Recommandation | Norme internationale et n'est pas une condition de conformité obligatoire.

Annexe A

Définitions de l'ASN.1

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Le module ASN.1 suivant donne les spécifications ASN.1 définitives pour le corps de la présente Recommandation | Norme internationale.

```
Notation {joint-iso-ccitt genericULS (20)
        modules (1) notation (1)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

-- EXPORTE tout --

IMPORTS
  -- Des normes d'annuaire: --
  informationFramework, selectedAttributeTypes,
  authenticationFramework
      FROM UsefulDefinitions {joint-iso-ccitt ds (5) module (1)
            usefulDefinitions (0) 2}
  Name
      FROM InformationFramework          informationFramework
  UniqueIdentifier
      FROM SelectedAttributeTypes        selectedAttributeTypes
  AlgorithmIdentifier
      FROM AuthenticationFramework       authenticationFramework

  -- Des autres modules GULS: --
  genericProtectingTransferSyntax
      FROM ObjectIdentifiers {joint-iso-ccitt genericULS (20)
      modules (1) objectIdentifiers (0)}
  SyntaxStructure { }
      FROM GenericProtectingTransferSyntax
            genericProtectingTransferSyntax;

-- ************************************************ --
-- Notation pour l'identité de sécurité et les identificateurs SA --
-- ************************************************ --

-- Les valeurs de type SecurityIdentity sont utilisées pour identifier les entités
-- qui attribuent des identificateurs d'associations de sécurité
-- établies extérieurement et pour d'autres besoins liés à la
-- sécurité et nécessitant des identificateurs globalement uniques.

SecurityIdentity ::=  CHOICE
{
  directoryName          Name,
  objectIdentifier       OBJECT IDENTIFIER
}
ExternalSAID ::=  SEQUENCE
{
  localSAID              INTEGER,
  assignerIdentity       SecurityIdentity   OPTIONAL
  -- Identité du système qui a attribué la valeur entière
}
```

```
-- ****************************************** --
-- Notation pour spécifier les échanges de sécurité --
-- ****************************************** --

SECURITY-EXCHANGE ::= CLASS
-- Cette définition de classe d'objet d'information doit être utilisée pour
-- spécifier une instance particulière d'un échange de sécurité.
{
  &SE-Items        SEC-EXCHG-ITEM,
  -- Ceci est un ensemble d'objets informationnels comprenant un
  -- ensemble d'items d'échange de sécurité
  &sE-Identifier   Identifier        UNIQUE
  -- Un identificateur local ou global pour l'échange de
  -- sécurité en question
}
WITH SYNTAX
-- La syntaxe suivante sert à spécifier
-- un échange de sécurité particulier.
{
  SE-ITEMS         &SE-Items
  IDENTIFIER       &sE-Identifier
}
Identifier ::= CHOICE
{
  local            INTEGER,
  global           OBJECT IDENTIFIER
}

SEC-EXCHG-ITEM ::= CLASS
{
  &ItemType,
      -- Type ASN.1 de cet item d'échange
  &itemId          INTEGER,
      -- Identificateur de cet item, soit 1, 2, 3...
  &Errors          SE-ERROR          OPTIONAL
      -- Liste facultative des erreurs pouvant résulter
      -- du transfert de cet item
}
WITH SYNTAX
{
  ITEM-TYPE        &ItemType
  ITEM-ID          &itemId
  [ERRORS          &Errors]
}

SE-ERROR ::= CLASS
{
  &ParameterType   OPTIONAL,
  -- Type ASN.1 d'un paramètre qui accompagnera la signalisation
  -- d'une situation d'erreur à l'expéditeur de l'item SEI
  &errorCode       Identifier        UNIQUE
  -- Un identificateur utilisé pour signaler la situation d'erreur
  -- à l'expéditeur de l'item SEI
}
WITH SYNTAX
{
  [PARAMETER       &ParameterType]
  ERROR-CODE       &errorCode
}
```

```
-- ********************************************** --
-- Notation pour spécifier les transformations de sécurité --
-- ********************************************** --

SECURITY-TRANSFORMATION ::= CLASS
-- Cette définition de classe d'objet informationnel sera utilisée
-- pour spécifier une instance particulière d'une transformation de sécurité.
{
   &sT-Identifier      OBJECT IDENTIFIER UNIQUE,
   -- Identificateur à utiliser pour signaler l'application de la
   -- transformation de sécurité en question
   &initialEncodingRules OBJECT IDENTIFIER
        DEFAULT {joint-iso-ccitt asn1 (1) ber-derived (2)
        canonical-encoding (0)},
   -- Les règles de codage initiales par défaut doivent produire une chaîne
   -- binaire avant d'appliquer le processus de codage
   -- de la transformation de sécurité.
   &StaticUnprotectedParm OPTIONAL,
   -- Type ASN.1 pour acheminer des paramètres statiques non protégés
   &DynamicUnprotectedParm OPTIONAL,
   -- Type ASN.1 pour acheminer des paramètres dynamiques non protégés
   &XformedDataType,
   -- Type ASN.1 de la valeur ASN.1 produite par le processus de
   -- codage de la transformation de sécurité
   &QualifierType  OPTIONAL
   -- &QualifierType spécifie le type ASN.1 du paramètre
   -- qualificateur utilisé avec la notation PROTECTED-Q.

}
WITH SYNTAX
-- La syntaxe suivante est utilisée pour spécifier une transformation
-- de sécurité particulière.
{
   IDENTIFIER                          &sT-Identifier
   [ INITIAL-ENCODING-RULES            &initialEncodingRules ]
   [ STATIC-UNPROT-PARM                &StaticUnprotectedParm ]
   [ DYNAMIC-UNPROT-PARM               &DynamicUnprotectedParm ]
   XFORMED-DATA-TYPE                   &XformedDataType
   [ QUALIFIER-TYPE                    &QualifierType ]
}


-- ********************************************** --
-- Notation pour spécifier la protection sélective des champs --
-- ********************************************** --

PROTECTED {BaseType, PROTECTION-MAPPING: protectionReqd} ::=
CHOICE
{
   dirEncrypt BIT STRING (CONSTRAINED BY {BaseType
            -- dirEncrypt doit être utilisé avec
            -- dirEncryptedTransformation seulement,
            -- et produit le même codage que le
            -- type ENCRYPTED de X.509/9594-8 -- },
   dirSign    SEQUENCE
          {
             baseType BaseType  OPTIONAL,
               -- doit être présent pour dirSignedTransformation
               -- et doit être omis pour
               -- dirSignatureTransformation
```

```
            algorithmId AlgorithmIdentifier,
            encipheredHash BIT STRING (CONSTRAINED BY
                {BaseType -- contient le hachage chiffré d'une
                          -- valeur de BaseType --})
        }
        -- dirSign doit être utilisé avec
        -- dirSignedTransformation ou
        -- dirSignatureTransformation seulement, et produit
        -- le même codage que le type
        -- SIGNED ou SIGNATURE X.509/9594-8 correspondant --,
  noTransform [0] BaseType,
        -- noTransform n'appelle pas de transformation de sécurité.
        -- Selon la politique de sécurité, noTransform peut être utilisé
        -- si une protection adéquate est assurée par les couches basses
        -- et si toute application au travers de laquelle les données peuvent
        -- être relayées garantit le maintien de la protection
        -- requise. Cette variante ne peut être utilisée que
        -- si protectionReqd.&bypassPermitted a la valeur TRUE,
  direct    [1] SyntaxStructure
            {{protectionReqd.&SecurityTransformation}},
        -- direct produit une valeur de syntaxe de transfert de protection
        -- qui est codée en utilisant les mêmes règles que l'ASN.1
        -- enveloppant. (Le type SyntaxStructure est importé de la
        -- Rec. UIT-T X.833 | ISO/CEI 11586-3)
    embedded   [2] EMBEDDED PDV (WITH COMPONENTS {
                identification (WITH COMPONENTS {
                    presentation-context-id,
                    context-negotiation (WITH COMPONENTS {
                        transfer-syntax (CONSTRAINED BY
                          {OBJECT IDENTIFIER :
                          protectionReqd.&protTransferSyntax})}),
                    transfer-syntax (CONSTRAINED BY
                        {OBJECT IDENTIFIER :
                        protectionReqd.&protTransferSyntax})}),
          data-value (WITH COMPONENTS {notation (BaseType)})
          -- La valeur de données est une valeur de type BaseType
          })
}
-- BaseType est le type à protéger, et protectionReqd est un objet ASN.1
-- de la classe PROTECTION-MAPPING. L'emploi de PROTECTED
-- nécessite d'importer dans le module d'utilisateur le type paramétré
-- PROTECTED, avec la définition nécessaire de
-- l'objet PROTECTION-MAPPING.

PROTECTED-Q {BaseType, PROTECTION-MAPPING: protectionReqd,
  PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier}  ::=
    PROTECTED {BaseType, protectionReqd} (CONSTRAINED BY
    {PROTECTION-MAPPING.&SecurityTransformation.&QualifierType: qualifier
        -- La valeur du qualificateur doit être communiquée
        -- à la transformation de sécurité utilisée
        })
  -- BaseType est le type à protéger, et protectionReqd est un objet ASN.1
  -- de la classe PROTECTION-MAPPING. L'emploi de PROTECTED
  -- nécessite d'importer dans le module d'utilisateur le type paramétré
  -- PROTECTED, avec la définition nécessaire de
  -- l'objet PROTECTION-MAPPING.
```

```
-- ****************************************** --
-- Notation pour spécifier les mappages de protection --
-- ****************************************** --

PROTECTION-MAPPING ::=  CLASS
{
    &SecurityTransformation SECURITY-TRANSFORMATION,
     -- &SecurityTransformation spécifie un ensemble d'objets ASN.1 de la classe
     -- SECURITY-TRANSFORMATION. L'emploi du mappage de
     -- protection en question sous-entend l'utilisation
     -- de l'une des transformations spécifiées, le choix étant
     -- laissé au système de codage. Les règles du choix entre
     -- ces transformations peuvent être spécifiées dans des observations.
    &protTransferSyntax OBJECT IDENTIFIER
       DEFAULT {joint-iso-ccitt genericULS (20)
       generalTransferSyntax (2)},
     -- Identifie la syntaxe de protection de transfert qu'il y a lieu
     -- d'utiliser dans un codage EMBEDDED PDV pour
     -- l'option "embedded".
    &bypassPermitted BOOLEAN DEFAULT FALSE
     -- indique s'il est permis de passer outre à la protection
}
WITH SYNTAX
{
    SECURITY-TRANSFORMATION              &SecurityTransformation
    [ PROTECTING-TRANSFER-SYNTAX         &protTransferSyntax ]
    [ BYPASS-PERMITTED                   &bypassPermitted ]
}

END
```

Annexe B

Enregistrement des échanges de sécurité et des transformations de sécurité

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

B.1 Introduction

L'identification des échanges de sécurité et des transformations de sécurité en vue de leur utilisation conforme aux diverses parties de la spécification générique de la sécurité des couches supérieures nécessite que l'on nomme de tels objets informationnels de manière non ambiguë. La présente annexe spécifie les procédures d'attribution de tels noms.

B.2 Procédures d'enregistrement

Le présent paragraphe spécifie les procédures d'enregistrement des échanges de sécurité et des transformations de sécurité spécifiées:

a) dans les Recommandations UIT-T | Normes internationales;

b) par certaines organisations qui en ont besoin.

B.2.1 Enregistrement dans les Recommandations UIT-T | Normes internationales

Dans certains cas les noms des échanges de sécurité ou des transformations de sécurité sont spécifiés dans des Recommandations UIT-T | Normes internationales faisant référence à la présente Recommandation UIT-T | Norme internationale. Le nom doit être défini conformément à la Rec. X.660 du CCITT | ISO/CEI 9834-1. Les autorités d'enregistrement internationales responsables de ces types d'objets informationnels ne sont actuellement pas concernées.

La Recommandation UIT-T | Norme internationale faisant référence à la présente Recommandation UIT-T | Norme internationale devra affecter un nom conformément à la Rec. X.660 du CCITT | ISO/CEI 9834-1, sans qu'il soit nécessaire de se référer à la Rec. X.660 du CCITT | ISO/CEI 9834-1.

B.2.2 Enregistrement par une organisation qui en a le besoin

L'attribution de noms aux spécifications d'échanges de sécurité ou de transformations de sécurité devra être conforme aux procédures générales et à la forme spécifiées dans la Rec. X.660 du CCITT | ISO/CEI 9834-1.

Les organisations qui désirent attribuer de tels noms trouveront un supérieur approprié dans l'arbre de dénomination de la Rec. X.660 du CCITT | ISO/CEI 9834-1 et demanderont qu'un arc leur soit attribué.

NOTE – Ceci inclut les organes nationaux de l'ISO/CEI, les organisations ayant des indicateurs de codes internationaux attribués conformément à la Norme ISO 6523, les administrations de télécommunication et les exploitations privées reconnues (EPR).

B.3 Autres registres applicables

Les définitions des transformations de sécurité peuvent, mais cela n'est pas une obligation, utiliser les entrées du registre des algorithmes cryptographiques établi conformément à la Norme ISO/CEI 9979, en vue de leur utilisation possible comme paramètres des transformations de sécurité.

Annexe C

Spécification des échanges de sécurité

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Les échanges de sécurité peuvent être définis dans les Recommandations UIT-T | Normes internationales ou dans un autre cadre et être enregistrés par toute organisation qui a la capacité d'attribuer des identificateurs d'objet. Les définitions des échanges de sécurité devraient être rendues aussi universelles que possible afin de pouvoir être utilisées dans de nombreuses applications. La présente annexe définit certains échanges de sécurité qui sont généralement considérés comme utiles. Cela ne sous-entend aucunement que les applications ou leurs réalisations sont supposées utiliser les échanges de sécurité spécifiques définis ici plutôt que d'autres.

C.1 Echange d'authentification de l'annuaire (unilatéral)

L'échange de sécurité «dirAuthenticationOneWay» est fondé sur l'échange d'authentification utilisé dans le protocole d'Annuaire (Rec. UIT-T X.511 | ISO/CEI 9594-3) pour une authentification unilatérale, simple ou poussée, d'entité. Les précisions concernant l'item «credentials» («accréditifs») sont données dans la Rec. UIT-T X.511 | ISO/CEI 9594-3; la sémantique associée est décrite dans la Rec. UIT-T X.509 | ISO/CEI 9594-8.

```
dirAuthenticationOneWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS    {credentials}
  IDENTIFIER   global : {securityExchanges dir-authent-one-way (1)}
}
credentials    SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID     1
}
```

Cet échange de sécurité fait intervenir un item SEI transféré du déclarant au vérificateur. Aucune erreur n'est définie; la signalisation des situations d'erreur est laissée à d'autres protocoles d'application.

C.1.1 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions suivantes:

- *déclaration* – Le réalisateur doit préciser si l'application joue le rôle d'entité appelante (qui lance l'échange de sécurité), d'entité appelée (qui répond à l'échange lancé par l'autre système), ou les deux rôles;
- *conformité statique* – Une réalisation jouant le rôle d'entité appelante doit être capable de produire l'item d'échange de sécurité «credentials». Une réalisation jouant le rôle d'entité appelée doit être capable de traiter l'item d'échange de sécurité «credentials»;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe, la Rec. UIT-T X.511 | ISO/CEI 9594-3 et la Rec. UIT-T X.509 | ISO/CEI 9594-8.

C.2 Echange d'authentification de l'annuaire (bilatéral)

L'échange de sécurité «dirAuthenticationTwoWay» est fondé sur l'échange d'authentification utilisé dans le protocole d'Annuaire (Rec. UIT-T X.511 | ISO/CEI 9594-3) pour l'authentification mutuelle, simple ou poussée, des entités. Des précisions concernant l'item «credentials» («accréditifs») sont données dans la Rec. UIT-T X.511 | ISO/CEI 9594-3. La sémantique associée est décrite dans la Rec. UIT-T X.509 | ISO/CEI 9594-8.

```
dirAuthenticationTwoWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS    {initiatorCredentials | responderCredentials}
  IDENTIFIER   global : {securityExchanges dir-authent-two-way (2)}
}
initiatorCredentials     SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        1
  ERRORS  {authenticationFailure}
}
responderCredentials     SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE      DirectoryAbstractService.Credentials
  ITEM-ID          2
}
authenticationFailure    SE-ERROR ::=
{
  PARAMETER    DirectoryAbstractService.SecurityProblem
  ERROR-CODE    local : 1
}
```

L'échange de sécurité fait intervenir deux items d'échange de sécurité, dont le premier est transféré de l'entité appelante à l'entité appelée. Si une erreur est détectée, après le premier transfert, l'entité appelée doit mettre un terme à l'échange de sécurité. Facultativement, elle peut utiliser le code d'erreur «authenticationFailure» ou peut mettre fin à l'échange sans donner de raison. En l'absence d'erreur après le premier transfert, l'item SEI «responderCredentials» est transféré de l'entité appelée à l'entité appelante. Aucune erreur n'est définie pour le deuxième transfert d'item SEI; la signalisation des situations d'erreur relève d'un autre protocole d'application.

C.2.1 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle d'entité appelante (qui initie l'échange de sécurité), d'entité appelée (qui répond à l'échange lancé par l'autre système), ou les deux rôles;
- *conformité statique* – Une réalisation jouant le rôle d'entité appelante doit être capable de produire l'item d'échange de sécurité «initiatorCredentials» et être capable de traiter l'item d'échange de sécurité «responderCredentials». Une réalisation jouant le rôle d'entité appelée doit être capable de produire l'item d'échange de sécurité «responderCredentials» et de traiter l'item d'échange de sécurité «initiatorCredentials»;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe, la Rec. UIT-T X.511 | ISO/CEI 9594-3 et la Rec. UIT-T X.509 | ISO/CEI 9594-8.

C.3 Echange de sécurité à négociation simple

Un contexte d'application peut avoir la capacité de prendre en charge plus d'un échange de sécurité afin d'assurer les mêmes services de sécurité dans différents protocoles ou mécanismes de sécurité. La prise en charge d'échanges ou de mécanismes de sécurité autres permet l'interfonctionnement d'entités homologues qui mettent en œuvre l'une de ces autres possibilités.

Pour déterminer les échanges de sécurité qu'il y a lieu d'employer au moment opportun, on dispose de la Negotiation-SE. Celle-ci, qui est un objet de la classe SECURITY-EXCHANGE, est utilisée pour négocier des échanges de sécurité particuliers; cet objet informationnel est constitué d'un ou de plusieurs identificateurs d'échange de sécurité. La Negotiation-SE est utilisée par l'entité appelante pour proposer un ou plusieurs échanges de sécurité et par l'entité demandée pour indiquer lequel des choix proposés sera employé dans les opérations qui suivent. Negotiation-SE peut être utilisée à tout moment pour modifier les échanges de sécurité en cours d'utilisation.

Les contextes d'application qui nécessitent cette négociation doivent spécifier qu'ils utilisent Negotiation-SE.

Negotiation-SE est formée de deux items SEI, «offeredIds» et «acceptedIds», comme indiqué ci-dessous.

```
simpleNegotiationSE  SECURITY-EXCHANGE   ::=
{
  SE-ITEMS     {offeredIds | acceptedIds}
  IDENTIFIER   global : {securityExchanges simple-negotiation-se (3)}
}
offeredIds       SEC-EXCHG-ITEM     ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID      1
}
acceptedIds    SEC-EXCHG-ITEM     ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID      2
}

Negotiation-SEI ::=  SEQUENCE OF OBJECT IDENTIFIER
```

C.3.1 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle d'entité appelante (qui initie l'échange de sécurité), d'entité appelée (qui répond à l'échange lancé par l'autre système), ou les deux rôles;
- *conformité statique* – Une réalisation jouant le rôle d'entité appelante doit être capable de produire l'item d'échange de sécurité «offeredIds» et être capable de traiter l'item d'échange de sécurité «acceptedIds». Une réalisation jouant le rôle d'entité appelée doit être capable de produire l'item d'échange de sécurité «acceptedIds» et de traiter l'item d'échange de sécurité «offeredIds»;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe.

C.4 Spécification ASN.1 définitive

```
GulsSecurityExchanges {joint-iso-ccitt genericULS (20)
        modules (1) gulsSecurityExchanges (2)}
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

 -- EXPORTE tout --

IMPORTS
  securityExchanges, notation
    FROM ObjectIdentifiers {joint-iso-ccitt genericULS (20)
    modules (1) objectIdentifiers (0)}
  SECURITY-EXCHANGE, SEC-EXCHG-ITEM, SE-ERROR
    FROM Notation notation
  Credentials, SecurityProblem
    FROM DirectoryAbstractService {joint-iso-ccitt ds (5)
    module (1) directoryAbstractService (2) 2};
```

```
-- ****************************************** --
-- Echange d'authentification d'annuaire (unidirectionnel) --
-- ****************************************** --

dirAuthenticationOneWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS      {credentials}
  IDENTIFIER   global : {securityExchanges dir-authent-one-way (1)}
}
credentials       SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        1
}


-- ****************************************** --
-- Echange d'authentification d'annuaire (bidirectionnel) --
-- ****************************************** --

dirAuthenticationTwoWay   SECURITY-EXCHANGE  ::=
{
  SE-ITEMS      {initiatorCredentials | responderCredentials}
  IDENTIFIER   global : {securityExchanges dir-authent-two-way (2)}
}
initiatorCredentials            SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        1
  ERRORS  {authenticationFailure}
}
responderCredentials          SEC-EXCHG-ITEM   ::=
{
  ITEM-TYPE    DirectoryAbstractService.Credentials
  ITEM-ID        2
}
authenticationFailure    SE-ERROR ::=
{
  PARAMETER    DirectoryAbstractService.SecurityProblem
  ERROR-CODE   local : 1
}

-- ************************** --
-- Echange de négociation simple --
-- ************************** --

simpleNegotiationSE  SECURITY-EXCHANGE   ::=
{
  SE-ITEMS     {offeredIds | acceptedIds }
  IDENTIFIER   global : {securityExchanges simple-negotiation-se (3)}
}
offeredIds        SEC-EXCHG-ITEM     ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID        1
}
acceptedIds      SEC-EXCHG-ITEM     ::=
{
  ITEM-TYPE    Negotiation-SEI
  ITEM-ID        2
}

Negotiation-SEI ::=  SEQUENCE OF OBJECT IDENTIFIER

END
```

Annexe D

Spécification des transformations de sécurité

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Les transformations de sécurité peuvent être définies dans les Recommandations UIT-T | Normes internationales ou dans un autre cadre et être enregistrées par une organisation conformément aux dispositions de l'Annexe B. Les définitions des transformations de sécurité devraient être rendues aussi universelles que possible afin qu'elles puissent être utilisées dans de nombreuses applications. La présente annexe définit certaines transformations de sécurité qui sont généralement considérées comme utiles. Ceci ne sous-entend aucunement que les applications ou leurs réalisations sont supposées utiliser les transformations de sécurité spécifiques définies ici plutôt que d'autres.

D.1 Transformation de sécurité "Directory ENCRYPTED"

La transformation de sécurité "Directory ENCRYPTED" est fonctionnellement équivalente au type paramétré ENCRYPTED défini dans la Rec. UIT-T X.509 | ISO/CEI 9594-8. Elle pourvoit au chiffrement et au déchiffrement.

```
dirEncryptedTransformation SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-encrypted (1) }
   -- Cette transformation transforme une chaîne d'octets en
   -- nouvelle chaîne binaire au moyen d'un processus de chiffrement.
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber (1)}
  XFORMED-DATA-TYPE BIT STRING
}
```

D.1.1 Autres précisions

Processus de codage:	processus de chiffrement fondé sur tout algorithme qui a été choisi.
Apports locaux au processus de codage:	algorithme, paramètres de l'algorithme, renseignements sur la clé de chiffrement.
Processus de décodage:	processus de déchiffrement fondé sur le même algorithme.
Apports locaux au processus de décodage:	algorithme, paramètres de l'algorithme, renseignements sur la clé de déchiffrement.
Résultats du processus de décodage:	élément récupéré, en tant que valeur de type ASN.1, qu'il y a lieu de protéger.
Paramètres:	aucun.
Qualificateurs de transformation:	aucun.
Erreurs:	aucun comportement d'erreur spécifié.
Services de sécurité:	confidentialité.

D.1.2 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle de codeur, de décodeur ou ces deux rôles;
- *conformité statique* – Une réalisation qui joue le rôle de codeur doit être capable de produire l'item transformé. Une réalisation qui joue le rôle de décodeur doit être capable de traiter l'item transformé;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe.

D.2 Transformation de sécurité "Directory SIGNED"

La transformation de sécurité "Directory SIGNED" est fonctionnellement équivalente au type paramétré SIGNED défini dans la Rec. UIT-T X.509 | ISO/CEI 9594-8. Elle pourvoit à la signature numérique avec l'appendice, l'item transformé incluant à la fois les données non protégées à signer et l'appendice de signature.

```
dirSignedTransformation   SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-signed (2) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            distinguished-encoding (1)}
  XFORMED-DATA-TYPE SEQUENCE
  {
    toBeSigned          ABSTRACT-SYNTAX.&Type (CONSTRAINED BY {
     -- Ce type doit être le type
     -- "to-be-signed" -- } ),
    algorithmId         AlgorithmIdentifier,
     -- des algorithmes utilisés pour calculer la signature --
    encipheredHash      BIT STRING
  }
}
```

D.2.1 Autres précisions

Processus de codage: le processus de codage est appliqué à la forme codée complète (étiquette-longueur-valeur), selon les règles DER de l'ASN.1, d'une valeur de type ASN.1 simple (l'"élément non protégé"), ce qui produit un "élément transformé" (une valeur d'un type SEQUENCE tel que défini ci-dessus). L'élément non protégé codé est soumis à une fonction (telle que le hachage) qui produit une chaîne d'octets intermédiaire. Celle-ci est codée au moyen des règles de codage de base de l'ASN.1 et le résultat est chiffré pour obtenir une chaîne binaire "encipheredHash". L'élément transformé est ensuite structuré.

Apports locaux au processus de codage: identificateur de hachage et algorithme de chiffrement, paramètres de l'algorithme, renseignements sur la clé de chiffrement.

Processus de décodage: la valeur de l'élément non protégé est extraite de l'item transformé et restituée. S'il y a lieu de vérifier la signature, le processus ci-après a également lieu. La vérification nécessite la forme codée DER de l'item non protégé, qui peut être obtenue à partir de l'item transformé, mais nécessite éventuellement un décodage et un recodage DER. Les octets sont soumis à une fonction (telle que le hachage) produisant une chaîne binaire intermédiaire. La valeur "encipheredHash" est déchiffrée et décodée au moyen des règles de codage de base de l'ASN.1 et le résultat est comparé à la chaîne d'octets intermédiaire. Si les deux sont identiques, la signature est correcte. Une signature incorrecte est signalée par une erreur.

Apports locaux au processus de décodage: identificateur de hachage et algorithme de chiffrement, paramètres de l'algorithme, renseignements sur la clé de déchiffrement. A noter que l'algorithme et ses paramètres peuvent être obtenus à partir de l'item transformé, mais comme ils ont été stockés/transférés sans protection, il est recommandé d'obtenir ces valeurs sous forme d'apports locaux, éventuellement tirés de champs acheminés dans un item non protégé.

Résultats du processus de décodage: item non protégé récupéré, sous forme de valeur de type ASN.1. Par ailleurs, un des deux résultats suivants, voire les deux, peut être obtenu facultativement:

a) un indicateur montrant si la signature était correcte ou non;

b) une copie de l'élément transformé ou de la valeur "encipheredHash" qui sera stockée localement en vue d'une éventuelle vérification de signature ultérieure.

Paramètres:	aucun.
Qualificateurs de transformation:	aucun.
Erreurs:	une situation d'erreur survient si la signature s'avère incorrecte.
Services de sécurité:	authentification de l'origine des données, intégrité des données et (dans certaines situations) non-répudiation.

D.2.2 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle de codeur, de décodeur ou ces deux rôles;
- *conformité statique* – Une réalisation qui joue le rôle de codeur doit être capable de produire l'item transformé. Une réalisation qui joue le rôle de décodeur doit être capable de traiter l'item transformé;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe.

D.3 Transformation de sécurité "Directory SIGNATURE"

La transformation de sécurité "Directory SIGNATURE" est fonctionnellement équivalente au type paramétré SIGNATURE défini dans la Rec. UIT-T X.509 | ISO/CEI 9594-8. Elle pourvoit à la signature numérique avec l'appendice, l'item transformé comportant l'appendice de signature mais non les données non protégées qui doivent être signées.

```
dirSignatureTransformation SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-signature (3) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            distinguished-encoding (1)}
  XFORMED-DATA-TYPE SEQUENCE
  {
    algorithmId    AlgorithmIdentifier,
     -- des algorithmes utilisés pour calculer la signature --
    encipheredHash BIT STRING
  }
}
```

D.3.1 Autres précisions

Processus de codage:	le processus de codage est appliqué à la forme codée complète (étiquette-longueur-valeur), selon les règles DER de l'ASN.1, d'une valeur de type ASN.1 simple (l'"élément non protégé"), ce qui produit un "élément transformé" (une valeur d'un type SEQUENCE tel que défini ci-dessus). L'item non protégé codé est soumis à une fonction (telle que le hachage) qui produit une chaîne d'octets intermédiaire. Celle-ci est codée au moyen des règles de codage de base de l'ASN.1 et le résultat est chiffré pour obtenir une chaîne binaire "encipheredHash". L'élément transformé est ensuite structuré.
Apports locaux au processus de codage:	identificateur de hachage et algorithme de chiffrement, paramètres de l'algorithme, renseignements sur la clé de chiffrement.
Processus de décodage:	s'il y a lieu de vérifier la signature, le processus ci-après a lieu. La vérification nécessite la forme codée DER de l'élément non protégé, qui est obtenue sous forme d'apport local. Les octets sont soumis à une fonction (telle que le hachage) qui produit une chaîne d'octets intermédiaire. La valeur "encipheredHash" est déchiffrée et décodée au moyen des règles de codage de base de l'ASN.1 et le résultat est comparé à la chaîne d'octets intermédiaire. Si les deux sont identiques, la signature est correcte. Une signature incorrecte est signalée par une erreur.

Apports locaux au processus de décodage:	élément non protégé, identificateur de hachage et algorithme de chiffrement, paramètres de l'algorithme, renseignements sur la clé de déchiffrement. A noter que l'algorithme et ses paramètres peuvent être obtenus à partir de l'élément transformé, mais comme ils ont été stockés/transférés sans protection, il est recommandé d'obtenir ces valeurs sous forme d'apports locaux, éventuellement tirés de champs acheminés dans l'item non protégé.
Résultats du processus de décodage:	un des deux résultats suivants, voire les deux, peut facultativement être obtenu: a) un indicateur montrant si la signature était correcte ou non; b) une copie de l'élément transformé ou de la valeur "encipheredHash" qui sera stockée localement en vue d'une éventuelle vérification de signature ultérieure.
Paramètres:	aucun.
Qualificateurs de transformation:	aucun.
Erreurs:	une situation d'erreur survient si la signature s'avère incorrecte.
Services de sécurité:	authentification de l'origine des données, intégrité des données et (dans certaines situations) non-répudiation.

D.3.2 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle de codeur, de décodeur ou ces deux rôles;
- *conformité statique* – Une réalisation qui joue le rôle de codeur doit être capable de produire l'item transformé. Une réalisation qui joue le rôle de décodeur doit être capable de traiter l'item transformé;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe.

D.4 Transformation de sécurité "GULS SIGNED"

La transformation de sécurité "GULS SIGNED" pourvoit à la signature numérique ou au cachetage avec appendice, l'item transformé comprenant à la fois les données non protégées à signer et l'appendice signature/cachetage. Elle a une fonction comparable à celle de la transformation de sécurité Directory SIGNED et présente les particularités suivantes:

- elle peut prendre en charge une signature ou une technique de cachetage basée sur un appendice, c'est-à-dire qu'elle n'est pas limitée à une technique chiffrement-hachage comme Directory SIGNED;
- elle supprime l'obligation de se limiter aux règles de codage distinctives; on peut utiliser n'importe quelle règle de codage à valeur simple (y compris les règles de codage canoniques);
- elle prend en charge les paramètres protégés pour indiquer les règles de codage initiales, les identificateurs d'algorithme, les paramètres d'algorithme et les renseignements sur la clé;
- elle assure l'identification d'un algorithme d'une signature numérique et d'une fonction de condensation, par des identificateurs d'algorithme distincts;
- elle vérifie que la signature est calculée sur un codage des éléments désignés, identique au codage de transfert, afin de signaler éventuellement la nécessité pour le décodeur, de décoder et de recoder les données lors de la vérification de la signature.

D'autres mappages de protection ont été définis à l'Annexe E; ils permettent à la notation PROTECTED { BaseType, signed } de faire une projection sur la transformation de sécurité Directory SIGNED ou GULS SIGNED.

```
gulsSignedTransformation {KEY-INFORMATION: SupportedKIClasses}
          SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signed (4) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
              canonical-encoding (0) }
   -- Ce choix par défaut des règles de codage canoniques peut être annulé
   -- en utilisant un paramètre à protection statique (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
    intermediateValue   EMBEDDED PDV (WITH COMPONENTS {
        identification (WITH COMPONENTS
            {transfer-syntax (CONSTRAINED BY {
              -- La syntaxe de transfert à utiliser est celle qui
              -- est indiquée par la valeur initEncRules, à l'intérieur de
              -- la valeur intermédiaire -- }) PRESENT}),
        data-value (WITH COMPONENTS {notation (IntermediateType
          { {SupportedKIClasses} })})
            -- La valeur de données codée est une valeur de type IntermediateType
        }),
    appendix      BIT STRING (CONSTRAINED BY {
         -- la valeur de l'appendice doit être produite
         -- en suivant la procédure du projet DIS 11586-1 -- })
  }
}
IntermediateType {KEY-INFORMATION: SupportedKIClasses }::= SEQUENCE
{
  unprotectedItem   ABSTRACT-SYNTAX.&Type
                -- ce type peut être uniquement le type de
                -- l'item non protégé ou BIT STRING
                -- si l'item non protégé n'est pas tiré
                -- d'une syntaxe abstraite
                -- ASN.1 --,
  initEncRules      OBJECT IDENTIFIER DEFAULT
              {joint-iso-itu-t asn1 (1) ber-derived (2)
               canonical-encoding (0)},
  signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
                -- Identifie l'algorithme de signature
                -- ou de cachetage et peut acheminer
                -- les paramètres d'algorithme --
  hashAlgorithm     AlgorithmIdentifier OPTIONAL,
                -- Identifie une fonction de hachage qui sera
                -- utilisée si un hachage est requis et
                -- que l'identificateur signOrSealAlgorithm
                -- ne comprend pas une fonction
                -- de hachage particulière. Peut aussi
                -- acheminer des paramètres d'algorithme. --
  keyInformation    SEQUENCE
              {
                 kiClass              KEY-INFORMATION.&kiClass
                                      ({SupportedKIClasses}),
                 keyInfo              KEY-INFORMATION.&KiType
                                      ({SupportedKIClasses}
                                      {@.kiClass})
              } OPTIONAL
                -- Les renseignements sur la clé peuvent partir de
                -- l'hypothèse de plusieurs formats, pilotés par
                -- les membres pris en charge des renseignements
                -- de la classe d'objet KEY-INFORMATION (définie au départ
                -- du module ASN.1 définitif)
}
```

D.4.1 Autres précisions

Processus de codage:	le processus de codage est appliqué à une valeur de type ASN.1 simple (l'"élément non protégé"), ce qui produit un "élément transformé" (une valeur de type SEQUENCE tel que défini ci-dessus). (Si l'élément non protégé n'est pas tiré d'une spécification de syntaxe abstraite ASN.1, on peut considérer qu'elle est une valeur de type BIT STRING de l'ASN.1.) Tout d'abord est produite une "valeur intermédiaire" de type ASN.1 "IntermediateType". Celle-ci est codée au moyen des règles de codage initiales, déterminées selon le 7.1.4. Les octets qui en résultent (le codage "étiquette-longueur-valeur" complet) sont soumis à un processus de signature ou de cachetage, avec ou sans fonction de hachage. Cela produit une valeur d'appendice sous forme de chaîne binaire. L'item transformé est ensuite structuré. NOTE – Voici des exemples de "processus de signature ou de cachetage" pour différents algorithmes: a) calcul d'un code d'authentification de message, conformément à l'ISO 8730 (un type de sceau); b) concaténation du codage d'une BIT STRING contenant une valeur de clé secrète pour le codage de l'IntermediateType, suivie de l'application d'une fonction de condensation au résultat (un type de sceau); c) application d'une fonction de condensation au codage de l'IntermediateType, suivie de la signature de la valeur du code condensé résultant, en utilisant une signature numérique ou un algorithme de chiffrement à clé publique.
Apports locaux au processus de codage:	identificateur d'algorithme de signature ou de cachetage, identificateur (facultatif) de hachage et algorithme de chiffrement, paramètres de l'algorithme, renseignements sur la clé de signature/cachetage.
Processus de décodage:	la valeur de l'item non protégé est extraite de l'élément transformé et restituée. S'il y a lieu de vérifier la signature, le processus ci-après a également lieu. La vérification nécessite le codage de la valeur intermédiaire au moyen des règles de codage initiales ce qui peut être obtenu à partir de l'élément transformé, mais nécessite éventuellement un décodage et un recodage avec les règles voulues. Ensuite a lieu le processus de vérification de la signature ou du cachet.
Apports locaux au processus de décodage:	renseignements sur la clé de vérification de la signature/cachet. L'identificateur de l'algorithme de signature ou de cachetage, l'identificateur de l'algorithme de hachage et/ou les paramètres d'algorithme peuvent aussi être requis s'ils n'ont pas été acheminés comme des paramètres protégés.
Résultats du processus de décodage:	item non protégé récupéré, sous forme de valeur de type ASN.1. Par ailleurs, un des deux résultats suivants, voire les deux, peut facultativement être obtenu: a) un indicateur montrant si la signature était correcte ou non; b) une copie de l'élément transformé ou de la valeur d'appendice qui sera stockée localement en vue d'une éventuelle vérification de signature ultérieure.
Paramètres:	les paramètres statiques protégés optionnels sont: règles de codage initiales, identificateur d'algorithme de signature/cachetage, paramètres de l'algorithme de signature/cachetage, identificateur d'algorithme de hachage, paramètres de l'algorithme de hachage, renseignements sur la clé.
Qualificateurs de transformation:	aucun.
Erreurs:	une situation d'erreur survient si la signature ou le cachet s'avère incorrect.
Services de sécurité:	authentification de l'origine des données, intégrité des données et (dans certaines situations) non-répudiation.

D.4.2 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle de codeur, de décodeur ou ces deux rôles;
- *conformité statique* – Une réalisation qui joue le rôle de codeur doit être capable de produire l'item transformé. Une réalisation qui joue le rôle de décodeur doit être capable de traiter l'item transformé;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe.

D.5 Transformation de sécurité "GULS SIGNATURE"

La transformation de sécurité "GULS SIGNATURE" pourvoit à la signature numérique ou au cachetage avec appendice, l'item transformé comprenant l'appendice signature/cachetage mais non les données non protégées à signer. Elle a une fonction comparable à celle de la transformation de sécurité Directory SIGNATURE et présente les particularités suivantes:

- elle peut prendre en charge une signature ou une technique de cachetage basée sur un appendice, c'est-à-dire qu'elle n'est pas limitée à une technique chiffrement-hachage comme Directory SIGNATURE;
- elle supprime l'obligation de se limiter aux règles de codage distinctives; on peut utiliser n'importe quelle règle de codage à valeur simple (y compris les règles de codage canoniques);
- elle prend en charge les paramètres protégés pour indiquer les règles de codage initiales, les identificateurs d'algorithme, les paramètres d'algorithme et les renseignements sur la clé;
- elle pourvoit à l'identification d'un algorithme de signature numérique et d'une fonction de hachage par des identificateurs d'algorithme distincts;
- les processus de codage et de décodage ont été simplifiés.

D'autres mappages de protection ont été définis à l'Annexe E; ils permettent à la notation PROTECTED {BaseType, signed} de faire une projection sur la transformation de sécurité Directory SIGNATURE ou GULS SIGNATURE.

```
gulsSignatureTransformation {KEY-INFORMATION: SupportedKIClasses }
        SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signature (5) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
            canonical-encoding (0)}
   -- Ce choix par défaut des règles de codage canoniques peut être annulé
   -- en utilisant un paramètre à protection statique (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
  initEncRules     OBJECT IDENTIFIER DEFAULT
            {joint-iso-itu-t asn1 (1) ber-derived (2)
             canonical-encoding (0)},
  signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
              -- Identifie l'algorithme de signature ou de
              -- cachetage et peut acheminer des
              -- paramètres d'algorithme --
  hashAlgorithm     AlgorithmIdentifier OPTIONAL,
              -- Identifie une fonction de hachage qui sera
              -- utilisée si un hachage est requis et
              -- que l'identificateur signOrSealAlgorithm
              -- ne comprend pas une fonction
              -- de hachage particulière. Peut aussi
              -- acheminer des paramètres d'algorithme.--
```

```
    keyInformation    SEQUENCE
                {
                   kiClass              KEY-INFORMATION.&kiClass
                                        ({SupportedKIClasses}),
                   keyInfo              KEY-INFORMATION.&KiType
                                        ({SupportedKIClasses}
                                        {@.kiClass})
                } OPTIONAL,
                  -- Les renseignements sur la clé peuvent partir de
                  -- l'hypothèse de plusieurs formats, pilotés par
                  -- les membres pris en charge des renseignements
                  -- de la classe d'objet KEY-INFORMATION (définie au départ
                  -- du module ASN.1 définitif)
    appendix        BIT STRING (CONSTRAINED BY {
          -- la valeur d'appendice doit être produite conformément
          -- à la procédure spécifiée dans le projet DIS 11586-1 -- })
    }
}
```

D.5.1 Autres précisions

Processus de codage: le processus de codage est appliqué à une valeur de type ASN.1 simple (l'"élément non protégé"), ce qui produit un "élément transformé" (une valeur de type SEQUENCE tel que défini ci-dessus). (Si l'élément non protégé n'est pas tiré d'une spécification de syntaxe abstraite ASN.1, on peut considérer qu'elle est une valeur de type BIT STRING de l'ASN.1.) Tout d'abord est produite une "valeur intermédiaire" de type ASN.1 "IntermediateType" définie en D.4. Celle-ci est codée au moyen des règles de codage initiales, déterminées dans le 7.1.4. Les octets qui en résultent (le codage "étiquette-longueur-valeur" complet) sont soumis à un processus de signature ou de cachetage, avec ou sans fonction de hachage. Cela produit une valeur d'appendice sous forme de chaîne binaire. L'élément transformé est ensuite structuré.

NOTE – Voici des exemples de "processus de signature ou de cachetage" pour différents algorithmes:

a) calcul d'un code d'authentification de message, conformément à l'ISO 8730 (un type de sceau);

b) concaténation du codage d'une BIT STRING contenant une valeur de clé secrète pour le codage de l'IntermediateType, suivie de l'application d'une fonction de condensation au résultat (un type de sceau);

c) application d'une fonction de condensation au codage de l'IntermediateType, suivie de la signature de la valeur du code condensé résultant, en utilisant une signature numérique ou un algorithme de chiffrement à clé publique.

Apports locaux au processus de codage: identificateur d'algorithme de signature ou de cachetage, identificateur (facultatif) de hachage et algorithme de chiffrement, paramètres de l'algorithme, renseignements sur la clé de signature/cachetage.

Processus de décodage: s'il y a lieu de vérifier la signature, le processus ci-après a également lieu. La vérification nécessite le codage de la valeur intermédiaire au moyen des règles de codage initiales, ce qui requiert la valeur de l'item non protégé, qui est obtenue sous forme d'apport local. Les valeurs de paramètre protégées peuvent être obtenues à partir de l'item transformé, éventuellement avec un décodage et un recodage au moyen des règles appropriées. Ensuite a lieu la vérification de la signature ou du cachet.

Apports locaux au processus de décodage:	item non protégé, renseignements sur la clé de vérification de la signature/cachet. L'identificateur de l'algorithme de signature ou de cachetage, l'identificateur de l'algorithme de hachage et/ou les paramètres d'algorithme peuvent aussi être requis s'ils ne sont pas acheminés comme des paramètres protégés.
Résultats du processus de décodage:	un des deux résultats suivants, voire les deux, peut facultativement être obtenu: a) un indicateur montrant si la signature était correcte ou non; b) une copie de l'élément transformé ou de la valeur d'appendice qui sera stockée localement en vue d'une éventuelle vérification de signature ultérieure.
Paramètres:	les paramètres statiques protégés optionnels sont: règles de codage initiales, identificateur d'algorithme de signature/cachetage, paramètres de l'algorithme de signature/cachetage, identificateur d'algorithme de hachage, paramètres de l'algorithme de hachage, renseignements sur la clé.
Qualificateurs de transformation:	aucun.
Erreurs:	une situation d'erreur survient si la signature ou le cachet s'avère incorrect.
Services de sécurité:	authentification de l'origine des données, intégrité des données et (dans certaines situations) non-répudiation.

D.5.2 Conformité

Une réalisation déclarée conforme à la présente Définition de l'échange de sécurité doit satisfaire aux prescriptions de conformité suivantes:

- *déclaration* – Le réalisateur doit préciser si la réalisation joue le rôle de codeur, de décodeur ou ces deux rôles;
- *conformité statique* – Une réalisation qui joue le rôle de codeur doit être capable de produire l'item transformé. Une réalisation qui joue le rôle de décodeur doit être capable de traiter l'item transformé;
- *conformité dynamique* – Une réalisation doit mettre en œuvre les procédures applicables décrites dans la présente annexe.

D.6 Spécification ASN.1 définitive

```
GulsSecurityTransformations {joint-iso-itu-t genericULS (20)
        modules (1) gulsSecurityTransformations (3) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

 -- EXPORTE tout --

IMPORTS
  securityTransformations, notation
    FROM ObjectIdentifiers {joint-iso-itu-t genericULS (20)
    modules (1) objectIdentifiers (0) }
  SECURITY-TRANSFORMATION, SecurityIdentity
    FROM Notation notation
  AlgorithmIdentifier
    FROM AuthenticationFramework {joint-iso-itu-t ds (5)
    module (1) authenticationFramework(7) 2 };

-- ****************************************** --
-- Notation pour spécifier les renseignements sur la clé --
-- ****************************************** --

KEY-INFORMATION ::= CLASS
 -- Cette définition de classe d'objet d'information doit être utilisée pour
 -- spécifier des renseignements sur la clé relatifs à des classes
```

```
  -- particulières de mécanismes de protection (par exemple symétriques ou
  -- asymétriques). Elle est utile pour définir diverses transformations de sécurité.
  {
    &kiClass
      CHOICE
      {   local       INTEGER,
            -- les objets locaux ne peuvent être définis que dans
            -- ce module ASN.1.
          global   OBJECT IDENTIFIER
            -- les objets globaux sont définis ailleurs
      }UNIQUE,
    &KiType
  }
  WITH SYNTAX
  {
    KEY-INFO-CLASS     &kiClass
    KEY-INFO-TYPE      &KiType
  }

  symmetricKeyInformation KEY-INFORMATION ::= {
    KEY-INFO-CLASS       local: 0
    KEY-INFO-TYPE SEQUENCE
    {
      entityId             SecurityIdentity,
      keyIdentifier        INTEGER
    }
  }

  asymmetricKeyInformation KEY-INFORMATION ::= {
    KEY-INFO-CLASS       local: 1
    KEY-INFO-TYPE SEQUENCE
    {
      issuerCAName      SecurityIdentity OPTIONAL,
      certSerialNumber     INTEGER OPTIONAL,
      signerName        SecurityIdentity OPTIONAL,
      keyIdentifier        BIT STRING OPTIONAL
    }
  }

  -- ****************************************** --
  -- Transformation de sécurité "Directory ENCRYPTED" --
  -- ****************************************** --

  dirEncryptedTransformation SECURITY-TRANSFORMATION ::=
  {
    IDENTIFIER {securityTransformations dir-encrypted (1) }
     -- Cette transformation transforme une chaîne d'octets en nouvelle
     -- chaîne binaire au moyen d'un processus de chiffrement.
    INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber (1) }
    XFORMED-DATA-TYPE BIT STRING
  }

  -- ****************************************** --
  -- Transformation de sécurité "Directory SIGNED" --
  -- ****************************************** --

  dirSignedTransformation   SECURITY-TRANSFORMATION ::=
  {
    IDENTIFIER {securityTransformations dir-signed (2) }
    INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
                distinguished-encoding (1)}
    XFORMED-DATA-TYPE SEQUENCE
    {
      toBeSigned           ABSTRACT-SYNTAX.&Type (CONSTRAINED BY {
       -- ce type est limité au type "to-be-signed" -- } ),
```

```
    algorithmId          AlgorithmIdentifier,
     -- des algorithmes utilisés pour calculer la signature --
    encipheredHash BIT STRING
  }
}
-- ****************************************** --
-- Transformation de sécurité "Directory SIGNATURE" --
-- ****************************************** --

dirSignatureTransformation SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations dir-signature (3) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
              distinguished-encoding (1)}
  XFORMED-DATA-TYPE SEQUENCE
  {
    algorithmId        AlgorithmIdentifier,
     -- des algorithmes utilisés pour calculer la signature --
    encipheredHash BIT STRING
  }
}

-- ********************************** --
-- Transformation de sécurité "GULS SIGNED" --
-- ********************************** --

gulsSignedTransformation {KEY-INFORMATION: SupportedKIClasses }
          SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signed (4) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
              canonical-encoding (0)}
   -- Ce choix par défaut pour les règles de codage initiales peut être
   -- supplanté par l'utilisation d'un paramètre de protection statique (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
    intermediateValue   EMBEDDED PDV (WITH COMPONENTS {
        identification (WITH COMPONENTS
            {transfer-syntax (CONSTRAINED BY {
              -- La syntaxe de transfert à utiliser est celle
              -- indiquée par la valeur initEncRules, figurant à l'intérieur de
              -- la valeur intermédiaire -- }) PRESENT}),
        data-value (WITH COMPONENTS {notation (IntermediateType
          { {SupportedKIClasses} })})
            -- La valeur codée est une valeur de type
            -- IntermediateType
        }),
    appendix      BIT STRING (CONSTRAINED BY {
         -- la valeur d'appendice doit être produite conformément
         -- à la procédure spécifiée dans le D.4 du projet DIS 11586-1 -- })
  }
}
IntermediateType {KEY-INFORMATION: SupportedKIClasses }::= SEQUENCE
{
  unprotectedItem   ABSTRACT-SYNTAX.&Type
                -- ce type est limité à celui de l'item
                -- non protégé, ou à BIT STRING si
                -- l'item non protégé n'est pas tiré d'une
                -- syntaxe abstraite ASN.1 --,
  initEncRules     OBJECT IDENTIFIER DEFAULT
              {joint-iso-itu-t asn1 (1) ber-derived (2)
               canonical-encoding (0)},
```

```
    signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
                -- Identifie l'algorithme de signature
                -- ou de cachetage et peut acheminer
                -- des paramètres d'algorithme --
    hashAlgorithm     AlgorithmIdentifier OPTIONAL,
                -- Identifie une fonction de hachage
                -- qui sera utilisée si une telle fonction
                -- est requise et que l'identificateur
                -- signOrSealAlgorithm n'indique
                -- pas de fonction de hachage particulière.
                -- Peut aussi acheminer des paramètres d'algorithme.--
    keyInformation    SEQUENCE
                {
                   kiClass    KEY-INFORMATION.&kiClass
                              ({SupportedKIClasses}),
                   keyInfo    KEY-INFORMATION.&KiType
                              ({SupportedKIClasses}
                              {@.kiClass})
                } OPTIONAL
                  -- Les renseignements sur la clé peuvent partir de
                  -- l'hypothèse de plusieurs formats, régis par
                  -- les membres pris en charge de la classe d'objet informationnel
                  -- KEY-INFORMATION (définie au début
                  -- du module ASN.1 définitif)
}


-- ************************************** --
-- Transformation de sécurité "GULS SIGNATURE" --
-- ************************************** --

gulsSignatureTransformation {KEY-INFORMATION: SupportedKIClasses }
          SECURITY-TRANSFORMATION ::=
{
  IDENTIFIER {securityTransformations guls-signature (5) }
  INITIAL-ENCODING-RULES {joint-iso-itu-t asn1 (1) ber-derived (2)
                canonical-encoding (0)}
   -- Ce choix par défaut pour les règles de codage initiales peut être
   -- neutralisé au moyen d'un paramètre de protection statique (initEncRules).
  XFORMED-DATA-TYPE SEQUENCE
  {
  initEncRules      OBJECT IDENTIFIER DEFAULT
                {joint-iso-itu-t asn1 (1) ber-derived (2)
                 canonical-encoding (0)},
  signOrSealAlgorithm  AlgorithmIdentifier OPTIONAL,
                -- Identifie l'algorithme de signature
                -- ou de cachetage et peut acheminer
                -- des paramètres d'algorithme --
  hashAlgorithm     AlgorithmIdentifier OPTIONAL,
                -- Identifie une fonction de hachage
                -- qui sera utilisée en cas de besoin quand l'identificateur
                -- signOrSealAlgorithm ne comporte
                -- pas de fonction de hachage particulière.
                -- Peut également acheminer des paramètres d'algorithme. --
  keyInformation    SEQUENCE
                {
                   kiClass    KEY-INFORMATION.&kiClass
                              ({SupportedKIClasses}),
                   keyInfo    KEY-INFORMATION.&KiType
                              ({SupportedKIClasses}
                              {@.kiClass})
                } OPTIONAL,
```

```
                -- Les renseignements sur la clé peuvent partir de
                -- l'hypothèse de plusieurs formats, régis par
                -- les membres pris en charge de la classe d'objet informationnel
                -- KEY-INFORMATION (définie au début
                -- du module ASN.1 définitif)
  appendix      BIT STRING (CONSTRAINED BY {
        -- la valeur d'appendice doit être produite conformément à la
        -- procédure spécifiée dans le D.5 du projet DIS 11586-1 -- }) 
  }
}

END
```

Annexe E

Spécification des mappages de protection

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

Les mappages de protection sont définis dans des modules ASN.1. Tout module de ce type peut être défini dans des Recommandations UIT-T | Normes internationales ou dans un autre cadre et enregistré par toute organisation qui a la capacité d'attribuer des identificateurs d'objet. Les définitions des mappages de protection doivent être rendues aussi universelles que possible afin qu'elles puissent être utilisées dans de nombreuses applications. La présente annexe définit quelques mappages de protection dont on considère généralement qu'ils sont utiles. Cela ne sous-entend aucunement que les applications ou leurs réalisations sont supposées utiliser les mappages de protection spécifiques définis ici de préférence à d'autres.

```
DirectoryProtectionMappings {joint-iso-itu-t genericULS (20)
        modules (1) dirProtectionMappings (4) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
 -- Ces mappages de protection produisent des codages
 -- compatibles au niveau binaire des types paramétrés dans
 -- le cadre d'authentification de l'annuaire

 -- EXPORTE tout --

IMPORTS
  notation, gulsSecurityTransformations
    FROM ObjectIdentifiers {joint-iso-itu-t genericULS (20)
    modules (1) objectIdentifiers (0) }
  PROTECTION-MAPPING
    FROM Notation notation
  dirEncryptedTransformation, dirSignedTransformation,
  dirSignatureTransformation
    FROM GulsSecurityTransformations
          gulsSecurityTransformations;

 -- ************************************ --
 -- Mappage de protection "Directory encrypted" --
 -- ************************************ --

 -- Ce mappage de protection permet à la notation
 -- PROTECTED {BaseType, encrypted}
 -- de remplacer la notation
 -- ENCRYPTED {BaseType}
 -- qui figure dans la Rec. UIT-T X.509 | ISO/CEI 9594-8:1994 et de
 -- produire un codage binaire identique.
 -- Service de sécurité: confidentialité.

encrypted PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {dirEncryptedTransformation }
}


 -- ********************************** --
 -- Mappage de protection "Directory signed" --
 -- ********************************** --

 -- Ce mappage de protection permet à la notation
 -- PROTECTED {BaseType, signed}
 -- de remplacer la notation
 -- SIGNED {BaseType}
```

```
-- qui figure dans la Rec. UIT-T X.509 | ISO/CEI 9594-8:1994 et de
-- produire un codage binaire identique.
-- Service de sécurité: authentification de l'origine des données,
-- intégrité et (dans certaines situations) la non-répudiation.
signed PROTECTION-MAPPING  ::=
{
  SECURITY-TRANSFORMATION {dirSignedTransformation }
}

-- ************************************* --
-- Mappage de protection "Directory signature" --
-- ************************************* --

-- Ce mappage de protection permet à la notation
-- PROTECTED {BaseType, signature}
-- de remplacer la notation SIGNATURE {BaseType}
-- qui figure dans la Rec. UIT-T X.509 | ISO/CEI 9594-8.
-- Service de sécurité: authentification de l'origine des données,
-- intégrité et (dans certaines situations) non-répudiation.

signature PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION  {dirSignatureTransformation }
}
END


GULSProtectionMappings {joint-iso-itu-t genericULS (20)
        modules (1) gulsProtectionMappings (5) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN
-- Ces mappages de protection sont plus universels que les précédents, qui sont destinés
-- spécifiquement à produire des codages binaires identiques à ceux des types
-- paramétrés du cadre d'authentification de l'annuaire.

-- EXPORTE tout --

IMPORTS
  notation, gulsSecurityTransformations
    FROM ObjectIdentifiers {joint-iso-itu-t genericULS (20)
    modules (1) objectIdentifiers (0) }
  PROTECTION-MAPPING
    FROM Notation notation
  dirEncryptedTransformation, gulsSignedTransformation,
  gulsSignatureTransformation, symmetricKeyInformation,
  asymmetricKeyInformation
    FROM GulsSecurityTransformations
            gulsSecurityTransformations;


-- ********************************** --
-- Mappage de protection "confidentiality" --
-- ********************************** --

-- Ce mappage de protection permet à la notation
-- PROTECTED {BaseType, confidentiality} d'être projetée
-- sur dirEncryptedTransformation ou sur no transformation
-- au choix du système de codage, selon la politique
-- de sécurité locale et d'autres considérations locales.
-- Service de sécurité: confidentialité.
```

```
confidentiality PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {dirEncryptedTransformation }
  BYPASS-PERMITTED TRUE
}


-- ****************************** --
-- Mappage de protection "GULS signed" --
-- ****************************** --

-- Ce mappage de protection permet à la notation
-- PROTECTED { BaseType, signed } d'être projetée
-- sur gulsSignedTransformation
-- Service de sécurité: authentification de l'origine des données, intégrité des données et
-- (dans certaines situations) non-répudiation.

signed PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {gulsSignedTransformation
    {{symmetricKeyInformation | asymmetricKeyInformation }}}
}

-- ********************************* --
-- Mappage de protection "GULS signature" --
-- ********************************* --

-- Ce mappage de protection permet à la notation
-- PROTECTED { BaseType, signature }
-- d'être projetée sur gulsSignatureTransformation.
-- Service de sécurité: authentification de l'origine des données, intégrité des données et
-- (dans certaines situations) non-répudiation.

signature PROTECTION-MAPPING ::=
{
  SECURITY-TRANSFORMATION {gulsSignatureTransformation
    {{symmetricKeyInformation | asymmetricKeyInformation }}}
}

END
```

Annexe F

Utilisation de l'identificateur d'objet

(Cette annexe fait partie intégrante de la présente Recommandation | Norme internationale)

La présente annexe donne des informations sur les niveaux supérieurs du sous-arbre d'identificateur d'objet dans lequel résident tous les identificateurs d'objet attribués dans la présente série de Spécifications. Elle le fait en fournissant un module ASN.1 appelé ObjectIdentifiers dans lequel des noms sont attribués à tous les nœuds non-feuilles du sous-arbre. La série complète de modules ASN.1 définis pour la présente série de Normes est également identifiée.

```
ObjectIdentifiers {joint-iso-itu-t genericULS (20)
      modules (1) objectIdentifiers (0) }
DEFINITIONS AUTOMATIC TAGS ::=
BEGIN

 -- EXPORTE tout --

genericULS              OBJECT IDENTIFIER ::=
                        {joint-iso-itu-t genericULS (20) }

 -- Catégories d'objets informationnels --

modules                 OBJECT IDENTIFIER ::= {genericULS 1}
generalTransferSyntax   OBJECT IDENTIFIER ::= {genericULS 2}
specificTransferSyntax  OBJECT IDENTIFIER ::= {genericULS 3}
securityExchanges       OBJECT IDENTIFIER ::= {genericULS 4}
securityTransformations OBJECT IDENTIFIER ::= {genericULS 5}

 -- modules ASN.1 --

objectIdentifiers       OBJECT IDENTIFIER ::= {modules 0}
notation                OBJECT IDENTIFIER ::= {modules 1}
gulsSecurityExchanges   OBJECT IDENTIFIER ::= {modules 2}
gulsSecurityTransformations
                        OBJECT IDENTIFIER ::= {modules 3}
dirProtectionMappings
                        OBJECT IDENTIFIER ::= {modules 4}
gulsProtectionMappings
                        OBJECT IDENTIFIER ::= {modules 5}
seseAPDUs               OBJECT IDENTIFIER ::= {modules 6}
genericProtectingTransferSyntax
                        OBJECT IDENTIFIER ::= {modules 7}
END
```

Annexe G

Lignes directrices pour l'utilisation des moyens de sécurité génériques des couches supérieures

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

G.1 Introduction

La présente annexe explique comment les normes GULS peuvent être utilisées pour assurer la sécurité d'une application particulière en partant de l'hypothèse que les outils GULS conviennent pour assurer la sécurité de l'application en question.

Il est souhaitable que le concepteur d'un protocole d'application OSI utilise les mêmes solutions de sécurité que celles qui sont utilisées dans d'autres protocoles d'application OSI. Cela n'est généralement pas tout à fait le cas étant donné que les applications ont des impératifs de sécurité différents et que les solutions de sécurité nécessitent des adaptations pour répondre à ces besoins différents. Toutefois, il est généralement possible d'adopter des solutions communes fondées sur l'identification des impératifs de sécurité communs.

L'objet de la présente série de Recommandations | Normes internationales est de fournir une série de moyens protocolaires de sécurité pouvant contribuer à l'incorporation des solutions de sécurité dans tout protocole d'application et qui encourage à adopter des solutions de sécurité communes dans les diverses applications. Toutefois, ces spécifications ne fournissent pas toutes les spécifications nécessaires pour atteindre des solutions de sécurité communes.

G.2 Moyens génériques fournis

Les moyens fournis par les normes GULS sont:

- un moyen général de réaliser des composantes de protocole de couche Application en prenant en charge l'échange des informations liées à la sécurité entre un couple d'invocations d'entité d'application en communication (le *concept de l'échange de sécurité*, qui est pris en charge par l'élément *SESE);*
- une approche générale de l'utilisation des moyens de la couche Présentation pour effectuer des transformations liées à la sécurité sur des éléments d'information afin de protéger ceux-ci (la *syntaxe de transfert protectrice générique*);
- des outils de notation syntaxique abstraite, pour spécifier l'application sélective de la protection de sécurité à des champs déterminés de ce protocole (type *PROTECTED* paramétré et variante *PROTECTED-Q* de ce type).

Un autre moyen de sécurité générique de cette nature est l'unité fonctionnelle d'authentification de l'élément ACSE. Bien que ce moyen soit défini dans la Rec. UIT-T X.217 | ISO/CEI 8649 et la Rec. X.227 du CCITT | ISO/CEI 8650-1 plutôt que dans la présente Recommandation | Norme internationale, la présente annexe traitera de l'utilisation de ce moyen dans la mise au point des solutions de sécurité.

Il est prévu que les moyens décrits seront utilisés par les concepteurs de nouvelles applications quand ils traiteront les besoins de sécurité. Toutefois, ces moyens peuvent aussi être utilisés pour ajouter des caractéristiques de sécurité aux protocoles d'application OSI existants. Dans une certaine mesure, cela peut être fait en construisant un nouveau contexte d'application qui englobe de tels moyens sans nécessairement modifier les spécifications des éléments ASE existants. Toutefois, pour assurer certains services de sécurité (tels que la confidentialité ou l'intégrité sélective des champs), les modifications des autres spécifications ASE seront nécessaires.

G.3 Aspects des solutions de sécurité qui ne figurent pas dans la présente Recommandation | Norme internationale

La portée de la présente Recommandation | Norme internationale est limitée à la communication des informations associées à la fourniture des services de sécurité, c'est-à-dire qu'elle ne s'étend pas à tous les détails de la fourniture d'un service de sécurité ou de la mise en œuvre d'un mécanisme de sécurité quelconque. Les aspects généraux des mécanismes de sécurité sont décrits dans les Recommandations | Normes internationales relatives au cadre de sécurité (Rec. UIT-T X.811, X.812, X.813, X.814, X.815 | ISO/CEI 10181). Des Recommandations | Normes internationales relatives à certains mécanismes de sécurité spécifiques et aux techniques de sécurité qui les prennent en charge sont élaborées par le JTC 1/SC 27 de l'ISO/CEI.

La mise en application des moyens génériques décrits dans les présentes Recommandations | Normes internationales dépendent en particulier de l'un des deux éléments suivants, voire des deux:

- les spécifications des *échanges de sécurité* particuliers conçus pour prendre en charge des mécanismes de sécurité spécifiques (tels qu'un échange d'authentification spécifique);
- les spécifications de *transformations de sécurité* particulières qui transforment les données d'utilisateur, à des fins de protection, d'une manière donnée (un processus de chiffrement, par exemple).

De telles spécifications ne sont pas fournies dans la présente Recommandation | Norme internationale (exception faite de certaines instances généralement jugées utiles définies dans les Annexes C et D). Toutefois, la présente Recommandation | Norme internationale n'inclut pas les outils et les lignes directrices destinés à faciliter les réalisations de telles spécifications. Il convient de noter que de telles spécifications doivent pouvoir être utilisées dans de nombreuses applications différentes, conjointement avec les moyens décrits dans la présente Recommandation | Norme internationale.

Par ailleurs, la présente Recommandation | Norme internationale ne spécifie pas les procédures d'établissement des associations de sécurité établies extérieurement.

La présente Recommandation | Norme internationale ne comprend pas la définition d'une interface de service avec les échanges de sécurité qui est indépendante du mécanisme utilisé.

G.4 Utilisation des moyens GULS pour fournir des services de sécurité

Le texte qui suit est une indication de la manière dont les moyens génériques décrits dans les présentes Recommandations | Normes internationales peuvent être utilisés pour assurer la fourniture des services de sécurité identifiés dans la Rec. X.800 du CCITT | ISO 7498-2 pour la couche Application. Ces services auront pour effet de parer aux faiblesses identifiées par des groupes d'applications spécifiques.

Les détails de la fourniture des services de sécurité ci-dessous utilisant les outils GULS peuvent être traités séparément dans d'autres Recommandations UIT-T | Normes internationales.

G.4.1 Authentification d'entité

L'authentification d'entité (décrite dans la Rec. UIT-T X.811 | ISO/CEI 10181-2) comporte généralement un *échange d'authentification* qui est un échange à n transferts d'informations d'authentification entre les deux parties (n est généralement égal à 1, 2 ou 3, mais peut être plus grand). Aussi, peut-on considérer un échange d'authentification comme un cas particulier d'un échange de sécurité.

Il y a deux moyens de prendre en charge un échange d'authentification en utilisant les moyens de sécurité génériques des couches supérieures:

- dans le cas particulier où l'échange d'authentification est limité à un ou deux transferts, et où il ne peut avoir lieu que conjointement avec l'établissement d'association d'application OSI, il peut être acheminé en utilisant l'unité fonctionnelle d'authentification d'ACSE;
- dans les cas où les restrictions ci-dessus ne s'appliquent pas, l'échange d'authentification peut être acheminé au moyen de l'élément SESE.

On notera que le type d'identité en cours d'authentification est immatériel et qu'il n'est pas nécessairement celui d'une entité OSI. Par ailleurs, l'authentification d'entité ne doit pas nécessairement survenir au début d'une association (elle peut par exemple survenir au début d'un dialogue de TP ou à n'importe quel moment de l'association).

L'authentification de l'entité peut également comporter une communication avec un tiers. Le protocole à cet effet pourrait être un protocole d'application, auquel cas les échanges de sécurité peuvent également être employés dans ce protocole.

G.4.2 Authentification de l'origine des données

Une méthode courante d'authentification de l'origine des données consiste à joindre une signature ou un cachet à l'item dont la source est en cours d'authentification. Cela peut être fait en acheminant l'item dans une association de sécurité qui utilise une transformation de sécurité de type à signature ou à cachet.

Pour protéger de cette manière une catégorie d'unités PDU complètes, la spécification du contexte d'application devrait contenir les règles indiquant que de telles unités PDU peuvent être acheminées dans un contexte de protection de présentation. Pour protéger un item d'information individuel dans une syntaxe abstraite, on peut utiliser le type paramétré PROTECTED.

G.4.3 Contrôle d'accès

De nombreux aspects du contrôle d'accès sont spécifiques aux applications et ne peuvent dès lors être traités d'une façon générique. Toutefois, on peut communiquer l'information de contrôle d'accès (se rapportant au fait d'accorder, de mettre en vigueur ou de révoquer les droits du contrôle d'accès) au moyen d'un échange de sécurité. A titre d'exemple, le transfert d'un certificat de contrôle d'accès peut être considéré comme un échange de sécurité simple (unidirectionnel). Un tel certificat peut alors être rattaché à toute autre unité PDU en l'acheminant au moyen des services d'échange de sécurité de l'élément SESE. Un exemple de l'utilisation de l'élément SESE à cette fin est donné au I.4.

L'authentification de l'intégrité et/ou de l'origine des données des informations de contrôle d'accès qui ont été échangées est, elle aussi, généralement très importante. On peut assurer la protection nécessaire en acheminant l'information de contrôle d'accès dans une association de sécurité qui utilise une transformation de sécurité de type à signature ou à cachet.

G.4.4 Confidentialité avec et sans connexion

On peut assurer la confidentialité d'une unité PDU complète en l'acheminant dans une association de sécurité qui utilise une transformation de sécurité de type chiffré. Pour protéger de cette façon une catégorie d'unités PDU complètes, la spécification du contexte d'application devra contenir des règles indiquant que de telles unités PDU doivent être acheminées dans un contexte de protection de présentation.

G.4.5 Confidentialité sélective des champs

La confidentialité d'un champ de protocole quelconque peut être assurée en acheminant ce champ dans une association de sécurité qui utilise une transformation de sécurité de type chiffré. Pour identifier les items d'information individuels qui, dans une syntaxe abstraite, nécessitent une telle protection, on peut utiliser le type paramétré PROTECTED.

G.4.6 Confidentialité du flux du trafic

Un processus de codage de transformation peut pourvoir au rattachement de données de remplissage à un item protégé mais non à la production d'unités PDU contenant uniquement des données de remplissage.

G.4.7 Intégrité avec et sans connexion

On peut assurer l'intégrité d'une unité PDU complète en l'acheminant dans une association de sécurité qui utilise une transformation de sécurité de type à signature ou à cachet. Pour protéger de cette façon une catégorie d'unités PDU complètes, la spécification du contexte d'application devra contenir des règles indiquant que de telles unités PDU doivent être acheminées dans un contexte de protection de présentation.

G.4.8 Intégrité sélective des champs

On peut assurer l'intégrité de tout champ de protocole en acheminant dans une association de sécurité qui utilise une transformation de type à signature ou à cachet. Pour identifier les éléments d'information individuels qui, dans une syntaxe abstraite, nécessitent une telle protection, on pourra utiliser le type paramétré PROTECTED.

G.4.9 Non-répudiation

Pour assurer un service de non-répudiation (avec preuve de l'origine ou preuve de la remise), il faut généralement appliquer aux données communiquées l'authentification de l'intégrité et/ou de l'origine des données. On peut assurer la protection nécessaire en acheminant les données dans une association de sécurité qui utilise une transformation de sécurité de type à signature ou à cachet.

Certains mécanismes de non-répudiation sont fondés sur l'utilisation de signatures ne pouvant être répudiées appliquées aux données communiquées. On peut être fait en acheminant les données dans une association de sécurité qui utilise une transformation de sécurité de type à signature et une technique de chiffrement asymétrique.

G.4.10 Vérification

La fourniture d'un service de vérification de la sécurité, requiert généralement d'autres services de sécurité que ceux décrits de G.4.1 à G.4.9. Le SESE peut être utilisé pour échanger des informations, tels que des messages d'alarme de sécurité et de vérification de la sécurité, entre entités. (Toutefois, il existe également d'autres Recommandations | Normes internationales concernant ce type d'échange d'informations, par exemple la Rec. X.736 du CCITT | ISO/CEI 10164-7 et la Rec. X.740 du CCITT | ISO/CEI 10164-8.)

G.5 Gestion des clés

La gestion des clés est un domaine complexe dont de nombreux aspects ne relèvent pas du domaine d'application de l'OSI. Toutefois, l'emploi de nombreux types de fonction de transformation de protection dans la couche Présentation dépendra des clés qui auront été établies.

Les clés peuvent être établies de diverses façons, à savoir:

a) par distribution manuelle ou d'autres moyens entièrement hors de la portée de l'OSI;

b) par l'établissement de clés dans une association séparée (antérieure ou chevauchante), par exemple en utilisant les services de gestion du système OSI;

c) par l'établissement de clés au sein de la même association mais avant que la clé ne soit requise par la transformation. Cela peut faire intervenir, par exemple, un échange d'extraction de clés Diffie-Hellman où l'envoi d'une clé protégée pour des raisons de confidentialité sous le couvert d'une autre transformation et/ou d'une autre clé.

   Dans le cas c), l'extraction de la clé ou l'échange de distribution peut être fait en tant qu'échange de sécurité et utiliser les services de l'échange de sécurité de l'élément SESE. Cela peut avoir lieu dans le cadre du protocole assurant l'établissement d'une association de sécurité établie extérieurement.

Une extraction de clé ou une distribution peut faire partie intégrante d'un échange de sécurité assurant d'autres services tels que l'authentification de l'entité.

Les paramètres dynamiques de la transformation de sécurité acheminés dans la syntaxe de transfert de sécurité peuvent également appartenir à la gestion des clés, par exemple en indiquant qu'une clé particulière peut être utilisée à partir d'un point donné.

G.6 Lignes directrices pour spécifier les contextes d'application

Généralement, l'utilisation de l'élément SESE nécessite que l'on introduise des règles spéciales, qui ne font pas partie de la spécification de l'élément ASE, dans une spécification de contexte d'application. De telles règles doivent spécifier:

a) *les éléments ASE* – L'introduction de l'élément SESE en tant que l'un des éléments ASE dans le contexte d'application;

b) *échanges de sécurité* – L'ensemble particulier d'échanges de sécurité qu'il y a lieu de prendre en charge, ce qui sous-entend une syntaxe abstraite SESE spécifique;

c) *mappages SESE PDU* – Le mappage des unités SESE PDU sur d'autres services, à savoir le service P-DATA, ou en tant que valeur de données de présentation imbriquée dans l'unité PDU d'un autre élément ASE;

d) *contrainte de concaténation de valeur PDV* – Impératifs de concaténation des unités SESE PDU particulières avec les valeurs de données de présentation d'autres éléments ASE;

e) *contraintes d'intégration de valeur PDV* – Impératifs d'intégration d'autres valeurs de données de présentation dans les unités SESE PDU;

f) *contraintes de procédure* – Règles relatives aux interactions de la machine d'état SESE avec les machines d'état d'autres éléments ASE, pour assurer par exemple que l'état d'autres machines de protocole ASE est clairement défini et non dans une impasse à l'aboutissement réussi ou à l'interruption de chaque échange de sécurité;

g) *contraintes de contexte de présentation* – Impératifs d'établissement de syntaxes de transfert particulières pour les syntaxes abstraites particulières.

G.7 Exemple

Supposons que l'on désire construire un nouveau contexte d'application pour le service OSI de transfert, accès et gestion de fichiers (FTAM) défini dans l'ISO 8571, qui ajouterait trois caractéristiques de sécurité au protocole de base FTAM:

a) un échange d'authentification mutuelle poussé qui sera utilisé conjointement avec l'établissement de l'association;

b) un certificat de contrôle d'accès, dont le format est défini dans une quelconque autre norme, qu'il y a lieu de rattacher à chaque demande F-SELECT ou F-CREATE;

c) la confidentialité et la protection de l'intégrité doivent être appliquées à toutes les données contenues dans les fichiers qui sont transmis.

Il est souhaitable de réaliser cela en modifiant la syntaxe abstraite FTAM ASE.

La première étape consiste à identifier, ou si nécessaire à spécifier, les échanges de sécurité qui sont nécessaires. Un échange de sécurité bidirectionnel est requis dans le cas a) et un échange de sécurité unidirectionnel est requis pour la caractéristique b). Si l'authentification seule est requise, on peut utiliser pour le point a) l'échange de sécurité dirAuthenticationTwoWay défini à l'Annexe C. Sinon, on peut utiliser un échange de sécurité qui allie l'authentification et l'établissement de clés, cas dans lequel la ou les clés tirées de l'échange pourraient être utilisées pour fournir la caractéristique c). Pour les besoins du présent exemple, on partira de l'hypothèse d'un échange de sécurité dirAuthenticationTwoWay. L'échange de sécurité pour la caractéristique b) pourrait être l'échange boundAccessControlCert défini dans l'exemple I.4.

L'étape suivante consiste à spécifier une syntaxe abstraite SESE prenant en charge ces échanges de sécurité. L'exemple qui sera donné plus loin dans la Partie 3 des présentes spécifications montre la manière de procéder à cet effet.

La dernière étape consiste à spécifier le contexte d'application requis. Conformément aux lignes directrices du G.6, la spécification comportera les règles suivantes:

a) *éléments ASE* – L'ensemble des ASE comprend le SESE, ainsi que les ASE FTAM et ACSE (non modifiés);

b) *échanges de sécurité* – Les échanges de sécurité dirAuthenticationTwoWay et boundAccessControlCert sont pris en charge;

c) *mappages SESE PDU* – Les unités SE-TRANSFER PDU acheminant l'échange de sécurité dirAuthenticationTwoWay fait un mappage sur la demande A-ASSOCIATE et sur les unités PDU données en réponse (par exemple comme composants additionnels pour le champ d'information d'utilisateur); l'unité SE-TRANSFER PDU acheminant l'échange de sécurité boundAccessControlCert est mappée sur P-DATA;

d) *contraintes de concaténation de valeur PDV* – Aucune;

e) *contraintes d'intégration de valeur PDV* – Chaque unité FTAM PDU (ou groupe d'unités PDU) contenant une demande F-SELECT ou une demande F-CREATE est intégrée dans une unité SE-TRANSFER PDU acheminant un échange de sécurité boundAccessControlCert;

f) *contraintes de procédure* – Toute situation d'erreur rencontrée dans l'échange de sécurité dirAuthenticationTwoWay se traduit par l'interruption de l'association d'application;

g) *contraintes de contexte de présentation* – Le contexte de présentation utilisé pour transférer les données contenues dans des fichiers doit utiliser une syntaxe de protection de transfert avec un mappage de protection faisant intervenir la confidentialité et la protection de l'intégrité.

Un nouvel identificateur d'objet est attribué à ce contexte d'application.

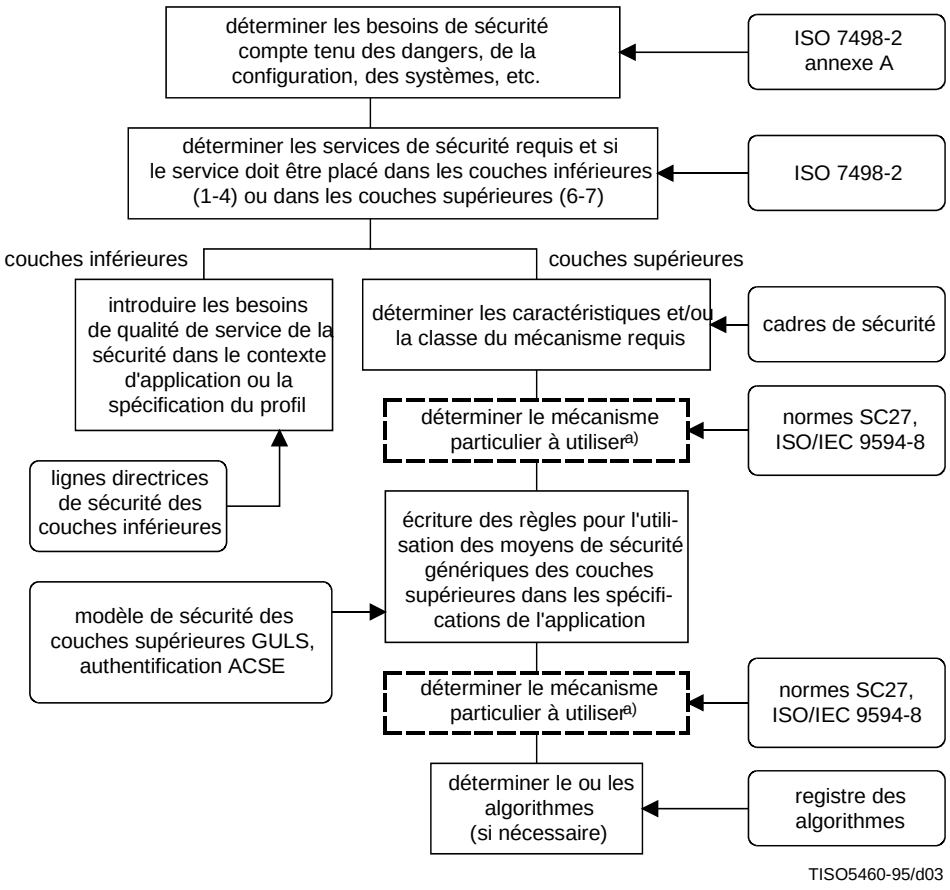
Annexe H

Relations avec d'autres normes

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

La présente annexe traite de la relation entre les spécifications GULS et d'autres normes, en partant de l'hypothèse que les outils GULS conviennent pour assurer la sécurité de l'application en question.

La Figure H.1 illustre le processus d'ensemble de l'incorporation de la sécurité dans une norme de protocole d'application.



a) Certains aspects de la détermination du mécanisme peuvent être reportés à un stade de profilage suivant la construction du protocole.

Figure H.1 – Lignes directrices pour intégrer la sécurité dans un protocole de couche Application

La Figure H.2 montre où se situent les moyens de protection GULS dans le processus global. Les observations ci-après se réfèrent aux blocs de la Figure H.2.

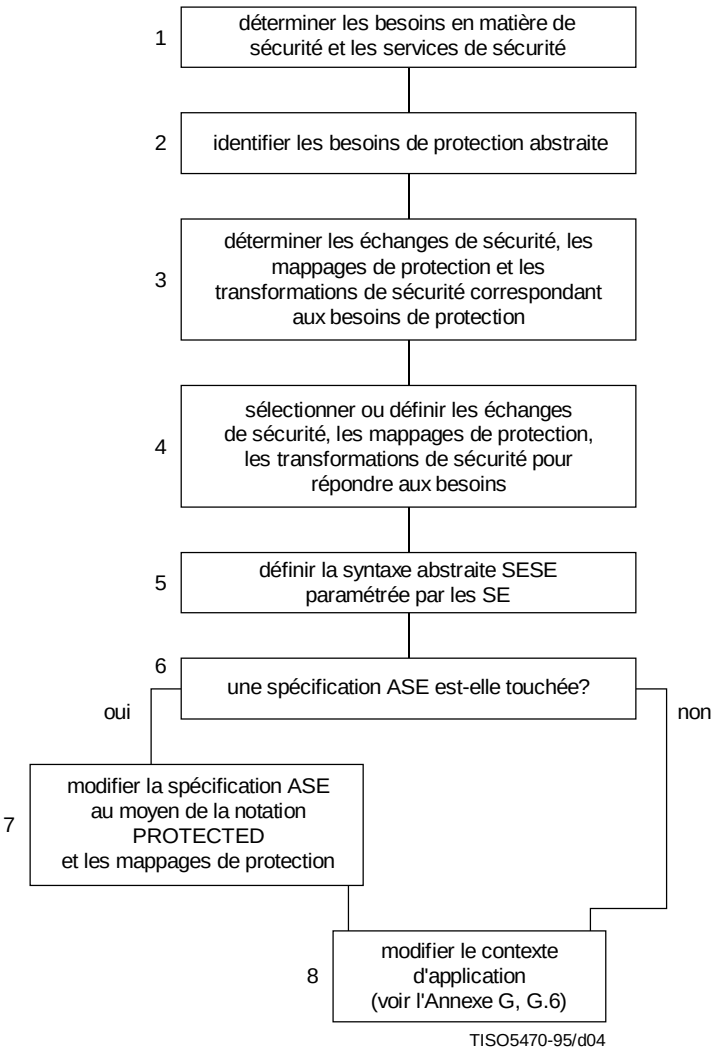


Figure H.2 – Incorporation de la sécurité GULS dans un protocole d'application OSI

bloc 4:

Les échanges de sécurité, les transformations de sécurité et les mappages de protection peuvent être spécifiés au moyen de nombreux types d'organisation différents. Le but général est que de telles spécifications puissent être utilisées dans des applications différentes, au lieu d'avoir à produire pour chaque cas de nouvelles spécifications remplissant les mêmes fonctions de base. Un réalisateur de protocole d'application devrait rechercher les spécifications existantes en se référant aux sources suivantes:

- annexes de la présente partie de la spécification de la sécurité générique des couches supérieures;
- spécifications contenues dans d'autres Recommandations UIT-T ou Normes internationales, soit sous forme de spécifications utilisable par plusieurs applications, soit en tant qu'applications particulières de l'OSI;
- des spécifications enregistrées existantes, telles que des spécifications mises au point et enregistrées par des organes de description.

A défaut de spécification utilisable, l'organisation mettra au point sa propre spécification, puis la normalisera et l'enregistrera pour utilisation dans d'autres applications.

bloc 6:

Généralement, une spécification ASE ne sera modifiée que s'il y a lieu d'apporter un changement à la spécification de la syntaxe abstraite. Cela est uniquement nécessaire quand on introduit des fonctions de sécurité sélective des champs (confidentialité, intégrité ou authentification de l'origine des données) à une granularité inférieure à celle d'une valeur de donnée de présentation. Dans d'autres cas, de nouveaux services de sécurité peuvent être introduits par des modifications des spécifications du contexte d'application n'ayant pas d'effet sur les spécifications ASE.

Pour spécifier la prise en charge de la sécurité d'une application OSI il sera nécessaire de produire:

a) des spécifications de protocoles afin d'assurer la sécurité par l'utilisation de classes de mécanisme particulières. Celles-ci sont notamment:

   – les échanges de sécurité pouvant être spécifiés au moyen de la notation SECURITY-EXCHANGE définie au 6.2;

   – les transformations de sécurité pouvant être spécifiées au moyen de la notation SECURITY-TRANSFORMATION définie au 7.2.

   Il faudra produire éventuellement des spécifications supplémentaires pour définir:

   – l'utilisation de services assurés par d'autres processus d'application OSI tels que le processus d'annuaire et la gestion des clés;

   – les interactions et les interdépendances des transformations de sécurité, les échanges de sécurité et l'utilisation d'autres processus d'application OSI.

   Dans la mesure du possible ces spécifications devraient être applicables à une gamme d'applications OSI.

b) les spécifications incorporées dans les spécifications de protocole d'application OSI pour mettre les dispositions de sécurité en relation avec les objets spécifiques du protocole d'application. Ceux-ci sont notamment:

   – *des protections sélectives des champs nécessaires pour les objets de données d'application* – Ceux-ci peuvent être spécifiés au moyen de la notation PROTECTED ou PROTECTED-Q définie aux 8.1 et 8.2;

   – *des besoins d'établissement d'associations de sécurité* – Les outils nécessaires à la spécification de tels besoins peuvent faire l'objet d'une normalisation séparée.

   Dans la mesure du possible, ceci devrait se faire en termes indépendants de classes de mécanisme particulières.

c) les spécifications pour l'application des classes de mécanisme spécifiques pour asseoir des applications OSI spécifiques. Celles-ci peuvent inclure:

   – des contextes ASO spécifiant l'utilisation des ASE/ASO (par exemple SESE) de sécurité avec d'autres ASE/ASO;

   – le mappage des types de protection requis sur les transformations de sécurité pouvant être spécifiées au moyen de la notation PROTECTION-MAPPING définie au 8.4;

   – le besoin d'appliquer des transformations de sécurité spécifiques à toutes les valeurs PDV de syntaxes abstraites particulières.

Annexe I

Exemples d'utilisation des moyens de sécurité génériques des couches supérieures

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

I.1 Exemple d'utilisation de la notation PROTECTED intégrée

A titre d'exemple illustrant l'utilisation du type paramétré PROTECTED, supposons qu'un concepteur de protocole d'application ait à spécifier une unité PDU ayant les caractéristiques suivantes:

a) l'ensemble de l'unité PDU doit être scellé dans un mécanisme d'intégrité;

b) l'unité PDU contient des champs individuels ayant les caractéristiques suivantes:

1) un champ (de type TypeOne) qui ne requiert aucune autre protection de sécurité;

2) un autre champ (de type TypeTwo) qui doit avoir une protection de confidentialité au moyen d'un algorithme symétrique; la clé de chiffrement doit également être acheminée dans l'unité PDU, chiffrée au moyen d'un algorithme asymétrique sous la clé publique du destinataire;

3) un autre champ (de type TypeThree) doit être signé au moyen de la clé privée de l'expéditeur.

Cette unité PDU peut être spécifiée comme le type ASN.1 suivant:

```
SecurePDU ::= PROTECTED
{
    SEQUENCE
    {
        encipheredConfKey       EncipheredConfKey,
        confidentialInfo        ConfidentialInfo,
        signedInfo              SignedInfo,
        clearInfo               TypeOne
    },
    sealed
}
EncipheredConfKey ::=   PROTECTED { ConfKey, encipheredKey}
ConfidentialInfo ::=    PROTECTED { TypeTwo, enciphered }
SignedInfo ::=          PROTECTED { TypeThree, signed }

ConfKey ::= BIT STRING
    -- Valeur envoyée dans la valeur, déterminée aléatoirement, fournie
    -- et utilisée par la transformation de sécurité utilisée pour
    -- le mappage de protection "sym-enciphered".
```

Cette ASN.1 produira, pour chaque instance de type PROTECTED, un codage tel que spécifié dans l'article 8. L'ensemble de l'unité PDU constitue un seul codage de ce type. Les trois autres codages sont intégrés dans le premier. Chaque codage utilise un type de transformation différent. La protection assurée par le codage extérieur s'applique à l'ensemble des contenus intérieurs.

La spécification dépend des mappages protecteurs "encipheredKey", "enciphered", "signed" et "sealed" qui les projettent sur les transformations. Ces dernières définitions doivent se trouver dans le même module ASN.1 que SecurePDU ou pourraient être des paramètres de ce module fournis au cours d'une étape ultérieure de développement du contexte d'application complet.

Exemple d'une série de définitions PROTECTION-MAPPING:

```
encipheredKey PROTECTION-MAPPING ::=
{   -- chiffré au moyen d'un algorithme asymétrique utilisant la clé publique du destinataire
    SECURITY-TRANSFORMATION { dirEncryptedTransformation }
}
enciphered PROTECTION-MAPPING ::=
{   -- chiffré au moyen d'un algorithme symétrique; la clé utilisée
    -- est la dernière valeur remise dans le cadre du mappage de protection
    -- "pk-enciphered"
    SECURITY-TRANSFORMATION { dirEncryptedTransformation }
}
signed PROTECTION-MAPPING ::=
{   -- signé au moyen de la clé privée de l'expéditeur
    SECURITY-TRANSFORMATION { dirSignedTransformation }
}
```

```
sealed PROTECTION-MAPPING ::=
{   -- scellé sous un mécanisme d'intégrité
    SECURITY-TRANSFORMATION { sealedTransformation }
    -- "sealedTransformation" n'est pas encore défini dans
    -- la présente Spécification.
}
```

I.2 Utilisation de la notation PROTECTED avec un qualificateur de transformation – Exemple 1

Ceci est une illustration de l'utilisation du type paramétré PROTECTED-Q fondée sur l'exemple du I.1 mais avec des qualificateurs spécifiés pour être utilisés par des transformations de sécurité. Les qualificateurs indiquent soit un algorithme particulier, soit une source d'algorithmes pour chaque transformation de sécurité, ainsi que le type de clé à utiliser pour chaque transformation.

L'unité PDU peut être spécifiée comme un type ASN.1 se présentant de la manière suivante:

```
SecurePDU ::= PROTECTED-Q
{
    SEQUENCE
    {
        encipheredConfKey       EncipheredConfKey,
        confidentialInfo        ConfidentialInfo,
        signedInfo              SignedInfo,
        clearInfo               TypeOne
    },
    sealed, { sealAlgorithm, preEstablishedKey }
}
EncipheredConfKey ::=   PROTECTED-Q { ConfKey, encipheredKey,
                        { rsaAlgorithm, receiverAsymKeyPair }}
ConfidentialInfo ::=    PROTECTED-Q { TypeTwo, enciphered,
                        { deaAlgorithm, accompanyingEncipheredKey }}
SignedInfo ::=          PROTECTED-Q { TypeThree, signed,
                        { signAlgorithm, senderAsymKeyPair }}

ConfKey ::= BIT STRING

rsaAlgorithm    AlgorithmSelector       ::= specificAlgorithm: { iso ... }
deaAlgorithm    AlgorithmSelector       ::= specificAlgorithm: { iso ... }
signAlgorithm   AlgorithmSelector       ::= algorithmSource: userDependent
sealAlgorithm   AlgorithmSelector       ::= algorithmSource: systemDefault
```

Dans cet exemple, le type ASN.1 de tous les qualificateurs est:

```
QualifierType ::= SEQUENCE
{
    algorithmSelector   AlgorithmSelector,
    keySelector         KeySelector
}
AlgorithmSelector ::= CHOICE
{
    specificAlgorithm   OBJECT IDENTIFIER,
    algorithmSource     BIT STRING
    {
        systemDefault (0),
        -- Utiliser l'algorithme par défaut du système standard.
        userDependent (1)
        -- Choix de l'algorithme fondé sur l'information locale concernant l'utilisateur.
    }
}
KeySelector ::= BIT STRING
{
    preEstablishedKey               (0),
    -- La clé a été préalablement fixée entre les parties.
    userSuppliedKey                 (1),
    -- La clé est fournie par l'expéditeur.
```

```
        accompanyingEncipheredKey       (2),
        -- La clé accompagne le champ protégé, acheminé dans un autre champ
        -- PROTECTED utilisant le mappage de protection "encipheredKey" comme un
        -- autre composant de la même structure ASN.1 environnante.
        senderAsymKeyPair               (3),
        -- La clé de codage est la clé privée de l'expéditeur; la clé de décodage
        -- est la clé publique correspondante.
        receiverAsymKeyPair             (4)
        -- La clé de codage est la clé publique du destinataire; la
        -- clé de décodage est la clé privée correspondante.
}
```

Les définitions de mappage de protection doivent refléter l'utilisation possible des qualificateurs de transformation, par exemple:

```
encipheredKey PROTECTION-MAPPING ::=
{       -- chiffre une clé qui sera utilisée pour protéger un autre champ
        SECURITY-TRANSFORMATION { qualEncryptedTransformation }
        -- une variante de "dirEncryptedTransformation" qui accepte
        -- le ou les qualificateurs de source d'algorithme et/ou de
        -- clé de type QualifierType
}
enciphered PROTECTION-MAPPING ::=
{       -- chiffrement général
        SECURITY-TRANSFORMATION { qualEncryptedTransformation }
        -- une variante de "dirEncryptedTransformation" qui accepte
        -- le ou les qualificateurs de source d'algorithme et/ou de
        -- clé de type QualifierType
}
signed PROTECTION-MAPPING ::=
{       -- signature numérique générale
        SECURITY-TRANSFORMATION { qualSignedTransformation }
        -- une variante de "gulsSignedTransformation" qui accepte
        -- le ou les qualificateurs de source d'algorithme et/ou de
        -- clé de type QualifierType
}
sealed PROTECTION-MAPPING ::=
{       -- scellé au moyen d'un mécanisme d'intégrité
        SECURITY-TRANSFORMATION { qualSealedTransformation }
        -- une variante de "gulsSignedTransformation" qui accepte
        -- le ou les qualificateurs de source d'algorithme et/ou de
        -- clé de type QualifierType
}
```

I.3 Utilisation de la notation PROTECTED avec un qualificateur de transformation – Exemple 2

Le texte qui suit illustre l'utilisation du type paramétré PROTECTED-Q au moyen de l'identificateur d'association de sécurité en tant que qualificateur d'un besoin de protection de "confidentialité" (voir E.4).

Avant l'utilisation de la protection "confidentialité" des données, une association de sécurité est établie extérieurement entre les deux systèmes en communication. Cela établit la transformation de sécurité et les paramètres statiques nécessaires à la gestion de l'opération afin de fournir la protection de confidentialité requise (c'est-à-dire l'algorithme, le mode de fonctionnement et les clés nécessaires). Cela peut être réalisé par exemple par l'utilisation d'un protocole de couche Application OSI qui utilise l'élément SESE pour prendre en charge l'échange de sécurité nécessaire. En plus de l'établissement des paramètres statiques, ce protocole d'établissement de l'association de sécurité établit un identificateur d'association de sécurité, sa-id, qui peut être utilisé tant dans le système de codage que dans le système de décodage pour se référer à la série de paramètres statiques.

La spécification par laquelle un élément de données du type ClearInfo doit être protégé par "confidentialité" au moyen des paramètres statiques identifiés par l'identificateur pc-id prendra la forme:

```
PROTECTED-Q { ClearInfo, confidentiality, sa-id }
```

Dans cet exemple, le qualificateur est du type:

SecurityAssociationId ::= ExternalSAID

tel que défini dans le module de Notation ASN.1 de l'Annexe A.

I.4 Exemple d'utilisation combinée de l'échange de sécurité et de la notation PROTECTED

Dans cet exemple, le système A envoie au système B une demande d'accès au moyen d'un certificat de contrôle d'accès. Celui-ci est protégé contre toute utilisation non autorisée par la méthode décrite à l'Annexe B de la Rec. UIT-T X.812 | ISO/CEI 10181-3 (Cadre général du contrôle d'accès). Le certificat de contrôle d'accès contient une valeur de protection (PV) qui se rapporte à une valeur de contrôle (CV) liée à la relation suivante:

$$PV = OWF (CV),$$

où OWF représente une fonction unidirectionnelle. La connaissance de CV est la preuve de la propriété du certificat de contrôle d'accès. Cela signifie que la CV doit être envoyée à B sous forme chiffrée. On part du principe que A et B ont tous deux des paires de clés publiques. Il est nécessaire d'envoyer la CV non chiffrée sous la clé publique de B. Par ailleurs, le certificat de contrôle d'accès et la demande sont également envoyés, scellés sous la clé privée de A.

Ce besoin doit être satisfait en définissant un échange de sécurité qui achemine l'information de sécurité nécessaire avec la demande d'accès (généralement une valeur de données de présentation d'un élément ASE spécifique à l'application) intégrée dans l'échange de sécurité. La définition de l'échange de sécurité pourrait se présenter de la manière suivante:

```
boundAccessControlCert   SECURITY-EXCHANGE  ::=
{
    SE-ITEMS            { boundACC }
    IDENTIFIER                  { ... object identifier ... }
}
boundACC     SEC-EXCHG-ITEM   ::=
{
    ITEM-TYPE                   PROTECTED { SealedSequence, sealed }
    ITEM-ID  1
}
SealedSequence ::= SEQUENCE
{
    accessControlCert    AccessControlCert,
    encipheredCV         EncipheredCV,
    accessRequest        EMBEDDED PDV
                -- La PDU de demande d'accès est insérée ici
}
AccessControlCert        ::= PROTECTED {...certificate contents..., signed }

EncipheredCV             ::= PROTECTED { BIT STRING, encrypted }
```

Annexe J

Bibliographie

(Cette annexe ne fait pas partie intégrante de la présente Recommandation | Norme internationale)

– ISO 8730:1990, *Opérations bancaires – Spécifications liées à l'authentification des messages (service aux entreprises).*

– Recommandation X.227 du CCITT (1992), *Spécification du protocole en mode connexion applicable à l'élément de service contrôle d'association.*

ISO 8650-1:1988, *Systèmes de traitement de l'information – Interconnexion de systèmes ouverts – Spécification du protocole pour l'élément de service de contrôle d'association*.

– Recommandation X.736 du CCITT (1992) | ISO/CEI 10164-7:1992, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestions des systèmes: Fonction de signalisation des alarmes de sécurité.*

– Recommandation X.740 du CCITT (1992) | ISO/CEI 10164-8:1993, *Technologies de l'information – Interconnexion des systèmes ouverts – Gestions des systèmes: Fonction de piste de vérification de sécurité.*

– Recommandation UIT-T X.813[2)] | ISO/CEI 10181-4:...[2)], *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: Cadre de non-répudiation*.

– Recommandation UIT-T X.814[2)] | ISO/CEI 10181-5:...[2)], *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: Cadre de confidentialité*.

– Recommandation UIT-T X.815[2)] | ISO/CEI 10181-6:...[2)], *Technologies de l'information – Interconnexion des systèmes ouverts – Cadres de sécurité pour les systèmes ouverts: Cadre d'intégrité.*

2) Actuellement, à l'état de projet.