Groupe de travail Réseau
Request for Comments : 5126
RFC rendue obsolète : 3126
Catégorie : Information
Traduction Claude Brière de L'Isle

D. Pinkas, Bull SAS
N. Pope,Thales eSecurity
J. Ross, Security and Standards
février 2008

Signatures électroniques évoluées conformes à la syntaxe de message cryptographique (CAdES)

Statut du présent mémoire
Le présent mémoire donne des informaions pour la communauté de l'Internet. Il ne spécifie aucune sorte de norme de l'Internet. La distribution du présent mémoire n'est soumise à aucune restriction.

Résumé
Le présent document définit le format d'une signature électronique qui peut rester valide sur de longues périodes. Cela inclut la preuve de sa validité même si le signataire ou la partie qui vérifie tente ultérieurement de nier (c'est-à-dire, répudier) la validité de la signature.

Le format peut être considéré comme une extension des RFC 3852 et RFC 2634, où, lorsque approprié, des attributs signés et non signés ont été définis.

Le contenu de cette RFC pour information revient à une transposition de la spécification technique (TS, *Technical Specification*) ETSI TS 101 733 V.1.7.4 (CMS Advanced Electronic Signatures -- CAdES) et lui est techniquement équivalente. Le contenu technique de cette spécification est maintenu par ETSI. La TS ETSI et ses futures mises à jour sont disponibles gratuitement à : http://www.etsi.org/WebSite/Standards/StandardsDownload.aspx

Table des Matières

1. Introduction

Le présent document est destiné à couvrir les signatures électroniques pour divers types de transactions, incluant les transactions d'affaires (par exemple, applications de bons de commande, de contrats, et de factures) où la validité à long terme de ces signatures est importante. Cela inclut la preuve de la validité même si le signataire ou la partie qui vérifie ultérieurement tente de nier (c'est-à-dire, répudier ; voir la norme [ISO10181-5]) la validité de la signature.

Donc, le présent document peut être utilisé pour toute transaction entre un individu et une société, entre deux sociétés, entre un individu et un organisme gouvernemental, etc. Le présent document est indépendant de tout environnement ; il peut être appliqué à tout environnement, par exemple, de cartes à mémoire, de carte de module d'identité d'abonné de système mondial de télécommunications mobiles (GSM SIM, *Global System for Mobile Communication Subscriber Identity Module*) de programmes spéciaux pour les signatures électroniques, etc.

La Directive européenne sur un cadre communautaire pour les signatures électroniques définit une signature électronique comme "des données en forme électronique qui sont attachées ou logiquement associées à d'autres données électroniques et qui servent de méthode d'authentification".

Une signature électronique, comme utilisée dans le présent document, est une forme de signature électronique évoluée, comme défini dans la Directive.

2. Domaine d'application

Le domaine d'application du présent document couvre les formats de signature électronique seulement. Les aspects de politique de signature électronique sont définis dans la [RFC3125] et dans le rapport technique ETSI 102 272 [TR102272].

Le présent document définit un certain nombre de formats de signature électronique, incluant des signatures électroniques qui peuvent rester valides sur de longues périodes. Cela inclut la preuve de validité même si le signataire ou la partie qui la vérifie tente ultérieurement de nier (répudier) la validité de la signature électronique.

Le présent document spécifie l'utilisation de fournisseurs de service de confiance (par exemple, des autorités d'horodatage) et des données qui ont besoin d'être archivées (par exemple, des certificats croisés et des listes de révocation) pour satisfaire les exigences de signatures électroniques à long terme.

Une signature électronique, comme définie dans le présent document, peut être utilisée pour un arbitrage dans le cas de dispute entre le signataire et le vérificateur, qui peut survenir plus tard, même des années après.

Le présent document inclut le concept de politiques de signature qui peut être utilisé pour établir la cohérence technique lors de la validation des signatures électroniques, mais il ne rend pas obligatoire leur utilisation.

Le présent document se fonde sur l'utilisation de clés de chiffrement publiques pour produire les signatures numériques, prises en charge par des certificats de clé publique. Le présent document spécifie aussi l'utilisation de services d'horodatage et de marquage de l'heure pour prouver la validité d'une signature longtemps après la durée de vie normale d'éléments critiques d'une signature électronique. Ce document définit aussi, en option, des moyens de fournir une protection à très long terme contre la compromission de clé ou l'affaiblissement des algorithmes.

Le présent document s'appuie sur des normes existantes largement adoptées. Cela inclut :
- la [RFC3852] "Syntaxe de messages cryptographique (CMS)" ;
- la norme ISO/CEI 9594-8/Recommandation UIT-T X.509 [X.509] "Technologie de l'information - Interconnexion des systèmes ouverts - L'annuaire : cadre d'authentification" ;
- la [RFC3280] "Profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet" ;
- la [RFC3161] "Protocole d'horodatage (TSP) d'infrastructure de clé publique X.509 pour l'Internet".

Note : voir l'ensemble complet des références à la Section 11.

Le présent document décrit les formats pour les signatures électroniques évoluées en utilisant l'ASN.1 (notation n° 1 de syntaxe abstraite) [X.208]. L'ASN.1 est codé en utilisant [X.690].

Ces formats se fondent sur la syntaxe de message cryptographique (CMS, *Cryptographic Message Syntax*) définie dans la [RFC3852]. Ces signatures électroniques sont donc appelées des signatures électroniques évoluées conformes à la syntaxe de message cryptographique (CAdES, *CMS Advanced Electronic Signatures*).

Un autre document, TS 101 903 [TS101903], décrit les formats pour les signatures électroniques évoluées en XML (XAdES) construites sur XML DSIG comme spécifié dans [XMLDSIG].

De plus, le présent document identifie d'autres documents qui définissent les formats pour les certificats de clés publiques, les certificats d'attribut, et les listes de révocation de certificat et les protocoles qui les prennent en charge, incluant les protocoles à utiliser par des tiers de confiance pour prendre en charge le fonctionnement de la création et de la validation de signature électronique.

Les annexes pour information incluent :
- des illustrations des formes étendues de formats de signature électronique qui protègent contre diverses faiblesses et des exemples de processus de validation (Annexe B) ;
- des descriptions et explications de certains des concepts utilisés dans le présent document, donnant la raison des parties normatives du présent document (Annexe C) ;
- des informations sur les protocoles pour interopérer avec les fournisseurs de service de confiance (Annexe D) ;
- des lignes directrices sur les dénominations (Annexe E) ;
- un exemple de contenu structuré et MIME (Annexe F) ;
- les relations entre le présent document et la Directive sur les signatures électroniques et les initiatives de normalisation associées (Annexe G) ;
- les API pour prendre en charge la génération et la vérification des signatures électroniques (Annexe H) ;
- les algorithmes de chiffrement qui peuvent être utilisés (Annexe I) ; et
- les schémas de désignation (Annexe J).

3. Définitions et abréviations

3.1 Définitions

Pour les besoins du présent document, les termes et définitions suivants s'appliquent :

Arbitre : une entité d' arbitrage peut être utilisée pour arbitrer une dispute entre un signataire et un vérificateur quand il y a un désaccord sur la validité d'une signature numérique.

Autorité d'attribut (AA) : autorité qui alloue des privilèges en produisant des certificats d'attribut d'attributs.

Certificat d'autorité : certificat produit par une autorité (par exemple, une autorité de certification ou une autorité d'attribut).

Liste de révocation d'autorité d'attribut (AARL, *Attribute Authority Revocation List*) : liste de révocation qui contient les références des certificats produits aux AA qui ne sont plus considérés comme valides par l'autorité qui les a produits.

Liste de révocation de certificat d'attribut (ACRL, *Attribute Certificate Revocation List*) : liste de révocation qui contient les références aux certificats d'attribut qui ne sont plus considérés comme valides par l'autorité qui les a produits.

Liste de révocation d'autorité de certification (CARL, *Certification Authority Revocation List*) : liste de révocation qui contient une liste des certificats de clé publique produits aux autorités de certification, qui ne sont plus considérés comme valides par le producteur du certificat.

Autorité de certification (CA, *Certification Authority*) : autorité de confiance pour un ou plusieurs utilisateurs pour créer et allouer des certificats de clé publique ; facultativement, l'autorité de certification peut créer les clés de l'utilisateur. Note : voir la Recommendation UIT-T X.509 [X.509].

Liste de révocation de certificats (CRL, *Certificate Revocation List*) : liste signée qui indique un ensemble de certificats de clé publique qui ne sont plus considérés être valides par le producteur de certificats.

Signature numérique : données ajoutées à une unité de données, ou sa transformation cryptographique, qui permet à un receveur de l'unité de données de prouver la source et l'intégrité de l'unité de données et de la protéger contre la falsification, par exemple, par le receveur. Note : voir la norme ISO 7498-2 [ISO7498-2].

Signature électronique : données en forme électronique qui sont rattachées ou logiquement associées à d'autres données électroniques et qui servent de méthode d'authentification. Note : voir la Directive 1999/93/EC du Parlement européen et du Conseil du 13 décembre 1999 sur un cadre communautaire pour les signatures électroniques [EUDirective].

Signature électronique étendue : signature électronique améliorée en complétant les exigences de base avec des données supplémentaires, comme un jeton d'horoatage et des données de révocation de certificat, pour traiter les menaces courantes reconnues.

Signature électronique fondée sur une politique explicite (EPES, *Explicit Policy-based Electonic Signature*) : signature électronique où la politique de signature qui devra être utilisée pour la valider est explicitement spécifiée.

Période de grâce : période qui permet aux informations de révocation de certificat de se propager aux consommateurs d'assertions par le processus de révocation.

Vérification initiale : processus effectué par un vérificateur après qu'une signature électronique est générée afin de capturer des informations supplémentaires qui pourraient la rendre valide pour une vérification à long terme.

Certificat de clé publique (PKC, *Public Key Certificate*) : clés publiques d'un utilisateur, avec d'autres informations, rendues infalsifiables par le chiffrement avec la clé privée de l'autorité de certification qui l'a produite. Note : voir la Recommandation UIT-T X.509 [X.509].

Rivest-Shamir-Adleman (RSA) : algorithme de chiffrement asymétrique fondé sur la difficulté de factoriser de très grands nombres en utilisant une paire de clés : une clé privée et une clé publique.

Politique de signature : ensemble de règles pour la création et la validation d'une signature électronique qui définissent les exigences techniques et procédurales pour la création et la validation d'une signature électronique, afin de satisfaire les besoins particuliers d'un secteur commercial, et selon laquelle la signature peut être déterminée comme étant valide.

Producteur de politique de signature : entité qui définit et produit une politique de signature.

Politique de validation de signature : partie de la politique de signature qui spécifie les exigences techniques qui pèsent sur le signataire à la création d'une signature et sur le vérificateur quand il valide une signature.

Signataire : entité qui crée une signature électronique.

Vérification suivante : processus effectué par un vérificateur pour attester la validité de la signature. Note : la vérification suivante peut être faite des années après la production de la signature électronique par le signataire et complétée par la vérification initiale, et elle n'a pas besoin de capturer plus de données qu'au moment de la vérification initiale.

Jeton d'horodatage *(Time-Stamp Token)* : objet de données qui lie une représentation d'une donnée à un instant particulier, établissant donc la preuve que la donnée existait avant cet instant.

Marque temporelle *(Time-Mark)* : information dans un chemin d'audit provenant d'un fournisseur de service de confiance qui lie une représentation d'une données à un instant particulier, établissant donc la preuve que les données existaient avant cet instant.

Autorité de marquage temporel : tiers de confiance qui crée des enregistrements dans un chemin d'audit afin d'indiquer qu'une donnée existait avant un instant particulier.

Autorité d'horodatage (TSA, *Time-Stamping Authority*) : tiers de confiance qui crée des jetons d'horodatage afin d'indiquer qu'une donnée existait à un instant particulier.

Unité d'horodatage (TSU, *Time-Stamping Unit*) : ensemble de matérels et logiciels géré comme une unité et a une seule clé de signature de jeton d'horodatage active à tout instant.

Fournisseur de service de confiance (TSP, *Trusted Service Provider*) : entité qui aide à construire des relations de confiance en rendant disponibles ou en fournissant des informations à la demande.

Données de validation : données supplémentaires qui peuvent être utilisées par un vérificateur de signatures électroniques pour déterminer si la signature est valide.

Signature électronique valide : signature électronique qui réussit à la vérification de validation.

Vérificateur : entité qui vérifie la preuve. Note 1: voir la norme ISO/CEI 13888-1 [ISO13888-1]. Note 2 : dans le contexte du présent document, c'est une entité qui valide une signature électronique.

3.2 Abréviations

Pour les besoins du présent document, les abréviations suivantes s'appliquent :

AA *(Attribute Authority)* autorité d'attribut
AARL *(Attribute Authority Revocation List)* : liste de révocation d'autorité d'attribut
ACRL *(Attribute Certificate Revocation List)* : liste de révocation de certificat d'attribut
API *(Application Program Interface)* : interface de programme d'application
ASCII *(American Standard Code for Information Interchange)* : code standard américain pour les échanges d'information
ASN.1 *(Abstract Syntax Notation 1)* : notation numéro un de syntaxe abstraite
CA *(Certification Authority)* : autorité de certification
CAD *(Card Accepting Device)* : dispositif de validation de carte
CAdES *(CMS Advanced Electronic Signature)* : signature électronique évoluée fondée sur la CMS
CAdES-A *(CAdES with Archive Validation Data)* : CAdES avec données de validation d'archive
CAdES-BES *(CAdES Basic Electronic Signature)* : signature électronique CAdES de base
CAdES-C *(CAdES with Complete Validation Data)* : CAdES avec données de validation complètes
CAdES-EPES *(CAdES Explicit Policy Electronic Signature)* : signature électronique CAdES fondée sur une politique explicite
CAdES-T *(CAdES with Time)* : signature électronique avec heure
CAdES-X *(CAdES with eXtended Validation Data)* : CAdES avec données de validation étendues
CAdES-X Long *(CAdES avec EXtended Long Validation Data)* : CAdES avec données de validation longue étendues
CARL *(Certification Authority Revocation List)* : liste de révocation d'autorité de certification
CMS *(Cryptographic Message Syntax)* : syntaxe de message cryptographique
CRL *(Certificate Revocation List)* : liste de révocation de certificat
CWA *(Commité européen de normalisation Workshop Agreement)* : accord de l'atelier du CEN
DER *(Distinguished Encoding Rules (ASN.1))* : règles de codage distinctif (de l'ASN.1)
DSA *(Digital Signature Algorithm)* : algorithme de signature numérique
EDIFACT *(Electronic Data Interchange For Administration, Commerce and Transport)* : échange de données électroniques pour l'administration, le commerce et le transport
EESSI *(European Electronic Signature Standardization Initiative)* : initiative européenne de normalisation des signatures électroniques
EPES *(Explicit Policy-based Electronic Signature) :* signature électronique fondée sur une politique explicite
ES (*Electronic Signature) :* signature électronique
ESS *(Enhanced Security Services)* : services de sécurité améliorée (améliore la CMS)
MIME *(Multipurpose Internet Mail Extensions)* : extensions de messagerie Internet multi objets
OCSP *(Online Certificate Status Protocol)* : protocole d'état de certificat en ligne
OID *(Object IDentifier)* : identifiant d'objet
PKC *(Public Key Certificate)* : certificat de clé publique
PKIX *(Public Key Infrastructure using X.509)* : infrastructure de clé publique X.509 (groupe de travail de l'IETF)
RSA : Rivest-Shamir-Adleman
SHA-1 *(Secure Hash Algorithm 1)* : algorithme n° 1 de hachage sécurisé
TSA *(Time-Stamping Authority)* : autorité d'horodatage
TSP *(Trusted Service Provider)* : fournisseur de service de confiance
TST *(Time-Stamp Token)* : jeton d'horodatage
TSU *(Time-Stamping Unit)* : unité d'horodatage
URI *(Uniform Resource Identifier)* : identifiant de ressource universel
URL *(Uniform Resource Locator)* : adresse universelle
XML *(Extensible Markup Language)* : langage de balisage extensible
XMLDSIG *(XML Digital Signature)* : signature numérique XML

4. Vue d'ensemble

Le présent document définit un certain nombre de formats de signature électronique (ES) qui s'appuient sur la CMS [RFC3852] en ajoutant des attributs signés et non signés.

Cette Section fournit une introduction aux parties majeures impliquées (paragraphe 4.1) introduit le concept de politiques de signature (paragraphe 4.2) fournit une vue d'ensemble des divers formats d'ES (paragraphe 4.3) introduit le concept de données de validation, et fournit une vue d'ensemble des formats qui incorporent des données de validation (paragraphe 4.4) et présente les considérations pertinentes sur l'arbitrage (paragraphe 4.5) et pour le processus de validation (paragraphe 4.6).

Les spécifications des attributs figurent dans les Sections 5 et 6 ; les annexes C et D donnent les raisons des définitions des différentes formes d'ES.

4.1 Parties majeures

Les parties majeures impliquées dans une transaction d'affaire impliquant une signature électronique, comme définie dans le présent document, sont :
- le signataire,
- le vérificateur,
- les fournisseurs de service de confiance (TSP) et
- l'arbitre.

Le signataire est l'entité qui crée la signature électronique. Quand le signataire signe numériquement sur les données en utilisant le format prescrit, cela représente en engagement au nom de l'entité signataire sur les données signées.

Le vérificateur est l'entité qui valide la signature électronique ; il peut être une seule ou plusieurs entités.

Les fournisseurs de service de confiance (TSP) sont une ou plusieurs entités qui aident à construire dles relations de confiance entre le signataire et le vérificateur. Ils soutiennent le signataire et le vérificateur au moyen des services de soutien qui incluent des certificats d'utilisateur, des certificats croisés, des jetons d'horatage, des CRL, des ARL, et des réponses d'OCSP. Les TSP suivants sont utilisés pour prendre en charge les fonctions définies dans le présent document :
- autorités de certification,
- autorités d'enregistrement,
- producteurs de CRL,
- répondants à OCSP,
- autorités de répertoires (par exemple, un annuaire),
- autorités d'horodatage,
- autorités de marquage de l'heure,
- producteurs de politique de signature.

Les autorités de certification fournissent aux utilisateurs des certificats de clé publique et un service de révocation.

Les autorités d'enregistrement permettent l'identification et l'enregistrement des entités avant qu'une CA génère des certificats.

Les autorités de répertoires publient les CRL produites par les CA, les politiques de signature produites par les producteurs de politique de signature, et facultativement, les certificats de clé publique.

Les autorités d'horodatage attestent que des données ont été formées avant un instant donné de confiance.

Les autorités de marquage de l'heure enregistrent que des données ont été formées avant un instant donné de confiance.

Les producteurs de politique de signature définissent les politiques de signature à utiliser par les signataires et les vérificateurs.

Dans certains cas, les TSP supplémentaires suivants sont nécessires :
- autorités d'attribut : elles fournissent aux utilisateurs des attributs liés aux certificats de clé publique.

Un arbitre est une entité qui arbitre les disputes entre un signataire et un vérificateur.

4.2 Politiques de signature

Le présent document inclut le concept de politiques de signature qui peuvent être utilisées pour établir une cohérence technique lors de la validation de signatures électroniques.

Quand une politique de signature complète utilisée par le vérificateur est soit explicitement indiquée par le signataire, soit impliquée par les données à signer, un résultat cohérent peut être obtenu à la validation d'une signature électronique.

Quand la politique de signature utilisée par le vérificateur n'est ni indiquée par le signataire ni ne peut être déduite d'autres données, ou si la politique de signature est incomplète, alors les vérificateurs, incluant les arbitres, peuvent obtenir des résultats différents quand ils valident une signature électronique. Donc, des politiques de signature complètes qui assurent la cohérence de la validation de signature sont recommandées du point de vue aussi bien du signataire que du vérificateur.

On trouvera plus d'informations sur les politiques de signature dans :
- le rapport technique d'ETSI [TR102038],
- les paragraphes 5.8.1, C.1, et C.3.1 du présent document,
- la [RFC3125],
- le rapport technique d'ETSI [TR102272].

4.3 Formats de signature électronique

Ce paragraphe fournit une vue d'ensemble de deux formes de signature électronique évoluée fondée sur la CMS spécifiées dans le présent document, à savoir la signature électronique CAdES de base (CAdES-BES) et la signature électronique CAdES fondée sur une politique explicite (CAdES-EPES). La conformité au présent document rend obligatoire que le signataire crée un de ces formats.

4.3.1 Signature électronique CAdES de base (CAdES-BES)

Une signature électronique CAdES de base (CAdES-BES) en accord avec le présent document, contient :
- les données signées d'utilisateur (par exemple, le document du signataire) comme défini dans la CMS [RFC3852] ;
- une collection d'attributs signés obligatoires, comme défini dans la CMS [RFC3852] et dans ESS [RFC2634] ;
- des attributs signés obligatoires supplémentaires, définis dans le présent document ; et
- la valeur de la signature numérique calculée sur les données d'utilisateur et, quand ils sont présents, les attributs signés, comme défini dans la CMS [RFC3852].

Une signature électronique CAdES de base (CAdES-BES) en accord avec le présent document, peut contenir :
- une collection d'attributs signés supplémentaires, et
- une collection d'attributs non signés facultatifs.

Les attributs signés obligatoires sont:
- Type de contenu. Il est défini dans la [RFC3852] et spécifie le type de la valeur EncapsulatedContentInfo *(informations de contenu encapsulé)* signée. Les détails sont fournis au paragraphe 5.7.1 du présent document. La raison de son inclusion est fournie à l'Annexe C.3.7.
- Résumé de message. Il est défini dans la [RFC3852] et spécifie le résumé de message de la chaîne d'octets eContent au sein des informations de contenu encapsulé qui sont signées. Les détails sont fournis au paragraphe 5.7.2.
- Certificat de signature ESS OU certificat de signature ESS version 2. L'attribut Certificat de signature ESS est défini dans les services de sécurité améliorés (ESS), [RFC2634], et permet seulement l'utilisation de SHA-1 comme algorithme de résumé. L'attribut Certificat de signature ESS version 2 est défini dans "Mise à jour des services de sécurité améliorée (ESS) : ajout du choix d'algorithme CertID", [RFC5035], et permet l'utilisation de tout algorithme de résumé. Une CAdES-BES qui revendique la conformité au présent document doit inclure l'un d'eux. Le paragraphe 5.7.3 fournit les détails de ces attributs. La raison de son inclusion est donnée à l'Annexe C.3.3. Des attributs facultatifs signés peuvent être ajoutés à la CAdES-BES, incluant des attributs signés facultatifs définis dans la CMS [RFC3852], dans ESS [RFC2634], et dans le présent document. La liste des attributs facultatifs qui figure ci-dessus donne les attributs facultatifs qui sont définis à la Section 5 et dont la raison est donnée à l'Annexe C.
- Moment de la signature, défini dans la CMS [RFC3852], indique le moment de la signature, telle que revendiqué par le signataire. Les détails et une courte raison sont fournis au paragraphe 5.9.1. L'Annexe C.3.6 fournit la raison.
- Indications de contenu, défini dans ESS [RFC2634], fournit des informations qui décrivent le contenu signé le plus interne d'un message multi couches où un contenu est encapsulé dans un autre. Le paragraphe 5.10.1 fournit les détails de la spécification. L'Annexe C.3.8 fournit la raison.
- Référence de contenu est défini dans ESS [RFC2634] et peut être incorporé comme moyen de lier les messages de demande et de réponse dans un échange entre deux parties. Le paragraphe 5.10.1 fournit les détails de la spécification et l'Annexe C.3.9 donne la raison.
- Identifiant de contenu est défini dans ESS [RFC2634] et contient un identifiant qui peut être utilisé ultérieurement sur l'attribut de référence de contenu précédent. Le paragraphe 5.10.2 fournit les détails de la spécification.

- Indication de type d'engagement : cet attribut est défini par le présent document comme moyen d'indiquer l'engagement pris par le signataire lors de la production de la signature. Le paragraphe 5.11.1 fournit les détails de la spécification. L'Annexe C.3.2 en donne la raison.
- Localisation du signataire : cet attribut est défini par le présent document. Il permet au signataire d'indiquer l'endroit où le signataire a délibérément produit la signature. Le paragraphe 5.11.2 fournit les détails de la spécification. L'Annexe C.3.5 en donne la raison.
- Attributs du signataire : cet attribut est défini par le présent document. Il permet qu'un rôle revendiqué ou certifié soit incorporé dans les informations signées. Le paragraphe 5.11.3 fournit les détails de la spécification. L'Annexe C.3.4 en fournit la raison.
- Horodatage de contenu : cet attribut est défini par le présent document. Il permet qu'un jeton d'horodatage des données soit signé pour être incorporé dans les informations signées. Il fournit la preuve de l'existence des données avant la création de la signature. Le paragraphe 5.11.4 fournit les détails de la spécification. l'Annexe C.3.6 en donne la raison. Une forme CAdES-BES peut aussi incorporer des instances d'attributs non signés, comme défini dans la CMS [RFC3852]) et ESS [RFC2634].
- Contre signature, défini dans la CMS [RFC3852] ; il peut être incorporé chaque fois que des signatures incorporées (c'est-à-dire, une signature sur une signature précédente) sont nécessaires. Le paragraphe 5.9.2 donne les détails de la spécification. L'Annexe C.5 en donne la raison.

La structure de CAdES-BES est illustrée à la Figure 1.

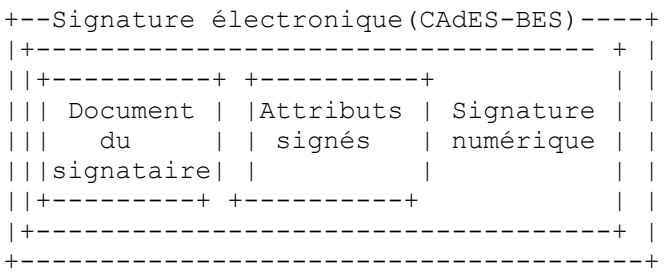


Figure 1 : Illustration d'une CAdES-BES

Les exigences de conformité d'un signataire d'une CAdES-BES sont définies au paragraphe 8.1.

Note : La CAdES-BES est le format minimum pour une signature électronique générée par le signataire. D'elle-même, elle ne fournit pas assez d'informations pour être vérifiée dans le long terme. Par exemple, des informations de révocation produites par des informations pertinentes d'état de certificat du producteur doivent être disponibles pour la validation à long terme (voir le paragraphe 4.4.2).

La CAdES-BES satisfait aux exigences légales pour les signatures électroniques, comme définies dans la Directive européenne sur les signatures électroniques, (voir à l'Annexe C la discussion sur les relations entre le présent document et la Directive). Elle fournit l'authentification et la protection d'intégrité de base.

La sémantique des données signées d'une CAdES-BES ou de son contexte peut implicitement indiquer une politique de signature au vérificateur.

La spécification du contenu des politiques de signature sort du domaine d'application du présent document. Cependant, plus d'informations sur les politiques de signature sont fournies dans le rapport technique d'ETSI [TR102038], la [RFC3125], et les paragraphes 5.8.1, C.1, et C.3.1 du présent document.

4.3.2 Signature électronique CAdES fondée sur une politique explicite (CAdES-EPES)

Une signature électronique CAdES fondée sur une politique explicite (CAdES-EPES), en accord avec le présent document, étend la définition d'une signature électronique pour se conformer à la politique de signature identifiée.

Une signature électronique CAdES fondée sur une politique explicite (CAdES-EPES) incorpore un attribut signé (sigPolicyID) qui indique la politique de signature qui devra être utilisée pour valider la signature électronique. Cet attribut signé est protégé par la signature. La signature peut aussi avoir d'autres attributs signés exigés pour se conformer à la politique de signature obligatoire.

Le paragraphe 5.7.3 fournit les détails de la spécification de l'attribut Identifiant de politique de signature. L'Annexe C.1 en fournit une courte raison. La spécification du contenu des politiques de signature sort du domaine d'application du présent document.

Plus d'informations sur les politiques de signature figurent dans le rapport tecnique d'ETSI [TR102038] et aux paragraphes 5.8.1, C.1, et C.3.1 du présent document.

La structure de la CAdES-EPES est illustrée à la Figure 2.

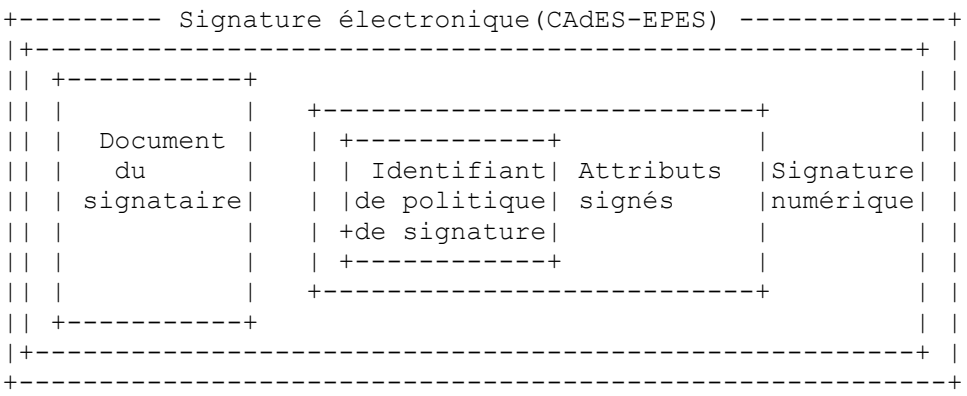


Figure 2 : Illustration d'une CAdES-EPES

Les exigences de conformité du signataire d'une CAdES-EPES sont définies au paragraphe 8.2.

4.4 Formats de signature électronique avec données de validation

La validation d'une signature électronique, en accord avec le présent document, exige des données supplémentaires nécessaires pour valider la signature électronique. Ces données supplémentaires sont appelées des données de validation, et incluent :
- des certificats de clé publique (PKC, *Public Key Certificate*) ;
- des infomations d'état de révocation pour chaque PKC ;
- des horodatages de confiance appliqués à la signature numérique, autrement une marque de temps devra être disponible dans un enregistrement d'audit ;
- quand c'est approprié, les détails d'une politique de signature à utiliser pour vérifier la signature électronique.

Les données de validation peuvent être collectées par le signataire et/ou le vérificateur. Quand l'attribut Identifiant de politique de signature signé est présent, il devra satisfaire les exigences de la politique de signature.

Les données de validation incluent les certificats de CA ainsi que les informations d'état de révocation sous la forme de listes de révocation de certificats (CRL, *Certificate Revocation List*) ou d'informations d'état de certificat (OCSP) fournies par un service en ligne. Les données de validation incluent aussi la preuve que la signature a été créée avant un instant particulier ; ce peut être un jeton d'horodatage ou une marque temporelle.

Le présent document définit des attributs non signés, capables de contenir des données de validation, qui peuvent être ajoutés à la CAdES-BES et CAdES-EPES, conduisant à des formats de signature électronique qui incluent des données de validation. Les paragraphes qui suivent résument ces formats et leurs caractéristiques les plus pertinentes.

4.4.1 Signature électronique avec heure (CAdES-T)

Une signature électronique avec heure (CAdES-T), en accord avec le présent document, est quand il exite une heure de confiance associée à l'ES.

L'heure de confiance peut être fournie par :
- un attribut Horodatage comme attribut non signé ajouté à l'ES, et
- une marque de temps de l'ES fournie par un fournisseur de service de confiance..

L'attribut Horodatage contient un jeton d'horodatage de la valeur de la signature électronique. Le paragraphe 6.1.1 fournit les détails de la spécification. L'Annexe C.4.3 en fournit la raison.

Une marque de temps fournie par un service de confiance aurait un effet similaire à celui de l'attribut Horodatage de signature, mais dans ce cas, aucun attribut n'est ajouté à l'ES, car il est de la responsabilité du TSP de fournir la preuve d'une marque de temps quand il est exigé qu'il le fasse. La gestion des marques de temps sort du domaine d'application du présent document.

L'heure de confiance fournit les étapes initiales vers la fourniture de la validité à long terme. Les signatures électroniques avec l'attribut Horodatage ou une marque de temps sur une BES/EPES, formant la CAdES-T sont illustrées à la Figure 3.

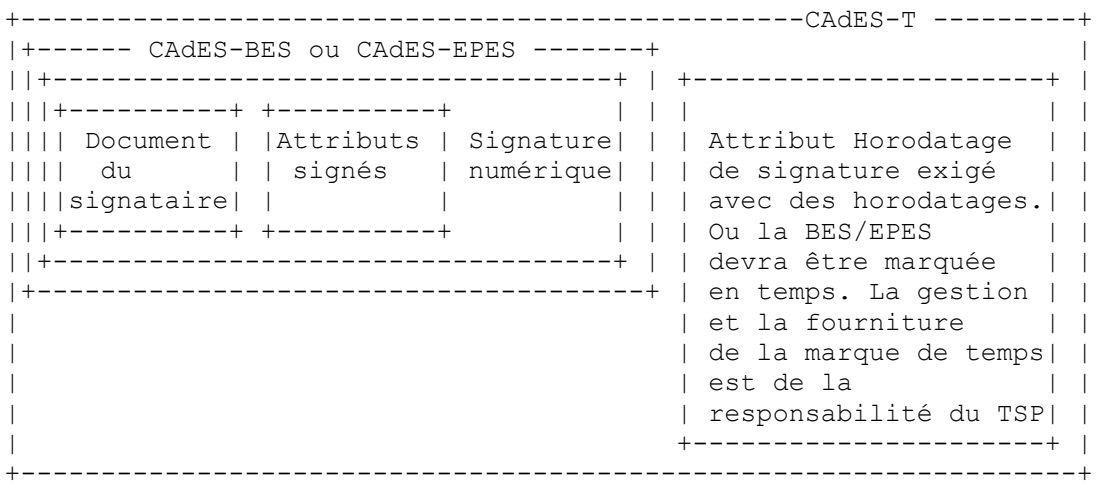


Figure 3 : Illustration des formats de CAdES-T

Note 1 : un jeton d'horodatage est ajouté à la CAdES-BES ou CAdES-EPES comme attribut non signé.
Note 2 : les jetons d'horodatage qui peuvent eux mêmes inclure des attributs non signés exigés pour valider le jeton d'horodatage, comme des attributs Références complètes de certificat et Références complètes de révocation, comme définis par le présent document.

4.4.2 ES avec références de données de validation complètes (CAdES-C)

Les signatures électroniques avec références complètes de données de validation (CAdES-C) en accord avec le présent document, ajoutent à la CAdES-T les attributs Références complètes de certificat et Références complètes de révocation, comme défini dans le présent document. L'attribut Références complètes de certificat contient les références de tous les certificats présents dans le chemin de certification utilisé pour vérifier la signature. L'attribut Références complètes de révocation contient les références aux CRL et/ou réponses OCSP utilisées pour vérifier la signature. Le paragraphe 6.2 fournit les détails de la spécification. La mémorisation des références permet de mémoriser ailleurs les valeurs du chemin de certification et des CRL ou réponses OCSP, réduisant la taille d'un format de signature électronique mémorisé.
Les paragraphes C.4.1 à C.4.2 donnent les raisons de l'usage des données de validation et quand il convient de générer la forme CAdES-C. Les signatures électroniques, avec les données de validation supplémentaires formant la CAdES-C, sont illustrées par la Figure 4.

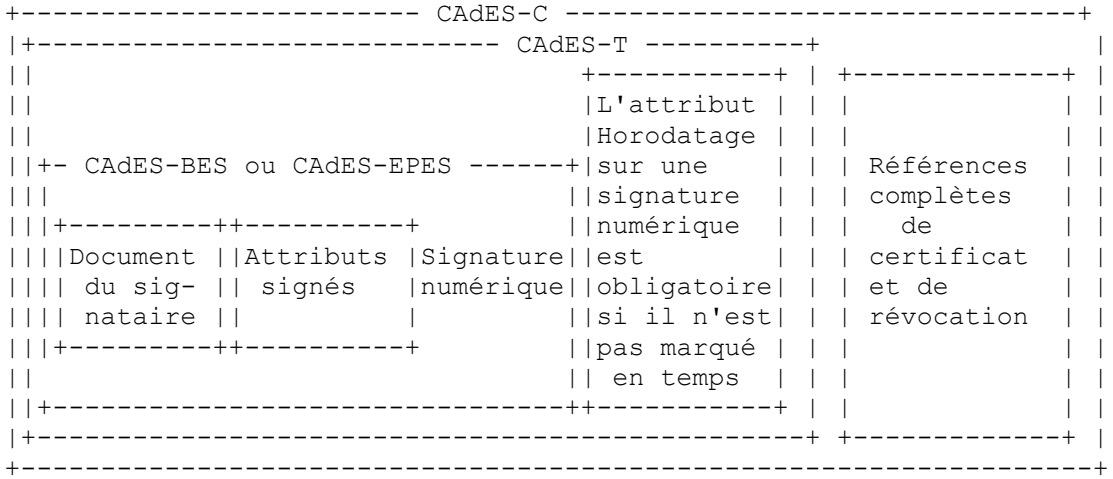


Figure 4 : Illustration du format CAdES-C

Note 1 : Les références complètes de certificat et de révocation sont ajoutées à la CAdES-T comme attribut non signé.

Note 2 : Au minimum, le signataire va fournir la CAdES-BES ou, quand il indique que la signature se conforme à une politique explicite de signature, la CAdES-EPES.

Note 3 : Pour réduire le risque de répudiation de la création de signature, l'indication de l'heure de confiance doit être aussi proche que possible de l'heure de création de la signature. Le signataire ou un TSP pourrait fournir la CAdES-T ; sinon, le vérificateur devrait créer la CAdES-T à la première réception d'une signature électronique parce que la CAdES-T fournit une preuve indépendante de l'existence de la signature avant l'indication de l'heure de confiance.

Note 4 : Une CAdES-T avec indication d'heure de confiance doit être créée avant qu'un certificat ait été révoqué ou soit arrivé à expiration.

Note 5 : Le signataire et les TSP pourraient fournir la CAdES-C pour minimiser ce risque, et quand le signataire ne fourni pas de CAdES-C, le vérificateur devrait créer la CAdES-C quand le composant de révocation et les données de validation exigés deviennent disponibles ; cela peut exiger une période de grâce.

Note 6 : Une période grâce permet que les informations de révocation de certificat se propagent par le processus de révocation. Cette période pourrait s'étendre du moment où une entité autorisée demande la révocation du certificat au moment où l'information est disponible pour que le consommateur d'assertions l'utilise. Afin de s'assurer que le certificat n'a pas été révoqué au moment où la signature a été marquée en temps ou horodatée, les vérificateurs devraient attendre jusqu'à la fin de la période de grâce. Une politique de signature peut définir des valeurs spécifiques pour les périodes de grâce.

Une illustration d'une période de grâce est fournie à la Figure 5.

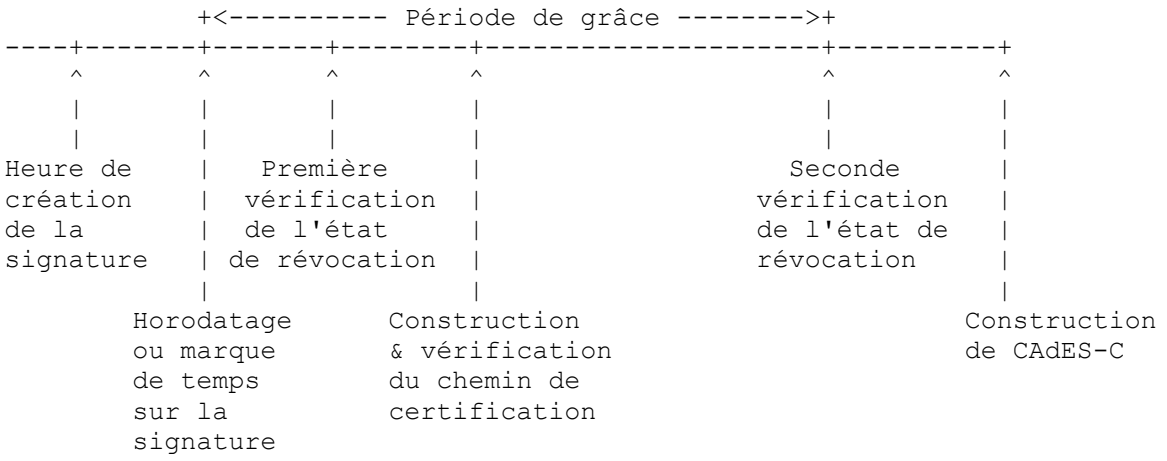


Figure 5 : Illustration d'une période de grâce

Note 7 : [CWA14171] spécifie un processus de validation de signature qui utilise CAdES-T, CAdES-C, et une période de grâce. L'Annexe B fournit un exemple de processus de validation. L'Annexe C.4 fournit des informations supplémentaires sur l'application de périodes de grâce durant le processus de validation.

Les exigences de conformité du vérificateur sont définies au paragraphe 8.3 pour une CAdES-C horodatée, et au paragraphe 8.4 pour une CAdES-C marquée en temps. Le présent document définit seulement les exigences de conformité pour le vérificateur jusqu'à une ES avec données de validation complètes (CAdES-C). Cela signifie qu'aucune des formes étendues et d'archive des signatures électroniques, comme défini aux paragraphes 4.4.3 et 4.4.4, n'a besoin d'être mise en œuvre pour réaliser la conformité au présent document.

4.4.3 Formats étendus de signature électronique

CAdES-C peut être extendue par l'ajout d'attributs non signés à la signature électronique. Le présent document définit divers attributs non signés applicables à toute vérification à long terme, et pour prévenir des situations désastreuses discutées à l'Annexe C. L'Annexe B fournit les détails des divers formats étendus, tous les attributs non signés requis pour

chaque type, et comment ils peuvent être utilisés au sein du processus de validation de signature électronique. Les paragraphes qui suivent donnent une vue d'ensemble des diverses formes de format de signature étendue dans le présent document.

4.4.3.1 EXtended Long signature électronique (CAdES-X Long)

Le format étendu long (CAdES-X Long) en accord avec le présent document, ajoute les attributs Valeurs de certificat et Valeurs de révocation au format CAdES-C. Le premier contient le chemin de certification entier nécessaire pour la vérifification de la signature ; le second contient les CRL et les réponses OCSP requises pour la validation de la signature. Cela fournit un répertoire connu des informations de certificat et de révocation nécessaires pour valider une CAdES-C et empêcher que ces informations soient perdues. Les paragraphes 6.3.3 et 6.3.4 donnent les détails de la spécification. L'Annexe B.1.1 donne des détails sur la production du format. Les Annexes C4.1 et C.4.2 en donnent la raison.

La structure du format CAdES-X Long est illustrée par la Figure 6.

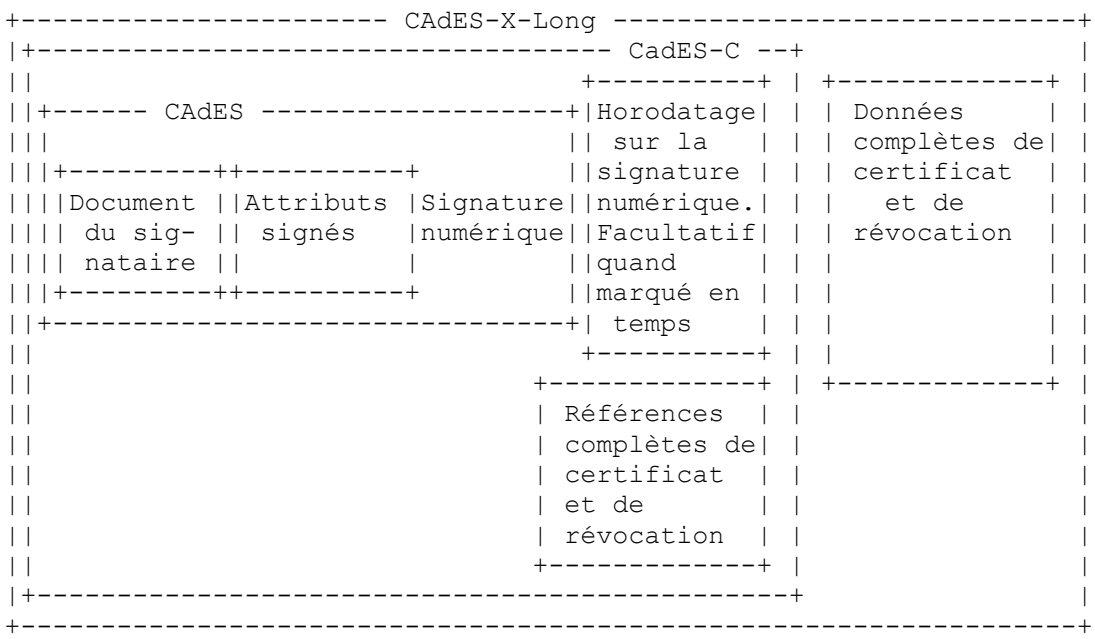


Figure 6 : Illustration de CAdES-X-Long

4.4.3.2 Signature électronique étendue avec temps de type 1 (CAdES-X Type 1)

Le format étendu avec temps de type 1 (CAdES-X Type 1) en accord avec le présent document, ajoute l'attribut Horodatage CAdES-C, dont le contenu est un jeton d'horodatage sur la CAdES-C elle-même, au format CAdES-C.

Cela fournit une protection d'intégrité et une heure de confiance sur tous les éléments et références. Cela peut protéger les certificats, les CRL, et les réponses OCSP en cas d'une compromission ultérieure d'une clé de CA, de CRL, ou de producteur OCSP. Le paragraphe 6.3.5 fournit les détails de la spécification. L'Annexe B.1.2 donne les détails de la production du processus d'horodatage. L'Annexe C.4.4.1 fournit la raison.

La structure du format CAdES-X Type 1 est illustrée à la Figure 7.

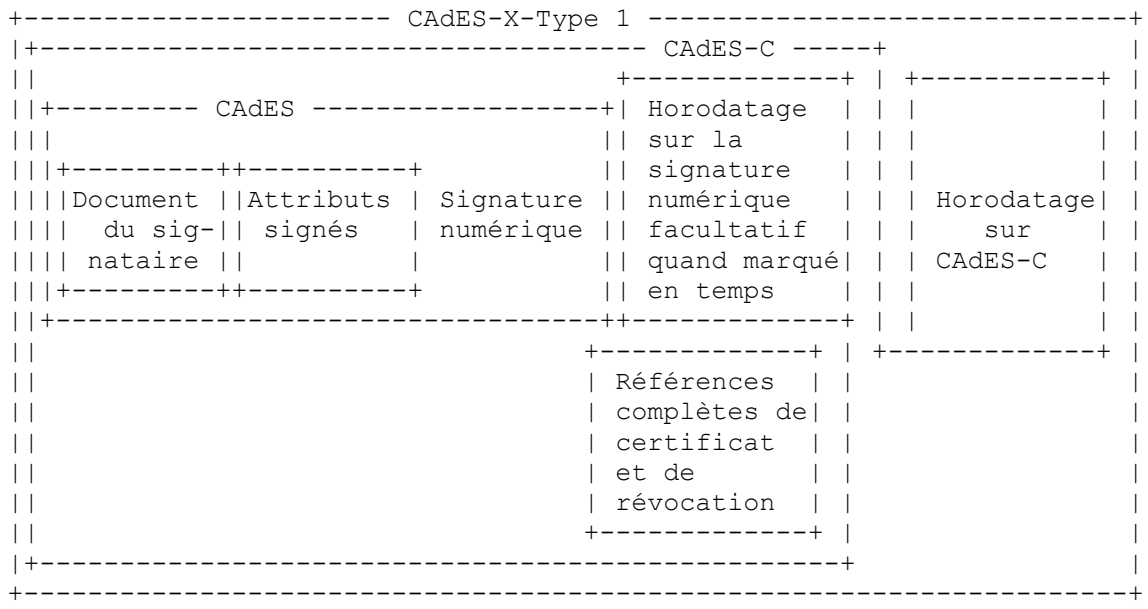


Figure 7 : Illustration de CAdES-X Type 1

4.4.3.3 Signature électronique étendue avec temps de type 2 (CAdES-X Type 2)

Le format étendu avec temps de type 2 (CAdES-X Type 2) en accord avec le présent document, ajoute au format CAdES-C l'attribut Références de crl/cert horodaté CAdES-C, dont le contenu est un jeton d'horodatage sur le chemin de certification et les références d'informations de révocation. Cela fournit une protection de l'intégrité et de l'heure de confiance sur toutes les références. Il peut protéger les certificats, les CRL et les réponses OCSP dans le cas d'une compromission ultérieure d'une clé de CA, d'une clé de CRL ou d'une clé de producteur d'OCSP.

Les deux CAdES-X Type 1 et CAdES-X Type 2 contrent les mêmes menaces, et l'usage de l'une ou l'autre dépend de l'environnement. Le paragraphe 6.3.5 fournit les détails de spécification. L'Annexe B.1.3 donne les détails sur la production du processus d'horodatage. L'Annexe C.4.4.2 fournit la raison.
La structure du format de CAdES-X Type 2 est illustrée à la Figure 8.

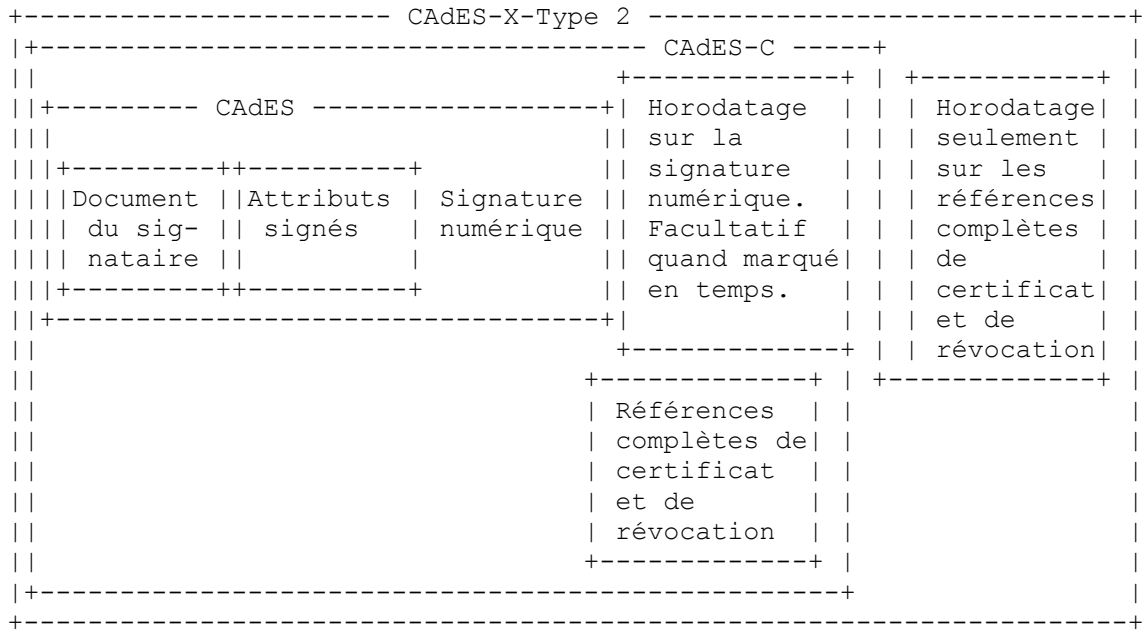


Figure 8 : Illustration de CAdES-X Type 2

4.4.3.4 Signature électronique étendue longue avec temps (CAdES-X Long Type 1 ou 2)

L'extension longue avec temps (CAdES-X Long Type 1 ou 2) en accord avec le présent document, est une combinaison de CAdES-X Long et d'un des deux types précédents (CAdES-X Type 1 et CAdES-X Type 2). L'Annexe B.1.4 donne des détails sur la production du processus d'horoatage. L'Annexe C.4.8 en donne la raison.

La structure du format CAdES-X Long Type 1 et CAdES-X Long Type 2 est illustrée par la Figure 9.

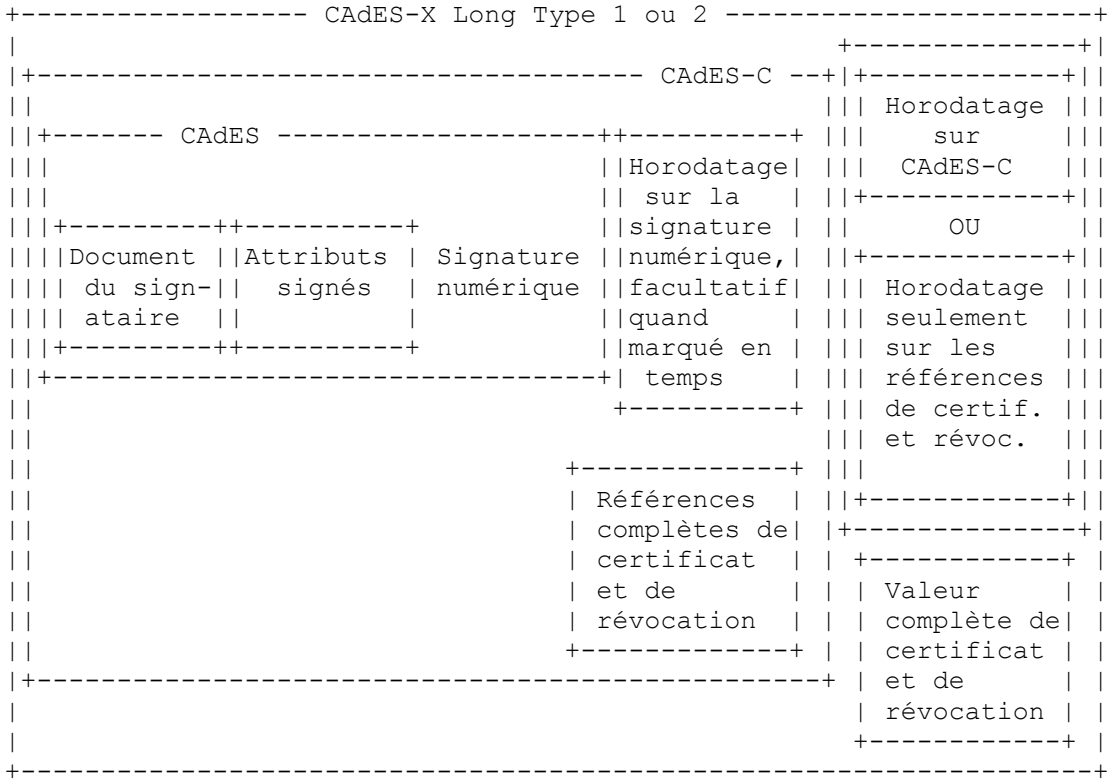
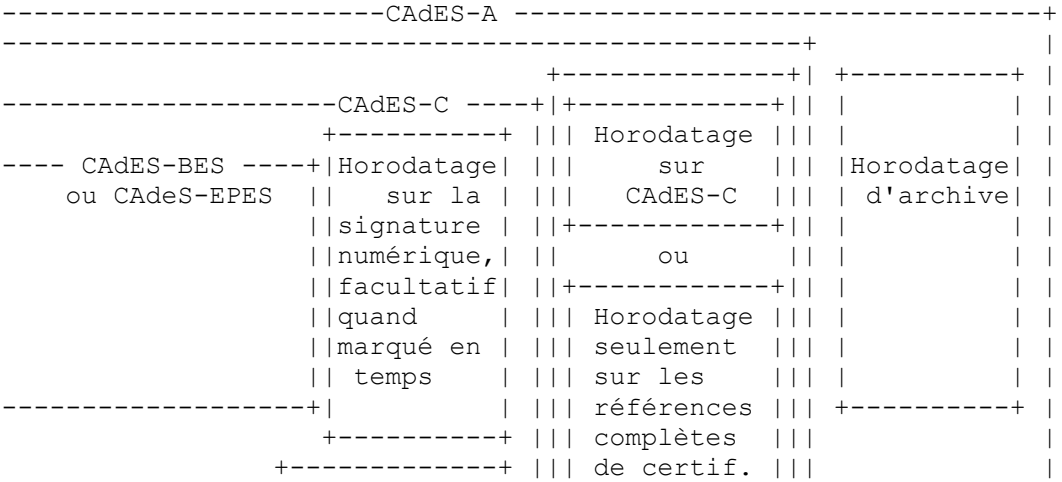


Figure 9 : Illustration de CAdES-X Long Type 1 et de CAdES Long Type 2

4.4.4 Signature électronique d'archivage (CAdES-A)

La forme d'archivage (CAdES-A) en accord avec le présent document, se construit sur une CAdES-X Long ou a CAdES-X Long Type 1 ou 2 en ajoutant un ou plusieurs attributs Horodatage d'archive. Cette forme est utilisée pour l'archivage à long terme de signatures. Les horodatages successifs protègent tout le matériel contre des algorithmes de hachage vulnérables ou la violation du matériel ou des algorithmes de chiffrement. Le paragraphe 6.4 contient les détails de la spécification. Les paragraphes C.4.5 et C.4.8 donnent la raison.

La structure de la forme CAdES-A est illustrée par la Figure 10.



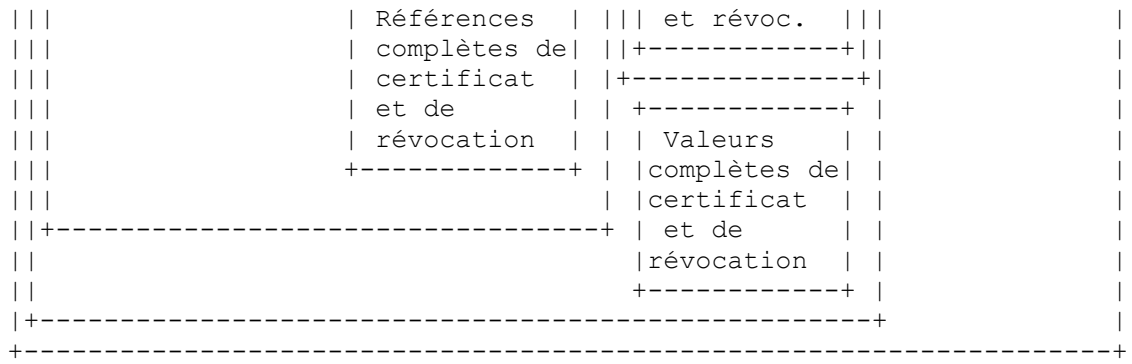


Figure 10 : Illustration of CAdES-A

4.5 Arbitrage

CAdES-C peut être utilisée pour l'arbitrage si il devait y avoir une dispute entre le signataire et le vérificateur, pourvu que :

- l'arbitre sache où récupérer le certificat du signataire (si il n'est pas déjà présent) tous les certificats croisés et les CRL, ACRL, ou réponse OCSP requis référencés dans la CAdES-C ;
- quand l'horodatage est utilisé dans la CAdES-T, le certificat provenant de la TSU qui a produit le jeton d'horodatage dans le format CAdES-T est encore dans sa période de validité ;
- quand l'horodatage est utilisé dans la CAdES-T, le certificat provenant de la TSU qui a produit le jeton d'horodatage dans le format CAdES-T n'est pas révoqué au moment de l'arbitrage ;
- quand le marquage de temps est utilisé dans la CAdES-T, un chemin d'audit fiable provenant de l'autorité de marquage de temps est disponible pour l'examen du temps ;
- aucune des clés privées correspondant aux certificats utilisés pour vérifier la chaîne de signature n'a jamais été compromise ;
- le chiffrement utilisé au moment de la construction de la CAdES-C n'a pas été cassé au moment où l'arbitrage est effectué ; et
- si la politique de signature peut être explicitement ou implicitement identifiée, alors un arbitre est capable de déterminer les règles nécessaires pour valider la signature électronique.

4.6 Processus de validation

Le processus de validation valide une signature électronique ; l'état de sortie du processus de validation peut être :
- invalide,
- validation incomplète,
- valide.

Une réponse invalide indique que soit le format de signature est incorrect, soit que la valeur de la signature numérique n'a pas réussi la vérification (par exemple, la vérification d'intégrité sur la valeur de signature numérique échoue, ou un des certificats dont dépend la vérification de signature numérique est connu pour être invalide ou révoqué).

Une réponse de validation incomplète indique que l'état de validation de la signature est actuellement inconnu. Dans le cas d'une validation incomplète, des informations supplémentaires peuvent être rendues disponibles à l'application ou à l'utilisateur, leur permettant de décider que faire avec la signature électronique. Dans le cas d'une validation incomplète, la signature électronique peut être vérifiée à nouveau plus tard quand des informations supplémentaires seront disponibles.

Note : par exemple, une validation incomplète peut être parce que tous les certificats requis ne sont pas disponibles ou que la période de grâce n'est pas achevée.

Une réponse valide indique que la signature a passé avec succès la vérification, et qu'elle se conforme à la politique de validation de signature.

Des exemples de séquence de validation sont illustrés dans l'Annexe B.

5. Attributs de signature électronique

Cette Section s'appuie sur la syntaxe de message cryptographique (CMS) existante, comme définie dans la [RFC3852], et des services de sécurité améliorée (ESS) comme définie dans la [RFC2634]. La structure globale d'une signature électronique est comme défini dans la CMS. La signature électronique (ES) utilise les attributs définis dans la CMS, ESS, et le présent document. Le présent document définit des attributs d'ES qu'il utilise et qui ne sont pas définis ailleurs.

L'ensemble d'attributs obligatoires et la valeur de la signature numérique sont définis comme la signature électronique minimum requise par le présent document. Une politique de signature peut rendre obligatoires que d'autres attributs signés soient présents.

5.1 Syntaxe générale

La syntaxe générale de l'ES est définie dans la CMS [RFC3852].

Note : la CMS définit des types de contenu pour id-data, id-signedData, id-envelopedData, id-digestedData, id-encryptedData, et id-authenticatedData. Bien que la CMS permette que d'autres documents définissent d'autres types de contenu, le type ASN.1 défini ne devrait pas être d'un type CHOIX. Le présent document ne définit pas d'autres types de contenu.

5.2 Type de contenu de données

Le type de contenu des données de l'ES est défini dans la CMS [RFC3852].

Note : Si le type de contenu est id-data, il est recommandé que le contenu soit codé en utilisant MIME, et que le type MIME soit utilisé pour identifier le format de présentation des données. Voir à l'Annexe F.1 un exemple d'utilisaion de MIME pour identifier le type de codége.

5.3 Type de contenu Signed-data

Le type de contenu Signed-data de l'ES est défini dans la CMS [RFC3852].

5.4 Type SignedData

La syntaxe de SignedData dans l'ES est défini dans la CMS [RFC3852].

Les champs de type SignedData sont comme défini dans la CMS [RFC3852].

L'identification du certificat d'un signataire utilisé pour créer la signature est toujours signée (voir le paragraphe 5.7.3). La politique de validation peut spécifier des exigences sur la présence de certains certificats. Le cas limite où il n'y a pas de signataire n'est pas valide dans le présent document.

5.5 Type EncapsulatedContentInfo

La syntaxe du type d'ES EncapsulatedContentInfo *(informations de contenu encapsulé)* est comme défini dans la CMS [RFC3852].

Pour les besoins de la validation à long terme, comme définie dans le présent document, il est conseillé que le eContent soit présent, ou que les données qui sont signées soient archivées d'une façon telle que cela préserve tout le codage des données. Il est important que la CHAINE D'OCTETS utilisée pour générer la signature reste la même chaque fois que le vérificateur ou un arbitre valide la signature.

Note : le eContent est facultatif dans la CMS :

- Quand il est présent, cela permet que les données signées soient encapsulées dans la structure SignedData qui contient alors les données signées et la signature. Cependant, les données signées peuvent seulement être accédées par un vérificateur capable de décoder la structure SignedData codée en ASN.1.
- Quand il manque, cela permet que les données signées soient envoyées ou mémorisées séparément de la signature, et la structure SignedData contient seulement la signature. C'est, dans le cas de la signature, seulement les données qui sont signées qui ont besoin d'être mémorisées et distribuées de façon à préserver tout codage des données.

Le cas limite où il n'y a pas de signataire n'est pas valide dans le présent document.

5.6 Type SignerInfo

La syntaxe du type d'ES SignerInfo est comme défini dans la CMS [RFC3852].

Les informations par signataire sont représentées dans le type SignerInfo. Dans le cas de plusieurs signatures indépendantes (voir l'Annexe B.5) il y a une instance de ce champ pour chaque signataire.

Les champs de type SignerInfo ont la signification définie dans la CMS [RFC3852], mais le champ signedAttrs devra contenir les attributs suivants :
- content-type *(type de contenu)* comme défini au paragraphe 5.7.1,
- message-digest *(résumé de message)* comme défini au paragraphe 5.7.2,
- signing-certificate *(certificat de signature)* comme défini au paragraphe 5.7.3.

5.6.1 Processus de calcul de résumé de message

Le processus de calcul du résumé de message est comme défini dans la CMS [RFC3852].

5.6.2 Processus de génération de signature de message

L'entrée du processus de génération de signature de message est comme défini dans la CMS [RFC3852].

5.6.3 Processus de vérification de signature de message

Les procédures pour la vérification de signature de message sont définies dans la CMS [RFC3852] et améliorées dans le présent document : l'entrée du processus de vérification de signature doit être la clé publique du signataire, qui doit être vérifiée comme correcte en utilisant l'attribut de référence du certificat de signature contenant une référence au certificat de signature, c'est-à-dire, quand on utilise SigningCertificateV2 de la RFC 5035 [X.690] ou SigningCertificate de ESS [RFC2634], la clé publique provenant du premier certificat identifié dans la séquence des identifiants de certificat provenant du SigningCertificate doit être la clé utilisée pour vérifier la signature numérique.

5.7 Attributs ES obligatoires de base présents

Les attributs suivants doivent être présents avec les données signées définies dans le présent document. Les attributs sont définis dans la CMS [RFC3852].

5.7.1 content-type

L'attribut content-type indique le type du contenu signé. La syntaxe du type d'attribut contenu-type est comme défini dans la CMS [RFC3852] au paragraphe 11.1.

Note 1 : comme mentionné dans la [RFC3852], l'attribut content-type doit avoir sa valeur (c'est-à-dire, ContentType) égale au eContentType de la valeur des EncapsulatedContentInfo signées.

Note 2 : pour les mises en œuvre qui prennent en charge la génération de signature, si l'attribut content-type est id-data, il est alors recommandé que le eContent soit codé en utilisant MIME. Pour les mises en œuvre qui prennent en charge la vérification de signature, si les données signées (c'est-à-dire, eContent) sont codées avec MIME, alors l'OID de l'attribut content-type doit être id-data. Dans les deux cas, le ou les content-type MIME doivent être utilisés pour identifier le format de présentation des données. Voir les détails sur l'utilisation de MIME à l'Annexe F.

5.7.2 Message-digest

La syntaxe du type d'attribut message-digest de l'ES est comme défini dans la CMS [RFC3852].

5.7.3 Attributs de référence de certificat de signature

Les attributs de référence de certificat de signature sont pris en charge en utilisant l'attribut ESS signing-certificate ou l'attribut ESS-signing-certificate-v2.

Ces attributs doivent contenir une référence au certificat du signataire ; ils sont conçus pour empêcher des attaques simples de substitution et reproduction et pour permettre qu'un ensemble restreint de certificats soit utilisé pour vérifier une signature. Ils ont une forme compacte (plus courte que le certificat complet) qui permet qu'un certificate soit identifié sans ambiguïté.

Un, et un seul des attributs alternatifs suivants devra être présent avec les signedData, définies par le présent document :
- l'attribut ESS signing-certificate, défini dans ESS [RFC2634], doit être utilisé si l'algorithme de hachage SHA-1 est utilisé,
- l'attribut ESS signing-certificate-v2, défini dans "Mise à jour des services de sécurité améliorée (ESS) : ajout du choix d'algorithme CertID", [RFC5035], qui devra être utilisé quand d'autres algorithmes sont utilisés.

Le certificat à utiliser pour vérifier la signature devra être identifié dans la séquence (c'est-à-dire, le certificat provenant du signataire) et la séquence ne devra pas être vide. La politique de validation de signature peut rendre obligatoire que d'autres certificats soient présents qui peuvent inclure tous les certificats jusqu'à l'ancre de confiance.

5.7.3.1 ESS signing-certificate Attribute Definition

La syntaxe du type d'attribut signing-certificate de l'ES est comme défini dans les services de sécurité améliorés (ESS), [RFC2634], et qualifiés plus précisément dans le présent document.

La séquence du champ Informations de politique n'est pas utilisée dans le présent document.

L'attribut ESS signing-certificate devra être un attribut signé. Le codage de ESSCertID pour ce certificat doit inclure le champ issuerSerial.

Si il est présent, le issuerAndSerialNumber dans le champ SignerIdentifier des SignerInfo devra correspondre au champ issuerSerial présent dans ESSCertID. De plus, le certHash provenant de ESSCertID devra correspondre au hachage SHA-1 du certificat. Le certificat identifié devra être utilisé durant le processus de vérification de signature. Si le hachage du certificat ne correspond pas au certificat utilisé pour vérifier la signature, la signature devra être considérée comme invalide.

Note : lorsque un certificat d'attribut est utilisé par le signataire pour associer un rôle, ou d'autres attributs du signataire, à la signature électronique, il est placé dans le champ Attributs de signataire comme défini au paragraphe 5.8.3.

5.7.3.2 Définition de l'attribut ESS signing-certificate-v2

L'attribut ESS signing-certificate-v2 est similaire à l'attribut ESS signing-certificate défini ci-dessus, sauf que cet attribut peut être utilisé avec des algorithmes de hachage autres que SHA-1.

La syntaxe du type d'attribut signing-certificate-v2 de l'ES est comme défini dans la "Mise à jour des services de sécurité améliorée (ESS) : ajout du choix d'algorithme CertID", [RFC5035], et qualifiée plus précisément dans le présent document.

La séquence du champ Informations de politique n'est pas utilisée dans le présent document.

Cet attribut devra être utilisé de la même manière que défini ci-dessus pour l'attribut ESS signing-certificate.

L'identifiant d'objet pour cet attribut est :
IDENTIFIANT D'OBJET id-aa-signingCertificateV2 ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-aa(2) 47 }

Si iI est présent, le issuerAndSerialNumber dans le champ SignerIdentifier des SignerInfo devra corrrespondre au champ issuerSerial présent dans ESSCertIDv2. De plus, le certHash de ESSCertIDv2 devra correspondre au hachage du certificat calculé en utilisant la fonction de hachage spécifiée dans le champ hashAlgorithm. Le certificat identifié devra être utilisé durant le processus de vérification de signature. Si le hachage du certificat ne correspond pas au certificat utilisé pour vérifier la signature, la signature devra être considérée comme invalide.

Note 1 : lorsque un certificat d'attribut est utilisé par le signataire pour associer un rôle, ou d'autres attributs du signataire, à la signature électronique, il est placé dans les attributs de signataire comme défini au paragraphe 5.8.3.

Note 2 : la RFC 3126 utilisait d'autres attributs de certificat de signature (voir le paragraphe 5.7.3.3) pour le même objet. Cette utilisation est maintenant déconseillée, car cette structure est plus simple.

5.7.3.3 Autre définition de l'attribut signing-certificate

La RFC 3126 utilisait un autre attribut signing-certificate comme solution de remplacement au signing-certificate ESS quand des algorithmes de hachage autres que SHA-1 étaient utilisés. Cette utilisation est maintenant déconseillée, car la structure de l'attribut signing-certificate-v2 est plus simple. Sa description est cependant toujours présente dans cette version pour la rétro compatibilité.

IDENTIFIANT D'OBJET id-aa-ets-otherSigCert ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-aa(2) 19 }

L'autre valeur de l'attribut other-signing-certificate a la syntaxe ASN.1 de OtherSigningCertificate :

```
OtherSigningCertificate ::= SEQUENCE {
  certs     SEQUENCE DE OtherCertID,
  policies  SEQUENCE DE PolicyInformation  FACULTATIF
                                                    -- NON UTILISÉ DANS LE PRÉSENT DOCUMENT }

OtherCertID ::= SEQUENCE {
  otherCertHash       OtherHash,
  issuerSerial        IssuerSerial FACULTATIF }

OtherHash ::= CHOIX {
  sha1Hash OtherHashValue,                  -- cela contient un hachage SHA-1.
  otherHash OtherHashAlgAndValue}

OtherHashValue ::= CHAINE D'OCTETS

OtherHashAlgAndValue ::= SEQUENCE {
  hashAlgorithm     AlgorithmIdentifier,
  hashValue         OtherHashValue }
```

5.8 Attributs obligatoires supplémentaires pour signatures électroniques fondées explicitement sur la politique

5.8.1 signature-policy-identifier

Le présent document rend obligatoire que pour CAdES-EPES, une référence à la politique de signature soit incluse dans les signedData. Cette référence est explicitement identifiée. Une politique de signature définit les règles de création et validation d'une signature électronique, et est incluse comme attribut signé avec chaque signature électronique explicitement fondée sur une politique. L'identifiant de politique de signature devra être un attribut signé.

L'identifiant d'objet suivant identifie l'attribut signature-policy-identifier :

IDENTIFIANT D'OBJET id-aa-ets-sigPolicyId ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-aa(2) 15 }

Les valeurs d'attribut signature-policy-identifier ont le type ASN.1 SignaturePolicyIdentifier :

```
SignaturePolicyIdentifier ::= CHOIX {
   signaturePolicyId          SignaturePolicyId,
   signaturePolicyImplied   SignaturePolicyImplied          -- non utilisé dans cette version.
}

SignaturePolicyId ::= SEQUENCE {
   sigPolicyId             SigPolicyId,
   sigPolicyHash           SigPolicyHash,
   sigPolicyQualifiers   SEQUENCE TAILLE (1..MAX) DE SigPolicyQualifierInfo FACULTATIF}

SignaturePolicyImplied ::= NULL
```

Le champ sigPolicyId contient un identifiant d'objet qui identifie de façon univoque une version spécifique de la politique de signature. La syntaxe de ce champ est la suivante :

SigPolicyId ::= IDENTIFIANT D'OBJET

Le champ sigPolicyHash contient facultativement l'identifiant de l'algorithme de hachage et le hachage de la valeur de la politique de signature. La hashValue au sein du sigPolicyHash peut être réglée à zéro pour indiquer que la valeur du hachage de politique n'est pas connue.

Note : l'utilisation d'une valeur de sigPolicyHash de zéro assure de la rétro compatibilité avec les versions antérieures du présent document. Si sigPolicyHash est zéro, alors la valeur du hachage ne devrait pas être vérifiée par rapport à la valeur de hachage calculée de la politique de signature.

Si la politique de signature est définie en utilisant l'ASN.1, le hachage est alors calculé sur la valeur sans les champs Type et Longueur extérieurs, et l'algorithme de hachage devra être comme spécifié dans le champ sigPolicyHash.

Si la politique de signature est définie en utilisant une autre structure, le type de structure et l'algorithme de hachage devront être spécifiés au titre de la politique de signature, ou indiqués en utilisant un qualificatif de politique de signature.

```
SigPolicyHash ::= OtherHashAlgAndValue

OtherHashAlgAndValue ::= SEQUENCE {
  hashAlgorithm   AlgorithmIdentifier,
  hashValue       OtherHashValue }

OtherHashValue ::= CHAINE D'OCTETS
```

Un identifiant de politique de signature peut être qualifié avec d'autres informations sur le qualificatif. La sémantique et la syntaxe du qualificatif sont comme associées à l'identifiant d'objet dans le champ sigPolicyQualifierId.La syntaxe générale de ce qualificatif est la suivante :

```
SigPolicyQualifierInfo ::= SEQUENCE {
      sigPolicyQualifierId   SigPolicyQualifierId,
      sigQualifier           TOUT DEFINI PAR sigPolicyQualifierId }
```

Le présent document spécifie les qualificatifs suivants :
- spuri : il contient la référence d'URI ou d'URL de la politique de signature, et
- sp-user-notice : il contient une remarque pour l'utilisateur qui devrait être affichée chaque fois que la signature est validée.

sigpolicyQualifierIds est défini dans le présent document :

```
SigPolicyQualifierId ::= IDENTIFIANT D'OBJET

IDENTIFIANT D'OBJET id-spq-ets-uri ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-spq(5) 1 }

SPuri ::= IA5String
IDENTIFIANT D'OBJET id-spq-ets-unotice ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-spq(5) 2 }

SPUserNotice ::= SEQUENCE {
   noticeRef       NoticeReference FACULTATIF,
   explicitText    DisplayText FACULTATIF}

NoticeReference ::= SEQUENCE {
   organization      DisplayText,
   noticeNumbers     SEQUENCE DE ENTIER }

DisplayText ::= CHOIX {
   visibleString   VisibleString  (TAILLE (1..200)),
   bmpString       BMPString      (TAILLE (1..200)),
   utf8String      UTF8String     (TAILLE (1..200)) }
```

5.9 Attributs facultatifs importés de la CMS

Les attributs suivants peuvent être présents avec les données signées ; les attributs sont définis dans la CMS [RFC3852] et importés dans le présent document. Lorsque approprié, les attributs sont qualifiés et profilés par le présent document.

5.9.1 signing-time

L'attribut signing-time spécifie le moment où le signataire prétend avoir effectué le processus de signature.

Les valeurs de l'attribut signing-time pour l'ES ont le type ASN.1 SigningTime comme défini dans la CMS [RFC3852].

Note : la [RFC3852] déclare que les dates entre le 1er janvier 1950 et le 31 décembre 2049 (inclus) doivent être codées en UTCTime. Toutes les dates avec des valeurs d'année avant 1950 ou après 2049 doivent être codées comme GeneralizedTime.

5.9.2 countersignature

Les valeurs de l'attribut countersignature pour une ES ont le type ASN.1 de CounterSignature, comme défini dans la CMS [RFC3852]. Un attribut countersignature devra être un attribut non signé.

5.10 Attributs facultatifs importés de ESS

Les attributs suivants peuvent être présents avec les données signées définies par le présent document. Les attributs sont définis dans ESS et sont importés dans le péesent document et sont qualifiés et profilés de façon appropriée par le présent document.

5.10.1 Attribut content-reference

L'attribut content-reference est un lien d'une SignedData à une autre. Il peut être utilisé pour lier une réponse au message original auquel il se réfère, ou pour incorporer par référence une SignedData dans une autre. L'attribut content-reference devra être un attribut signé.

Les valeurs d'attribut content-reference pour une ES ont le type ASN.1 de ContentReference, comme défini dans ESS [RFC2634].

L'attribut content-reference devra être utilisé comme défini dans ESS [RFC2634].

5.10.2 Attribut content-identifier

L'attribut content-identifier fournit un identifiant pour le contenu signé, à utiliser quand une référence à ce contenu peut être ultérieurement requise, par exemple, dans l'attribut content-reference dans d'autres données signées envoyées ultérieurement. L'attribut content-identifier devra être un attribut signé.

Les valeurs du type d'attribut content-identifier pour l'ES ont le type ASN.1 de ContentIdentifier, comme défini dans la [RFC2634].

L'attribut minimal content-identifier devrait contenir un enchaînement d'informations d'identification spécifiques de l'utilisateur (comme un nom d'utilisateur ou des informations d'identification de matériel de clé publique) une chaîne GeneralizedTime, et un nombre aléatoire.

5.10.3 Attribut content-hints

L'attribut content-hints fournit des informations sur le contenu signé le plus interne d'un message multi couches où un contenu est encapsulé dans un autre.

La syntaxe du type d'attribut content-hints de l'ES est comme défini dans ESS [RFC2634].

Quand il est utilisé pour indiquer le format précis des données à présenter à l'utilisateur, les règles suivantes s'appliquent :

- le contentType indique le type du contenu associé. C'est un identifiant d'objet (c'est-à-dire, une chaîne unique d'entiers) alloué par une autorité qui définit le type de contenu et
- quand le contentType est id-data, la contentDescription devra définir le format de présentation ; le format peut être défini par des types MIME.

Quand le format du contenu est défini par des types MIME, les règles suivantes s'appliquent :
- le contentType devra être id-data, comme défini dans la CMS [RFC3852] ;
- la contentDescription devra être utilisée pour indiquer le codage des données, en accord avec les règles définies dans la [RFC2045] ; voir à l'Annexe F un exemple de contenu structuré et MIME.

Note 1 : IDENTIFIANT D'OBJET id-data ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs7(7) 1 }

Note 2 : contentDescription est facultaif dans ESS [RFC2634]. Il peut être utilisé pour compléter des contentTypes définis ailleurs ; de telles définitions sortent du domaine d'application du présent document.

5.11 Attributs facultatifs supplémentaires définis dans le présent document

Ce paragraphe définit un certain nombre d'attributs qui peuvent être utilisés pour indiquer des informations supplémentaires à un vérificateur :
a) le type d'engagement du signataire, et/ou
b) la localisation prétendue où la signature est effectuée, et/ou
c) les attributs revendiqués ou certifiés du signataire, et/ou
d) un horodatage du contenu appliqué avant la signature du contenu.

5.11.1 Attribut commitment-type-indication

Il peut y avoir des situations où un signataire veut indiquer explicitement à un vérificateur qu'en signant les données, il illustre un type d'engagement au nom du signataire. L'attribut commitment-type-indication porte de telles informations.

L'attribut commitment-type-indication devra être un attribut signé. Le type d'engagement peut être :
- défini au titre de la politique de signature, et dans ce cas, le type d'engagement a une sémantique précise qui est définie au titre de la politique de signature,
- être un type enregistré, et dans ce cas, le type d'engagement a une sémantique précise définie par l'enregistrement, sous les règles de l'autorité d'enregistrement. Cette autorité d'enregistrement peut être une association commerciale ou une autorité réglementaire.

La politique de signature spécifie un ensemble d'attributs qu'elle "reconnaît". Cet ensemble "reconnu" inclut tous lestypes d'engagements définis au titre de la politique de signature, ainsi que tous les types d'engagement définis en externe que la politique peut choisir de reconnaître. Seuls les types d'engagement reconnus sont permis dans ce champ.

L'identifiant d'objet suivant identifie l'attribut commitment-type-indication :

IDENTIFIANT D'OBJET id-aa-ets-commitmentType ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 16}

Les valeurs d'attribut commitment-type-indication ont le type ASN.1 de CommitmentTypeIndication.

```
CommitmentTypeIndication ::= SEQUENCE {
  commitmentTypeId CommitmentTypeIdentifier,
  commitmentTypeQualifier SEQUENCE TAILLE (1..MAX) DE CommitmentTypeQualifier FACULTATIF}

CommitmentTypeIdentifier ::= IDENTIFIANT D'OBJET

CommitmentTypeQualifier ::= SEQUENCE {
   commitmentTypeIdentifier   CommitmentTypeIdentifier,
   qualifier                  TOUT DÉFINI PAR commitmentTypeIdentifier }
```

L'utilisation des qualificatifs de type d'engagement sort du domaine d'application du présent document.

Les types génériques d'engagement suivants sont définis dans le présent document :

IDENTIFIANT D'OBJET id-cti-ets-proofOfOrigin ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 1}

IDENTIFIANT D'OBJET id-cti-ets-proofOfReceipt ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 2}

IDENTIFIANT D'OBJET id-cti-ets-proofOfDelivery ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 3}

IDENTIFIANT D'OBJET id-cti-ets-proofOfSender ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 4}

IDENTIFIANT D'OBJET id-cti-ets-proofOfApproval ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 5}

IDENTIFIANT D'OBJET id-cti-ets-proofOfCreation ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 6}

Ces types génériques d'engagement ont la signification suivante :
Preuve d'origine indique que le signataire reconnaît avoir créé, approuvé, et envoyé le message.
Preuve de réception indique que le signataire reconnaît avoir reçu le contenu du message.
Preuve de livraison indique que le TSP qui fournit cette indication a livré un message dans une mémorisation locale accessible au receveur du message.
Preuve d'envoyeur indique que l'entité qui fournit cette indication a envoyé le message (mais ne l'a pas nécessairement créé).
Preuve d'approbation indique que le signataire a approuvé le contenu du message.
Preuve de céeation indique que le signataire a créé le message (mais pas nécessairement approuvé, ni envoyé).

5.11.2 Attribut signer-location

L'attribut signer-location spécifie un mnémonique pour une adresse associée au signataire à une localisation géographique particulière (par exemple, une ville). Le mnémonique est enregistré dans le pays dans lequel le signataire est situé et est utilisé dans la fourniture du service de télégrammes public (en accord avec la Recommandation UIT-T F.1 [F.1]).

L'attribut signer-location devra être un attribut signé. L'identifiant d'objet suivant identifie l'attribut signer-location :

IDENTIFIANT D'OBJET id-aa-ets-signerLocation ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 17}

Les valeurs de l'attribut Signer-location ont le type ASN.1 de SignerLocation :

```
SignerLocation ::= SEQUENCE {                          -- au moins un des suivants devra être présent:
  countryName    [0]    DirectoryString FACULTATIF,    -- comme utilisé pour désigner un pays dans X.500
  localityName   [X.509]    DirectoryString FACULTATIF, -- comme utilisé pour désigner une localité dans X.500
  postalAdddress [RFC3280]    PostalAddress FACULTATIF }

PostalAddress ::= SEQUENCE TAILLE(1..6) DE DirectoryString
```

5.11.3 Attribut signer-attributes

L'attribut signer-attributes spécifie des attributs supplémentaires du signataire (par exemple, le rôle). Ils peuvent être :
- des attributs revendiqués par le signataire, ou
- des attributs certifiés du signataire.

L'attribut signer-attributes devra être un attribut signé. L'identifiant d'objet suivant identifie l'attribut signer-attribute :

IDENTIFIANT D'OBJET id-aa-ets-signerAttr ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 18}

Les valeurs de signer-attributes ont le type ASN.1 de SignerAttribute :

```
SignerAttribute ::= SEQUENCE DE CHOIX {
   claimedAttributes        [0]   ClaimedAttributes,
   certifiedAttributes   [X.509]   CertifiedAttributes }

ClaimedAttributes ::= SEQUENCE DE Attribute

CertifiedAttributes ::= AttributeCertificate                    -- comme défini dans la RFC 3281 : voir le paragraphe 4.1.
```

Note 1 : un seul signer-attributes peut être utilisé.
Note 2 : Attribute et AttributeCertificate sont comme défini respectivement dans les Recommandations [X.501] et [X.509].

5.11.4 Attribut content-horodatage

L'attribut Horodatage de contenu est un attribut qui est le jeton d'horodatage du contenu des données signées avant qu'il soit signé. L'attribut Horodatage de contenu deva être un attribut signé.

L'identifiant d'objet suivant identifie l'attribut Horodatage de contenu :

IDENTIFIANT D'OBJET id-aa-ets-contentTimestamp ::= { iso(1) member- body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 20}

Les valeurs de l'attribut Horodatage de contenu ont le type ASN.1 de ContentTimestamp :
ContentTimestamp ::= TimeStampToken

La valeur de messageImprint de TimeStampToken (comme décrit dans la [RFC3161]) devra être un hachage de la valeur du champ eContent au sein de encapContentInfo dans les données signées.

Pour plus d'informations sur la définition de TimeStampToken, voir le paragraphe 7.4.

Note : l'horodatage du contenu indique que les informations signées ont été formées avant la date incluse dans l'horodatage de contenu.

5.12 Prise en charge de signatures multiples

5.12.1 Signatures indépendantes

Plusieurs signatures indépendantes (voir l'Annexe B.5) sont prises en charge par des SignerInfo indépendantes provenant de chaque signataire.

Chaque SignerInfo devra inclure tous les attributs requis par le présent document et devra être traité indépendamment par le vérificateur.

Note : des signatures indépendantes peuvent être utilisées pour fournir des signatures indépendantes provenant des différentes parties avec différents attributs signés, ou pour fournir de multiples signatures provenant de la même partie en utilisant des altgorithmes de signature différents, et dans ce cas, les autres attributs, à l'exclusion des valeurs de temps et des informations de politique de signature, vont généralement être les mêmes.

5.12.2 Signatures incorporées

Plusieurs signatures incorporées (voir l'Annexe C.5) sont prises en charge en utilisant l'attribut countersignature non signé (voir le paragraphe 5.9.2). Chaque contre signature est portée dans un attribut countersignature porté comme attribut non signé aux SignerInfo auquelles la contre signature est appliquée.

Note : les contre signatures peuvent être utilisées pour fournir des signatures provenant de différentes parties avec différents attributs signés, ou pour fournir plusieurs signatures provenant de la même partie en utilisant des algorithmes de signature différents, et dans ce cas, les autres attributs, à l'exclusion des valeurs de temps et les informations de politique de signature, vont généralement être les mêmes.

6. Attributs supplémentaires de validation de signature électronique

Cette section spécifie des attributs qui contiennent différents types de données de validation. Ces attributs s'appuient sur la signature électronique spécifiée à la Section 5. Cela inclut :
- l'horodatage de signature appliqué à la valeur de signature électronique ou une marque de temps dans un chemin d'audit. Ceci est défini comme la signature électronique avec temps (CAdES-T) ; et
- les références complètes de données de validation qui comprennent l'horodatage de la valeur de signature, plus les références de toutes les informations de certificats (complete-certificate-references) et de révocation (complete-revocation-references) utilisées pour la validation complète de la signature électronique. Ceci est défini comme la signature électronique avec références complètes de données (CAdES-C).

Note 1 : les formats pour CAdES-T sont illustrés au paraagraphe 4.4, et les attributs sont définis au paragraphe 6.1.1.

Note 2 : les formats pour CAdES-C sont illustrés au paragraphe 4.4. Les attributs requis pour le format de signature CAdES-C sont définis aux paragraphes 6.2.1 à 6.2.2 ; les attributs facultatifs le sont aux paragraphes 6.2.3 et 6.2.4.

De plus, les formes étendues facultatives de données de validation sont aussi définies ; voir à l'Annexe B ure vue d'ensemble des formes étendues de données de validation :
- CAdES-X avec horodatage : il y a deux types d'horodatage utilisés dans les données de validation étendues définies par le présent document :
  - Type 1 (CAdES-X Type 1) : comporte un horodatage sur l'ES avec données de validation complètee (CAdES-C) ; et
  - Type 2 (CAdES-X Type2) : comporte un horodatage sur les références de chemin de certification et les références d'informations de révocation utilisées pour prendre en charge la CAdES-C.

Note 3 : les formats pour la CAdES-X de type 1 et la CAdES-X de type 2 sont illustrés respectivement aux paragraphes B.1.2 et B.1.3.

- CAdES-X Long : comporte les références complètes de données de validation (CAdES-C), plus les valeurs réelles de toutes les informations de certificats et de révocation utilisées dans la CAdES-C.

Note 4 : les formats pour CAdES-X Long sont illustrés à l'Annexe B.1.1.

- CAdES-X Long Type 1 ou CAdES-X Long Type 2 : comporte un horodatage X (de type 1 ou de type 2) plus les valeurs réelles de toutes les informations de certificats et de révocation utilisées dans la CAdES-C conformément à CAdES-X Long.

Cette Section spécifie aussi les structures de données utilisées dans le format d'archive de données de validation (CAdES-A) des formes étendues :
Les formes d'archive de signature électronique (CAdES-A) comportent :
- les références complètes de données de validation (CAdES-C),
- les valeurs de certificat et de révocation (comme dans une CAdES-X Long),
- tout horodatage existant de signature électronique étendue (CAdES-X Type 1 ou CAdES-X Type 2), si il en est présent, et
- les données signées d'utilisateur et un horodatage d'archive supplémentaire appliqué sur toutes ces données.

Un horodatage d'archive peut être appliqué de façon répétée après de longues périodes pour maintenir la validité quand la signature électronique et les algorithmes d'horodatage s'affaiblissent.

Les données supplémentaires requises pour créer les formes de signature électronique identifiées ci-dessus sont portées comme des attributs non signés associés à une signature individuelle et sont placées dans le champ unsignedAttrs de SignerInfo. Donc, tous les attributs définis dans la Section 6 sont des attributs non signés.

Note 5 : lorsque plusieurs signatures doivent être prises en charge, comme décrit au paragraphe 5.12, chaque signature a un SignerInfo séparé. Donc, chaque signature exige ses propres valeurs d'attribut non signé pour créer les CAdES-T, CAdES-C, etc.

Note 6 : les attributs facultatifs des données de validation étendues sont définis aux paragraphes 6.3 et 6.4.

6.1 Attribut Horodatage de signature (CAdES-T)

Une signature électronique avec horodatage est une signature électronique pour laquelle une partie, mais pas toutes, des données supplémentaires requises pour la validation est disponible (c'est-à-dire, certaines informations de certificats et de révocation sont disponibles, mais pas toutes).

La structure minimum d'horodatage de données de validation est :
- l'attribut horodatage de signature, comme défini au paragraphe 6.1.1, sur la valeur de l'ES.

6.1.1 Définition de l'attribut Horodatage de signature

L'attribut Horodatage de signature est un TimeStampToken *(jeton d'horodatage)* calculé sur la valeur de la signature pour un signataire spécifique ; c'est un attribut non signé. Plusieurs instances de cet attribut peuvent se produire avec une signature électronique, provenant de différents TSA.

L'identifiant d'objet suivant identifie l'attribut Horodatage de signature :

IDENTIFIANT D'OBJET id-aa-signatureTimeStampToken ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 14}

La valeur de l'attribut Horodatage de signature a le type ASN.1 de SignatureTimeStampToken :

```
SignatureTimeStampToken ::= TimeStampToken
```

La valeur du champ messageImprint au sein de TimeStampToken devra être un hachage de la valeur du champ Signature au sein de SignerInfo pour les signedData horodatées.

Pour plus d'informations et la définition de TimeStampToken, voir le paragraphe 7.4.

Note 1 : dans le cas de signatures multiples, il est possible d'avoir :
- un TimeStampToken calculé pour chacun de tous les signataires ; ou
- un TimeStampToken calculé sur la signature d'un seul signataire ; et
- pas de TimeStampToken sur une autre signature de signataire.

Note 2 : dans le cas de signatures multiples, plusieurs TST, produits par différentes TSA, peuvent être présents au sein des mêmes signerInfo (voir la [RFC3852]).

6.2 Références de données de validation complètes (CAdES-C)

Une signature électronique avec références complètes de données de validation (CAdES-C) est une signature électronique pour laquelle toutes les données supplémentaires requises pour la validation (c'est-à-dire, toutes les informations de certificats et de révocation) sont disponibles. Cette forme est construite sur la forme CAdES-T définie ci-dessus.

Au minimum, les données de validation complètes doivent inclure :
- une indication de temps, qui devra être soit un attribut Horodatage de signature, comme défini au paragraphe 6.1.1, soit une marque de temps effectuée par une autorité de marquage de temps,
- des références complètes de certificat, comme défini au paragraphe 6.2.1;
- des références complètes de révocation, comme défini au paragraphe 6.2.2.

6.2.1 Définition de l'attribut complete-certificate-references

L'attribut Références complètes de certificat est un attribut non signé. Il fait référence à l'ensemble complet des certificats de CA qui ont été utilisés pour valider une ES avec données de validation complètes jusqu'au certificat (non inclus) du signataire. Une seule instance de cet attribut devra apparaître dans une signature électronique.

Note 1 : le certificat du signataire est référencé dans l'attribut Certificat de signature (paragraphe 5.7.3).

IDENTIFIANT D'OBJET id-aa-ets-certificateRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 21}

La valeur de l'attribut Références complètes de certificat a la syntaxe ASN.1 de CompleteCertificateRefs :

```
CompleteCertificateRefs ::=  SEQUENCE DE OtherCertID
```

OtherCertID est défini au paragraphe 5.7.3.3.

IssuerSerial devra être présent dans OtherCertID. Le hachage de certificat devra correspondre au hachage du certificat référencé.

Note 2 : des copies des valeurs de certificat peuvent être détenues en utilisant l'attribut Valeurs de certificat défini au paragraphe 6.3.3.

Cet attribut peut inclure des références à la chaîne de certification pour toute TSU qui fournit des jetons d'horodatage. Dans ce cas, l'attribut non signé devra être ajouté aux données signées du jeton d'horodatage pertinent comme attribut non signé dans le champ signerInfos.

6.2.2 Définition de l'attribut complete-revocation-references

L'attribut Références complètes de révocation est un attribut non signé. Une seule instance de cet attribut devra se produire dans une signature électronique. Il fait référence à l'ensemble complet de CRL, ACRL, ou réponses OCSP qui a été utilisé dans la validation du signataire, et aux certificats de CA utilisés dans une ES avec données de validation complètes.

Cet attribut indique que le vérificateur a pris toutes les mesures raisonnables pour rassembler les informations de révocation disponibles. Les références mémorisées dans cet attribut peuvent être utilisées pour restituer les informations référencées, si elles ne sont pas mémorisées dans la structure de CMS, mais ailleurs.

L'identifiant d'objet suivant identifie l'attribut Références complètes de révocation :

IDENTIFIANT D'OBJET id-aa-ets-revocationRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 22}

La valeur de l'attribut Références complètes de révocation a la syntaxe ASN.1 de CompleteRevocationRefs :

```
CompleteRevocationRefs ::=  SEQUENCE DE CrlOcspRef

CrlOcspRef ::= SEQUENCE {
    crlids       [0]  CRLListID    FACULTATIF,
    ocspids      [X.509]  OcspListID   FACULTATIF,
    otherRev     [RFC3280]  OtherRevRefs FACULTATIF
}
```

CompleteRevocationRefs devra contenir une CrlOcspRef pour le certificat de signature, suivi d'une pour chaque OtherCertID dans l'attribut CompleteCertificateRefs. Le second champ CrlOcspRef et les suivants devront être dans le même ordre que le OtherCertID auquel ils se rapportent. Au moins un de CRLListID ou OcspListID ou OtherRevRefs devrait être présent pour toutes les CA, sauf celle "de confiance" pour le chemin de certification.

```
CRLListID ::= SEQUENCE {
   crls             SEQUENCE DE CrlValidatedID }

CrlValidatedID ::=  SEQUENCE {
    crlHash              OtherHash,
    crlIdentifier        CrlIdentifier FACULTATIF }

CrlIdentifier ::= SEQUENCE {
   crlissuer            Name,
   crlIssuedTime        UTCTime,
   crlNumber            ENTIER FACULTATIF }

OcspListID ::=  SEQUENCE {
   ocspResponses        SEQUENCE DE OcspResponsesID }

OcspResponsesID ::=  SEQUENCE {
```

```
  ocspIdentifier          OcspIdentifier,
  ocspRepHash             OtherHash    FACULTATIF
}

OcspIdentifier ::= SEQUENCE {
  ocspResponderID    ResponderID,                -- comme dans les données de réponse OCSP
  producedAt         GeneralizedTime             -- comme dans les données de réponse OCSP
}
```

Quand on crée un crlValidatedID, le crlHash est calculé sur la CRL entière codée en DER incluant la signature. Le crlIdentifier va normalement être présent sauf si la CRL peut être déduite d'autres informations.

Le crlIdentifier est pour identifier la CRL en utilisant le nom du producteur et l'heure de production de la CRL, qui devra correspondre à l'heure de thisUpdate contenue dans la CRL produite, et si il est présent, au numéro de CRL. L'attribut crlListID est un attribut non signé. Dans le cas où la CRL identifiée est une CRL delta, les références à l'ensemble des CRL pour fournir une liste de révocation complète devra être incluse.

Le OcspIdentifier est pour identifier la réponse OCSP en utilisant le nom du producteur et l'heure de production de la réponse OCSP, qui devra correspondre à l'instant de production tel que contenu dans la réponse OCSP produite. Comme il peut être nécessaire de faire la différence entre deux réponses OCSP reçues dans la même seconde, le hachage de la réponse contenue dans le OcspResponsesID peut être nécessaire pour résoudre l'ambiguïté.

Note 1 : des copies des valeurs de la CRL et des réponses OCSP peuvent être détenues en utilisant l'attribut Valeurs de révocation défini au paragraphe 6.3.4.

Note 2 : il est recommandé que cet attribut soit utilisé de préférence à OtherRevocationInfoFormat spécifié dans la RFC 3852 pour conserver la rétro compatibilité avec la version précédente de cette spécification.

La syntaxe et la sémantique des autres références de révocation sortent du domaine d'application du présent document. La définition de la syntaxe des autres formes d'informations de révocation est comme identifiée par OtherRevRefType.

Cet attribut peut inclure les références à l'ensemble complet de CRL, ACRL, ou réponses OCSP qui a été utilisé pour vérifier la chaîne de certification pour toutes les TSU qui fournissent des jetons d'horodatage. Dans ce cas, l'attribut non signé devra être ajouté aux données signées du jeton d'horodatage pertinent comme unsignedAttrs dans le champ signerInfos.

6.2.3 Définition de l'attribut attribute-certificate-references

Cet attribut est seulement utilisé quand un certificat d'attribut d'utilisateur est présent dans la signature électronique.

L'attribut attribute-certificate-references est un attribut non signé. Il fait référence à l'ensemble complet de certificats d'AA qui a été utilisé pour valider le certificat d'attribut. Une seule instance de cet attribut devra apparaître dans une signature électronique.

IDENTIFIANT D'OBJET id-aa-ets-attrCertificateRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 44}

La valeur de l'attribut attribute-certificate-references a la syntaxe ASN.1 de AttributeCertificateRefs :

```
AttributeCertificateRefs ::=  SEQUENCE DE OtherCertID
```

OtherCertID est défini au paragraphe 5.7.3.3.

Note : des copies des valeurs du certificat peuvent être détenues en utilisant l'attribut Valeurs de certificat défini au paragraphe 6.3.3.

6.2.4 Définition de l'attribut attribute-revocation-references

Cet attribut est seulement utilisé quand un certificat d'attribut d'utilisateur est présent dans la signature électronique et quand ce certificat d'attribut peut être révoqué.

L'attribut attribute-revocation-references est un attribut non signé. Une seule instance de cet attribut devra apparaître dans une signature électronique. Il fait référence à l'ensemble complet d'ACRL ou de réponses OCSP qui a été utilisé dans la validation du certificat d'attribut. Cet attribut peut être utilisé pour illustrer que le vérificateur a pris toutes les mesures raisonnables pour collecter les informations de révocation disponibles.

L'identifiant d'objet suivant identifie l'attribut attribute-revocation-references :

IDENTIFIANT D'OBJET id-aa-ets-attrRevocationRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 45}

La valeur d''attribut attribute-revocation-references a la syntaxe ASN.1 de AttributeRevocationRefs :

```
AttributeRevocationRefs ::=  SEQUENCE DE CrlOcspRef
```

6.3 Données de validation étendues (CAdES-X)

Ce paragraphe spécifie un certain nombre d'attributs facultatifs qui sont utilisés par les formes étendues de signatures électroniques (voir à l'Annexe B une vue d'ensemble de ces formes de données de validation).

6.3.1 Données de validation avec horodatage (CAdES-X type 1 ou type 2)

Les données de validation étendues peuvent inclure un des attributs supplémentaires suivants, formant une CAdES-X à données de validation avec horodatage (CAdES-X Type 1 ou CAdES-X Type 2) pour fournir une protection supplémentaire contre la compromission de CA et assurer l'intégrité des données de validation utilisées :
- Horodatage CAdES-C, comme défini au paragraphe 6.3.5 (CAdES-X Type 1) ; ou
- Références de certificats et CRL horodatés, comme défini au paragraphe 6.3.6 (CAdES-X Type 2).

6.3.2 Données de validation longues (CAdES-X long, CAdES-X long type 1 ou 2)

Les données de validation étendues peuvent aussi inclure les informations supplémentaires suivantes, formant une CAdES-X Long, à utiliser si les processus ultérieurs de validation ne peuvent pas avoir accès à ces informations :
- valeurs de certificat, comme défini au paragraphe 6.3.3 ; et
- valeurs de révocation, comme défini au paragraphe 6.3.4.

Les données de validation étendues peuvent, en plus des valeurs de certificat et des valeurs de révocation, comme défini aux paragraphes 6.3.3 et 6.3.4, inclure un des attributs supplémentaires suivants, formant une CAdES-X longue de type 1 ou une CAdES-X longue de type 2.
- Horodatage de CAdES-C, comme défini au paragraphe 6.3.3 (CAdES-X longue de type 1) ; ou
- Références de certificats et CRL horodatées, comme défini au paragraphe 6.3.4 (CAdES-X longue de type 2).

La CAdES-X longue de type 1 ou la CAdES-X longue de type 2 fournit une protection supplémentaire contre la compromission ultérieure de la CA et fournit la protection de l'intégrité des données de validation utilisées.

Note 1 : la CAdES-X-Long fournit une preuve à long terme de la validité de la signature aussi longtemps que les clés de CA, les clés de producteur de CRL, et les clés de répondant OCSP ne sont pas compromises et sont résistantes aux attaques cryptographiques.

Note 2 : tant que les données de l'horodatage restent valides, la CAdES-X longue de type 1 et la CAdES-X longue de type 2 fournissent les importantes propriétés suivantes pour les signatures à longue durée : que ayant été trouvées valides une fois, elles devront continuer de l'être des mois ou des années plus tard, longtemps après l'expiration de la période de validité des certificats, ou après la compromission de la clé d'utilisateur.

6.3.3 Définition de l'attribut certificate-values

Cet attribut peut être utilisé pour contenir des informations de certificat requises pour les formes suivantes de signature électronique étendue : CAdES-X longue, ES-X longue de type 1, et CAdES-X longue de type 2 ; voir à l'Annexe B.1.1 une illustration de ces formes de signature électronique.

L'attribut Valeurs de certificat est un attribut non signé. Une seule instance de cet attribut devra apparaître dans une signature électronique. Il contient les valeurs des certificats référencés dans l'attribut Références complètes de certificat.

Note : Si un certificat d'attribut est utilisé, il n'est pas fourni dans cette structure mais devra être fourni par le signataire comme attribut du signataire (voir le paragraphe 5.11.3).

L'identifiant d'objet suivant identifie l'attribut Valeurs de certificat :

IDENTIFIANT D'OBJET id-aa-ets-certValues ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 23}

La valeur de l'attribut Valeurs de certificat a la syntaxe ASN.1 de CertificateValues :

```
CertificateValues ::=  SEQUENCE DE Certificate
```

Certificate est défini au paragraphe 7.1, (qui est comme défini dans la Recommandation UIT-T X.509 [X.509]).

Cet attribut peut inclure les informations de certification pour toute TSU qui a fourni des jetons d'horodatage, si ces certificats ne sont pas déjà inclus dans les TST au titre des signatures de TSU. Dans ce cas, l'attribut non signé devra être ajouté aux données signées du jeton d'horodatage pertinent.

6.3.4 Définition de l'attribut Vakeurs de révocation

Cet attribut est utilisé pour contenir les informations de révocation requises pour les formes suivantes de signature électronique étendue : CAdES-X Long, ES X-Long Type 1, et CAdES-X Long Type 2 ; voir à l'Annexe B.1.1 une illustration de cette forme de signature électronique.

L'attribut Valeurs de révocation est un attribut non signé. Une seule instance de cet attribut devra apparaître dans une signature électronique. Il contient les valeurs des CRL et OCSP référencées dans l'attribut Références complètes de révocation.

Note : Il est recommandé que cet attribut soit utilisé de préférence à OtherRevocationInfoFormat spécifié dans la RFC 3852 pour conserver la rétro compatibilité avec la version antérieure de cette spécification.

L'identifiant d'objet suivant identifie l'attribut Valeurs de révocation :

IDENTIFIANT D'OBJET id-aa-ets-revocationValues ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 24}

La valeur de l'attribut Valeurs de révocation a la syntaxe ASN.1 de RevocationValues :

```
RevocationValues ::=  SEQUENCE {
  crlVals          [0] SEQUENCE DE CertificateList FACULTATIF,
  ocspVals         [X.509] SEQUENCE DE BasicOCSPResponse FACULTATIF,
  otherRevVals   [RFC3280] OtherRevVals FACULTATIF }

OtherRevVals ::= SEQUENCE {
  OtherRevValType   OtherRevValType,
  OtherRevVals         TOUT DÉFINI PAR OtherRevValType }

OtherRevValType ::= IDENTIFIANT D'OBJET
```

La syntaxe et la sémantique des autres valeurs de révocation (OtherRevVals) sortent du domaine d'application du présent document.

La définition de la syntaxe des autres formes d'informations de révocation est comme identifié par OtherRevRefType.

CertificateList est défini au paragraphe 7.2 (qui est comme défini dans la Recommandation UIT-T X.509 [X.509]).

BasicOCSPResponse est défini au paragraphe 7.3 (qui est comme défini dans la [RFC2560]).

Cet attribut peut inclure les valeurs de données de révocation incluant les CRL et réponses OCSP pour toute les TSU qui ont fourni des jetons d'horodatage, si ces certificats ne sont pas déjà inclus dans les TST au titre des signatures des TSU. Dans ce cas, l'attribut non signé devra être ajouté aux données signées du jeton d'horodatage pertinent.

6.3.5 Définition de l'attribut Horodatage de CAdES-C

Cet attribut est utilisé pour protéger de la compromission de la clé de CA.

Cet attribut est utilisé pour l'horodatage de la signature électronique complète (CAdES-C). Il est utilisé dans les formes suivantes de signature électronique étendue : CAdES-X Type 1 et CAdES-X Long Type 1 ; voir à l'Annexe B.1.2 une illustration de cette forme de signature électronique.

L'attribut Horodatage de CAdES-C est un attribut non signé. C'est un jeton d'horodatage du hachage de la signature électronique et des données de validation complètes (CAdES-C). C'est un attribut TimeStampToken d'utilisation particulière qui horodate la CAdES-C. Plusieurs instances de cet attribut peuvent apparaître dans une signature électronique provenant de TSA différentes.

Note 1 : il est recommandé que les attributs qui sont horodatés soient codés en DER. Si le DER n'est pas employé, le codage binaire des structures ASN.1 horodatées devrait être préservé pour assurer que le re-calcul des hachages de données est cohérent.

Note 2 : chaque attribut est inclus dans le hachage avec le type d'attribut et les valeurs d'attribut (incluant le type et la longueur) mais dans le type et la longueur des la SEQUENCE externe.

L'identifiant d'objet suivant identifie l'attribut Horodatage de CAdES-C:

IDENTIFIANT D'OBJET id-aa-ets-escTimeStamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 25}

La valeur de l'attribut CAdES-C-timestamp a la syntaxe ASN.1 de ESCTimeStampToken :

```
ESCTimeStampToken ::= TimeStampToken
```

La valeur du champ messageImprint au sein de TimeStampToken devra être un hachage des valeurs enchaînées (sans le codage de type ou de longueur pour cette valeur) des objets de données suivants :
- OCTETSTRING du champ SignatureValue au sein de SignerInfo ;
- Horodatage de signature, ou marque de temps effectuée par une autorité de marquage de temps ;
- attribut Références complètes de certificat ; et
- attribut Références complètes de révocation.

Voir plus d'informations et la définition de TimeStampToken, au paragraphe 7.4.

6.3.6 Définition de l'attribut time-stamped-certs-crls-references

Cet attribut est utilisé pour protéger contre la compromission de la clé de CA. Cet attribut est utilisé pour l'horodatage des références de certificat et de révocation. Il est utilisé dans les formes suivantes de signature électronique étendue : CAdES-X de type 2 et CAdES-X longue de type 2 ; voir à l'Annexe B.1.3 une illustration de cette forme de signature électronique.

Un attribut time-stamped-certs-crls-references est un attribut non signé. C'est un jeton d'horodatage produit pour une liste de certificats et réponses OCSP et/ou CRL référencés pour protéger contre certaines CA compromises. Sa syntaxe est la suivante :

Note 1: il est recommandé que les attributs horodatés soient codés en DER. Si DER n'est pas employé, alors le codage binaire des structures ASN.1 à horodater devrait être préservé pour assurer la cohérence du recalcul du hachage des données.

Note 2 : chaque attribut est inclus dans le hachage avec le attrType et attrValues (incluant type et longueur) mais sans le type et la longueur de la SEQUENCE externe.

L'identifiant d'objet suivant identifie l'attribut time-stamped-certs-crls-references :

IDENTIFIANT D'OBJET id-aa-ets-certCRLTimestamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 26}

La valeur de l'attribut a la syntaxe ASN.1 de TimestampedCertsCRLs :

TimestampedCertsCRLs ::= TimeStampToken

La valeur du champ messageImprint au sein du TimeStampToken devra être un hachage des valeurs enchaînées (sans le codage du type ou de la longueur pour cette valeur) des objets de données suivants, comme présents dans l'ES avec données de validation complètes (CAdES-C) :
- attribut Références complètes de certificat; et
- attribut Références complètes de révocation.

6.4 Données de validation d'archive

Lorsque il est exigé d'une signature électronique qu'elle dure très longtemps, et que le jeton d'horodatage sur une signature électronique court le danger d'être invalidé à cause de la faiblesse de l'algorithme ou des limites de la période de validité du certificat de TSA, il peut être nécessaire d'horodater plusieurs fois la signature électronique. Quand c'est exigé, un attribut Archive d'horodatage peut être nécessaire pour la forme d'archive de la signature électronique (CAdES-A). Cet attribut d'archive d'horodatage peut être appliqué de façon répétée sur une certaine période.

6.4.1 Définition de l'attribut horodatage d'archive

L'attribut horodatage d'archive est un jeton d'horodatage de beaucoup d'éléments des données signées dans la signature électronique. Si les attributs certificate-values et revocation-values ne sont pas présents dans la CAdES-BES ou CAdES-EPES, ils ne devront alors pas être ajoutés à la signature électronique avant le calcul du jeton d'horodatage d'archive.

L'attribut horodatage d'archive est un attribut non signé. Plusieurs instances de cet attribut peuvent apparaître dans une signature électronique aussi bien dans le temps que provenant de différentes TSU.

L'identifiant d'objet suivant identifie l'attribut incorporé horodatage d'archive :

IDENTIFIANT D'OBJET id-aa-ets-archiveTimestampV2 ::= iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 48}

Les valeurs de l'attribut horodatage d'archive ont la syntaxe ASN.1 de ArchiveTimeStampToken :

ArchiveTimeStampToken ::= TimeStampToken

La valeur du champ messageImprint au sein de TimeStampToken devra être un hachage de l'enchaînement de :
- l'élément encapContentInfo de la séquences des données signées ;
- tout contenu externe protégé par la signature, si l'élément eContent de encapContentInfo est omis ;
- les éléments Certificates et crls de la séquence des données signées, quand elle est présente, et ;
- tous les éléments de données dans la séquence SignerInfo incluant tous les attributs signés et non signés.

Note 1 : un attribut archiveTimestamp de remplacement, identifié par un identifiant d'objet de { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 27, est défini dans les versions antérieures du [TS101733] et dans la RFC 3126.
L'attribut archiveTimestamp, défini dans les versions de TS 101 733 antérieures à 1.5.1 et dans la RFC 3126, n'est pas compatible avec l'attribut défini dans le document actuel. L'attribut archiveTimestamp, défini dans les versions 1.5.1 à 1.6.3 du TS 101 733, est compatible avec le document actuel si le contenu est interne à encapContentInfo. Sauf si la version de TS 101 733 employée par le signataire est connue de tous les receveurs, l'utilisation de l'attribut archiveTimestamp défini dans les versions antérieures du TS 101 733 est déconseillée.

Note 2 : les contre signatures détenues comme des attributs countersignature n'exigent pas d'horodatage d'archive indépendant, car elles sont protégées par l'horodatage d'archive contre la structure de données signées contenante.

Note 3 : sauf si DER est utilisé partout, il est recommandé que le codage binaire des structures ASN.1 horodatées soit préservé quand elles sont archivées pour assurer que le recalcul du hachage des données est cohérent.

Note 4 : le hachage est calculé sur les éléments de données enchaînées comme ils sont reçus/mémorisés, incluant le codage de type et de longueur.

Note 5 : bien qu'il soit recommandé que les attributs non signés soient codés en DER, cela ne peut pas être garanti de façon générale sauf accord préalable. Plus d'informations et la définition de TimeStampToken figurent au paragraphe 7.4. L'horadatage devrait être créé en utilisant des algorithmes plus forts (ou de plus grandes longueurs de clé) que dans les signatures électroniques originales et dans les algorithmes d'horodatage faibles.

Note 6 : cette forme d'ES fournit aussi une protection contre la compromission de clé de TSP.

ArchiveTimeStamp va être ajouté comme attribut non signé dans la séquence des informations de signataire. Pour la validation d'un ArchiveTimeStamp, les éléments de données des SignerInfo doivent être enchaînées, en excluant tous les attributs ArchivTimeStampToken ultérieurs.

Les informations de certificats et de révocation nécessaires pour valider le ArchiveTimeStamp devront être fournies par une des méthodes suivantes :
- la TSU fournit les informations dans les données signées du jeton d'horodatage ;
- l'ajout de l'attribut Références complètes de certificat et de l'attribut Références complètes de révocation du TSP comme attribut non signé au sein de TimeStampToken, quand les informations requises sont mémorisées ailleurs ; ou
- l'ajout de l'attribut Valeurs de certificat et de l'attribut Valeurs de révocation du TSP comme attribut non signé au sein du TimeStampToken, quand les informations requises sont mémorisées ailleurs.

7. Autres structures de données standard

7.1 Format de certificat de clé publique

La syntaxe de base de certificat X.509 v3 est définie dans la Recommandation UIT-T X.509 [X.509]. Un profil du certificat X.509 v3 est défini dans la [RFC3280].

7.2 Format de liste de révocation de certificats

La syntaxe de CRL X.509 v2 est définie dans la Recommandation UIT-T X.509 [X.509]. Un profil de CRL X.509 v2 est défini dans la [RFC3280].

7.3 Format de réponse OCSP

Le format d'un jeton OCSP est défini dans la [RFC2560].

7.4 Format de jeton d'horodatage

Le format d'un type de jeton d'horodatage est défini dans la [RFC3161] et un profil dans [TS101861].

7.5 Formats de nom et d'attribut

La syntaxe de la désignation et des autres attributs est définie dans la Recommandation UIT-T X.509 [X.509].

Note : le nom utilisé par le signataire, détenu comme le sujet dans le certificat du signataire, est alloué et vérifié à l'enregistrement auprès de l'autorité de certification, directement ou indirectement à travers une autorité d'enregistrement, avant d'être produit avec un certificat.

Le présent document ne fait aucune restriction sur la forme du nom. Le nom du sujet peut être un nom distinctif, comme défini dans la Recommandation UIT-T X.500 [X.500], contenu dans le champ Sujet du certificat, ou toutes autres formes de nom contenu dans le champ d'extension de certificat subjectAltName, comme défini dans la Recommandation UIT-T X.509 [X.509]. Dans le cas où le sujet n'a pas de nom distinctif, le nom du sujet peut être une séquence vide et l'extension subjectAltName devra être critique.

Toutes les autorités de certification, autorités d'attribut, et autorités d'horodatage devront utiliser des noms distinctifs dans le champ Sujet de leurs certificats.

Le nom distinctif devra inclure des identifiants de l'organisation qui fournit le service et la juridiction légale (par exemple, le pays) sous laquelle il opère.

Quand un signataire signe comme individu, mais souhaite aussi s'identifier lui-même comme agissant au nom d'une organisation, il peut être nécessaire de fournir deux formes indépendantes d'identification. La première identité, qui est directement associée à la clé de signature, l'identifie comme individu. La seconde, qui est gérée indépendamment, identifie cette personne comme agissant au titre de l'organisation, éventuellement avec un certain rôle. Dans ce cas, une des deux identités est portée dans le champ subject/subjectAltName du certificat du signataire comme décrit ci-dessus.

Le présent document ne spécifie pas le format de l'attribut du signataire qui peut être inclus dans les certificats de clé publique.

Note : l'attribut du signataire peut être pris en charge en utilisant un rôle revendiqué dans le champ de CMS Attributs signés, ou en plaçant un certificat d'attribut contenant un rôle certifié dans le champ de CMS Attributs signés ; voir au paragraphe 7.6.

7.6 AttributeCertificate

La syntaxe du type AttributeCertificate est définie dans la [RFC3281].

8. Exigences de conformité

Pour les mises en œuvre qui prennent en charge la génération de signature, le présent document définit les exigences de conformité pour la génération des deux formes de base de signature électronique, une des deux formes doit être mise en œuvre.

Pour les mises en œuvre qui prennent en charge la vérification de signature, le présent document définit les exigences de conformité pour la vérification des deux formes de base de signature électronique, une des deux formes doit être mise en œuvre.

Le présent document définit seulement les exigences de conformité jusqu'à une ES avec données de validation complètes (CAdES-C). Cela signifie qu'aucune des formes étendues et d'archive de la signature électronique (CAdES-X, CAdES-A) n'a besoin d'être mise en œuvre pour obtenir la conformité au présent document.

À la vérification, l'inclusion des attributs facultatifs signés et non signés doit être prise en charge seulement dans la mesure où la signature est vérifiable. La sémantique des attributs facultatifs peut n'être pas prise en charge, sauf si une politique de signature spécifie le contraire.

8.1 Signature électronique CAdES-Basic (CAdES-BES)

Un système qui prend en charge les signataires CAdES-BES, conformément au présent document, devra, au minimum, prendre en charge la génération d'une signature électronique consistant en les composants suivants :

- la syntaxe générale de la CMS et du type de contenu, comme défini dans la [RFC3852] (paragraphes 5.1 et 5.2) ;
- les données signées de CMS, comme défini dans la [RFC3852], avec la version réglée à 3 et au moins un SignerInfo présent (voir les paragraphes 5.3 à 5.6) ;
- les attributs de CMS suivants, comme défini dans la [RFC3852] :
  - content-type ; qui doit toujours être présent (paragraphe 5.7.1) ; et
  - message-digest ; qui doit toujours être présent (paragraphe 5.7.2).
- un des attributs suivants, comme défini dans le présent document :
  - signing-certificate : comme défini au paragraphe 5.7.3.1 ; ou
  - signing-certificate-v2 : comme défini au paragraphe 5.7.3.2.

Note : la RFC 3126 utilisait un autre attribut signing-certificate (voir le paragraphe 5.7.3.3). Cette utilisation est maintenant déconseillée, car la structure de l'attribut signing-certificate-v2 est plus simple que celle de l'autre attribut signing-certificate.

8.2 Signature électronique CAdES-Explicit Policy-based signature électronique

Un système qui prend en charge des signataires sur la base de la politique, en accord avec le présent document, devra, au minimum, prendre en charge la génération d'une signature électronique comportant les composants précédents définis pour le signataire de base, plus :
- les attributs suivants, comme défini au paragraphe 5.9 :
  - signature-policy-identifier ; il doit toujours être présent (voir le paragraphe 5.8.1).

8.3 Vérification utilisant l'horodatage

Un système qui prend en charge les vérificateurs, conformément au présent document, avec les facilités d'horodatage devra, au minimum, prendre en charge :
- la vérification des composants obligatoires d'une signature électronique, comme défini au paragraphe 8.1 ;
- l'attribut Horodatage de signature, comme défini au paragraphe 6.1.1 ;
- l'attribut Références complètes de certificat, comme défini au paragraphe 6.2.1 ;
- l'attribut Références complètes de révocation, comme défini au paragraphe 6.2.2 ;
- les certificats de clé publique, comme défini dans la Recommandation UIT-T X.509 [X.509] (voir le paragraphe 8.1) ;
- et l'un de :
  - liste de révocation de certificat, comme défini dans la Recommandation UIT-T X.509 [X.509] (paragraphe 8.2) ; ou
  - protocole d'état de certificat en ligne, comme défini dans la [RFC2560] (voir le paragraphe 8.3).

8.4 Vérification utilisant des enregistrements sûrs

Un système qui prend en charge les vérificateurs, conformément au présent document, devra, au minimum, prendre en charge :
- la vérification des composants obligatoires d'une signature électronique, comme défini au paragraphe 8.1 ;
- l'attribut Références complètes de certificat, comme défini au paragraphe 6.2.1 ;
- l'attribut Références complètes de révocation, comme défini au paragraphe 6.2.2 ;
- un enregistrement de la signature électronique et du moment où la signature a été validée pour la première fois, en utilisant les informations référencées de certificats et de révocation, doit être conservé, de telle façon que les enregistrements ne puissent pas être modifiés de façon indétectable ;
- les certificats de clé publique, comme défini dans la Recommandation UIT-T X.509 [X.509] (voir le paragraphe 8.1) ;
- et l'un de :
  - liste de révocation de certificat, comme défini dans la Recommandation UIT-T X.509 [X.509] (paragraphe 8.2) ; ou
  - protocole d'état de certificat en ligne, comme défini dans la [RFC2560] (voir le paragraphe 8.3).

9. Références

9.1 Références normatives

[F.1] Recommandation UIT-T F.1, "Dispositions opérationnelles pour le service international de télégrammes publics".

[RFC2045] N. Freed et N. Borenstein, "Extensions de messagerie Internet multi-objets (MIME) Partie 1 : Format des corps de message Internet", novembre 1996. *(D. S., MàJ par* 2184, 2231, 5335.)

[RFC2560] M. Myers, R. Ankney, A. Malpani, S. Galperin et C. Adams, "Protocole d'état de certificat en ligne d'infrastructure de clé publique X.509 pour l'Internet - OCSP", juin 1999. *(P.S.) (Remplacée par* RFC6960)

[RFC2634] P. Hoffman, éd., "Services de sécurité améliorés pour S/MIME", juin 1999. (*MàJ par* RFC5035) *(P.S.)*

[RFC3161] C. Adams, P. Cain, D. Pinkas et R. Zuccherato, "Protocole d'horodatage (TSP) d'infrastructure de clé publique X.509 pour l'Internet", août 2001.

[RFC3280] R. Housley, W. Polk, W. Ford et D. Solo, "Profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet", avril 2002. *(Obsolète, voir* RFC5280)

[RFC3281] S. Farrell et R. Housley, "Profil de certificat d'attribut Internet pour l'autorisation", avril 2002. *(Obsolète, voir* RFC5755)

[RFC3370] R. Housley, "Algorithmes de syntaxe de message cryptographique (CMS)", août 2002. *(P.S. ; MàJ par* RFC8702*)*

[RFC3852] R. Housley, "Syntaxe de message cryptographique (CMS)", juillet 2004. *(Obsolète, voir la* RFC5652)

[RFC5035] J. Schaad, "Mise à jour des services de sécurité améliorée (ESS) : ajout du choix d'algorithme CertID", août 2007. *(P.S.)*

[X.208] Recommandation UIT-T X.208, "Spécification de la notation numéro un de syntaxe abstraite (ASN.1)", Genève, novembre 1988.

[X.500] Recommandation UIT-T X.500, "L'annuaire : aperçu général des concepts, modèles et services", 1993.

[X.501] Recommandation UIT-T X.501, "L'annuaire : modèles", 1993.

[X.509] Recommandation UIT-T X.509, "L'annuaire : cadre d'authentification", Genève, 1988.

[X.680] Recommandation UIT-T X.680 | ISO/CEI 8824-1:2002 "Technologies de l'information - Notation de syntaxe abstraite n°1 (ASN.1) : Spécification de la notation de base". (07/2002)

[X.690] Recommandation UIT-T X.690 | ISO/CEI 8825-1:2002, "Technologies de l'information - Règles de codage de l'ASN.1 : Spécification des règles de codage de base (BER), règles de codage canoniques (CER) et règles de codage distinctives (DER)", (07/2002).

9.2 Références pour information

[CWA14171] CEN Workshop Agreement CWA 14171, "Lignes directrices générales pour la vérification de signature électronique".

[EUDirective] Directive 1999/93/EC du Parlement européen et du Conseil du 13 décembre 1999 sur un cadre communautaire pour les signatures électroniques.

[ISO7498-2] Norme ISO/CEI 7498-2 (1989), "Systèmes de traitement de l'information - Interconnexion des systèmes ouverts- Modèle de référence de base - Partie 2 : Architecture de sécurité".

[ISO9796-2] Norme ISO/CEI 9796-2 (2002), "Technologie de l'information - Techniques de sécurité - Schéma de signature numérique donnant la récupération de message - Partie 2 : mécanismes fondés sur la mise en facteur d'entiers".

[ISO9796-4] Norme ISO/CEI 9796-4 (1998), "Technologie de l'information - Techniques de sécurité - Schéma de signature numérique donnant la récupération de message - Partie 4 : mécanismes fondés sur le logarithme dicret".

[ISO10118-1] Norme ISO/CEI 10118-1 (2000), "Technologie de l'information - Techniques de sécurité - Fonctions de hachage - Partie 1 : Généralités".

[ISO10118-2] Norme ISO/CEI 10118-2 (2000), "Technologie de l'information - Techniques de sécurité - Fonctions de hachage - Partie 2 : fonctions de hachage utilisant un algorithme de chiffrement de bloc à n bits".

[ISO10118-3] Norme ISO/CEI 10118-3 (2004), "Technologie de l'information - Techniques de sécurité - Fonctions de hachage - Partie 3 : fonctions de hachage dédiées".

[ISO10118-4] Norme ISO/CEI 10118-4 (1998), "Technologie de l'information - Techniques de sécurité - Fonctions de hachage - Partie 4 : fonctions de hachage utilisant une arithmétique modulaire".

[ISO10181-5] Norme ISO/CEI 10181-5, "Cadres de sécurité dans les systèmes ouverts - cadre de non répudiation", avril 1997.

[ISO13888-1] Norme ISO/CEI 13888-1 (2004), "Technologie de l'information - Techniques de sécurité - non répudiation - Partie 1 : Généralités".

[ISO14888-1] Norme ISO/CEI 14888-1 (1998): "Technologie de l'information - Techniques de sécurité - signatures numériques avec appendice - Partie 1 : Généralités".

[ISO14888-2] Norme ISO/CEI 14888-2 (1999): "Technologie de l'information - Techniques de sécurité - signatures numériques avec appendice - Partie 2 : mécanismes fondés sur l'identité".

[ISO14888-3] Norme ISO/CEI 14888-3 (1998): "Technologie de l'information - Techniques de sécurité - signatures numériques avec appendice - Partie 3 : mécanismes fondés sur le certificat".

[ISO15946-2] Norme ISO/CEI 15946-2 (2002): "Technologie de l'information - Techniques de sécurité - techniques de chiffrement fondés sur les courbes elliptiques - Partie 2 : signatures numériques".

[P1363] IEEE P1363 (2000): "Standard Specifications for Public-Key Cryptography".

[RFC2479] C. Adams, "Interface de programme d'application de service générique de sécurité d'unité de données indépendante (IDUP-GSS-API) ", décembre 1998. *(Information)*

[RFC2743] J. Linn, "Interface générique de programme d'application de service de sécurité, version 2, mise à jour 1", janvier 2000. (*MàJ par* RFC5554)

[RFC3125] J. Ross, D. Pinkas et N. Pope, "Politiques de signature électronique", septembre 2001. *(Exp.)*

[RFC3447] J. Jonsson et B. Kaliski, "Normes de cryptographie à clés publiques (PKCS) n° 1 : Spécifications de la cryptographie RSA version 2.1", février 2003. *(Obsolète, remplacée par* RFC8017) *(Information)*

[RFC3494] K. Zeilenga, "Vers le dépassement du protocole léger d'accès à un répertoire version 2 (LDAPv2)", mars 2003.

[RFC3851] B. Ramsdell, "Spécification du message d'extensions de messagerie Internet multi-objets/sécurisé (S/MIME) version 3.1", juillet 2004. *(Obsolète, voir* RFC5751)

[RFC4210] C. Adams et autres, "Protocole de gestion de certificat (CMP) d'infrastructure de clé publique X.509 pour l'Internet", septembre 2005. (*MàJ par la* RFC6712) *(P.S.)*

[RFC4346] T. Dierks et E. Rescorla, "Protocole de sécurité de la couche Transport (TLS) version 1.1", avril 2006. *(Remplace* RFC2246 *; Remplacée par* RFC5246 *; MàJ par* RFC4366, 4680, 4681, 5746, 6176, 7465, 7507, 7919)

[RFC4523] K. Zeilenga, "Protocole léger d'accès à un répertoire (LDAP) : Définitions de schémas pou les certificats X.509", juin 2006.

[TS101733] ETSI TS 101 733 V.1.7.3 (2005-06) "Formats de signature électronique".

[TS101861] ETSI TS 101 861: "Time stamping profile".

[TS101903] ETSI TS 101 903: "XML Advanced Electronic Signatures (XAdES)".

[TR102038] ETSI TR 102 038: "Electronic Signatures and Infrastructures (ESI); XML format for signature policies".

[TR102272] ETSI TR 102 272 V1.1.1 (2003-12). "Electronic Signatures and Infrastructures (ESI); ASN.1 format for signature policies".

[XMLDSIG] XMLDSIG, W3C/IETF Recommendation, "XML-Signature Syntax and Processing". février 2002.

[X9.30-1] ANSI X9.30-1 (1997): "Public Key Cryptography for the Financial Services Industry - Part 1: The Digital Signature Algorithm (DSA)".

[X9.30-2] ANSI X9.30-2 (1997): "Public Key Cryptography for the Financial Services Industry - Part 2: The Secure Hash Algorithm (SHA-1)".

[X9.31-1] ANSI X9.31-1 (1997): "Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry - Part 1: The RSA Signature Algorithm".

[X9.31-2] ANSI X9.31-2 (1996): "Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry - Part 2: Hash Algorithms".

[X9.62] ANSI X9.62 (1998): "Public Key Cryptography for the Financial Services Industry - The Elliptic Curve Digital Signature Algorithm (ECDSA)".

Les spécifications techniques ETSI peuvent être téléchargées gratuitement via la zone de téléchargement des services et produits à : http://www.etsi.org/WebSite/Standards/StandardsDownload.aspx

Annexe A (normative) : Définitions ASN.1

Cette Annexe fournit un récapitulatif de toutes les définitions de syntaxe ASN.1 pour la nouvelle syntaxe définie dans le présent document.

A.1 Définitions de format de signature utilisant la syntaxe d'ASN.1 X.208

Note : le module ASN.1 défini dans l'Annexe A.1 en utilisant la syntaxe définie dans la Recommandation UIT-T [X.208] a la préséance sur celle définie à l'Annexe A.2 en cas de conflit.

ETS-ElectronicSignatureFormats-ExplicitSyntax88 { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-mod(0) eSignature-explicit88(28)}

ÉTIQUETTES EXPLICITES DE DÉFINITIONS ::=

DÉBUT

-- EXPORTE TOUT

IMPORTE

-- Syntaxe de message cryptographique (CMS) : RFC 3852

ContentInfo, ContentType, id-data, id-signedData, SignedData, EncapsulatedContentInfo, SignerInfo, id-contentType, id-messageDigest, MessageDigest, id-signingTime, SigningTime, id-countersignature, Countersignature
DE CryptographicMessageSyntax2004 { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) modules(0) cms-2004(24) }

-- Attributs ESS définis : ESS Update
-- RFC 5035 (Ajout de l'agilité d'algorithme CertID)

id-aa-signingCertificate, SigningCertificate, IssuerSerial, id-aa-contentReference, ContentReference, id-aa-contentIdentifier, ContentIdentifier, id-aa-signingCertificateV2
DE ExtendedSecurityServices-2006 { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) modules(0) id-mod-ess-2006(30) }

-- Profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat pour l'Internet : RFC 3280

Certificate, AlgorithmIdentifier, CertificateList, Name, DirectoryString, Attribute, BMPString, UTF8String

DE PKIX1Explicit88 {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-explicit(18)}

GeneralNames, GeneralName, PolicyInformation
DE PKIX1Implicit88 {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-implicit (19)}

-- Profil de certificat d'attribut Internet pour l'autorisation - RFC 3281

AttributeCertificate

DE PKIXAttributeCertificate {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-attribute-cert(12)}

-- OCSP - RFC 2560

BasicOCSPResponse, ResponderID
DE OCSP {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-ocsp(14)}

-- Protocole d'horodatage - RFC 3161

TimeStampToken
DE PKIXTSP {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-tsp(13)}

-- Définitions des arcs d'identifiant d'objet utilisées dans le présent document
-- ==========

-- OID utilisé pour référencer les mécanismes de signature électronique fondés sur le présent document pour être utilisés avec l'API Protection d'unité de données indépendantes (IDUP, *Independent Data Unit Protection*) (voir l'Annexe D).

IDENTIFIANT D'OBJET id-etsi-es-IDUP-Mechanism-v1 ::= { itu-t(0) identified-organization(4) etsi(0) electronic-signature-standard (1733) part1 (1) idupMechanism (4) etsiESv1(1) }

-- Attributs d'ES de base de CMS définis dans le présent document
-- ==========

-- OtherSigningCertificate - déconseillé

IDENTIFIANT D'OBJET id-aa-ets-otherSigCert ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-aa(2) 19 }

```
OtherSigningCertificate ::=  SEQUENCE {
  certs         SEQUENCE DE OtherCertID,
  policies      SEQUENCE DE PolicyInformation FACULTATIF
-- NON UTILISÉ DANS LE PRÉSENT DOCUMENT
}

OtherCertID ::= SEQUENCE {
  otherCertHash       OtherHash,
  issuerSerial        IssuerSerial FACULTATIF }

OtherHash ::= CHOIX {
   sha1Hash    OtherHashValue,                    -- contient un hachage SHA-1.
   otherHash   OtherHashAlgAndValue}
```

-- Attributs de politique d'ES définis dans le présent document
-- Attributs de base obligatoires de signature électronique comme ci-dessus avec en plus :
-- Attributs Signature-policy-identifier

IDENTIFIANT D'OBJET id-aa-ets-sigPolicyId ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16) id-aa(2) 15 }

```
SignaturePolicy ::= CHOIX {
  signaturePolicyId           SignaturePolicyId,
  signaturePolicyImplied      SignaturePolicyImplied          -- non utilisé dans cette version.
}

SignaturePolicyId ::= SEQUENCE {
  sigPolicyId       SigPolicyId,
  sigPolicyHash     SigPolicyHash,
  sigPolicyQualifiers   SEQUENCE TAILLE (1..MAX) DE SigPolicyQualifierInfo FACULTATIF
}
```

```
SignaturePolicyImplied ::= NULL

SigPolicyId ::= IDENTIFIANT D'OBJET

SigPolicyHash ::= OtherHashAlgAndValue

OtherHashAlgAndValue ::= SEQUENCE {
  hashAlgorithm   AlgorithmIdentifier,
  hashValue       OtherHashValue }

OtherHashValue ::= CHAINE D'OCTETS

SigPolicyQualifierInfo ::= SEQUENCE {
  sigPolicyQualifierId  SigPolicyQualifierId,
  sigQualifier          TOUS DÉFINIS PAR sigPolicyQualifierId }

SigPolicyQualifierId ::= IDENTIFIANT D'OBJET

IDENTIFIANT D'OBJET id-spq-ets-uri ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16)
id-spq(5) 1 }

SPuri ::= IA5String

IDENTIFIANT D'OBJET id-spq-ets-unotice ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
smime(16) id-spq(5) 2 }

SPUserNotice ::= SEQUENCE {
   noticeRef       NoticeReference FACULTATIF,
   explicitText    DisplayText FACULTATIF}

NoticeReference ::= SEQUENCE {
  organization      DisplayText,
  noticeNumbers    SEQUENCE DE ENTIER }

DisplayText ::= CHOIX {
  visibleString    VisibleString  (TAILLE (1..200)),
  bmpString        BMPString      (TAILLE (1..200)),
  utf8String       UTF8String     (TAILLE (1..200)) }

-- Attributs facultatifs de signature électronique

-- Attribut Commitment-type

IDENTIFIANT D'OBJET id-aa-ets-commitmentType ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 16}

CommitmentTypeIndication ::= SEQUENCE {
 commitmentTypeId CommitmentTypeIdentifier,
 commitmentTypeQualifier SEQUENCE TAILLE (1..MAX) DE CommitmentTypeQualifier FACULTATIF}

CommitmentTypeIdentifier ::= IDENTIFIANT D'OBJET

CommitmentTypeQualifier ::= SEQUENCE {
  commitmentTypeIdentifier CommitmentTypeIdentifier,
  qualifier   TOUS DÉFINIS PAR  commitmentTypeIdentifier }

IDENTIFIANT D'OBJET id-cti-ets-proofOfOrigin ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) cti(6) 1}

IDENTIFIANT D'OBJET id-cti-ets-proofOfReceipt ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) cti(6) 2}
```

IDENTIFIANT D'OBJET id-cti-ets-proofOfDelivery ::={ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 3}

IDENTIFIANT D'OBJET id-cti-ets-proofOfSender ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 4}

IDENTIFIANT D'OBJET id-cti-ets-proofOfApproval ::={ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 5}

IDENTIFIANT D'OBJET id-cti-ets-proofOfCreation ::={ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 6}

-- Attribut Signer-location

IDENTIFIANT D'OBJET id-aa-ets-signerLocation ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 17}

SignerLocation ::= SEQUENCE { -- au moins un des suivants devra être présent.
countryName [0] DirectoryString FACULTATIF, -- comme utilisé pour désigner un pays dans X.500.
localityName [X.509] DirectoryString FACULTATIF, -- comme utilisé pour désigner une localité dans X.500.
postalAdddress [RFC3280] PostalAddress FACULTATIF }

PostalAddress ::= SEQUENCE TAILLE(1..6) DE DirectoryString

-- Attribut Signer-attributes

IDENTIFIANT D'OBJET id-aa-ets-signerAttr ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 18}

SignerAttribute ::= SEQUENCE DE CHOIX {
claimedAttributes [0] ClaimedAttributes,
certifiedAttributes [X.509] CertifiedAttributes }

ClaimedAttributes ::= SEQUENCE DE Attribute

CertifiedAttributes ::= AttributeCertificate -- comme défini dans la RFC 3281,voir le paragraphe 4.1.

-- Attribut Content-Timestamp

IDENTIFIANT D'OBJET id-aa-ets-contentTimestamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 20}

ContentTimestamp ::= TimeStampToken

-- Attribut Signature-TimeStamp

IDENTIFIANT D'OBJET id-aa-signatureTimeStampToken ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 14}

SignatureTimeStampToken ::= TimeStampToken

-- Attribut Complete-certificate-references

IDENTIFIANT D'OBJET id-aa-ets-certificateRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 21}

CompleteCertificateRefs ::= SEQUENCE DE OtherCertID

-- Attribut Complete-revocation-references

```
IDENTIFIANT D'OBJET id-aa-ets-revocationRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-aa(2) 22}

CompleteRevocationRefs ::=  SEQUENCE DE CrlOcspRef

CrlOcspRef ::= SEQUENCE {
  crlids        [0] CRLListID   FACULTATIF,
  ocspids       [X.509] OcspListID  FACULTATIF,
  otherRev      [RFC3280] OtherRevRefs FACULTATIF
}

CRLListID ::=  SEQUENCE {
  crls       SEQUENCE DE CrlValidatedID}

CrlValidatedID ::=  SEQUENCE {
  crlHash               OtherHash,
  crlIdentifier         CrlIdentifier FACULTATIF}

CrlIdentifier ::= SEQUENCE {
  crlissuer             Name,
  crlIssuedTime     UTCTime,
  crlNumber         ENTIER FACULTATIF }

OcspListID ::=  SEQUENCE {
   ocspResponses       SEQUENCE DE OcspResponsesID}

OcspResponsesID ::=  SEQUENCE {
   ocspIdentifier        OcspIdentifier,
   ocspRepHash           OtherHash    FACULTATIF
}

OcspIdentifier ::= SEQUENCE {
  ocspResponderID     ResponderID,                    -- comme dans les données de réponse OCSP.
  producedAt          GeneralizedTime                 -- comme dans les données de réponse OCSP.
}

OtherRevRefs ::= SEQUENCE {
   otherRevRefType   OtherRevRefType,
   otherRevRefs         TOUS DÉFINIS PAR otherRevRefType
}

OtherRevRefType ::= IDENTIFIANT D'OBJET

-- Attribut Certificate-values

IDENTIFIANT D'OBJET id-aa-ets-certValues ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-aa(2) 23}

CertificateValues ::=  SEQUENCE DE Certificate

-- Valeurs de révocation de Certificate-attribut

IDENTIFIANT D'OBJET id-aa-ets-revocationValues ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 24}

RevocationValues ::=  SEQUENCE {
  crlVals            [0] SEQUENCE DE CertificateList FACULTATIF,
  ocspVals          [X.509] SEQUENCE DE BasicOCSPResponse FACULTATIF,
  otherRevVals   [RFC3280] OtherRevVals FACULTATIF}

OtherRevVals ::= SEQUENCE {
   otherRevValType   OtherRevValType,
```

```
   otherRevVals         TOUS DÉFINIS PAR otherRevValType
}

OtherRevValType ::= IDENTIFIANT D'OBJET

-- Attribut Horodatage de CAdES-C

IDENTIFIANT D'OBJET id-aa-ets-escTimeStamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-aa(2) 25}

ESCTimeStampToken ::= TimeStampToken

-- Certificats et CRL horodatés

IDENTIFIANT D'OBJET id-aa-ets-certCRLTimestamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 26}

TimestampedCertsCRLs ::= TimeStampToken

-- Attribut Horodatage d'archive
IDENTIFIANT D'OBJET id-aa-ets-archiveTimestampV2  ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
pkcs-9(9) smime(16) id-aa(2) 48}

ArchiveTimeStampToken ::= TimeStampToken

-- Attribut Attribute-certificate-references

IDENTIFIANT D'OBJET id-aa-ets-attrCertificateRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 44}

AttributeCertificateRefs ::= SEQUENCE DE OtherCertID

-- Attribut Attribute-revocation-references

IDENTIFIANT D'OBJET id-aa-ets-attrRevocationRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 45}

AttributeRevocationRefs ::=  SEQUENCE DE CrlOcspRef

FIN
```

A.2 Définitions de format de signature utilisant la syntaxe d'ASN.1 X.680

Note : le module ASN.1 défini dans l'Annexe A.1 a la préséance sur celui défini dans l'Annexe A.2 en utilisant la syntaxe définie dans la Recommandation UIT-T X.680 (1997) [X.680] en cas de conflit.

```
ETS-ElectronicSignatureFormats-ExplicitSyntax97 { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-mod(0) eSignature-explicit97(29)}

ÉTIQUETTES EXPLICITES DE DÉFINITIONS ::=

DÉBUT

-- EXPORTE TOUT -

IMPORTE

-- Syntaxe de message cryptographique (CMS) : RFC 3852

ContentInfo, ContentType, id-data, id-signedData, SignedData, EncapsulatedContentInfo, SignerInfo, id-contentType, id-
messageDigest, MessageDigest, id-signingTime, SigningTime, id-countersignature, Countersignature
```

DE CryptographicMessageSyntax2004 { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) modules(0) cms-2004(24) }

-- Attributs définis dans ESS : mise à jour de ESS
-- RFC 5035 (Ajout de l'agilité de l'algorithme CertID)

id-aa-signingCertificate, SigningCertificate, IssuerSerial,
id-aa-contentReference, ContentReference, id-aa-contentIdentifier,
ContentIdentifier, id-aa-signingCertificateV2
DE ExtendedSecurityServices-2006
{ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) modules(0) id-mod-ess-2006(30) }

-- Infrastructure de clé publique Internet X.509
-- Profil de certificat et de CRL : RFC 3280
Certificate, AlgorithmIdentifier, CertificateList, Name, Attribute

DE PKIX1Explicit88 {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-explicit(18)}

GeneralNames, GeneralName, PolicyInformation
DE PKIX1Implicit88 {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-implicit(19)}

-- Profil de certificat d'attribut Internet pour autorisation - RFC 3281

AttributeCertificate
DE PKIXAttributeCertificate {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-attribute-cert(12)}

-- OCSP RFC 2560

BasicOCSPResponse, ResponderID
DE OCSP {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-ocsp(14)}

-- RFC 3161 Infrastructure Internet de clé publique X.509
-- Protocole d'horodatage

TimeStampToken
DE PKIXTSP {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-tsp(13)}

-- X.520

DirectoryString {}
DE SelectedAttributeTypes {joint-iso-itu-t ds(5) module(1) selectedAttributeTypes(5) 4}

;

-- Définitions des arcs d'identifiant d'objet utilisés dans le présent document
-- ==

-- OID utilisés pour référencer les mécanismes de signature électronique fondés sur le présent document à utiliser avec l'API IDUP (voir l'Annexe D)

IDENTIFIANT D'OBJET id-etsi-es-IDUP-Mechanism-v1 ::= { itu-t(0) identified-organization(4) etsi(0) electronic-signature-standard (1733) part1 (1) idupMechanism (4) etsiESv1(1) }

-- Attributs d'ES de base définis dans le présent document
-- ==

-- Attributs de CMS définis dans le présent document

```
-- OtherSigningCertificate - déconseillé

IDENTIFIANT D'OBJET id-aa-ets-otherSigCert ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
smime(16) id-aa(2) 19 }

OtherSigningCertificate ::=  SEQUENCE {
  certs       SEQUENCE DE OtherCertID,
  policies    SEQUENCE DE PolicyInformation FACULTATIF        -- non utilisé dans le présent document
}

OtherCertID ::= SEQUENCE {
  otherCertHash       OtherHash,
  issuerSerial        IssuerSerial FACULTATIF }

OtherHash ::= CHOIX {
  sha1Hash OtherHashValue,                          - contient un hachage SHA-1.
  otherHash OtherHashAlgAndValue}

-- Attributs de politique d'ES définis dans le présent document
-- ===========================================================

-- Attributs de base obligatoire de signature électronique, en plus.
-- Identifiant de politique de signature Policy

IDENTIFIANT D'OBJET id-aa-ets-sigPolicyId ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
smime(16) id-aa(2) 15 }

SignaturePolicy ::= CHOIX {
  signaturePolicyId         SignaturePolicyId,
  signaturePolicyImplied    SignaturePolicyImplied              -- non utilisé dans cette version.
}

SignaturePolicyId ::= SEQUENCE {
  sigPolicyId             SigPolicyId,
  sigPolicyHash           SigPolicyHash,
  sigPolicyQualifiers     SEQUENCE TAILLE (1..MAX) DE SigPolicyQualifierInfo FACULTATIF
}

SignaturePolicyImplied ::= NULL

SigPolicyId ::= IDENTIFIANT D'OBJET

SigPolicyHash ::= OtherHashAlgAndValue

OtherHashAlgAndValue ::= SEQUENCE {
  hashAlgorithm   AlgorithmIdentifier,
  hashValue       OtherHashValue
}

OtherHashValue ::= CHAINE D'OCTETS

SigPolicyQualifierInfo ::= SEQUENCE {
  sigPolicyQualifierId   SIG-POLICY-QUALIFIER.&id ({SupportedSigPolicyQualifiers}),
  qualifier              SIG-POLICY-QUALIFIER.&Qualifier ({SupportedSigPolicyQualifiers}
                         {@sigPolicyQualifierId})FACULTATIF }

  SupportedSigPolicyQualifiers SIG-POLICY-QUALIFIER ::= { noticeToUser | pointerToSigPolSpec }

  SIG-POLICY-QUALIFIER ::= CLASS {
    &id           IDENTIFIANT D'OBJET UNIQUE,
    &Qualifier    FACULTATIF }
```

```
  WITH SYNTAX {
    SIG-POLICY-QUALIFIER-ID     &id
    [SIG-QUALIFIER-TYPE &Qualifier] }

noticeToUser SIG-POLICY-QUALIFIER ::= {
    SIG-POLICY-QUALIFIER-ID id-spq-ets-unotice SIG-QUALIFIER-TYPE SPUserNotice }

pointerToSigPolSpec SIG-POLICY-QUALIFIER ::= {
    SIG-POLICY-QUALIFIER-ID id-spq-ets-uri SIG-QUALIFIER-TYPE SPuri }

IDENTIFIANT D'OBJET id-spq-ets-uri ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9) smime(16)
id-spq(5) 1 }

SPuri ::= IA5String

IDENTIFIANT D'OBJET id-spq-ets-unotice ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs9(9)
smime(16) id-spq(5) 2 }

SPUserNotice ::= SEQUENCE {
   noticeRef       NoticeReference FACULTATIF,
   explicitText    DisplayText FACULTATIF}

NoticeReference ::= SEQUENCE {
    organization       DisplayText,
    noticeNumbers    SEQUENCE DE ENTIER }

DisplayText ::= CHOIX {
    visibleString    VisibleString   (TAILLE (1..200)),
    bmpString        BMPString       (TAILLE (1..200)),
    utf8String       UTF8String      (TAILLE (1..200)) }

-- Attributs facultatifs de signature électronique

-- Type d'engagement

IDENTIFIANT D'OBJET id-aa-ets-commitmentType ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 16}

CommitmentTypeIndication ::= SEQUENCE {
  commitmentTypeId CommitmentTypeIdentifier,
  commitmentTypeQualifier SEQUENCE TAILLE (1..MAX) DE CommitmentTypeQualifier FACULTATIF}

CommitmentTypeIdentifier ::= IDENTIFIANT D'OBJET

CommitmentTypeQualifier ::= SEQUENCE {
  commitmentQualifierId   COMMITMENT-QUALIFIER.&id,
  qualifier                       COMMITMENT-QUALIFIER.&Qualifier FACULTATIF }

COMMITMENT-QUALIFIER ::= CLASS {
    &id              IDENTIFIANT D'OBJET UNIQUE,
    &Qualifier     FACULTATIF }
WITH SYNTAX {
    COMMITMENT-QUALIFIER-ID     &id
    [COMMITMENT-TYPE &Qualifier] }

IDENTIFIANT D'OBJET id-cti-ets-proofOfOrigin ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) cti(6) 1}

IDENTIFIANT D'OBJET id-cti-ets-proofOfReceipt ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) cti(6) 2}
```

IDENTIFIANT D'OBJET id-cti-ets-proofOfDelivery ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 3}

IDENTIFIANT D'OBJET id-cti-ets-proofOfSender ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 4}

IDENTIFIANT D'OBJET id-cti-ets-proofOfApproval ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 5}

IDENTIFIANT D'OBJET id-cti-ets-proofOfCreation ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) cti(6) 6}

-- Localisation du signataire

IDENTIFIANT D'OBJET id-aa-ets-signerLocation ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 17}

SignerLocation ::= SEQUENCE { -- au moins un des suivants devra être présent.
countryName [0] DirectoryString{maxSize} FACULTATIF, -- comme utilisé pour nommer un pays dans X.520
localityName [X.509] DirectoryString{maxSize} FACULTATIF, -- comme utilisé pour nommer une localité dans X.520
postalAdddress [RFC3280] PostalAddress FACULTATIF }

PostalAddress ::= SEQUENCE TAILLE(1..6) DE DirectoryString{maxSize}
-- paramètre maxSize comme spécifié dans X.683

-- Attributs de signataire

IDENTIFIANT D'OBJET id-aa-ets-signerAttr ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 18}

SignerAttribute ::= SEQUENCE DE CHOIX {
claimedAttributes [0] ClaimedAttributes,
certifiedAttributes [X.509] CertifiedAttributes }

ClaimedAttributes ::= SEQUENCE DE Attribute

CertifiedAttributes ::= AttributeCertificate -- comme défini dans la RFC 3281 ; voir le paragraphe 4.1.

-- Horodatage du contenu

IDENTIFIANT D'OBJET id-aa-ets-contentTimestamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 20}

ContentTimestamp ::= TimeStampToken

-- Horodatage de signature

IDENTIFIANT D'OBJET id-aa-signatureTimeStampToken ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 14}

SignatureTimeStampToken ::= TimeStampToken

-- Références complètes de certificat.

IDENTIFIANT D'OBJET id-aa-ets-certificateRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-aa(2) 21}

CompleteCertificateRefs ::= SEQUENCE DE OtherCertID

-- Références complètes de révocation

```
IDENTIFIANT D'OBJET id-aa-ets-revocationRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-aa(2) 22}

CompleteRevocationRefs ::=  SEQUENCE DE CrlOcspRef

CrlOcspRef ::= SEQUENCE {
  crlids        [0] CRLListID   FACULTATIF,
  ocspids       [X.509] OcspListID  FACULTATIF,
  otherRev     [RFC3280] OtherRevRefs FACULTATIF
}

CRLListID ::=  SEQUENCE {
  crls       SEQUENCE DE CrlValidatedID
}

CrlValidatedID ::=  SEQUENCE {
  crlHash               OtherHash,
  crlIdentifier          CrlIdentifier FACULTATIF   }

CrlIdentifier ::= SEQUENCE {
   crlissuer             Name,
   crlIssuedTime       UTCTime,
   crlNumber            ENTIER FACULTATIF
}

OcspListID ::=  SEQUENCE {
   ocspResponses       SEQUENCE DE OcspResponsesID
}

OcspResponsesID ::=  SEQUENCE {
   ocspIdentifier           OcspIdentifier,
   ocspRepHash             OtherHash    FACULTATIF
}

OcspIdentifier ::= SEQUENCE {
  ocspResponderID     ResponderID,                    -- comme dans les données de réponse OCSP.
    producedAt         GeneralizedTime                -- comme dans les données de réponse OCSP.
}

OtherRevRefs ::= SEQUENCE {
  otherRevRefType   OTHER-REVOCATION-REF.&id,
  otherRevRefs        SEQUENCE DE OTHER-REVOCATION-REF.&Type
}

OTHER-REVOCATION-REF ::= CLASS {
    &Type,
    &id   IDENTIFIANT D'OBJET UNIQUE }
  WITH SYNTAX {
    WITH SYNTAX &Type ID &id }

-- Valeurs de certificat

IDENTIFIANT D'OBJET id-aa-ets-certValues ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-aa(2) 23}

CertificateValues ::=  SEQUENCE DE Certificate

-- Valeurs de révocation de certificat

IDENTIFIANT D'OBJET id-aa-ets-revocationValues ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 24}
```

```
RevocationValues ::=  SEQUENCE {
  crlVals           [0] SEQUENCE DE CertificateList FACULTATIF,
  ocspVals          [X.509] SEQUENCE DE BasicOCSPResponse FACULTATIF,
  otherRevVals    [RFC3280] OtherRevVals FACULTATIF
}

OtherRevVals ::= SEQUENCE {
   otherRevValType   OTHER-REVOCATION-VAL.&id,
   otherRevVals         SEQUENCE DE OTHER-REVOCATION-REF.&Type
}

THER-REVOCATION-VAL ::= CLASS {
     &Type,
     &id   IDENTIFIANT D'OBJET UNIQUE }
   WITH SYNTAX {
     WITH SYNTAX &Type ID &id }

-- Horodatage CAdES-C
IDENTIFIANT D'OBJET id-aa-ets-escTimeStamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9)
smime(16) id-aa(2) 25}

ESCTimeStampToken ::= TimeStampToken

-- Certificats et CRL horodatés
IDENTIFIANT D'OBJET id-aa-ets-certCRLTimestamp ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 26}

TimestampedCertsCRLs ::= TimeStampToken

-- Horodatage d'archive
IDENTIFIANT D'OBJET id-aa-ets-archiveTimestampV2   ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1)
pkcs-9(9) smime(16) id-aa(2) 48}

ArchiveTimeStampToken ::= TimeStampToken

-- Références de certificat d'attribut
IDENTIFIANT D'OBJET id-aa-ets-attrCertificateRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 44}

AttributeCertificateRefs ::=  SEQUENCE DE OtherCertID

-- Références de révocation d'attribut
IDENTIFIANT D'OBJET id-aa-ets-attrRevocationRefs ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-
9(9) smime(16) id-aa(2) 45}

AttributeRevocationRefs ::= SEQUENCE DE CrlOcspRef

FIN
```

Annexe B (pour information) : Formes étendues de signatures électroniques

La Section 4 fournit une vue d'ensemble des divers formats de signatures électroniques inclus dans le présent document. Cette annexe fait la liste des attributs qui doivent être présents dans les divers formats étendus de signature électronique et fournit des exemples de séquences de validation utilisant les formats étendus.

B.1 Formes étendues de données de validation

Les données de validation complètes (CAdES-C) décrites au paragraphe 4.3 et illustrées à la Figure 3 peuvent être étendues pour créer des signatures électroniques avec données de validation étendues. Certaines formes de signature électronique qui incluent la validation étendue sont expliquées ci-dessous.

Une signature électronique X-Long (CAdES-X Long) est la CAdES-C avec les valeurs des informations de certificats et de révocation.

Cette forme de signature électronique peut être utile quand le vérificateur n'a pas d'accès direct aux informations suivantes :
- le certificat du signataire,
- tous les certificats de CA qui constituent le chemin de certification complet,
- toutes les informations d'état de révocation associées, comme référencé dans la CAdES-C.

Dans certaines situations, des horodatages supplémentaires peuvent être créés et ajoutés aux signatures électroniques comme attributs supplémentaires. Par exemple :
- horodatage de toutes les données de validation détenues avec l'ES (CAdES-C) ; ces données de validation étendues sont appelées une CAdES-X de type 1, ou
- horodatage des données de référence individuelles comme utilisées pour la validation complète. Cette forme de données de validation étendues est appelé une CAdES-X de type 2.

Note 1 : les avantages/inconvénients des CAdES-X de type 1 et de type 2 sont discutés à l'Annexe C.4.4.

Les formes d''horodatage ci-dessus peuvent être utiles quand il est exigé de contrer le risque que des clés de CA compromises soient utilisées dans la chaîne de certificats.

Une combinaison des deux formats ci-dessus peut être utilisée. Cette forme de données de validation étendues est appelée une ES X-long de type 1 ou CAdES-X long de type 2. Cette forme de signature électronique peut être utile quand le vérificateur a besoin des valeurs et de la preuve de quand les données de validation ont existé.

Note 2 : les avantages/inconvénients des CAdES-X long de type 1 et CAdES-X long de type 2 sont discutés à l'Annexe C.4.6.

B.1.1 CAdES-X long

Une signature électronique avec les données de validation supplémentaires formant une CAdES-X long est illustrée à la Figure B.1 et comprend :
- une CAdES-BES ou CAdES-EPES, comme défini aux paragraphes 4.3, 5.7, ou 5.8,
- un attribut Références complètes de certificat, comme défini au paragraphe 6.2.1,
- un attribut Références complètes de révocation, comme défini au paragraphe 6.2.2.

Les attributs suivants sont exigés si un TSP ne fournit pas une marque de temps de l'ES :
- attribut Horodatage de signature, comme défini au paragraphe 6.1.1.

Les attributs suivants sont exigés si les valeurs de certificat et de révocation complètes ne sont pas déjà incluses dans la CAdES-BES ou CAdES-EPES :
- attribut Valeurs de certificat, comme défini au paragraphe 6.3.3;
- attribut Valeurs de révocation, comme défini au paragraphe 6.3.4.

Si des certificats d'attributs sont utilisés, alors les attributs suivants peuvent être présents :
- attribut attribute-certificate-references, défini au paragraphe 6.2.3;
- attribut attribute-revocation-references, comme défini au paragraphe 6.2.4.

D'autres attributs non signés peuvent être présents, mais ne sont pas exigés.

Note : les références de certificat et révocation d'attribut sont seulement présentes si un certificat d'attribut d'utilisateur est présent dans la signature électronique ; voir les paragraphes 6.2.2 et 6.2.3.

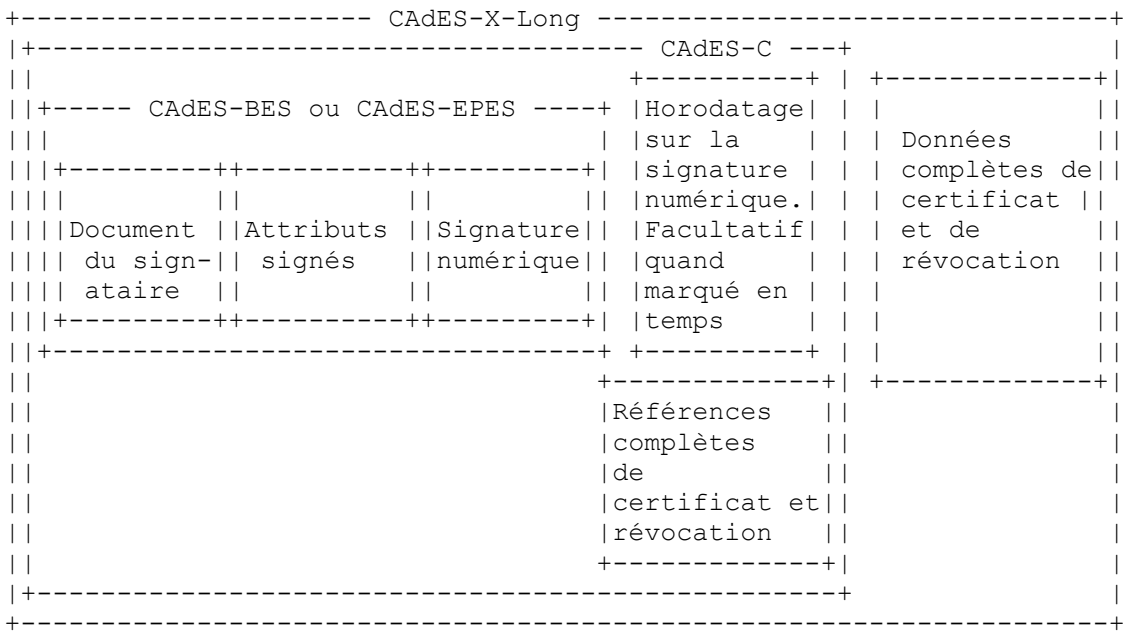


Figure B.1 : Illustration de CAdES-X-Long

B.1.2 CAdES-X type 1

Une signature électronique avec les données de validation supplémentaires formant les données de validation étendues - Type 1 X est illustrée à la Figure B.2 et comprend :
- la CAdES-BES ou CAdES-EPES, comme défini aux paragraphes 4.2, 5.7, ou 5.8,
- un attribut Références complètes de certificat, comme défini au paragraphe 6.2.1,
- un attribut Références complètes de révocation, comme défini au paragraphe 6.2.2,
- un attribut Horodatage de CAdES-C, comme défini au paragraphe 6.3.5.

Les attributs suivants sont exigés si un TSP ne fournit pas de marque de temps de l'ES :
- attribut Horodatage de signature, comme défini au paragraphe 6.1.1.

Si des certificats d'attributs sont utilisés, les attributs suivants peuvent être présents :
- attribute-certificate-references, défini au paragraphe 6.2.3;
- attribute-revocation-references, comme défini au paragraphe 6.2.4.

D'autres attributs non signés peuvent être présents, mais ne sont pas exigés.

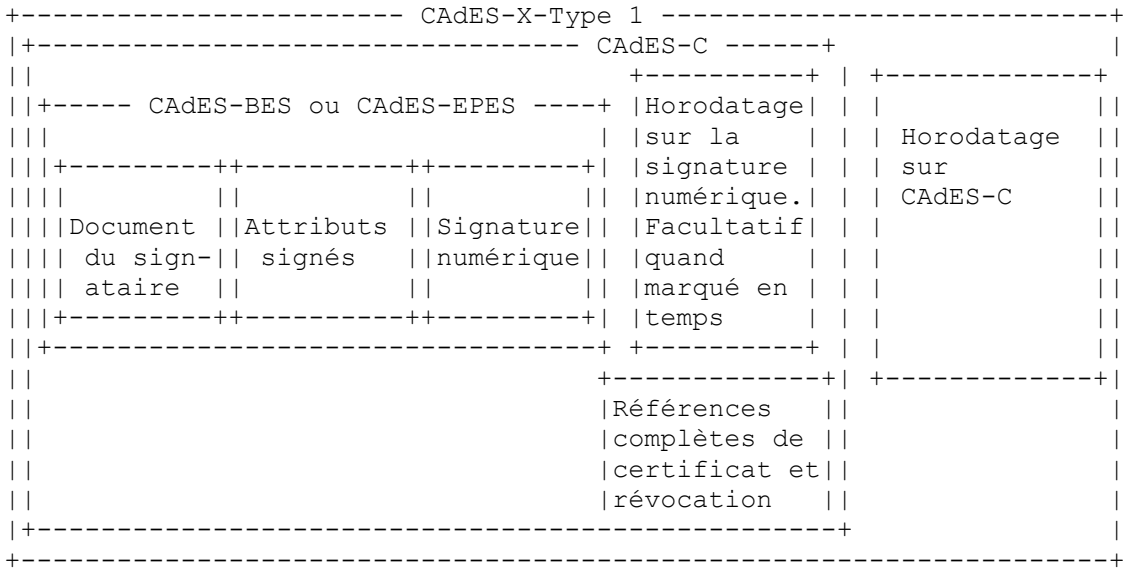


Figure B.2 : Illustration de CAdES-X Type 1

B.1.3 CAdES-X type 2

Une signature électronique avec des données de validation supplémentaires formant les données de validation étendues - type 2 X est illustrée à la Figure B.3 et comporte :
- une CAdES-BES ou CAdES-EPES, comme défini aux paragraphes 4.2, 5.7, ou 5.8,
- un attribut Références complètes de certificat, comme défini au paragraphe 6.2.1,
- un attribut Références complètes de révocation, comme défini au paragraphe 6.2.2,
- un attribut time-stamped-certs-crls-references, comme défini au paragraphe 6.3.6.

Les attributs suivants sont exigés si un TSP ne fournit pas une marque de temps de l'ES :
- attribut Horodatage de signature, comme défini au paragraphe 6.1.1.

Si des certificats d'attribut sont utilisés, alors les attributs suivants peuvent être présents :
- attribut attribute-certificate-references, défini au paragraphe 6.2.3;
- attribut attribute-revocation-references, comme défini au paragraphe 6.2.4.

D'autres attributs non signés peuvent être présents, mais ne sont pas exigés.

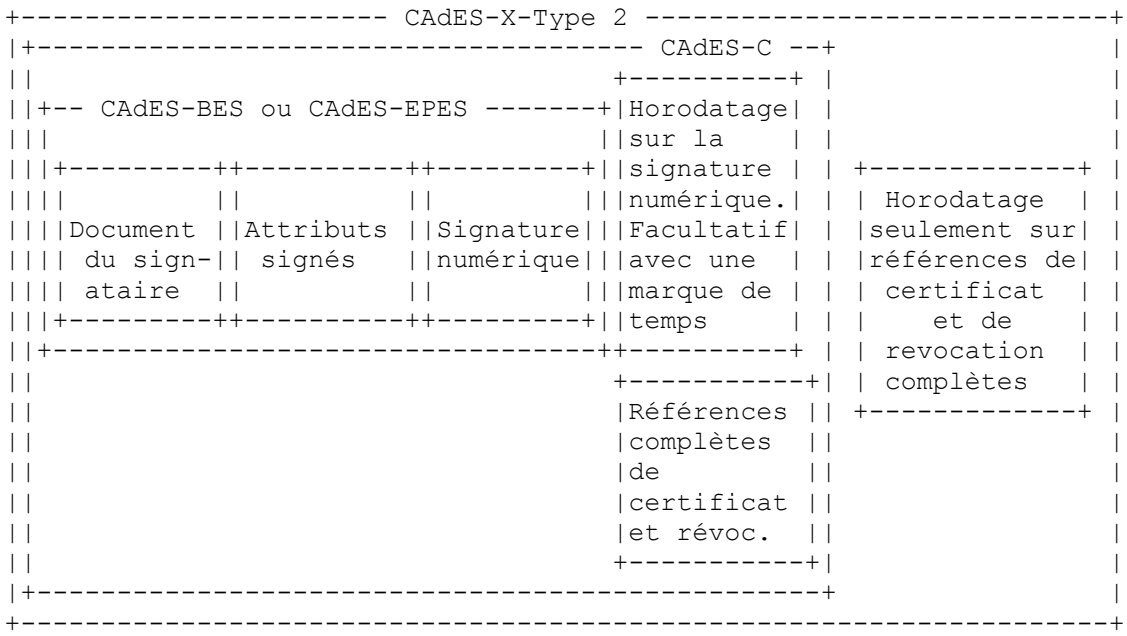


Figure B.3 : Illustration de CAdES-X type 2

B.1.4 CAdES-X longue type 1 et CAdES-X longue type 2

Une signature électronique avec les données de validation supplémentaires formant la CAdES-X longue type 1 et la CAdES-X longue type 2 est illustrée à la Figure B.4 et comprend :
- une CAdES-BES ou CAdES-EPES, comme défini au paragraphes 4.3, 5.7, ou 5.8,
- un attribut Références complètes de certificat, comme défini au paragraphe 6.2.1,
- un attribut Références complètes de révocation, comme défini au paragraphe 6.2.2.

Les attributs suivants sont exigés si un TSP ne fournit pas de marque de temps de l'ES :
- attribut Horodatage de signature, comme défini au paragraphe 6.1.1.

Les attributs suivants sont exigés si les valeurs complètes de certificat et de révocation ne sont pas déjà incluses dans la CAdES-BES ou CAdES-EPES :
- attribut Valeurs de certificat, comme défini au paragraphe 6.3.3;
- attribut Valeurs de révocation, comme défini au paragraphe 6.3.4.

Si des certificats d'attributs sont utilisés, alors les attributs suivants peuvent être présents :
- attribut attribute-certificate-references, défini au paragraphe 6.2.3,

- attribut attribute-revocation-references, comme défini au paragraphe 6.2.4.

De plus, un des attributs suivants est exigé :
- attribut CAdES-C-Timestamp, comme défini au paragraphe 6.3.5,
- attribut time-stamped-certs-crls-references, comme défini au paragraphe 6.3.6.

D'autres attributs non signés peuvent être présents, mais ne sont pas exigés.

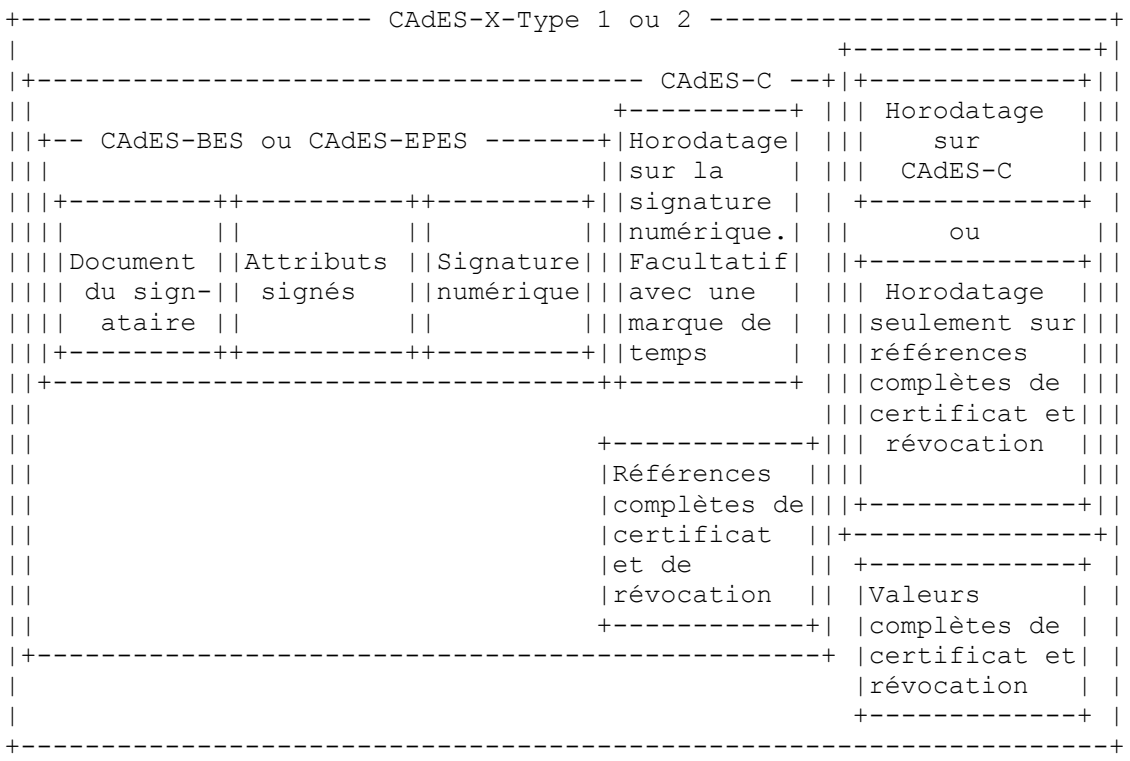


Figure B.4 : Illustration de CAdES-X longue type 1 et CAdES-X longue type 2

B.2 Extensions d'hoirodatage

Chaque instance de l'attribut Horodatage peut inclure, comme attributs non signés dans les signedData de l'horodatage, les attributs suivants relatifs à la TSU :
- un attribut Références complètes de certificat de la TSU, comme défini au paragraphe 6.2.1,
- un attribut Références complètes de révocation de la TSU, comme défini au paragraphe 6.2.2,
- un attribut Valeurs de certificat de la TSU, comme défini au paragraphe 6.3.3,
- un attribut Valeurs de révocation de la TSU, comme défini au paragraphe 6.3.4.

D'autres attributs non signés peuvent être présents, mais ne sont pas exigés.

B.3 Données de validation d'archive (CAdES-A)

Avant que les algorithmes, clés, et autres données cryptographiques utilisées au moment de la construction de la CAdES-C deviennent faibles et que les fonctions cryptographiques deviennent vulnérables, ou que les certificats qui prennent en charge l'horodatage précédent arrivent à expiration, les données signées, la CAdES-C, et toutes les informations supplémentaires (c'est-à-dire, toute CAdES-X) devraient être horodatées. Si possible, cela devrait utiliser des algorithmes plus forts (ou des clés plus longues) que dans l'horodatage original. Ces données supplémentaires et cet horodatage sont appelés des données de validation d'archive exigées pour le format Archive d'ES (CAdES-A). Le processus d'horoatage peut être répété chaque fois que la protection utilisée pour l'horodatage d'une CAdES-A précédente devient faible. Une CAdES-A peut donc porter plusieurs horodatages incorporés.

Un exemple de signature électronique (ES), avec les données de validation supplémentaires pour la CAdES-C et CAdES-X formant la CAdES-A est illustré à la Figure B.5.

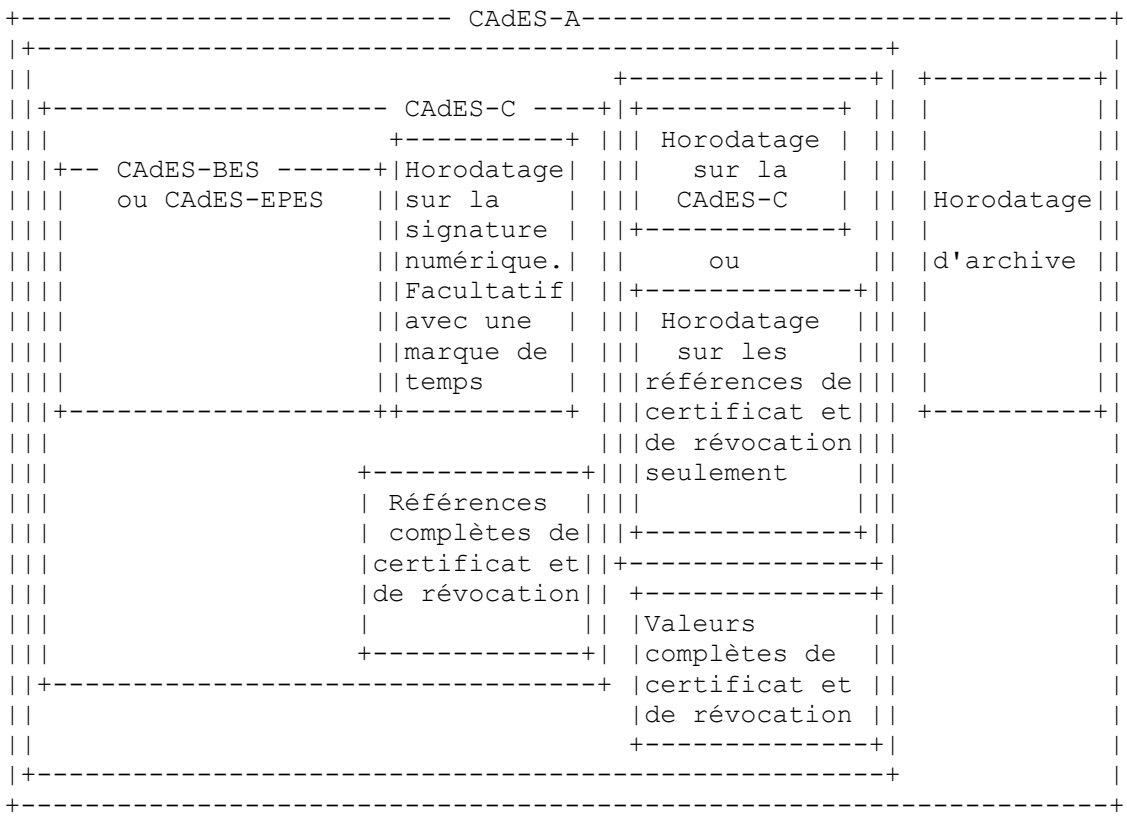


Figure B.5 : Illustration de CAdES-A

Une CAdES-A comporte les éléments suivants :
- la CAdES-BES ou CAdES-EPES, incluant leurs attributs signés et non signés,
- un attribut Références complètes de certificat, comme défini au paragraphe 6.2.1,
- un attribut Références complètes de révocation, comme défini au paragraphe 6.2.2.

L'attribut suivant est exigé si un TSP ne fournit pas de marque de temps de l'ES :
- attribut Horodatage de signature, comme défini au paragraphe 6.1.1.

Si des certificats d'attributs sont utilisés, alors les attributs suivants peuvent être présents :
- un attribut attribute-certificate-references, défini au paragraphe 6.2.3,
- un attribut attribute-revocation-references, comme défini au paragraphe 6.2.4.

Les attributs suivants sont exigés si les valeurs complètes de certificat et de révocation ne sont pas déjà incluses dans la CAdES-BES ou CAdES-EPES :
- attribut Valeurs de certificat, comme défini au paragraphe 6.3.3,
- attribut Valeurs de révocation, comme défini au paragraphe 6.3.4.

Au moins un des deux attributs suivants est exigé :
- attribut CAdES-C-Timestamp, comme défini au paragraphe 6.3.5,
- attribut time-stamped-certs-crls-references, comme défini au paragraphe 6.3.6.

L'attribut suivant est exigé :
- attribut archive-time-stamp, défini au paragraphe 6.4.1.

Plusieurs instances de l'attribut archive-time-stamp peuvent apparaître avec une signature électronique, à la fois dans le temps et provenant de TSU différentes. L'horodatage devrait être créé en utilisant des algorithmes plus forts (ou des clés plus longues) que dans les signatures électroniques ou horodatages originaux.

D'autres attributs non signés de l'ES peuvent être présents, mais ne sont pas exigés.

L'attribut archive-time-stamp va lui-même contenir les informations de certificat et de révocation exigées pour valider le archive-time-stamp ; cela peut inclure les attributs non signés suivants :

- attribut Références complètes de certificat de la TSU, comme défini au paragraphe 6.2.1,
- attribut Références complètes de révocation de la TSU, comme défini au paragraphe 6.2.2,
- attribut Valeurs de certificat de la TSU, comme défini au paragraphe 6.3.3,
- attribut Valeurs de révocation de la TSU, comme défini au paragraphe 6.3.4.

D'autres attributs non signés peuvent être présents, mais ne sont pas exigés.

B.4 Exemple de séquence de validation

Comme décrit précédemment, le signataire ou vérificateur initial peut collecter toutes les données supplémentaires qui forment la signature électronique. La Figure B.6 et la description qui suit montrent comment le processus de validation peut construire une signature électronique complète avec le temps.

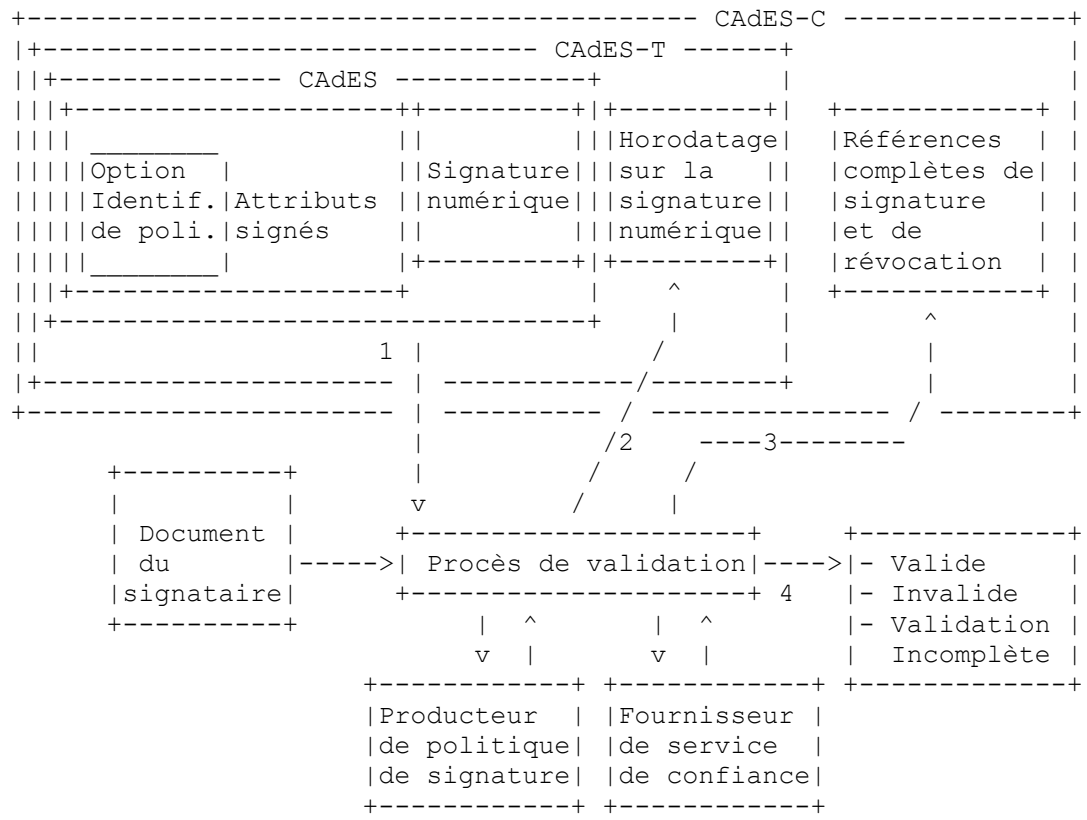


Figure B.6 : Illustration d'une séquence de validation CAdES

Peu après la réception de la signature électronique (CAdES) du signataire (1), la valeur de la signature numérique peut être vérifiée ; le processus de validation devra au moins ajouter un horodatage (2), sauf si le signataire en a fourni un qui est de confiance pour le vérificateur. Le processus de validation peut aussi valider la signature électronique en utilisant des données supplémentaires (par exemple, certificats, CRL, etc.) fournies par les fournisseurs de service de confiance. Quand c'est applicable, le processus de validation va aussi avoir besoin de se conformer aux exigences spécifiées dans une politique de signature. Si le résultat du processus de validation est "validation incomplète", alors le résultat de cette étape est la CAdES-T.

Pour s'assurer que l'état de validité est "Valide" ou "Invalide" et le communiquer à l'utilisateur (4), toutes les données supplémentaires exigées pour valider la CAdES-C doivent être disponibles (par exemple, le certificat complet et les informations de révocation).

Une fois que les données nécessaires pour que les références de données de validation complètes (CAdES-C) sont disponibles, le processus de validation devrait :
- obtenir tous les certificats et informations d'état de révocation supplémentaires nécessaires ;
- compléter toutes les vérifications de validation sur l'ES en utilisant les informations complètes de certificat et de révocation (si un horodatage n'est pas déjà présent, cela peut être ajouté à cette même étape, en combinant les processus de CAdES-T et de CAdES-C) ;
- enregistrer les références complètes de certificat et de révocation (3) ;

- indiquer l'état de validité à l'utilisateur (4).

En même temps que le processus de validation crée la CAdES-C, il peut fournir et/ou enregistrer les valeurs des informations de certificats et d'état de révocation utilisées dans la CAdES-C (5). Le résultat final est appelé CAdES-X long.

Ceci est illustré par la Figure B.7.

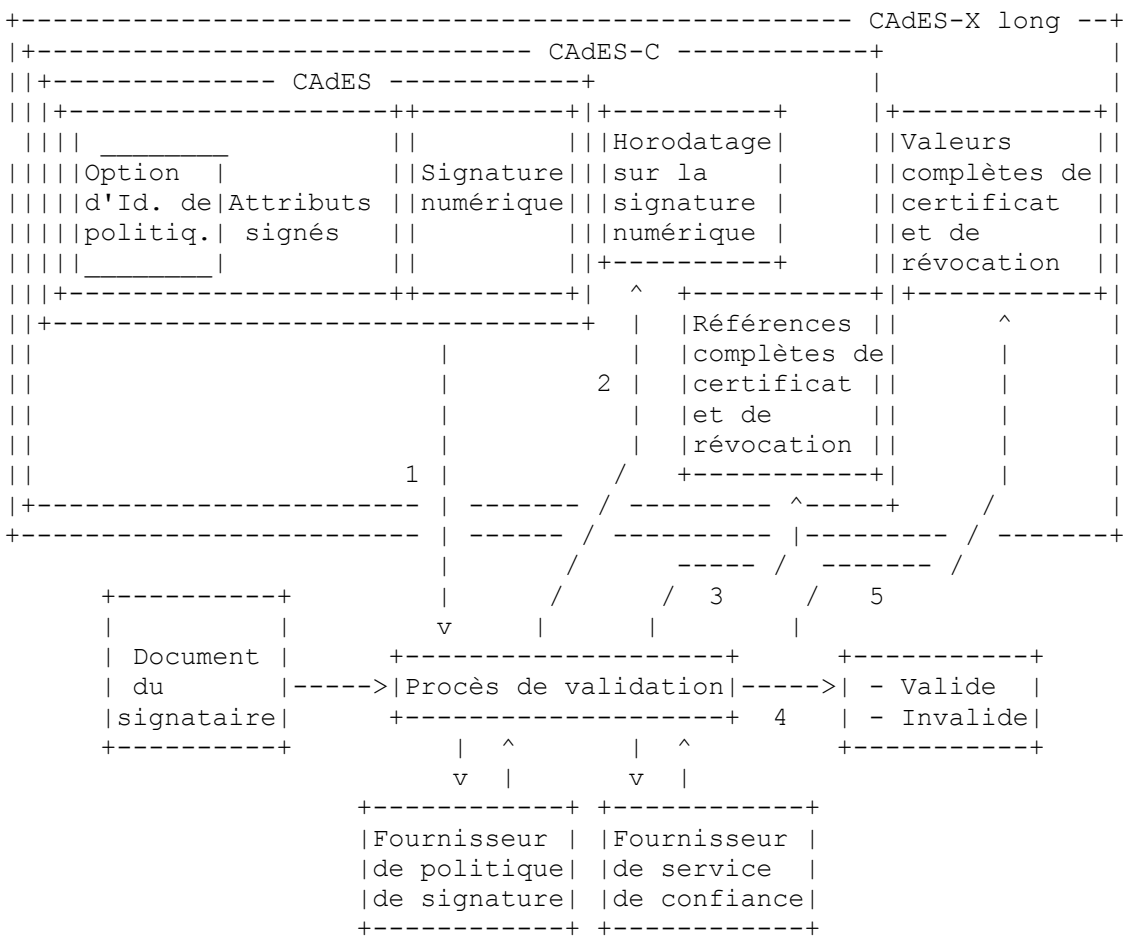
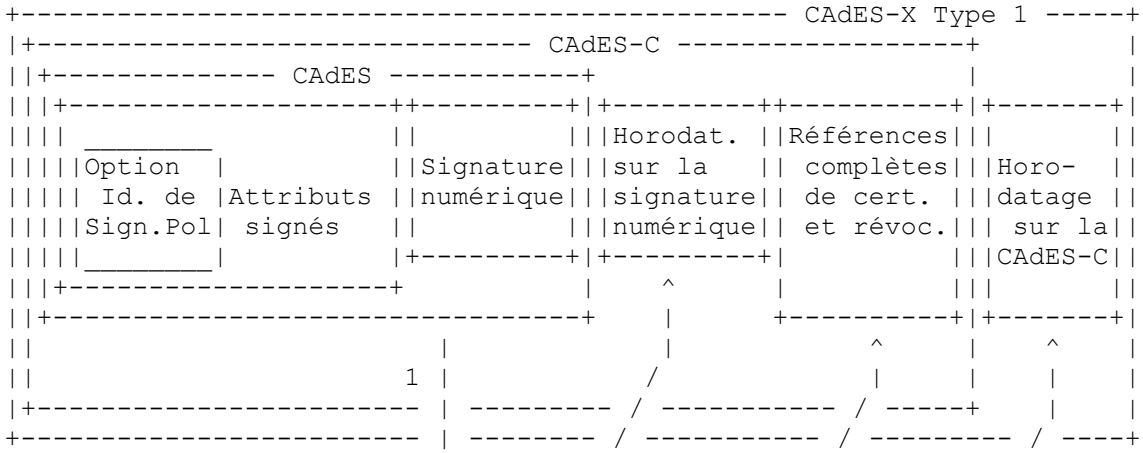


Figure B.7 : Illustration d'une séquence de validation de CAdES avec CAdES-X Long

Quand le processus de validation crée la CAdES-C, il peut aussi créer des formes étendues de données de validation.

Une première solution de remplacement est l'horodatage de toutes les données formant la CAdES-X de type 1.

C'est illustré par la Figure B.8.



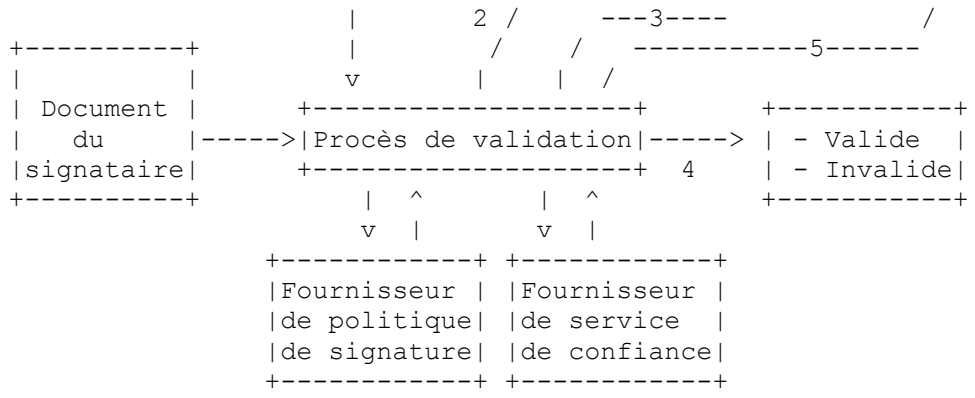


Figure B.8 : Illustration de CAdES avec données de validation étendues de CAdES-X type 1

Une autre solution de remplacement est l'horodatage des références d'informations de certificat et de révocation utilisées pour valider la signature électronique (mais pas la signature) (6). Le résultat final est appelé CAdES-X type 2.

C'est illustré par la Figure B.9.

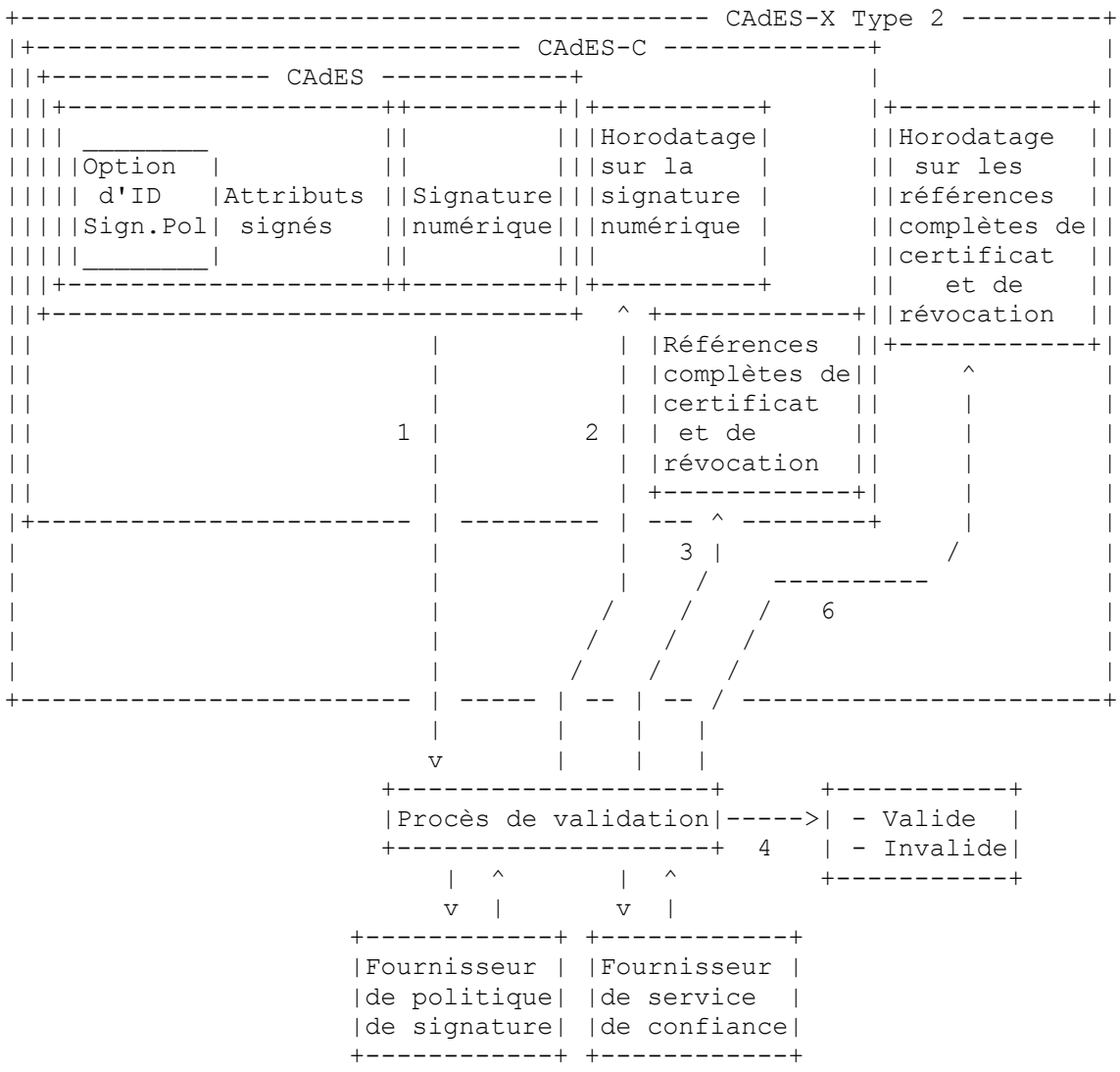


Figure B.9 : Illustration de CAdES avec données de validation étendues CAdES-X de type 2

Avant que les algorithmes utilisés dans une des signatures électroniques soient compromis ou rendus vulnérables, il peut être nécessaire d'horodater la signature électronique entière, incluant toutes les valeurs de la validation et les données d'utilisateur comme ES avec données de validation d'archive (CAdES-A) (7).

Une CAdES-A est illustrée à la Figure B.10.

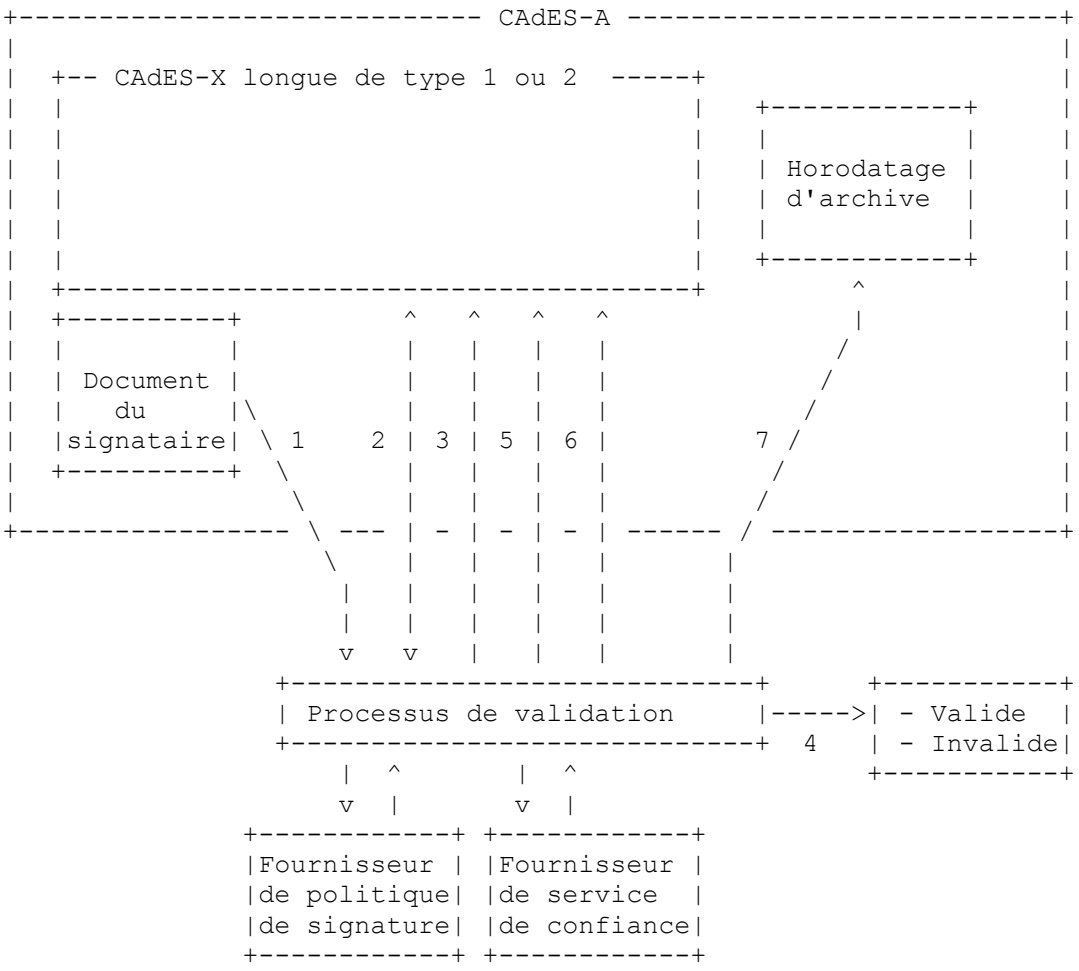


Figure B.10 : Illustration de CAdES-A

B.5. Caractéristiques facultatives supplémentaires

Le présent document définit aussi des caractéristiques supplémentaires facultatives pour :
- indiquer un type d'engagement pris par le signataire,
- indiquer le moment prétendu où la signature a été faite,
- indiquer la localisation prétendue du signataire,
- indiquer le rôle prétendu ou certifié sous lequel une signature a été créée,
- prendre en charge des contre-signatures,
- prendre en charge des signatures multiples.

Annexe C (pour information) : Description générale

Cette annexe explique certains des concepts et fournit la raison des parties normatives du présent document.

La spécification ci-dessous inclut une description de pourquoi et quand chaque composant d'une signature électronique est utile, avec une brève description des vulnérabilités et des menaces et la manière de les contrer.

C.1 Politique de signature

La politique de signature est un ensemble de règles pour la création et la validation d'une signature électronique, sous lesquelles la signature peut être déterminée comme valide. Un certain contexte légal/contractuel peut reconnaître une politique de signature particulière comme satisfaisant ses exigences. Une politique de signature peut être produite, par exemple, par une partie qui s'appuie sur les signatures électroniques et choisie par le signataire pour être utilisée avec cette partie intéressée. Autrement, une politique de signature peut être établie à travers une association de commerce électronique pour l'usage de ses membres. Le signataire et le vérificateur utilisent tous deux la même politique de signature.

La politique de signature peut être explicitement identifiée ou peut être impliquée par la sémantique des données signées et autres données externes, comme une référence à un contrat, qui lui-même se réfère à une politique de signature. Une politique de signature explicite a une référence unique au monde qui est liée à une signature électronique par le signataire au titre du calcul de signature.

La politique de signature doit être disponible sous une forme lisible par l'homme afin qu'il puisse être assuré qu'elle satisfait aux exigences du contexte légal et contractuel dans lequel elle est appliquée. Pour faciliter le traitement automatique d'une signature électronique, les parties de la politique de signature qui spécifient les règles électroniques pour la création et la validation de la signature électronique, doivent aussi être définies intégralement et dans une forme traitable par ordinateur.

La politique de signature inclut donc ce qui suit :
- des règles qui s'appliquent à la validation technique d'une signature particulière,
- des règles qui peuvent être impliquées par l'adoption d'une politique de certificat qui s'applique à la signature électronique (par exemple, des règles pour assurer le secret d'une clé de signature privée),
- des règles qui se rapportent à l'environnement utilisé par le signataire, par exemple, l'accord sur l'utilisation d'un dispositif de validation de carte (CAD, *Card Accepting Device*) en conjonction avec une carte à mémoire.

Par exemple, les règles majeures exigées pour la validation technique peuvent inclure :
- des clés racines reconnues ou "des autorités de certification de niveau supérieur",
- des politiques de certificat acceptables (si il en est),
- les extensions et valeurs de certificat nécessaires (si il en est),
- le besoin de l'état de révocation pour chaque composant de l'arborescence de certification,
- les TSA acceptables (si des jetons d'horodatage sont utilisés),
- les organisations acceptables pour garder les chemins d'audit avec des marques de temps (si des marques de temps sont utilisées),
- les AA acceptables (si il en est d'utilisées), et
- des règles définissant les composants de la signature électronique qui devront être fournis par le signataire avec les données exigées par le vérificateur quand il doit fournir une preuve à long terme.

C.2 Informations signées

Les informations signées peuvent être définies comme un message encapsulé dans MIME qui peut être utilisé pour signaler le format du contenu afin de choisir le bon affichage ou la bonne application. Elles peuvent être composées de données formatées, de texte libre, ou de champs provenant d'un formulaire électronique (e-form). Par exemple, le format "pdf" de Adobe(tm) ou le langage de balisage extensible (XML) peut être utilisé. L'Annexe D définit comment le contenu peut être structuré pour indiquer le type de données signées en utilisant MIME.

C.3 Composants d'une signature électronique

C.3.1 Référence à la politique de signature

Quand deux parties indépendantes veulent évaluer une signature électronique, il est fondamental qu'elles obtiennent le même résultat. Cette exigence peut être satisfaite en utilisant des politiques de signature complètes qui assurent la cohérence de la validation de signature. Les politiques de signature peuvent être identifiées implicitement par les données signées, ou elles peuvent être explicitement identifiées en utilisant la forme CAdES-EPES de signature électronique ; la CAdES-EPES rend obligatoire l'utilisation d'une politique de signature cohérente par le signataire et le vérificateur.

En signant sur l'identifiant de politique de signature dans la CAdES-EPES, le signataire indique explicitement qu'il a appliqué la politique de signature en créant la signature.

Afin d'identifier sans ambiguïté les détails d'une politique de signature explicite à utiliser pour vérifier une CAdES-EPES, la signature, un identifiant, et le hachage de la "politique de signature" doivent faire partie des données signées. Des

informations supplémentaires sur la politique explicite (par exemple, une référence au document sur la Toile) peuvent être portées comme "qualificatif" dans l'identifiant de politique de signature.

Afin d'identifier sans ambiguïté l'autorité responsable de la définition d'une politique de signature explicite, la "politique de signature" peut être signée.

C.3.2 Indication du type d'engagement

Le type d'engagement peut être indiqué dans la signature électronique soit :
- explicitement en utilisant une "indication de type d'engagement" dans la signature électronique,
- implicitement ou explicitement à partir de la sémantique des données signées.

Si le type d'engagement indiqué est explicite en utilisant une "indication de type d'engagement" dans la signature électronique, l'acceptation d'une signature vérifiée implique l'acceptation de la sémantique de ce type d'engagement. La sémantique des indications explicites de type d'engagement peut être soumise à l'accord du signataire et du vérificateur, spécifié au titre de la politique de signature ou enregistrée pour un usage générique à travers plusieurs politiques.

Si un format de signature électronique CAdES-EPES est utilisé et si la signature électronique inclut une indication de type d'engagement autre que une de celles reconnues sous la politique de signature, la signature devra être traitée comme invalide.

Comment l'engagement est indiqué en utilisant la sémantique des données signées sort du domaine d'application du présent document.

Note : des exemples d'engagement indiqué par la sémantique des données signées sont :
- un engagement explicite fait par le signataire indiqué par le type de données signées. Donc, la structure des données signées peut avoir un engagement explicite dans le contexte de l'application (par exemple, un ordre d'achat EDIFACT) ;
- un engagement implicite qui est un engagement fait par le signataire parce que les données signées ont une sémantique (signification) spécifique, qui est seulement interprétable par des humains, (c'est-à-dire, un texte libre).

C.3.3 Identifiant de certificat provenant du signataire

Dans de nombreux environnements réels, les utilisateurs vont être capables d'obtenir de différentes CA ou même de la même CA, différents certificats contenant la même clé publique pour différents noms. Le principal avantage est qu'un utilisateur peut utiliser la même clé privée pour différents objets. Plusieurs utilisations de la clé privée est un avantage quand une carte à mémoire est utilisée pour protéger la clé privée, car la mémorisation d'une carte à mémoire est toujours limitée. Quand plusieurs CA sont impliquées, chaque certificat différent peut contenir une identité différente, par exemple, comme citoyen d'une nation ou comme employé d'une compagnie. Donc, quand une clé privée est utilisée pour divers objets, le certificat est nécessaire pour préciser le contexte dans lequel la clé privée a été utilisée lors de la génération de la signature. Lorsque il y a la possibilité d'utiliser plusieurs clés privées, il est nécessaire que le signataire indique au vérificateur le certificat précis à utiliser.

De nombreux schémas courants ajoutent simplement le certificat après les données signées et sont donc vulnérables à des attaques de substitution. Si le certificat provenant du signataire a simplement été ajouté à la signature et n'est donc pas protégé par la signature, n'importe qui pourrait substituer un certificat à un autre, et le message paraîtrait être signé par quelqu'un d'autre. Afin de contrer cette sorte d'attaque, l'identifiant du signataire doit être protégé par la signature numérique du signataire.

Afin de d'identifier sans ambiguïté le certificat à utiliser pour la vérification de la signature, un identifiant du certificat provenant du signataire devra faire partie des données signées.

C.3.4 Attributs de rôle

Bien que le nom du signataire soit important, la position du signataire au sein d'une compagnie ou d'une organisation est aussi d'une importance capitale. Certaines informations (c'est-à-dire, un contrat) ne peuvent être valides que si elles sont signées par un utilisateur dans un rôle particulier, par exemple, un directeur des ventes. Dans de nombreux cas, qui est réellement le directeur des ventes n'a pas d'importance, mais être sûr que le signataire est habilité par cette compagnie à être le directeur des ventes est fondamental.

Le présent document définit deux façons différentes pour fournir cette caractéristique :
- en mettant un nom de rôle revendiqué dans le champ Attributs signés de CMS,
-en mettant un certificat d'attribut contenant un nom de rôle certifié dans le champ Attributs signés de CMS.

Note : une autre approche possible serait d'utiliser des attributs supplémentaires contenant les noms de rôles dans le certificat d'identité du signataire. Cependant, il a été décidé de ne pas suivre cette approche car elle complique de façon significative la gestion des certificats. Par exemple, utiliser des certificats séparés pour l'identité du signataire et les rôles signifie que de nouvelles clés d'identité n'ont pas besoin d'être produites si le rôle d'un utilisateur change.

C.3.4.1 Rôle revendiqué

Le signataire peut être de confiance pour déclarer son propre rôle sans aucun certificat pour corroborer cette revendication ; dans ce cas, le rôle revendiqué peut être ajouté à la signature comme attribut signé.

C.3.4.2 Rôle certifié

À la différence des certificats de clé publique qui lient un identifiant à une clé publique, les certificats d'attribut lient l'identifiant d'un certificat à certains attributs, comme un rôle. Un certificat d'attribut N'EST PAS produit par une CA mais par une autorité d'attribut (AA, *Attribute Authority*). L'autorité d'attribut, dans la plupart des cas, pourrait être sous le contrôle d'une organisation ou d'une compagnie qui est mieux placée pour savoir quels attributs sont pertinents pour un individu. L'autorité d'attribut peut utiliser ou pointer des certificats de clé publique produits par une CA, pourvu que la confiance appropriée puisse être portée à cette CA. Les certificats d'attribut peuvent avoir des périodes de validité variées. Cette période peut être assez courte, par exemple, un jour. Bien que cela exige qu'un nouveau certificat d'attribut soit obtenu chaque jour, valide pour ce jour, cela peut être avantageux car la révocation de tels certificats peut n'être pas nécessaire. Quand il signe, le signataire va devoir spécifier quel certificat d'attribut il choisit. Pour ce faire, le certificat d'attribut va devoir être inclus dans les données signées afin d'être protégé par la signature numérique du signataire.

Afin d'identifier sans ambigïté le ou les certificats d'attribut à utiliser pour la vérification de la signature, un identifiant du ou des certificats d'attribut provenant du signataire devra faire partie des données signées.

C.3.5 Localisation du signataire

Dans certaines transactions, la localisation prétendue du signataire au moment où il applique sa signature peut devoir être indiquée. Pour cette raison, un indicateur facultatif de localisation devra pouvoir être inclus.

Afin de fournir l'indication de la localisation du signataire au moment de l'application de la signature, un attribut de localisation peut être inclus dans la signature.

C.3.6 Heure de signature

Le présent document fournit la capacité d'inclure un instant prétendu de signature comme attribut d'une signature électronique.

En utilisant cet attribut, un signataire peut signer à un instant qui est l'instant prétendu de signature. Quand une ES avec temps est créée (CAdES-T) un horodatage de confiance est obtenu et ajouté à l'ES ou une marque de temps de confiance existe dans un chemin d'audit. Quand un vérificateur accepte une signature, les deux temps devront être dans des limites acceptables.

Un autre attribut facultatif est défini dans le présent document pour horodater le contenu et pour fournir la preuve de l'existence du contenu à l'instant indiqué par le jeton d'horodatage.

En utilisant cet attribut facultatif, une heure sûre de confiance peut être obtenue avant que le document soit signé et incluse sous la signature numérique. Cette solution exige une connexion en ligne avec un service d'horodatage de confiance avant de générer la signature et peut ne pas représenter l'heure précise de signature, car elle peut être obtenue à l'avance. Cependant, cet attribut facultatif peut être utilisé par le signataire pour prouver que l'objet signé existait avant la date incluse dans l'horodatage (voir le paragraphe 5.11.4).

C.3.7 Format du contenu

Quand on présente des données signées à un utilsateur humain, il peut être important qu'il n'y ait pas d'ambiguïté sur la présentation des informations signées au consommateur d'assertions. Afin que la représentation appropriée (texte, son, ou vidéo) soit choisie par le consommateur d'assertions quand des données (par opposition aux données qui n'ont pas été signées ou chiffrées) sont encapsulées dans le champ SignedData (indiqué par le eContentType au sein du EncapsulatedContentInfo réglé à id-data) d'autres informations de type devraient être utilisées pour identifier le type de document signé. Ceci est généralement réalisé en utilisant le type de contenu MIME et le mécanisme de codage défini dans la [RFC2045]). Plus d'informations sur l'utilisation de MIME sont données dans l'Annexe F.

C.3.8 content-hints

L'attribut contents-hints fournit des informations sur le contenu signé le plus interne d'un message multi couches où un contenu est encapsulé dans un autre. Cela peut être utile si les données signées sont elles-mêmes chiffrées.

C.3.9 Référence croisée de contenu

Quand la présentation de données signées est en relation avec d'autres données signées, il peut être important d'identifier les données signées auxquelles elle se rapporte. Les attributs content-reference et content-identifier, comme défini dans ESS [RFC2634] donnent la capacité de lier les messages de demande et de réponse dans un échange entre deux parties.

C.4 Composants des données de validation

C.4.1 Informations d'état de révocation

Un vérificateur va devoir s'assurer que le certificat du signataire était valide au moment de la signature. Cela peut être fait soit :
- en utilisant des listes de révocation de certificat (CRL, *Certificate Revocation List*) ;
- en utilisant les réponses d'un serveur en ligne d'état de certificat (par exemple, obtenues par le protocole OCSP).

Note 1 : l'instant de la signature peut n'être pas connu, de sorte que l'horodatage ou le marquage en temps peut être utilisé pour fournir l'indication de quand il est connu que la signature existait.

Note 2 : quand on valide une signature électronique et qu'on vérifie les informations d'état de révocation, si une "période de grâce" est exigée, elle doit être d'une longueur convenable pour permettre que l'autorité impliquée traite une demande de révocation de "dernière minute" et pour que la demande se propage à travers le système de révocation. Cette période de grâce est à ajouter au temps inclus avec le jeton d'horodatage ou la marque de temps, et donc les informations d'état de révocation devraient être prises après la fin de la période de grâce.

C.4.1.1 Informations de CRL

Quand on utilise des CRL pour obtenir les informations de révocation, un vérificateur va devoir s'assurer qu'il obtient, au moment de la première vérification, les informations de révocation de certificat appropriées de la CA du signataire. Ce devrait être fait aussitôt que possible pour minimiser le délai entre la génération et la vérification de la signature. Cependant, une "période de grâce" est exigée pour donner aux CA le temps de traiter les demandes de révocation.

Par exemple, une demande de révocation peut arriver à une CA juste avant qu'elle produise la prochaine CRL, et il peut n'y avoir pas assez de temps pour inclure les informations révisées d'état de révocation. Cela implique de vérifier que le numéro de série du certificat du signataire n'est pas inclus dans la CRL. Le signataire, le vérificateur initial, ou un vérificateur ultérieur peut obtenir cette CRL. Si elle est obtenue par le signataire, elle devra alors être portée au vérificateur. Il peut être convenable d'archiver la CRL pour faciliter les vérifications ou arbitrages suivants. Autrement, pourvu que la CRL soit archivée ailleurs, et soit accessible pour les besoins d'arbitrage, alors le numéro de série de la CRL utilisée peut être archivé avec la signature électronique vérifiée comme une forme de CAdES-C.

Même si le numéro de série du certificat apparaît dans la CRL avec l'état "suspendu" (c'est-à-dire, en garde) la signature n'est pas réputée valide car un certificat suspendu n'est pas supposé être utilisé même par son propriétaire légitime.

C.4.1.2 Informations d'OCSP

Quand on utilise OCSP pour obtenir les informations de révocation, un vérificateur va devoir s'assurer qu'il obtient, au moment de la première vérification, une réponse OCSP qui contient l'état "valide". Cela devrait être fait aussitôt que

possible après la génération de la signature, tout en fournissant une "période de grâce" convenable pour permettre à l'autorité impliquée de traiter une demande de révocation de "dernière minute". Le signataire, le vérificateur, ou tous autres tiers peuvent aller chercher cette réponse OCSP. Comme les réponses OCSP sont transitoires et ne sont donc pas archivées par un TSP, ni CA, il est de la responsabilité de chaque vérificateur de s'assurer qu'elles sont mémorisées dans un endroit sûr. La façon la plus simple est de les mémoriser associées à la signature électronique. Une autre solution serait de les mémoriser de telle façon qu'elles puissent être facilement restituées et qu'elles incorporent des réferences à elles dans la signature électronique elle-même comme une forme de CAdES-C.

De la même façon que dans le cas de la CRL, il peut arriver que le certificat soit déclaré invalide mais avec l'état secondaire de "suspendu". Dans ce cas, le même commentaire que pour la CRL s'applique.

C.4.2 Chemin de certification

Un vérificateur peut devoir s'assurer que le chemin de certification était valide au moment de la signature, jusqu'à un point de confiance, en accord avec :
- les contraintes de dénomination,
- les contraintes de politique de certificat,
- la politique de signature, quand applicable.

Comme le moment de la signature ne peut pas être connu avec certitude, une limite supérieure devrait être utilisée comme indiqué par l'horodatage ou la marque de temps.

Dans ce cas, il va être nécessaire de capturer tous les certificats du chemin de certification, en commençant avec ceux du signataire et en terminant par le certificat auto signé d'une racine de confiance ; quand c'est applicable, cela peut être spécifié au titre de la politique de signature. De plus, il va être nécessaire de capturer les listes de révocation d'autorité de certification (CARL, *Certificate Authority Revocation List*) pour prouver qu'aucune des CA de la chaîne n'était révoquée au moment de la signature. Là encore, tout ce matériel peut être incorporé dans la signature électronique (formes ES X). Une solution de remplacement serait de mémoriser ces informations de telle sorte qu'elles puissent être facilement restituées et d'incorporer des références à ces informations dans la signature électronique elle-même sous une forme de CAdES-C.

C.4.3 Horodatage des signatures à longue durée de vie

Une importante propriété pour les signatures à long terme est qu'une signature, ayant été trouvée valide une fois, devra continuer de l'être des mois ou des années plus tard.

Il peut être exigé d'un signataire, d'un vérificateur, ou des deux qu'ils fournissent, à la demande, la preuve qu'une signature numérique a été créée ou vérifiée durant la période de validité de tous les certificats qui constituent le chemin de certification. Dans ce cas, le signataire, le vérificateur, ou les deux vont devoir aussi fournir la preuve que le certificat du signataire et tous les certificats de CA utilisés pour former un chemin de certification valide n'avaient pas été révoqués quand la signature a été créée ou vérifiée.

Il ne serait pas acceptable de considérer une signature comme invalide si les clés ou les certificats ont ensuite été compromis. Donc, il est nécessaire d'être capable de démontrer que les clés de la signature étaient valides au moment de la création de la signature pour fournir la preuve à long terme de la validité d'une signature.

Il se pourrait qu'un certificat ait été valide au moment de la signature mais révoqué plus tard. Dans ce cas, la preuve devra être fournie que le document a été signé avant la révocation de la clé de signature. L'horodatage par une autorité d'horodatage (TSA, *Time-Stamping Authority*) peut fournir cette preuve. Un horodatage est obtenu en envoyant la valeur de hachage des données concernées à la TSA. L'horodatage retourné est un document signé qui contient la valeur du hachage, l'identité de la TSA, et le moment de l'horodatage. Cela prouve que les données concernées existaient avant l'instant du marquage. L'horodatage d'une signature numérique (en envoyant un hachage de la signature à la TSA) avant la révocation de la clé privée du signataire fournit la preuve que la signature avait été créée avant la révocation du certificat.

Si un receveur veut détenir une signature électronique valide, il va devoir s'assurer qu'il a obtenu pour elle un horodatage avant la révocation de cette clé (et de toute clé impliquée dans la validation). Plus tôt l'horodatage est obtenu après l'instant de la signature, meilleur c'est. Tout horodatage ou marque de temps qui est pris après la date d'expiration d'un certificat dans le chemin de certification n'a pas de valeur pour prouver la validité d'une signature.

Il est important de noter que les signatures peuvent être générées "hors ligne" et horodatées plus tard par n'importe qui, par exemple, par le signataire ou tout receveur intéressé par la valeur de la signature. L'horodatage peut donc être fourni par le signataire, avec le document signé, ou obtenu par le receveur suite à la réception du document signé.

L'horodatage N'est PAS un composant de la signature électronique de base, mais il est le composant essentiel de l'ES avec temps.

Il est exigé, dans le présent document, que si la valeur de la signature numérique d'un signataire doit être horodatée, le jeton d'horodatage soit produit par une source de confiance, appelée une autorité d'horodatage.

Le présent document exige que la valeur de la signature numérique du signataire soit horodatée par une source de confiance avant que la signature électronique puisse devenir une ES avec données de validation complètes. Des TSA acceptables peuvent être spécifiées dans une politique de validation de signature.

Cette technique est appelée une CAdES-C dans le présent document.

Si le signataire et le vérificateur devaient exiger que la valeur de l'horodatage de signature satisfasse les exigences de la politique de signature, la politique de signature peut spécifier un délai permis entre les deux horodatages.

C.4.4 Horodatage pour signature de longue durée avant compromission de clé de CA

Les signatures électroniques étendues horodatées sont nécessaires quand il y a une exigence de sauvegarde contre la possibilité de la compromission d'une clé de CA dans la chaîne de certificats. Il peut être exigé d'un vérificateur qu'il fournisse, à la demande, la preuve que le chemin de certification et les informations de révocation utilisées au moment de la signature étaient valides, même dans le cas où une des clés de production ou clé de répondeur OCSP est compromise ultérieurement.

Le présent document définit deux façons d'utiliser les horodatages pour se protéger contre cette compromission :
- l'horodatage de l'ES avec des données de validation complètes, quand une réponse OCSP est utilisée pour obtenir du signataire l'état du certificat (CAdES-X de type 1). Ce format convient pour être utilisé avec une réponse OCSP, et elle offre l'avantage supplémentaire de fournir la protection de l'intégrité sur des données ;
- l'horodatage de seulement le chemin de certification et des références d'informations de révocation quand une CRL est utilisée pour obtenir du signataire l'état du certificat (CAdES-X de type2). Il convient d'utiliser ce format avec les CRL, car les informations horodatées peuvent être utilisées pour plus d'une signature (quand les signataires ont leurs certificats produits par la même CA et quand les signatures peuvent être vérifiées en utilisant les mêmes CRL).

Note : le signataire, le vérificateur, ou les deux peuvent obtenir l'horodatage.

C.4.4.1 Horodatage de l'ES avec données de validation complètes (CAdES-X type 1)

Quand une réponse OCSP est utilisée, il est nécessaire d'horodater en particulier cette réponse dans le cas où la clé provenant du répondeur serait compromise. Comme les informations contenues dans la réponse OCSP sont spécifiques de l'utilisateur et de l'instant, un horodatage individuel est nécessaire pour chaque signature reçue. Au lieu de plaçer l'horodatage seulement sur les références de chemin de certification et les références d'informations de révocation, qui incluent la réponse OCSP, l'horodatage est placé sur la CAdES-C. Comme le chemin de certification et les références d'informations de révocation sont incluses dans l'ES avec données de validation complètes, ils sont aussi protégés. Pour le même prix cryptographique, cela fournit un mécanisme d'intégrité sur l'ES avec données de validation complètes. Toute modification peut être immédiatement détectée. On devrait remarquer que d'autres moyens de protection/détection de l'intégrité de l'ES avec données de validation complètes existent et pourraient être utilisés. Bien que cette technique exige un horodatage pour chaque signature, elle convient bien pour les utilisateurs individuels qui souhaitent avoir une copie protégée en intégrité de toutes les signatures validées qu'ils ont reçu.

En horodatant la signature électronique complète, incluant la signature numérique ainsi que les références aux certificats et aux informations d'état de révocation utilisées pour prendre en charge la validation de cette signature, l'horodatage assure qu'il n'y a pas d'ambiguïté dans les moyens de validation de cette signature.

Cette technique est appelée CAdES-X de type 1 dans le présent document.

Note : la confiance est réalisée dans les références en incluant un hachage des données référencées.

Si il est désiré pour une raison quelconque de garder une copie des données supplémentaires référencées, ces données supplémentaires peuvent être rattachées à la signature électronique, et dans ce cas la signature électronique devient une CAdES-X longue de type 1, comme défini par le présent document.

Une CAdES-X longue de type 1 est simplement l'enchaînement d'une CAdES-X de type 1, avec une copie des données supplémentaires référencées.

C.4.4.2 Horodatages des certificats et références d'informations de révocation (CAdES-X type 2)

Horodater chaque ES avec données de validation complètes, comme défini ci-dessus, peut n'être pas efficace, en particulier quand le même ensemble de certificats de CA et d'informations de CRL est utilisé pour valider de nombreuses signatures.

Horodater les certificats de CA va empêcher tout attaquant de produire des certificats de CA bogués qui pourraient prétendre avoir existé avant que la clé de CA ait été compromise. Tous les certificats de CA horodatés bogués vont montrer que le certificat a été créé après que la clé de CA légitime a été compromise. De la même façon, horodater les CRL de CA va empêcher tout attaquant de produire des CRL de CA boguées qui pourraient prétendre avoir existé avant la compromission de la clé de CA.

L'horodatage des certificats et CRL couramment utilisés peut être fait centralement, par exemple, à l'intérieur d'une compagnie ou par un fournisseur de services. Cette méthode réduit la quantité de données que le vérificateur doit horodater ; par exemple, elle pourrait être réduite à juste un horodatage par jour (c'est-à-dire, dans le cas où tous les signataires utilisent la même CA, et la CRL s'applique pour la journée entière). Les informations qui ont besoin d'être horodatées ne sont pas les certificats et CRL réels, mais les références non ambiguës à ces certificats et CRL.

Cette technique est appelée la CAdES-X de type 2 dans le présent document et exige ce qui suit :
- toutes les références de certificats de CA et les références d'informations de révocation (c'est-à-dire, les CRL) utilisées dans la validation de la CAdES-C sont couvertes par un ou plusieurs horodatages.

Donc, une CAdES-C avec une valeur d'horodatage de signature à l'instant T1 peut être prouvée valide si toutes les références de CA et de CRL sont horodatées à l'instant T1+.

C.4.5 Horodatage des archives de signature

Les avancées des capacités de calcul augmentent la probabilité d'être capable de casser les algorithmes et compromettre les clés. Il y a donc une exigence d'être capable de protéger les signatures électroniques contre cette possibilité.

Sur une longue période, des faiblesses peuvent apparaître dans les algorithmes de chiffrement utilisés pour créer une signature électronique (par exemple, du fait du temps disponible pour la cryptanalyse, ou des améliorations des techniques de cryptanalyse). Avant que de telles faiblesses deviennent probables, un vérificateur devrait prendre des mesures supplémentaires pour maintenir la validité de la signature électronique. Plusieurs techniques pourraient être utilisées pour atteindre ce but, selon la nature de la cryptographie affaiblie. Afin de simplifier les choses, une seule technique appelée archivage des données de validation, couvrant tous les cas, est utilisée dans le présent document.

L'archivage des données de validation consiste en les données de validation et les données complètes de certificat et de révocation, horodatées avec la signature électronique. L'archivage des données de validation est nécessaire si la fonction de hachage et les algorithmes de chiffrement qui ont été utilisés pour créer la signature ne sont plus sûrs. Aussi, si on ne peut pas supposer que la fonction de hachage utilisée par l'autorité d'horodatage est sûre, alors des horodatages incorporés de la signature électronique archivée sont exigés.

L'éventualité d'une compromission de la clé du fournisseur de services de confiance (TSP, *Trusted Service Provider*) devrait être significativement inférieure à celle des clés d'utilisateur parce que les TSP sont supposés utiliser une cryptographie plus forte et une meilleure protection des clés. On peut s'attendre à ce que de nouveaux algorithmes (ou des anciens avec de plus longues clés) soient utilisés. Dans ce cas, une séquence d'horodatages va protéger contre la falsification. Chaque horodatage doit être apposé avant la compromission de la clé de signature ou le cassage des algorithmes utilisés par l'autorité d'horodatage (TSA, *Time-Stamping Authority*). Les TSA devraient avoir des clés longues (par exemple, au moment de la rédaction du présent document c'était d'au moins 2048 bits pour l'algorithme RSA de signature) et/ou un "bon" ou différent algorithme.

Les horodatages incorporés vont aussi protéger le vérificateur contre la compromission des clés ou le craquage de l'algorithme sur les vieilles signatures électroniques.

Le processus va devoir être effectué et itéré avant que les algorithmes de chiffrement utilisés pour générer le précédent horodatage ne soient plus sûrs. L'archivage des données de validation peut donc porter plusieurs horodatages incorporés.

Cette technique est appelée CAdES-A dans le présent document.

C.4.6 Référence à des données supplémentaires

Lorsque ils utilisent les données de validation étendues CAdES-X de type 1 ou CAdES-X de type 2, les vérificateurs ont quand même besoin de garder trace de tous les composants qui ont été utilisés pour valider la signature, afin d'être capables de les restituer plus tard. Ces composants peuvent être archivés par une source externe, comme un fournisseur de service de confiance ; dans ce cas, les informations référencées qui sont fournies au titre de l'ES avec données de validation complètes (CAdES-C) sont adéquates. Les références réelles de certificats et de CRL dans la CAdES-C peuvent être rassemblées quand nécessaire pour un arbitrage.

Si les références aux données supplémentaires ne sont pas adéquates, alors les valeurs réelles de toutes les informations de certificats et de révocation exigées peuvent faire partie de la signature électronique. Cette technique est appelée la CAdES-X longue de type 1 ou CAdES-X longue de type 2 dans le présent document.

C.4.7 Horodatage pour reconnaissance mutuelle

Dans certains scénarios d'affaires, le signataire et le vérificateur ont tous deux besoin d'un horodatage sur leur propre copie de la valeur de signature. Idéalement, les deux horodatages devraient être aussi proches que possible l'un de l'autre.

Exemple : un contrat est signé par deux parties, A et B, représentant leurs organisations respectives ; pour horodater les données du signataire et du vérificateur, deux approches sont possibles :
- dans les termes du contrat, une TSA prédéfinie commune "de confiance" peut être utilisée ;
- si les deux organisations ont leur propre service d'horodatage, A et B peuvent avoir la transaction horodatée par ces deux services d'horodatage.

Dans ce dernier cas, la signature électronique va seulement être considérée comme valide si leurs deux horodatages ont été obtenus en temps utile (c'est-à-dire, il ne devrait pas y avoir un long délai entre l'obtention des deux horodatagess). Donc, ni A ni B ne peuvent répudier l'heure de signature indiquée par leur propre service d'horodatage. Donc, A et B n'ont pas besoin d' un accord sur une TSA commune "de confiance" pour obtenir une transaction valide.

Il est important de noter que les signatures peuvent être générées "hors ligne" et horodatées ensuite par n'importe qui, par exemple, par le signataire ou tout receveur intéressé à la validation de la signature. L'horodatage sur la signature provenant du signataire peut donc être fourni par le signataire, avec le document signé, et/ou être obtenu par le vérificateur suite à la réception du document signé.

Les scénarios d'affaire peuvent donc imposer qu'une ou plusieurs des méthodes d'horodatage de signature à long terme décrites ci-dessus soient utilisées. Cela peut faire partie d'une politique de validation de signature mutuellement acceptée qui fait partie d'un accord de politique de signature selon lequel des signatures numériques peuvent être utilisées pour prendre en charge les relations d'affaires entre les deux parties.

C.4.8 Clé TSA compromise

Les serveurs de TSA devraient être construits d'une façon telle qu'une fois la clé privée de signature installée, il y ait une probabilité minimale de compromission aussi longtemps que possible. Donc, la période de validité des clés de la TSA devrait être aussi longue que possible.

La CAdES-T et la CAdES-C contiennent toutes deux au moins un horodatage sur la signature du signataire. Afin de se protéger contre la compromission de la clé privée de signature utilisée pour produire cet horodatage, l'archivage des données de validation peut être utilisé quand une clé d'autorité d'horodatage différente est impliquée pour produire l'horodatage supplémentaire. Si il est estimé que la clé de TSA utilisée pour produire un horodatage antérieur pourrait être compromise (par exemple, en dehors de sa période de validité) alors la CAdES-A devrait être utilisée. Pour des périodes extrêmement longues, cela peut être appliqué de façon répétée en utilisant de nouvelles clés de TSA.

Cette technique est appelée dans le présent document une CAdES-A incorporée.

C.5 Signatures multiples

Certaines signatures électroniques peuvent seulement être valides si elles portent plus d'une signature. C'est généralement le cas quand un contrat est signé entre deux parties. L'ordre des signatures peut ou non être important, c'est-à-dire, l'une peut ou non devoir appliquée avant les autres.

Plusieurs formes de signatures multiples et contre signatures doivent être prises en charge qui entrent dans deux catégories de base :
- signatures indépendantes ;
- signatures incorporées.

Les signatures indépendantes sont des signatures parallèles où l'ordre des signatures n'est pas important. La capacité d'avoir plus d'une signature indépendante sur les mêmes données devra être fournie.

Les signatures incorporées sont appliquées les unes après les autres et sont utilisées lorsque l'ordre dans lequel les signatures sont appliquées est important. La capacité de signer sur les données signées devra être fournie.

Ces formes sont décrites au paragraphe 5.13. Tous les autres schémas de signatures multiples, par exemple, un document signé avec une contre signature, des doubles contre signatures, ou plusieurs signatures, peuvent être réduits à une ou plusieurs occurrences des deux cas ci-dessus.

Annexe D (pour information) : Protocoles de données pour interopérer avec les TSP

D.1 Protocoles de fonctionnement

Les protocoles suivants peuvent être utilisés par les signataires et vérificateurs pour interopérer avec des fournisseurs de services de confiance durant la création et la validation de signature électronique.

D.1.1 Restitution de certificat

Les certificats d'utilisateur, les certificats de CA, et les certificats croisés peuvent être restitués à partir d'un répertoire en utilisant le protocole léger d'accès à un répertoire (LDAP, *Lightweight Directory Access Protocol*) comme défini dans la [RFC3494], avec le schéma défini dans la [RFC4523].

D.1.2 Restitution de CRL

Les listes de révocation de certificat, incluant les variantes de listes de révocation d'autorité et les CRL partielles, peuvent être restituées à partir d'un répertoire en utilisant le protocole léger d'accès à un répertoire, comme défini dans la [RFC3494], avec le schéma défini dans la [RFC4523].

D.1.3 État de certificate en ligne

Une solution de remplacement à l'utilisation de listes de révocation de certificats est que l'état d'un certificat peut être vérifié en utilisant le protocole d'état de certificat en ligne (OCSP, *Online Certificate Status Protocol*) comme défini dans la [RFC2560].

D.1.4 Horodatage

Le service d'horodatage peut être joint en utilisant le protocole d'horodatage défini dans la [RFC3161].

D.2 Protocoles de gestion

Les signataires et les vérificateurs peuvent utiliser les protocoles de gestion suivants pour gérer l'utilisation des certificats.

D.2.1 Demande de révocation de certificate

La demande qu'un certificat soit révoqué peut être faite en utilisant les messages de demande et réponse de révocation définis dans la [RFC4210].

Annexe E (pour information) : Considérations sur la sécurité

E.1 Protection des clés privées

La sécurité du mécanisme de signature électronique défini dans le présent document dépend de la confidentialité de la clé privée du signataire. Les mises en œuvre devrait prendre des mesures pour s'assurer que les clés privées ne peuvent pas être compromises.

E.2 Choix des algorithmes

Les mises en œuvre devraient savoir que les algorithmes de chiffrement deviennent plus faibles avec le temps. Lorsque de nouvelles techniques de cryptanalyse sont développées et que les performances de calcul s'améliorent, le facteur de travail pour casser un algorithme cryptographique particulier va se réduire. Donc, les mises en œuvre d'algorithme cryptographique devraient être modulaires, permettant que de nouveaux algorithmes soient insérés directement. C'est-à-dire que les mises en œuvres devraient être prêtes à ce que l'ensemble des algorithmes de mise en œuvre obligatoire change avec le temps.

Annexe F (pour information) : Exemple structuré de contenu et MIME

F.1 Utilisation de MIME pour coder les données

Le contenu signé peut être structuré en utilisant MIME (Extensions multi-objets de messagerie Internet [RFC2045]). Alors que la structure MIME a été initialement développée pour la messagerie électronique de l'Internet, elle a un certain nombre de caractéristiques qui la rendent utile pour fournir une structure commune pour le codage d'une gamme de documents électroniques et autres données multimédia (par exemple, des photographies, des vidéos). Ces caractéristiques incluent :
- de fournir un moyen de signaler le type de "l'objet" porté (par exemple, du texte, une image, un fichier ZIP, des données d'application) ;
- de fournir un moyen d'associer un nom de fichier à un objet ;
- d'associer plusieurs objets indépendants (par exemple, un document et une image) pour former un objet multi-parties ;
- de traiter des données codée en texte ou en binaire et, si nécessaire, de recoder le binaire en texte.

Quand on code un seul objet, MIME consiste en :
- informations d'en-tête, suivies par;
- du contenu codé.

Cette structure peut être étendue pour prendre en charge un contenu multi-parties.

F.1.1 Informations d'en-tête

Un en-tête MIME inclut :

Les informations de version MIME : par exemple, MIME-Version: 1.0

Les informations de type de contenu, qui incluent des informations décrivant le contenu, suffisantes pour qu'il soit présenté à un utilisateur ou processus d'application, comme exigé. Cela inclut des informations sur le "type de support" (par exemple, du texte, une image, de l'audio) ou si les données sont à passer à un type particulier d'application. Dans le cas de texte, le type de contenu inclut des informations sur le jeu de caractères utilisé, par exemple, Content-Type: text/plain; charset="us-ascii".

Les informations de codage de contenu, qui définissent comment le contenu est codé (voir ci-dessous sur le codage accepté par MIME).

D'autres informations sur le contenu, comme une description ou un nom de fichier associé.

Un exemple d'en-tête MIME pour un objet de texte est :

```
Mime-Version: 1.0
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: quoted-printable
```

Un exemple d'en-tête MIME pour un fichier binaire contenant un document pdf est :

Content-Type: application/pdf
Content-Transfer-Encoding: base64
Content-Description: JCFV201.pdf
Content-Disposition: filename="JCFV201.pdf"

F.1.2 Codage de contenu

MIME prend en charge une gamme de mécanismes pour coder du texte et des données binaires.

Les données de texte peuvent être portées de façon transparente comme des lignes de données de texte codées en caractères ASCII à 7 ou 8 bits. MIME inclut aussi un codage "quoted-printable" qui convertit les caractères autres que l'ASCII de base en séquence ASCII.

Le binaire peut être porté soit :
- de façon transparente comme octets de 8 bits ;
- converti en ensemble de base de caractères utilisant un système appelé Base64.

Note : comme il y a des relais de messagerie qui peuvent seulement traiter le 7 bits ASCII, le codage Base64 est généralement utilisé sur l'Internet.

F.1.3 Contenu multi-parties

Plusieurs objets (par exemple, du texte et un fichier joint) peuvent être associés en utilisant un type de contenu spécial "multi-part". Ceci est indiqué par le type de contenu "multipart" avec une indication de la chaîne à utiliser qui indique une séparation entre chaque partie.

En plus d'un en-tête pour le contenu multipart global, chaque partie inclut ses propres informations d'en-tête indiquant le type de contenu et le codage internes.

Un exemple de contenu multipart est :

Mime-Version: 1.0
Content-Type: multipart/mixed; boundary="----
=_NextPart_000_01BC4599.98004A80"
Content-Transfer-Encoding: 7bit

------=_NextPart_000_01BC4599.98004A80
Content-Type: text/plain; charset=ISO-8859-1
Content-Transfer-Encoding: 7bit

Suite à votre demande, j'ai joint votre proposition de version de carte Java 2.0 API et le Java Card FAQ.

------=_NextPart_000_01BC4599.98004A80
Content-Type: application/pdf; name="JCFV201.pdf"
Content-Transfer-Encoding: base64
Content-Description: JCFV201.pdf
Content-Disposition: attachment; filename="JCFV201.pdf"

0M8R4KGxGuEAAAAAAAAAAAAAAAAAAAAPgADAP7/CQAGAAAAAAAAAAAAAAACAAAAAgAAAA
AAAAAAEAAAtAAAAAEAAAD+////AAAAAAMAAAAGAAAA//////////////////////////////////////AANhAAQAYg==

------=_NextPart_000_01BC4599.98004A80--

Le contenu multipart peut être incorporé. Ainsi un ensemble d'objets associés (par exemple, de texte HTML et des images) peut être traité comme une seule pièce jointe à un autre objet (par exemple, du texte).

Le type de contenu provenant de chaque partie du message S/MIME indique le type de contenu.

F.2 S/MIME

L'utilisation spécifique de MIME pour porter des données sécurisées par la CMS (étendue comme défini dans le présent document) est appelée S/MIME (voir la [RFC3851]).

S/MIME porte des signatures électroniques comme :
- un objet "application/pkcs7-mime" avec la CMS portée comme une pièce jointe binaire (PKCS7 est le nom de l'ancienne version de la CMS). Les données signées peuvent être incluses dans les données signées, qui elles-mêmes peuvent être incluses dans un seul objet S/MIME. Voir le paragraphe 3.4.2 de la [RFC3851] "Signature avec application/pkcs7-mime avec des données signées" et la Figure F.1 ci-dessous.
- ou un objet "multipart/signed" avec les données signées et la signature codées comme des objets MIME distincts.

Les données signées ne sont pas incluses dans SignedData, et la structure CMS inclut seulement la signature. Voir les paragraphe 3.4.3 de la [RFC3851]"Signature en utilisant le format multipart/signed" et la Figure F.2 ci-après.

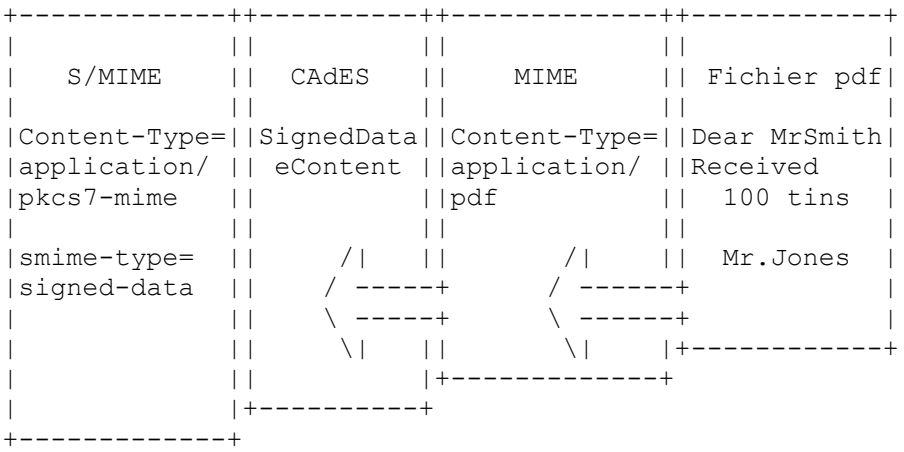


Figure F.1 : Signature utilisant application/pkcs7-mime

F.2.1 Utilisation de application/pkcs7-mime

Cette approche est similaire au traitement des données signées comme tout autre fichier binaire joint.

Un exemple de données signées codées en utilisant cette approche est :

Content-Type: application/pkcs7-mime; smime-type=signed-data;
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7m

567GhIGfHfYT6ghyHhHUujpfyF4f8HHGTrfvhJhjH776tbB9HG4VQbnj777n8HHGT9HG4VQpfyF467GhIGfHfYT6rfvb
nj756tbBghyHhHUujhJhjHHUujhJh4VQpfyF467GhIGfHfYGTrfvbnjT6jH7756tbB9H7n8HHGghyHh6YT64V0GhIGfHf
Qbnj75

F.2.2 Utilisation de application/pkcs7-signature

La CMS prend aussi en charge une autre structure où la signature et les données protégées sont des objets MIME séparés portés au sein d'un seul message. Dans ce cas, les données signées ne sont pas incluses dans SignedData, et la structure CMS inclut seulement la signature. Voir le paragraphe 3.4.3 de la [RFC3851], "Signature utilisant le format multipart/signed" et la Figure F.2 ci-après.

Un exemple de données signées codées en utilisant cette approche est :

Content-Type: multipart/signed;
protocol="application/pkcs7-signature";
micalg=sha1; boundary=boundary42

--boundary42
Content-Type: text/plain

C'est un message signé en clair.

--boundary42

Content-Type: application/pkcs7-signature; name=smime.p7s
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename=smime.p7s

ghyHhHUujhJhjH77n8HHGTrfvbnj756tbB9HG4VQpfyF467GhIGfHfYT64VQpfyF467GhIGfHfYT6jH77n8HHGghyHh
HUujhJh756tbB9HGTrfvbnjn8HHGTrfvhJhjH776tbB9HG4VQbnj7567GhIGfHfYT6ghyHhHUujpfyF47GhIGfHfYT64V
Qbnj756

--boundary42--

Avec cette seconde approche, les données signées passent à travers le processus de CMS et sont portées au titre d'une structure MIME multi-parties signée, comme illustré à la Figure F.2. La structure CMS contient juste la signature électronique.

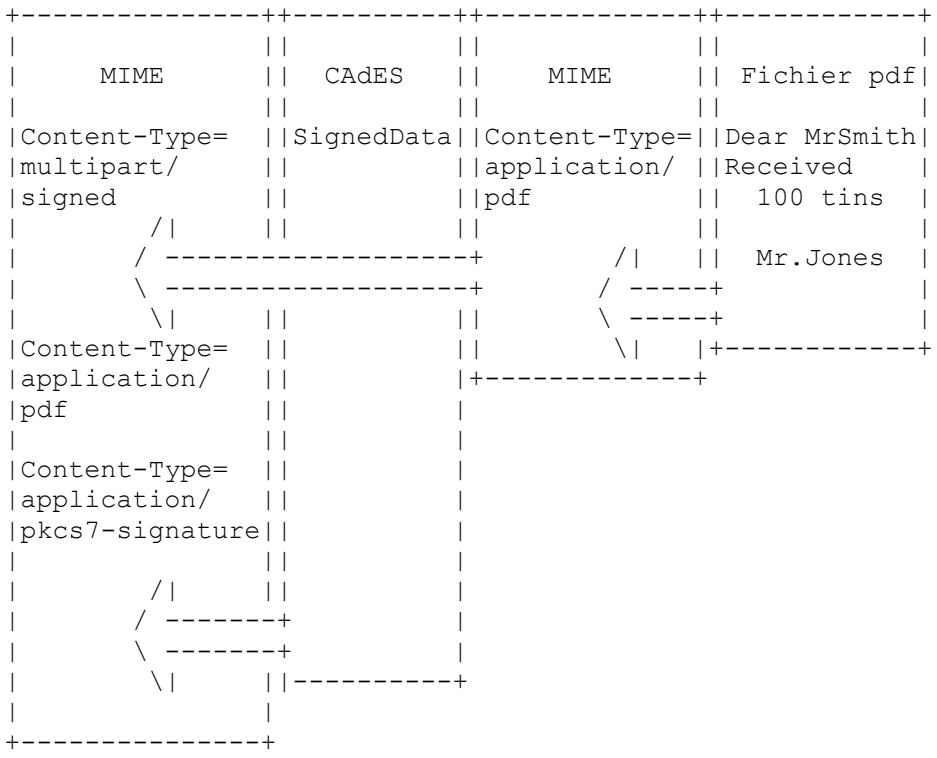


Figure F.2 : Signature utilisant application/pkcs7-signature

Cette seconde approche (multipart/signed) a l'avantage que les données signées peuvent être décodées par tout système compatible avec MIME même si il ne reconnaît pas les signatures électroniques codées avec la CMS.

Annexe G (pour information) : Relations avec la Directive européenne et EESSI

G.1 Introduction

Cette annexe fournit une indication sur les relations entre les signatures électroniques créées sous le présent document et les exigences de la Directive du Parlement Européen et du Conseil sur un cadre communautaire pour les signatures électroniques.

Note : la notice légale devrait être vue comme la législation nationale spécifique concernant l'utilisation des signatures électroniques.

Le présent document fait partie d'un ensemble de normes qui ont été définies sous "l'initiative européenne de normalisation des signatures électroniques" (EESSI) pour les produits et solutions de signature électronique conformes à la Directive européenne pour les signatures électroniques.

G.2 Les signatures électroniques et la directive

Cette directive définit les signatures électroniques comme "des données en forme électronique quii sont attachées ou logiquement associées à d'autres données électroniques et qui servent de méthode d'authentification".

La directive déclare qu'une signature électronique ne devrait pas être refusée comme "moyen légal efficace et admissible ce pruve dans des procédures légales" sur le seule base qu'elle est en forme électronique.

La directive identifie une signature électronique comme ayant une équivalence avec une signature manuscrite si elle satisfait les critéres spécifiques suivants :

- elle est une "signature électronique évoluée" avec les propriétés suivantes :
  a) elle est reliée de façon univoque au signataire ;
  b) elle est capable d'identifier le signataire ;
  c) elle est créée en utilisant des moyens que le signataire peut conserver sous son seul contrôle ;
  d) elle est liée aux données auxquelles elle se rapporte de telle façon que tout changement ultérieur des données est détectable.
- elle se fonde sur un certificat qui satisfait les critères détaillés donnés à l'Annexe I de la directive et est produite par un "fournisseur de service de certification" qui satisfait aux exigences données à l'Annexe II de la directive. Un tel certificat est appelé un "certificat qualifié" ;
- elle est créée par un "appareil", pour lequel des critères détaillés ssont donnés à l'Annexe III de la directive. Un tel appareil est appelé un "appareils sûr de création de signature".

Cette forme de signature électronique est appelée une "signature électronique qualifiée" dans EESSI (voir ci-dessous).

G.3 Les formats de signature électronique ETSI et la directive

Une signature électronique créée en accord avec le présent document est :

a) considérée être une "signature électronique" dans les termes de la Directive ;
b) considérée être une "signature électronique évoluée" dans les termes de la Directive ;
c) considérée être une "signature électronique qualifiée", pourvu que les exigences supplémentaires des Annexes I, II, et III de la Directive soient satisfaites. Les exigences des Annexes I, II, et III de la Directive sortent du domaine d'application du présent document, et sont l'objet de normalisation par ailleurs.

G.4 Standard EESSI et classes de signature électronique

G.4.1 Structure de la normalisation EESSI

EESSI cherche à normaliser plusieurs domaines. Voir les sites de ETSI et du CEN pour la dernière liste de normes et leurs versions :

- utilisation de certificats de clé publique X.509 comme certificats qualifiés,
- gestion de la sécurité et politique de certificat pour les CSP qui produisent des certificats qualifiés,
- exigences de sécurité pour systèmes de confiance utilisés par les CSP qui produisent des certificats qualifiés,
- exigences de sécurité pour les appareils de création de signature sûre,
- exigences de sécurité pour les systèmes de création de signature,
- procédures pour la vérification de signature électronique,
- syntaxe et formats de codage de signature électronique,
- protocole pour interopérer avec une autorité d'horodatage,
- exigences de politique pour les autorités d'horodatage,
- formats XML de signature électronique.

Chacune de ces normes vise une gamme d'exigences, incluant les exigences de signatures électroniques qualifiées, comme spécifié à l'article 5.1 de la Directive. Cependant, certaines d'entre elles visent aussi des exigences générales de signatures électroniques pour les affaires et le commerce électronique, qui entrent toutes dans la catégorie de l'article 5.2 de la Directive. Cette variation dans les exigences peut être identifiée comme des niveaux différents ou comme des options différentes.

G.4.2 Classes de signatures électroniques

Comme certaines de ces normes visent une gamme d'exigences, il peut être utile d'identifier un ensemble de normes pour viser les besoins spécifiques de certaines affaires. Un tel ensemble de normes et leur utilisation définit une classe de signature électronique. La première classe déjà identifiée est la signature électronique qualifiée, qui satisfait les exigences de l'article 5.1 de la Directive.

Un nombre limité de "classes de signatures électroniques" et des profils correspondants pourrait être défini en étroite coopération avec les acteurs du marché (affaires, utilisateurs, fournisseurs). Le besoin de telles normes est envisagé, en plus de celles des signatures électroniques qualifiées, dans des domaines tels que :
- différentes classes de signatures électroniques avec validité à long terme,
- signatures électroniques pour des transactions d'affaire avec valeur limitée.

G.4.3 Classes de signature électronique et format ETSI de signature électronique

Le format de signature électronique défini dans le présent document est applicable au domaine EESSI "signature électronique et formats de codage".

Une signature électronique produite par un signataire (voir la Section 5 et le paragraphe 10.1) est applicable à la classe proposée de signature électronique "signatures électroniques qualifiées satisfaisant à l'article 5.1".

Avec l'ajout d'attributs par le vérificateur (voir la Section 6 et le paragraphe 10.2) la signature électronique qualifiée prend en charge la validité à long terme.

Annexe H (information) : API pour la génération et la vérification des jetons de signatures électroniques

Bien que le présent document décrive les formats de données d'une signature électronique, la question est de savoir si il existe des API (interfaces de programmation d'application) capables de manipuler ces structures. Au moins deux de ces API ont été définies ; une par l'IETF et l'autre par le groupe de gestion d'objet (OMG, *Object Management Group)*.

H.1 Tramage des données

Pour être capable d'utiliser l'une ou l'autre de ces API, il va être nécessaire de tramer la structure de données de signature électronique définie précédemment en utilisant un format de jeton indépendant du mécanisme. Le paragraphe 3.1 de la [RFC2743] spécifie un niveau indépendant du mécanisme de représentation encapsulante pour le jeton initial d'une séquence d'établissement de contexte d'une GSS-API, incorporant un identifiant du type de mécanisme à utiliser sur ce contexte et permettant que les jetons soient interprétés sans ambiguïté.

Pour qu'ils puissent être traités par ces API, tous les formats de données de signature électronique qui sont définis dans le présent document devront être tramés conformément à cette description.

Le format de codage de l'étiquette jeton est dérivé de l'ASN.1 et du DER, mais sa représentation concrète est définie directement en termes d'octets plutôt qu'au niveau ASN.1, afin de faciliter l'interopérabilité de mise en œuvre sans utiliser de code général de traitement ASN.1. L'étiquette jeton consiste en les éléments suivants, dans cet ordre :

1) 0x60 : étiquette pour SEQUENCE de la RFC 2743 ; indique que la forme construite, de codage de longueur définie, suit.

2) octets de longueur de jeton, spécifiant la longueur des données qui suivent (c'est-à-dire, le total des longueurs des éléments 3 à 5 de cette liste, et l'objet jeton défini par le mécanisme qui suit l'étiquette). Cet élément comporte un nombre d'octets variable :
   a) si la valeur indiquée est de moins de 128, elle devra être représentée dans un seul octet avec le bit 8 (de poids fort) réglé à "0" et le reste des bits représentant la valeur ;
   b) si la valeur indiquée est 128 ou plus, elle devra être représentée en deux octets ou plus, avec le bit 8 du premier octet réglé à "1" et les bits restants du premier octet spécifiant le nombre d'octets supplémentaires. Les octets suivants portent la valeur, 8 bits par octet, avec le chiffre de poids fort en premier. Le nombre minimum d'octets devra être utilisé pour coder la longueur (c'est-à-dire, aucun octet représentant des zéros en tête ne devra être inclus dans le codage de longueur).

3) 0x06 : étiquette pour IDENTIFIANT D'OBJET.

4) longueur d'identifiant d'objet : longueur (nombre d'octets) de l'identifiant d'objet codé contenu dans l'élément 5, codé selon les règles décrites au 2a) et 2b) ci-dessus.

5) octets de l'identifiant d'objet : nombre d'octets variable, codé selon les règles BER d'ASN.1 :
   - le premier octet contient la somme de deux valeurs :
   (1) le composant de niveau supérieur de l'identifiant d'objet, multiplié par 40 (en décimal) et
   (2) le composant de second niveau de l'identifiant d'objet. Ce cas pariculier est le seul point au sein du codage d'un identifiant d'objet où un seul octet représente le contenu de plus d'un composant.
   - les octets suivants, si ils sont exigés, codent successivement les composants inférieurs dans l'identifiant d'objet représenté. Le codage d'un composant peut s'étendre sur plusieurs octets, codant 7 bits par octet (bit de poids fort en premier) et avec le bit 8 réglé à "1" sur tous les octets sauf le dernier du codage du composant. Le nombre minimum d'octets devra être utilisé pour coder chaque composant (c'est-à-dire, aucun octet représentant les zéros en tête ne devra être inclus dans le codage d'un composant).

Note : dans de nombreuses mises en œuvre, les éléments 3 à 5 peuvent être mémorisés et référencés comme une chaîne constante contiguë.

L'étiquette de jeton est immédiatement suivie par un objet jeton défini par le mécanisme. Noter qu'aucun spécificateur de taille indépendant n'intervient à la suite de la valeur d'identifiant d'objet pour indiquer la taille de l'objet jeton défini par le mécanisme.

Les jetons conformes au présent document devront avoir l'OID suivant afin d'être traités par les API IDUP :

IDENTIFIANT D'OBJET id-etsi-es-IDUP-Mechanism-v1 ::= { itu-t(0) identified-organization(4) etsi(0) electronic-signature-standard (1733) part1 (1) IDUPMechanism (4) etsiESv1(1) }

H.2 IDUP-GSS-API définies par l'IETF

Le groupe de travail CAT de l'IETF a produit, en décembre 1998, la [RFC2479] sous le nom de IDUP-GSS-API (protection d'unité de données indépendantes) capable de traiter le formatde données de signature électronique défini dans le présent document. L'API GSS IDUP inclut la prise en charge de services de non répudiation.

Elle prend en charge la génération de preuves, où les "preuves" sont des informations qui par elles-même, ou quand elles sont utilisées en conjonction avec d'autres informations, sont utilisées pour établir la preuve d'un événement ou action, ainsi que la vérification de la preuve.

IDUP prend en charge divers types de preuves. Tous les types définis dans IDUP sont pris en charge dans le présent document par le paramètre commitment-type *(type d'engagement)*.

Le paragraphe 2.3.3 de IDUP décrit les appels spécifiques nécessaires pour traiter les preuves (appels "EV"). Le groupe d'appels "EV" fournit une interface simple, de haut niveau avec les mécanismes IDUP sous-jacents quand des développeurs d'application ont besoin de traiter seulement la preuve, sans service de chiffrement ou d'intégrité.

Toutes les générations et vérifications sont effectuées en accord avec le contenu d'une politique de NR qui est référencée dans le contexte.

Get_token_details est utilisé pour retourner les attributs qui correspondent à un jeton d'entrée donné à une application. Comme les jetons IDUP-GSS-API sont destinés à être opaques à l'application appelante, cette fonction permet à l'application de déterminer les informations sur le jeton sans avoir à violer l'intention d'opacité de l'IDUP. Le type de mécanisme est d'une importance capitale, et l'application peut alors l'utiliser comme entrée à l'appel IDUP_Establish_Env() afin d'établir l'environnement correct dans lequel traiter le jeton.

Generate_token génère un jeton de non répudiation qui va utiliser l'environnement courant.

Verify_evidence vérifie le jeton de preuve en utilisant l'environnement courant. Cette opération retourne un code major_status qui peut être utilisé pour déterminer si la preuve contenue dans un jeton est complète (c'est-à-dire, peut être vérifiée avec succès plus tard (peut-être des années après)). Si la preuve d'un jeton n'est pas complète, le jeton peut être passé à une autre API, form_complete_pidu, pour la compléter. Cela arrive quand un état "conditionnellement valide" est retourné. Cet état correspond à l'état "validation incomplète" du présent document.

Form_complete_PIDU est utilisé principalement quand le jeton de preuve lui-même ne contient pas toutes les données exigées pour sa vérification, et il est prévu que certaines des données non mémorisées dans le jeton peuvent devenir inaccessibles durant l'intervalle entre la génération du jeton de preuve et la vérification sauf si elles sont mémorisées dans le jeton. L'opération Form_Complete_PIDU rassemble les informations manquantes et les inclut dans le jeton afin qu'il soit garanti que la vérification peut être possible à tout instant futur.

H.3. Interfaces de sécurité CORBA définie par l'OMG

Les interfaces de non répudiation ont été définies dans "CORBA Security", un document produit par le groupe de gestion d'objet (OMG, O*bject Management Group*). Ces interfaces sont décrites en langage de définition d'interface (IDL, *Interface Definition Language*) et sont facultatives.

Le traitement des "jetons" qui prennent en charge la non répudiation est fait à travers les interfaces suivantes :
- set_NR_features spécifie les caractéristiques à appliquer aux opérations futures de génération et vérification de preuve ;
- get_NR_features retourne les caractéristiques qui vont être appliquées aux futures opérations de génération et vérification de preuve ;
- generate_token génère un jeton de non répudiation utilisant les caractéristiques de non répudiation courantes ;
- verify_evidence vérifie le jeton de preuve en utilisant les caractéristiques de non répudiation courantes ;
- get_tokens_details retourne les informations sur un jeton de non répudiation d'entrée. Les informations retournées dépendent du type de jeton ;
- form_complete_evidence est utilisé quand le jeton de preuve lui-même ne contient pas toutes les données exigées pour sa vérification, et il est prévu que certaines des données non mémorisées dans le jeton peuvent devenir inaccessibles durant l'intervalle entre la génération du jeton de preuve et la vérification sauf si elles sont mémorisées dans le jeton. L'opération form_complete_evidence rassemble les informations manquantes et les inclut dans le jeton afin qu'il puisse être garanti que la vérification sera possible à tout instant futur.

Note : la similarité entre les deux ensembles d'API est remarquable.

Annexe I (information) : Algorithmes de chiffrement

La [RFC3370] décrit les conventions pour utiliser plusieurs algorithmes de chiffrement avec tla syntaxe de message crytographique (CMS). Seuls les algorithmes de hachage et de signature sont appropriés pour le présent document.

Depuis la publication de la [RFC3370], MD5 a été cassé. Cet algorithme n'est plus considéré comme approprié et a été supprimé de la liste des algorithmes.

I.1 Algorithmes de résumé

I.1.1 SHA-1

L'algorithme de résumé SHA-1 est défini dans la publication FIPS 180-1. L'identifiant d'algorithme pour SHA-1 est :

IDENTIFIANT D'OBJET sha-1 ::= { iso(1) identified-organization(3) oiw(14) secsig(3) algorithm(2) 26 }

Le champ de paramètre AlgorithmIdentifier est facultatif. Si il est présent, le champ de paramètres devra contenir un ASN.1 NULL. Les mises en œuvre devrait accepter des identifiants d'algorithme SHA-1 avec les paramètres absents ainsi qu'avec les paramètres NULL. Les mises en œuvre devraient générer les identifiants d'algorithme SHA-1 avec les paramètres NULL.

I.1.2 Généralités

Ci-dessous est une sélection de travaux faits dans le domaine des algorithmes de résumé ou, comme ils sont souvent appelés, des fonctions de hachage :

- ISO/CEI 10118-1 (1994) [ISO10118-1]: "Technologies de l'information - techniques de sécurité - fonctions de hachage - Partie 1 : Généralités". ISO/CEI 10118-1 contient les définitions et décrit les concepts de base.

- ISO/CEI 10118-2 (1994) [ISO10118-2]: "Technologies de l'information - techniques de sécurité - fonctions de hachage - Partie 2 : fonctions de hachage utilisant un algorithme de chiffrement de bloc de n bits". ISO/CEI 10118-2 spécifie deux façons de construire une fonction de hachage à partir d'un chiffrement de bloc.

- ISO/CEI 10118-3 (1997) [ISO10118-3]: "Technologies de l'information - techniques de sécurité - fonctions de hachage - Partie 3 : fonctions de hachage dédiées". ISO/CEI 10118-3 spécifie les fonctions de hachage dédiées suivantes :
  - SHA-1 (FIPS 180-1);
  - RIPEMD-128;
  - RIPEMD-160.

- ISO/CEI 10118-4 (1998) [ISO10118-4]: "Technologies de l'information - techniques de sécurité - fonctions de hachage - Partie 4 : fonctions de hachage utilisant une arithmétique modulaire".

- RFC 1320 (1992) : "Algorithme de résumé de message MD4". La RFC 1320 spécifie la fonction de hachage MD4. Aujourd'hui, MD4 est considéré comme périmé.

- RFC 1321 (1992) : "Algorithme de résumé de message MD5". La RFC 1321 (pour information) spécifie la fonction de hachage MD5. Aujourd'hui, MD5 n'est pas recommandé pour les nouvelles mises en œuvre.

- Publication FIPS 180-1 (1995): "Norme de hachage sûr". FIPS 180-1 spécifie l'algorithme de hachage sûr (SHA), une fonction de hachage dédiée développée pour être utilisée avec DSA. Le SHA d'origine, publié en 1993, a été légèrement révisé en 1995 et renommé SHA-1.

- ANSI X9.30-2 (1997) [X9.30-2]: "Public Key Cryptography for the Financial Services Industry - Part 2: The Secure Hash Algorithm (SHA-1)". X9.30-2 spécifie la version ANSI de SHA-1.

- ANSI X9.31-2 (1996) [X9.31-2]: "Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry - Part 2: Hash Algorithms". X9.31-2 spécifie les algorithmes de hachage.

I.2 Algorithmes de signature numérique

I.2.1 DSA

L'algorithme de signature DSA est défini dans FIPS Pub 186. DSA est toujours utilisé avec l'algorithme de résumé de message SHA-1. L'identifiant d'algorithme pour DSA est :

IDENTIFIANT D'OBJET id-dsa-with-sha1 ::= { iso(1) member-body(2) us(840) x9-57 (10040) x9cm(4) 3 }

Le champ de paramètres AlgorithmIdentifier ne devra pas être présent.

I.2.2 RSA

L'algorithme de signature RSA est défini dans la [RFC3447]. La [RFC3370] spécifie l'utilisation de l'algorithme de signature RSA avec l'algorithme SHA-1. L'identifiant d'algorithme pour RSA avec SHA-1 est :

IDENTIFIANT D'OBJET Sha1WithRSAEncryption ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 5 }

Note : la [RFC3370] recommande que MD5 ne soit pas utilisé pour les nouvelles mises en œuvre.

I.2.3 Généralités

Voici un choix de travaux qui ont été faits dans le domaine des mécanismes de signature numérique :

- FIPS Publication 186 (1994): "Digital Signature Standard". L'algorithme de signature numérique du NIST (DSA) est une variante du mécanisme de signature numérique fondée sur le logarithme discret de ElGamal. DSA exige une fonction de hachage de 160 bits et rend SHA-1 obligatoire.

- IEEE P1363 (2000) [P1363]: "Standard Specifications for Public-Key Cryptography". IEEE P1363 contient des mécanismes pour les signatures numériques, l'établissement de clés, et le chiffrement fondé sur trois familles de schémas de clé publique :
  - techniques "conventionnelles" de logarithme discret (DL) c'est-à-dire, accord de clé Diffie-Hellman (DH), accord de clé Menezes-Qu-Vanstone (MQV), algorithme de signature numérique (DSA), et signatures numériques Nyberg-Rueppel (NR) ;

- variantes fondées sur la courbe elliptique (EC) des mécanismes de DL spécifiés ci-dessus, c'est-à-dire, EC-DH, EC-MQV, EC-DSA, et EC-NR. Pour les courbes elliptiques, les options de mise en œuvre incluent mod p et la caractéristique 2 avec représentation en base polynomiale ou normale ;
- techniques fondées sur la factorisation d'entier, incluant le chiffrement RSA, les signatures numériques RSA, et le transport de clé fondé sur RSA.

- ISO/CEI 9796-2 (1997) [ISO9796-2]: "Technologies de l'information - techniques de sécurité - schémas de signature numérique donnant la récupération de message - Partie 2 : mécanismes utilisant une fonction de hachage". ISO/CEI 9796-2 spécifie des mécanismes de signature numérique avec récupération partielle de message qui se fondent aussi sur la technique RSA mais utilisent une fonction de hachage.

- ISO/CEI 9796-4 (1998) [ISO9796-4] : "schémas de signature numérique donnant la récupération de message - Partie 4: mécanismes fondés sur le logarithme discret". ISO/CEI 9796-4 spécifie des mécanismes de signature numérique avec récupération partielle de message qui se fondent sur les techniques de logarithme discret. Le document inclut le schéma de Nyberg-Rueppel.

- ISO/CEI 14888-1 [ISO14888-1]: "Signatures numériques avec appendice - Partie 1 : Généralités". ISO/CEI 14888-1 contient les définitions et décrit les concepts de base des signatures numériques avec appendice.

- ISO/CEI 14888-2 [ISO14888-2]: "Signatures numériques avec appendice - Partie 2 : mécanismes fondés sur l'identité". ISO/CEI 14888-2 spécifie des schémas de signature numérique avec appendice qui utilisent du matériel de chiffrement fondé sur l'identité. Le document inclut les techniques de connaissance zéro de Fiat-Shamir et Guillou-Quisquater.

- ISO/CEI 14888-3 [ISO14888-3]: "Signatures numériques avec appendice - Partie 3 : mécanismes fondés sur le certificat". ISO/CEI 14888-3 spécifie des schémas de signature numérique avec appendice qui utilisent du matériel de chiffrement fondé sur le certificat. Le document inclut cinq schémas :
  - DSA,
  - EC-DSA, analogue fondé sur la courbe elliptique de l'algorithme de signature numérique du NIST,
  - signatures Pointcheval-Vaudeney,
  - signatures RSA,
  - ESIGN.

- ISO/CEI 15946-2 (2002) [ISO15946-2] : "Techniques cryptographiques fondées sur les courbes elliptiques - Partie 2 : signatures numériques", spécifie des schémas de signature numérique avec appendice qui utilisent des courbes elliptiques. Le document inclut deux schémas :
  - EC-DSA, analogue fondé sur la courbe elliptique de l'algorithme de signature numérique du NIST,
  - EC-AMV, analogue fondé sur la courbe elliptique de l'algorithme de signature Agnew-Muller-Vanstone.

- ANSI X9.31-1 (1997) [X9.31-1]: "Public Key Cryptography Using Reversible Algorithms for the Financial Services Industry - Part 1: The RSA Signature Algorithm". ANSI X9.31-1 spécifie un mécanisme de signature numérique avec appendice utilisant la technique de clé publique RSA.

- ANSI X9.30-1 (1997) [X9.30-1]: "Public Key Cryptography Using Irreversible Algorithms for the Financial Services Industry - Part 1: The Digital Signature Algorithm (DSA)". ANSI X9.30-1 spécifie DSA, l'algorithme de signature numérique du NIST.

- ANSI X9.62 (1998) [X9.62]: "Public Key Cryptography for the Financial Services Industry - The Elliptic Curve Digital Signature Algorithm (ECDSA)". ANSI X9.62 spécifie l'algorithme de signature numérique à courbe elliptique, analogue à l'algorithme de signature numérique du NIST (DSA) utilisant des courbes elliptiques. Les appendices donnent des informations sur les mathématiques sous-jacentes pour la cryptographie à courbes elliptiques et donnent de nombreux exemples.

Annexe J (information) : Lignes directrices pour les désignations

J.1 Allocation de noms

Le nom du sujet devra être alloué par un schéma d'enregistrement administré par une autorité d'enregistrement (RA) pour assurer l'unicité. Cette RA peut être un corps indépendant ou une fonction assumée par une autorité de certification.

En plus d'assurer l'unicité, la RA devra vérifier que le nom alloué identifie de façon appropriiée le demandeur et que des vérifications d'authentification sont effectuées pour protéger contre l'usurpation.

Le nom alloué par une RA se fonde sur les informations d'enregistrement fournies par le demandeur, ou relatives au demandeur (par exemple, son nom personnel, sa date de naissance, l'adresse de sa résidence) et sur des informations allouées par la RA. Trois variantes existent couramment :

- le nom se fonde entièrement sur les informations d'enregistrement, qui l'identifient de façon univoque (par exemple, "Pierre Durand (né le) 6 juillet 1956") ;
- le nom se fonde sur des informations d'enregistrement, avec l'ajout de qualificatifs par l'autorité d'enregistrement pour assurer l'unicité (par exemple, "Pierre Durand 12") ;
- les informations d'enregistrement restent confidentielles pour l'autorité d'enregistrement et elle alloue un "pseudonyme".

J.2 Fourniture de l'accès aux information s d'enregistrement

Dans certaines circonstances, il peut être nécessaire que les informations utilisées durant l'enregistrement, mais non publiées dans le certificat, soient rendues disponibles à des tiers (par exemple, à un arbitre pour résoudre une dispute ou pour l'application de la loi). Ces informations d'enregistrement vont probablement inclure des informations personnelles et sensibles.

Donc, la RA a besoin d'établir une politique pour :
- si les informations d'enregistrement devraient être divulguées,
- à qui de telles informations devraient être divulguées,
- dans quelles circonstances de telles informations devraient être divulguées.

Cette politique peut être différente selon que la RA est utilisée seulement au sein d'une entreprise ou pour l'usage du public. La politique va devoir prendre en compte la règlementation nationale et en particulier sur la protection et la confidentialité des données.

Actuellement, la fourniture de l'accès à l'enregistrement est une affaire locale pour la RA. Cependant, si l'accès ouvert est exigé, des protocoles standard, comme HTTP -- RFC 2068 (protocole de transfert Hypertext) peuvent être employés avec l'ajout des mécanismes de sécurité nécessaires pour satisfaire les exigences de protection des données (par exemple, Sécurité de la couche transport [RFC4346]) avec authentification du client.

J.3 Schémas de désignation

J.3.1 Schémas de dénomination des citoyens individuels

Dans certains cas, le nom du sujet qui est contenu dans un certificat de clé publique peut n'être pas assez significatif. Cela peut arriver parce que il existe des homonymes ou parce que on utilise des pseudonymes. Une distinction pourrait être faite si plus d'attributs étaient présents. Cependant, ajouter plus d'attributs à un certificat de clé publique placé dans un répertoire public irait à l'encontre des exigences de protection de la confidentialité.

Dans tous les cas, l'autorité d'enregistrement va obtenir les informations au moment de l'enregistrement, mais toutes les informations ne vont pas être placées dans le certificat. Afin de réaliser un équilibre entre ces deux exigences opposées, les valeurs de hachage de certains attributs supplémentaires peuvent être placées dans un certificat de clé publique. Quand le possesseur du certificat fournit ces attributs supplémentaires, ils peuvent alors être vérifiés. Utiliser des attributs biométriques peut identifier sans ambiguïté une personne. Des exemples d'attributs biométriques qui peuvent être utilisés incluent des images ou une signature manuelle du possesseur du certificat.

Note : Utiliser des valeurs de hachage ne protège la confidentialité que si les entrées possibles sont assez grandes. Par exemple, utiliser le hachage du numéro de sécurité sociale d'une personne n'est généralement pas suffisant car il peut facilement être inversé.

Une image peut être utilisée si le vérificateur a rencontré une fois la personne et veut vérifier ultérieurement que le certificat qu'il a se rapporte bien à la personne qu'il a rencontré. Dans ce cas, au premier échange, l'image est envoyée, et le hachage contenu dans le certificat peut être utilisé par le vérificateur pour vérifier qu'il est la bonne personne. Au prochain échange, l'image n'a pas besoin d'être envoyée à nouveau.

Une signature manuelle peut être utilisée si un document signé a été reçu auparavant. Dans ce cas, au premier échange, le dessin de la signature manuelle est envoyé, et le hachage contenu dans le certificat peut être utilisé par le vérificateur pour vérifier que c'est la bonne signature manuelle. À l'échange suivant, la signature manuelle n'a pas besoin d'être envoyée à nouveau.

J.3.2 Schémas de dénomination des employés d'une organisation

Le nom d'un employé d'une organisation va probablement être une combinaison du nom de l'organisation et l'identifiant de l'employé au sein de cette organisation. Un nom d'organisation est généralement un nom enregistré, c'est-à-dire, un nom commercial ou d'affaire utilisé dans les affaires quotidiennes. Ce nom est enregistré par une autorité de désignation, qui garantit que le nom enregistré de l'organisation n'est pas ambigu et ne peut pas être confondu avec celui d'une autre organisation.

Afin d'avoir plus d'informations sur le nom enregistré d'une organisation, il est nécessaire de revenir à un répertoire publiquement disponible tenu par l'autorité de désignation.

L'identifiant peut être un nom ou un pseudonyme (par exemple, un surnom ou un numéro d'employé). Quand c'est un nom, il est supposé être assez descriptif pour identifier sans ambiguïté la personne. Quand c'est un pseudonyme, le certificat ne divulgue pas l'identité de la personne. Cependant, il assure que la personne a été correctement authentifiée au moment de l'enregistrement et donc peut être eligible à certains avantages obtenus implicitement ou explicitement par la possession du certificat. Dans les deux cas, cependant, cela peut être insuffisant à cause de l'existence d'homonymes.

Placer plus d'attributs dans le certificat peut être une solution, par exemple, en donnant l'unité d'organisation de la personne ou le nom d'une ville où est situé le bureau. Cependant, plus on place d'informations dans le certificat, plus il y a de problèmes si il y a un changement dans la structure de l'organisation ou le lieu de travail. Ce peut n'être donc pas la meilleure solution. Une solution de remplacement est de fournir plus d'attributs (comme l'unité d'organisation et le lieu de travail) par l'accès à un répertoire tenu par l'entreprise. Il est probable que, au moment de l'enregistrement, l'autorité d'enregistrement obtiendra plus d'informations que ce qui a été placé dans le certificat, si de telles informations supplémentaires sont placées dans un répertoire accessible seulement à l'entreprise.

Remerciements

Des remerciements particuliers sont dus à Russ Housley pour sa relecture du document.

Adresse des auteurs

John Ross
Security & Standards Consultancy Ltd
The Waterhouse Business Centre
2 Cromer Way
Chelmsford
Essex
CM1 2QE
United Kingdom
mél : ross@secstan.com

Nick Pope
Thales eSecurity
Meadow View House
Long Crendon
Aylesbury
Buck
HP18 9EQ
United Kingdom
mél : nick.pope@thales-esecurity.com

Denis Pinkas
Bull SAS
Rue Jean-Jaures
78340 Les Clayes sous Bois CEDEX
FRANCE
mél : Denis.Pinkas@bull.net

Déclaration complète de droits de reproduction

Des copies des dépôts d'IPR faites au secrétariat de l'IETF et toutes assurances de disponibilité de licences, ou le résultat de tentatives faites pour obtenir une licence ou permission générale d'utilisation de tels droits de propriété par ceux qui mettent en œuvre ou utilisent la présente spécification peuvent être obtenues sur le répertoire en ligne des IPR de l'IETF à http://www.ietf.org/ipr.

L'IETF invite toute partie intéressée à porter son attention sur tous copyrights, licences ou applications de licence, ou autres droits de propriété qui pourraient couvrir les technologies qui peuvent être nécessaires pour mettre en œuvre la présente norme. Prière d'adresser les informations à l'IETF à ietf-ipr@ietf.org.