Groupe de travail Réseau
Request for Comments : 5480
 RFC mise à jour : 3279
 Catégorie : Sur la voie de la normalisation
 Traduction Claude Brière de L'Isle

S. Turner, IECA
 D. Brown, Certicom
 K. Yiu, Microsoft
 R. Housley, Vigil Security
 T. Polk, NIST
 mars 2009

Syntaxe et sémantique du champ subjectPublicKeyInfo dans la cryptographie à courbe elliptique (ECC)

Statut du présent mémoire

Le présent document spécifie un protocole de l'Internet sur la voie de la normalisation pour la communauté de l'Internet, et appelle à des discussions et suggestions pour son amélioration. Prière de se référer à l'édition en cours des "Protocoles officiels de l'Internet" (STD 1) pour voir l'état de normalisation et le statut de ce protocole. La distribution du présent mémoire n'est soumise à aucune restriction.

Notice de droits de reproduction

Copyright (c) 2009 IETF Trust et les personnes identifiées comme auteurs du document. Tous droits réservés.

Le présent document est soumis au BCP 78 et aux dispositions légales de l'IETF Trust qui se rapportent aux documents de l'IETF (<http://trustee.ietf.org/license-info>) en vigueur à la date de publication de ce document. Prière de revoir ces documents avec attention, car ils décrivent vos droits et obligations par rapport à ce document.

Résumé

Le présent document spécifie la syntaxe et la sémantique du champ Informations de clé publique sujette dans les certificats qui prennent en charge la cryptographie à courbe elliptique. Le présent document met à jour le paragraphe 2.3.5, la section 5, et le module ASN.1 de la [RFC3279] "Algorithmes et identifiants pour le profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet".

Table des matières

1. Introduction.....	1
1.1 Terminologie.....	2
2. Champs d'information de clé publique sujette.....	2
2.1 Identifiants d'algorithme de clé publique de cryptographie à courbe elliptique.....	2
2.2 Clé publique sujette.....	4
3. Bits d'usage de clé.....	4
4. Considérations sur la sécurité.....	5
5. Considérations d'ASN.1.....	6
6. Considérations relatives à l'IANA.....	6
7. Remerciements.....	7
8. Références.....	7
8.1 Références normatives.....	7
8.2 Références pour information.....	7
Appendice A. Module ASN.1	7
Adresse des auteurs.....	13

1. Introduction

Le présent document spécifie le format du champ subjectPublicKeyInfo dans les certificats X.509 [RFC5280] qui utilisent la cryptographie à courbe elliptique (ECC, *Elliptic Curve Cryptography*). Il met à jour la [RFC3279]. Le présent document spécifie les formats de codage des clés publiques utilisées avec les algorithmes d'ECC suivants :

- o algorithme de signature numérique à courbe elliptique (ECDSA, *Elliptic Curve Digital Signature Algorithm*) ;
- o schémas de famille de courbe elliptique Diffie-Hellman (ECDH, *Elliptic Curve Diffie-Hellman*) ;
- o schémas de famille de courbe elliptique Menezes-Qu-Vanstone (ECMQV, *Elliptic Curve Menezes-Qu-Vanstone*).

Deux méthodes pour spécifier les algorithmes qui peuvent être utilisés avec le champ `subjectPublicKey` sont définies. Une méthode permet que la clé soit utilisée avec tout algorithme de ECC, tandis que l'autre méthode restreint l'usage de la clé à des algorithmes spécifiques. Pour favoriser l'interopérabilité, le présent document indique ce qui est exigé que mettent en œuvre les autorités de certification (CA, *Certification Authority*) qui utilisent les algorithmes d'ECC et les consommateurs d'assertions qui prétendent traiter les algorithmes d'ECC.

Le module ASN.1 [X.680] du présent document inclut l'ASN.1 pour les algorithmes d'ECC. Il inclut aussi l'ASN.1 pour les algorithmes non ECC définis dans les [RFC3279] et [RFC5758], bien que le texte associé ne soit pas affecté. En mettant à jour tout l'ASN.1 de la [RFC3279] dans le présent document, les mises en œuvre ont seulement besoin d'utiliser le module qui se trouve dans le présent document.

1.1 Terminologie

Les mots clés "DOIT", "NE DOIT PAS", "EXIGE", "DEVRA", "NE DEVRA PAS", "DEVRAIT", "NE DEVRAIT PAS", "RECOMMANDE", "PEUT", et "FACULTATIF" en majuscules dans ce document sont à interpréter comme décrit dans le BCP 14, [RFC2119].

2. Champs d'information de clé publique sujette

Dans le certificat X.509, le champ `subjectPublicKeyInfo` a le type `SubjectPublicKeyInfo`, qui a la syntaxe ASN.1 suivante :

```
SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm      AlgorithmIdentifier,
    subjectPublicKey  CHAÎNE BINAIRE
}
```

Les champs dans `SubjectPublicKeyInfo` ont la signification suivante :

- o `algorithm` est l'identifiant d'algorithme et les paramètres pour la clé publique ECC.
- o `subjectPublicKey` est la clé publique ECC. Voir au paragraphe 2.2.

Le type `AlgorithmIdentifier`, qui est inclus de la [RFC5280], est défini comme suit :

```
AlgorithmIdentifier ::= SEQUENCE {
    algorithm      IDENTIFIANT D'OBJET,
    parameters    TOUS CEUX DEFINIS PAR algorithm FACULTATIF }

```

Les champs dans `AlgorithmIdentifier` ont la signification suivante :

- o `algorithm` identifie l'algorithme de chiffrement avec un identifiant d'objet. Voir le paragraphe 2.1.
- o `parameters`, qui est facultatif, sont les paramètres associés pour l'identifiant d'algorithme dans le champ `algorithm`. Voir le paragraphe 2.1.1.

2.1 Identifiants d'algorithme de clé publique de cryptographie à courbe elliptique

Le champ `algorithm` de la structure `SubjectPublicKeyInfo` [RFC5280] indique l'algorithme et tous les paramètres associés pour la clé publique ECC (voir le paragraphe 2.2). Trois identifiants d'algorithme sont définis dans ce document :

- o `id-ecPublicKey` indique que les algorithmes qui peuvent être utilisés avec la clé publique sujette sont sans restriction. La clé est seulement restreinte par les valeurs indiquées dans l'extension de certificat d'usage de clé (voir la Section 3). `id-ecPublicKey` DOIT être pris en charge (voir le paragraphe 2.1.1). Cette valeur est aussi incluse dans les certificats quand une clé publique est utilisée avec ECDSA.
- o `id-ecDH` indique que l'algorithme qui peut être utilisé avec la clé publique sujette est restreinte à l'algorithme Diffie-Hellman à courbe elliptique (voir le paragraphe 2.1.2). `id-ecDH` PEUT être pris en charge.
- o `id-ecMQV` indique que l'algorithme qui peut être utilisé avec la clé publique sujette est restreint à l'algorithme d'accord de clé Menezes-Qu-Vanstone à courbe elliptique (voir le paragraphe 2.1.2). `id-ecMQV` PEUT être pris en charge.

2.1.1 Identifiant et paramètres d'algorithme sans restriction

L'identifiant d'algorithme "unrestricted" est :

```
IDENTIFIANT D'OBJET id-ecPublicKey ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) keyType(2) 1 }
```

La syntaxe de clé publique (ECPoint) est décrite au paragraphe 2.2.

Le paramètre pour id-ecPublicKey est comme suit et DOIT toujours être présent :

```
ECPParameters ::= CHOIX {
    namedCurve      IDENTIFIANT D'OBJET
    -- implicitCurve NUL
    -- specifiedCurve SpecifiedECDomain
}
-- implicitCurve et specifiedCurve NE DOIVENT PAS être utilisés dans PKIX.
-- Les détails pour SpecifiedECDomain se trouvent dans [X9.62].
-- Tout ajout futur à ce CHOIX devrait être coordonné avec ANSI X9.
```

Les champs dans ECPParameters ont la signification suivante :

- o namedCurve identifie toutes les valeurs requises pour un ensemble particulier de paramètres de domaine de courbes elliptiques à représenter par un identifiant d'objet. Ce choix DOIT être pris en charge. Voir le paragraphe 2.1.1.1.
- o implicitCurve permet d'hériter des paramètres de domaine de courbes elliptiques. Ce choix NE DOIT PAS être utilisé.
- o specifiedCurve, qui est du type SpecifiedECDomain (défini dans [X9.62]) permet de spécifier explicitement tous les paramètres de domaine de courbes elliptiques. Ce choix NE DOIT PAS être utilisé. Voir la Section 5, "Considérations d'ASN.1".

L'ajout de tout nouveau choix dans ECPParameters doit être coordonné avec ANSI X9.

Le champ AlgorithmIdentifier dans SubjectPublicKeyInfo est le seul endroit dans un certificat où les paramètres de domaine de courbes elliptiques peuvent être situés. Si les paramètres de domaine de courbes elliptiques ne sont pas présents, alors les clients DOIVENT rejeter les certificat.

2.1.1.1 Courbe désignée

Le champ namedCurve dans ECPParameters utilise des identifiants d'objet pour désigner les courbes bien connues. Le présent document publie les identifiants de courbes pour les quinze courbes recommandées par le NIST [FIPS186-3]. D'autres documents peuvent publier d'autres identifiants de nom de courbe. Les courbes désignées par le NIST sont :

- Noter que dans [X9.62] les courbes sont appelées "ansix9" plutôt que "sec". Par exemple, secp192r1 est la même courbe que ansix9p192r1.
- Noter que dans la [RFC3279] la courbe secp192r1 était appelée prime192v1 et la courbe secp256r1 était appelée prime256v1.
- Noter que [FIPS186-3] appelle secp192r1 "P-192", secp224r1 "P-224", secp256r1 "P-256", secp384r1 "P-384", et secp521r1 "P-521".

```
IDENTIFIANT D'OBJET secp192r1 ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) curves(3) prime(1) 1 }
```

```
IDENTIFIANT D'OBJET sect163k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 1 }
```

```
IDENTIFIANT D'OBJET sect163r2 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 15 }
```

```
IDENTIFIANT D'OBJET secp224r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 33 }
```

```
IDENTIFIANT D'OBJET sect233k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 26 }
```

```
IDENTIFIANT D'OBJET sect233r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 27 }
```

```
IDENTIFIANT D'OBJET secp256r1 ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) curves(3) prime(1) 7 }
```

IDENTIFIANT D'OBJET sect283k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 16 }

IDENTIFIANT D'OBJET sect283r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 17 }

IDENTIFIANT D'OBJET sect384r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 34 }

IDENTIFIANT D'OBJET sect409k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 36 }

IDENTIFIANT D'OBJET sect409r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 37 }

IDENTIFIANT D'OBJET sect521r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 35 }

IDENTIFIANT D'OBJET sect571k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 38 }

IDENTIFIANT D'OBJET sect571r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 39 }

2.1.2 Identifiants et paramètres d'algorithme restreint

Deux algorithmes "restreints" sont définis pour les algorithmes d'accord de clé : les schémas de famille d'accord de clé de courbe elliptique Diffie-Hellman (ECDH, *Elliptic Curve Diffie-Hellman*) et les schémas de famille d'accord de clé de courbe elliptique Menezes-Qu-Vanstone (ECMQV, *Elliptic Curve Menezes-Qu-Vanstone*). Les deux algorithmes sont identifiés par un identifiant d'objet et ont des paramètres. L'identifiant d'objet varie selon l'algorithme, mais les paramètres sont toujours des ECParameters et DOIVENT toujours être présents (voir le paragraphe 2.1.1).

L'algorithme ECDH utilise l'identifiant d'objet suivant :

IDENTIFIANT D'OBJET id-ecDH ::= { iso(1) identified-organization(3) certicom(132) schemes(1) ecdh(12) }

L'algorithme ECMQV utilise l'identifiant d'objet suivant :

IDENTIFIANT D'OBJET id-ecMQV ::= { iso(1) identified-organization(3) certicom(132) schemes(1) ecmqv(13) }

2.2 Clé publique sujette

Le champ subjectPublicKey de SubjectPublicKeyInfo est la clé publique ECC. Les clés publiques ECC ont la syntaxe suivante :

ECPoint ::= CHAÎNE D'OCTETS

Les mises en œuvre de cryptographie à courbe elliptique conformes au présent document DOIVENT prendre en charge la forme non compressée et PEUVENT prendre en charge la forme compressée de la clé publique ECC. La forme hybride de la clé publique ECC de [X9.62] NE DOIT PAS être utilisée. Comme spécifié dans [SEC1] :

- o La clé publique à courbe elliptique (une valeur de type ECPoint qui est une CHAÎNE D'OCTETS) est transposée en une subjectPublicKey (une valeur de type CHAÎNE BINAIRE) comme suit : le bit de plus fort poids de la valeur de CHAÎNE D'OCTETS devient le bit de poids fort de la valeur de CHAÎNE BINAIRE, et ainsi de suite ; le bit de moindre poids de la CHAÎNE D'OCTETS devient le bit de moindre poids de la CHAÎNE BINAIRE. Les sous programmes de conversion se trouvent aux paragraphes 2.3.1 et 2.3.2 de [SEC1].
- o Le premier octet de la CHAÎNE D'OCTETS indique si la clé est comprimée ou non comprimée. La forme non comprimée est indiquée par 0x04 et la forme comprimée est indiquée par 0x02 ou 0x03 (voir le paragraphe 2.3.3 de [SEC1]). La clé publique DOIT être rejetée si une autre valeur est incluse dans le premier octet.

3. Bits d'usage de clé

Si l'extension keyUsage est présente dans un certificat d'autorité de certification (CA, *Certification Authority*) qui indique id-ecPublicKey dans SubjectPublicKeyInfo, alors toute combinaison des valeurs suivantes PEUT être présente :

digitalSignature,
nonRepudiation,
keyAgreement,
keyCertSign,
cRLSign.

Si l'extension keyUsage de certificat de CA affirme keyAgreement, alors elle PEUT affirmer encipherOnly ou decipherOnly. Cependant, la présente spécification RECOMMANDE que si keyCertSign ou cRLSign est présent, alors keyAgreement, encipherOnly, et decipherOnly NE DEVRAIENT PAS être présents.

Si l'extension keyUsage est présente dans un certificat d'entité d'extrémité (EE, *End Entity*) qui indique id-ecPublicKey dans SubjectPublicKeyInfo, alors toute combinaison des valeurs suivantes PEUT être présente :

digitalSignature,
nonRepudiation,
keyAgreement.

Si l'extension keyUsage de certificat d'EE affirme keyAgreement, alors elle PEUT affirmer encipherOnly ou decipherOnly.

Si l'extension keyUsage est présente dans un certificat qui indique id-ecDH ou id-ecMQV dans SubjectPublicKeyInfo, alors ce qui suit DOIT être présent :

keyAgreement ;

un des suivants PEUT être présent :

encipherOnly,
decipherOnly.

Si l'extension keyUsage est présente dans un certificat qui indique id-ecDH ou id-ecMQV dans SubjectPublicKeyInfo, alors les valeurs suivantes NE DOIVENT PAS être présentes :

digitalSignature,
nonRepudiation,
keyTransport,
keyCertSign,
cRLSign.

4. Considérations sur la sécurité

Les considérations sur la sécurité de la [RFC3279] s'appliquent.

Quand on met en œuvre ECC dans des certificats X.509 et des listes de révocation de certificats (CRL) il y a trois choix relatifs aux algorithmes qui doivent être faits pour le champ signatureAlgorithm dans un certificat ou une liste de certificats :

- 1) Quelle est la taille de la clé publique ?
- 2) Quel est l'algorithme de hachage [FIPS180-3] ?
- 3) Quelle est la courbe ?

La CA doit considérer la force de la sécurité fournie par chacun de ces choix. La sécurité est mesurée en bits, où un fort chiffrement symétrique avec une clé de X bits est dite fournir X bits de sécurité. Il est recommandé que les bits de sécurité fournis par chaque choix soient en gros équivalents. Le tableau qui suit donne des bits de sécurité comparables minimums [SP800-57] pour les tailles de clé ECDSA et les algorithmes de résumé de message. Il fait aussi la liste des courbes (voir le paragraphe 2.1.1.1) pour les tailles de clés.

Minimum de bits de sécurité	Taille de clé ECDSA	Algorithmes de résumé de message	Courbes
80	160 à 223	SHA-1	sect163k1
		SHA-224	secp163r2
		SHA-256	secp192r1
		SHA-384, SHA-512	

112	224 à 255	SHA-224 SHA-256	secp224r1 sect233k1
128	256 à 383	SHA-384, SHA-512 SHA-256 SHA-384	sect233r1 secp256r1 sect283k1
192	384 à 511	SHA-512 SHA-384 SHA-512	sect283r1 secp384r1 sect409k1, sect409r1
256	512 et +	SHA-512	secp521r1 sect571k1, sect571r1

Pour favoriser l'interopérabilité, les choix suivants sont RECOMMANDÉS :

Minimum de bits de sécurité	Taille de clé ECDSA	Algorithmes de résumé de message	Courbes
80	192	SHA-256	secp192r1
112	224	SHA-256	secp224r1
128	256	SHA-256	secp256r1
192	384	SHA-384	secp384r1
256	512	SHA-512	secp521r1

Utiliser une plus grande valeur de hachage et la tronquer ensuite consomme plus de puissance de traitement que nécessaire. Ceci a plus d'importance sur les appareils qui ont des contraintes. Comme le signataire ne connaît pas l'environnement que le receveur va utiliser pour valider la signature, il vaut mieux utiliser une fonction de hachage qui fournit la taille de sortie de la valeur de hachage désirée, et pas plus.

Il y a des risques pour la sécurité à utiliser des clés qui ne sont pas associées à des courbes bien connues et largement reconnues. Par exemple, la courbe peut ne pas satisfaire la condition de Menezes-Okamoto-Vanstone (MOV) [X9.62] ou elle peut être vulnérable à l'attaque "Anomalous" [X9.62]. De plus, soit a) toutes les propriétés arithmétiques d'une clé publique ECC candidate doivent être validées pour assurer qu'elle a la représentation unique correcte dans le sous groupe (additif) correct (et donc est aussi dans le groupe EC correct) spécifié par les paramètres de domaine ECC associé, soit b) certaines des propriétés arithmétiques d'une clé publique ECC candidate doivent être validées pour assurer qu'elle est dans le groupe correct (mais pas nécessairement le sous groupe correct) spécifié par les paramètres de domaine ECC associé [SP800-56A].

Comme noté dans la [RFC3279], l'utilisation de MD2 et MD5 pour de nouvelles applications est déconseillée. Il est quand même raisonnable d'utiliser MD2 et MD5 pour vérifier les signatures existantes.

5. Considérations d'ASN.1

[X9.62] définit des options supplémentaires pour ECPParameters et ECDSA-Sig-Value [RFC3279]. Si une mise en œuvre a besoin d'utiliser ces options, on utilise alors le module [X9.62] ASN.1. La présente RFC contient un sous ensemble conforme du module ASN.1 défini dans [X9.62].

Si une mise en œuvre génère un codage PER [X.691] qui utilise le module ASN.1 qui se trouve dans la présente spécification, elle pourrait ne pas réaliser le même résultat codé qu'une qui utilise le module [X9.62]. PER n'est pas exigé par les environnements PKIX ni S/MIME. Si un environnement de mise en œuvre exige PER, alors les soucis de mise en œuvre seront moins probables avec l'utilisation du module [X9.62].

6. Considérations relatives à l'IANA

Le présent document fait une large utilisation des identifiants d'objet pour enregistrer les types de clé publique, les courbes elliptiques, et les algorithmes. La plupart sont enregistrés dans l'arc ANSI X9.62, à l'exception des algorithmes de hachage (qui sont dans l'arc du NIST) et de beaucoup des courbes (qui sont dans l'arc Certicom Inc. ; ces courbes ont été adoptées par l'ANSI et le NIST). De plus, un identifiant d'objet est utilisé pour identifier le module ASN.1 qui se trouve à l'Appendice A. Il est défini dans un arc délégué par l'IANA au groupe de travail PKIX. Aucune autre action de l'IANA n'est nécessaire pour ce document ou une de ses mises à jour prévues.

7. Remerciements

Les auteurs tiennent à remercier Stephen Farrell, Alfred Hoenes, Johannes Merkle, Jim Schaad, et Carl Wallace de leurs précieux apports.

8. Références

8.1 Références normatives

- [FIPS180-3] National Institute of Standards and Technology (NIST), FIPS Publication 180-3 : "Secure Hash Standard", octobre 2008.
- [FIPS186-3] National Institute of Standards and Technology (NIST), FIPS Publication 186-3: "Digital Signature Standard", (projet) novembre 2008.
- [RFC2119] S. Bradner, "[Mots clés à utiliser](#) dans les RFC pour indiquer les niveaux d'exigence", BCP 14, mars 1997. (MàJ par [RFC8174](#))
- [RFC3279] L. Bassham, W. Polk et R. Housley, "[Algorithmes et identifiants](#) pour le profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet", avril 2002.
- [RFC4055] J. Schaad et autres, "[Algorithmes et identifiants supplémentaires pour la cryptographie RSA](#) à utiliser dans le profil de certificat d'infrastructure de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet", juin 2005.
- [RFC5280] D. Cooper et autres, "[Profil de certificat d'infrastructure](#) de clé publique X.509 et de liste de révocation de certificat (CRL) pour l'Internet", mai 2008. (P.S. ; Remplace les [RFC3280](#), [RFC4325](#), [RFC4630](#) ; MàJ par [RFC8398](#), [8399](#))
- [SEC1] Standards for Efficient Cryptography Group (SECG), "SEC1: Elliptic Curve Cryptography", Version 1.0, septembre 2000.
- [X9.62] American National Standards Institute (ANSI), ANS X9.62-2005 : "The Elliptic Curve Digital Signature Algorithm (ECDSA)", 2005.
- [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)

8.2 Références pour information

- [RFC5758] Q. Dang, S. Santesson, K. Moriarty, D. Brown, T. Polk, "Infrastructure de clés publiques X.509 sur Internet : Algorithmes et identifiants supplémentaires pour DSA et ECDSA", janvier 2010. (MàJ [RFC3279](#)). (P. S.)
- [SP800-56A] National Institute of Standards and Technology (NIST), Special Publication 800-56A: "Recommendation for Pair-wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revised)", mars 2007.
- [SP800-57] National Institute of Standards and Technology (NIST), Special Publication 800-57: "Recommendation for Key Management - Part 1 (Revised)", mars 2007.
- [X.691] Recommandation UIT-T X.691 | ISO/CEI 8825-2:2002, "Technologies de l'information - Règles de codage de l'ASN.1 : Spécification des règles de codage de paquet", (07/2002).

Appendice A. Module ASN.1

PKIX1Algorithms2008 { iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0)

45 }

ÉTIQUETTES DE DÉFINITIONS EXPLICITES ::=

DÉBUT

-- EXPORTE TOUT ;

IMPORTE

-- De la [RFC4055]

id-sha224, id-sha256, id-sha384, id-sha512

DE PKIX1-PSS-OAEP-Algorithms

{ iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) pkix(7) id-mod(0) id-mod-pkix1-rsa-pkalg(33) }

;

--

-- Algorithmes de résumé de message

--

-- MD-2

-- Les paramètres sont NULS

IDENTIFIANT D'OBJET id-md2 ::= { iso(1) member-body(2) us(840) rsadsi(113549) digestAlgorithm(2) 2 }

-- MD-5

-- Les paramètres sont NULS

IDENTIFIANT D'OBJET id-md5 ::= { iso(1) member-body(2) us(840) rsadsi(113549) digestAlgorithm(2) 5 }

-- SHA-1

-- Les paramètres sont de préférence absents

IDENTIFIANT D'OBJET id-sha1 ::= { iso(1) identified-organization(3) oiw(14) secsig(3) algorithm(2) 26 }

-- SHA-224

-- Les paramètres sont de préférence absents

-- IDENTIFIANT D'OBJET id-sha224 ::= { joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101)

-- csor(3) nistalgorithm(4) hashalgs(2) 4 }

-- SHA-256

-- Les paramètres sont de préférence absents

-- IDENTIFIANT D'OBJET id-sha256 ::= {

-- joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101) csor(3) nistalgorithm(4) hashalgs(2) 1 }

-- SHA-384

-- Les paramètres sont de préférence absents

-- IDENTIFIANT D'OBJET id-sha384 ::= {

-- joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101) csor(3) nistalgorithm(4) hashalgs(2) 2 }

-- SHA-512

-- Les paramètres sont de préférence absents

-- IDENTIFIANT D'OBJET id-sha512 ::= {

-- joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101) csor(3) nistalgorithm(4) hashalgs(2) 3 }

--

-- Algorithmes de clé publique (PK)

--

-- Algorithme et clé RSA PK

IDENTIFIANT D'OBJET rsaEncryption ::= { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 1 }

```
RSAPublicKey ::= SEQUENCE {
    modulus      ENTIER -- n
    publicExponent ENTIER -- e
}
```

-- Algorithme DSA PK, clé, et paramètres

IDENTIFIANT D'OBJET id-dsa ::= { iso(1) member-body(2) us(840) x9-57(10040) x9algorithm(4) 1 }

DSAPublicKey ::= ENTIER -- clé publique, y

```
DSS-Parms ::= SEQUENCE {
    p ENTIER,
    q ENTIER,
    g ENTIER
}
```

-- Algorithme Diffie-Hellman PK, clé, et paramètres

IDENTIFIANT D'OBJET dhpublishnumber ::= { iso(1) member-body(2) us(840) ansi-x942(10046) number-type(2) 1 }

DHPublicKey ::= ENTIER -- clé publique, $y = g^x \text{ mod } p$

```
DomainParameters ::= SEQUENCE {
    p      ENTIER,           -- nombre premier impair,  $p = jq + 1$ 
    g      ENTIER,           -- générateur, g
    q      ENTIER,           -- facteur de  $p - 1$ 
    j      ENTIER FACULTATIF, -- facteur de sous groupe,  $j \geq 2$ 
    validationParms ValidationParms FACULTATIF
}
```

```
ValidationParms ::= SEQUENCE {
    seed      CHAÎNE BINAIRE,
    pgenCounter ENTIER
}
```

-- Algorithme et paramètres KEA PK

IDENTIFIANT D'OBJET id-keyExchangeAlgorithm ::= { joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101) dod(2) infosec(1) algorithms(1) 22 }

KEA-Parms-Id ::= CHAÎNE D'OCTETS

-- Paragraphe 2.1.1 Identifiant d'algorithme non restreint, clé, et paramètres

-- (Les clés ECDSA utilisent id-ecPublicKey)

IDENTIFIANT D'OBJET id-ecPublicKey ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) keyType(2) 1 }

ECPoint ::= CHAÎNE D'OCTETS

-- Paramètres pour restreint et non restreint

```
ECPParameters ::= CHOIX {
    namedCurve      IDENTIFIANT D'OBJET
```

```

-- implicitCurve NUL
-- specifiedCurve SpecifiedECDomain
}
-- implicitCurve et specifiedCurve NE DOIVENT PAS être utilisées dans PKIX.
-- Les détails pour SpecifiedECDomain se trouvent dans [X9.62].
-- Tout ajout futur à ce CHOIX devrait être coordonné avec ANSI X9.

-- Paragraphe 2.1.2 Identifiants d'algorithme restreint, clé, et paramètres : ECDH

IDENTIFIANT D'OBJET id-ecDH ::= { iso(1) identified-organization(3) certicom(132) schemes(1) ecdh(12) }

-- ECPPoint ::= CHAÎNE D'OCTETS

-- Les paramètres sont ECPParameters.

-- Paragraphe 2.1.2 Identifiants d'algorithme restreint, clé, et paramètres : ECMQV

IDENTIFIANT D'OBJET id-ecMQV ::= { iso(1) identified-organization(3) certicom(132) schemes(1) ecmqv(13) }

-- ECPPoint ::= CHAÎNE D'OCTETS

-- Les paramètres sont ECPParameters.

--
-- Algorithmes de signature
--
-- RSA avec MD-2
-- Les paramètres sont NULS

IDENTIFIANT D'OBJET md2WithRSAEncryption ::= {
  iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 2 }

-- RSA with MD-5
-- Les paramètres sont NULS

IDENTIFIANT D'OBJET md5WithRSAEncryption ::= {
  iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 4 }

-- RSA avec SHA-1
-- Les paramètres sont NULS

IDENTIFIANT D'OBJET sha1WithRSAEncryption ::= {
  iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-1(1) 5 }

-- DSA avec SHA-1
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET id-dsa-with-sha1 ::= { iso(1) member-body(2) us(840) x9-57(10040) x9algorithm(4) 3 }

-- DSA avec SHA-224
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET id-dsa-with-sha224 ::= {
  joint-iso-ccitt(2) country(16) us(840) organization(1) gov(101) csor(3) algorithms(4) id-dsa-with-sha2(3) 1 }

-- DSA avec SHA-256
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET id-dsa-with-sha256 ::= {
  joint-iso-ccitt(2) country(16) us(840) organization(1) gov(101) csor(3) algorithms(4) id-dsa-with-sha2(3) 2 }

```

```

-- ECDSA avec SHA-1
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET cdsa-with-SHA1 ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) signatures(4) 1 }

-- ECDSA avec SHA-224
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET ecdsa-with-SHA224 ::= {
    iso(1) member-body(2) us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 1 }

-- ECDSA avec SHA-256
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET ecdsa-with-SHA256 ::= {
    iso(1) member-body(2) us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 2 }

-- ECDSA avec SHA-384
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET ecdsa-with-SHA384 ::= {
    iso(1) member-body(2) us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 3 }

-- ECDSA avec SHA-512
-- Les paramètres sont ABSENTS

IDENTIFIANT D'OBJET ecdsa-with-SHA512 ::= {
    iso(1) member-body(2) us(840) ansi-X9-62(10045) signatures(4) ecdsa-with-SHA2(3) 4 }

--
-- Valeurs de signature
--

-- DSA

DSA-Sig-Value ::= SEQUENCE {
    r ENTIER,
    s ENTIER
}

-- ECDSA

ECDSA-Sig-Value ::= SEQUENCE {
    r ENTIER,
    s ENTIER
}

--
-- Courbes elliptiques désignées
--

-- Noter que dans [X9.62] les courbes sont appelées "ansiX9" et non "sec". Par exemple secp192r1 est la même courbe que
ansi9p192r1.

-- Noter que dans la [RFC3279] la courbe secp192r1 est appelée "prime192v1" et la courbe secp256r1 est appelée
prime256v1.

-- Noter que [FIPS186-3] appelle secp192r1 P-192, secp224r1 P-224, secp256r1 P-256, secp384r1 P-384, et secp521r1 P-
521.

```

IDENTIFIANT D'OBJET secp192r1 ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) curves(3) prime(1) 1 }

IDENTIFIANT D'OBJET sect163k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 1 }

IDENTIFIANT D'OBJET sect163r2 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 15 }

IDENTIFIANT D'OBJET secp224r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 33 }

IDENTIFIANT D'OBJET sect233k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 26 }

IDENTIFIANT D'OBJET sect233r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 27 }

IDENTIFIANT D'OBJET secp256r1 ::= { iso(1) member-body(2) us(840) ansi-X9-62(10045) curves(3) prime(1) 7 }

IDENTIFIANT D'OBJET sect283k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 16 }

IDENTIFIANT D'OBJET sect283r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 17 }

IDENTIFIANT D'OBJET secp384r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 34 }

IDENTIFIANT D'OBJET sect409k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 36 }

IDENTIFIANT D'OBJET sect409r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 37 }

IDENTIFIANT D'OBJET secp521r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 35 }

IDENTIFIANT D'OBJET sect571k1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 38 }

IDENTIFIANT D'OBJET sect571r1 ::= { iso(1) identified-organization(3) certicom(132) curve(0) 39 }

FIN

Adresse des auteurs

Sean Turner
IECA, Inc.
3057 Nutley Street, Suite 106
Fairfax, VA 22031
USA
mél : turners@ieca.com

Kelvin Yiu
Microsoft
One Microsoft Way
Redmond, WA 98052-6399
USA
mél : kelviny@microsoft.com

Daniel R. L. Brown
Certicom Corp
5520 Explorer Drive #400
Mississauga, ON L4W 5L1
CANADA
mél : dbrown@certicom.com

Russ Housley
Vigil Security, LLC
918 Spring Knoll Drive
Herndon, VA 20170
USA
mél : housley@vigilsec.com

Tim Polk
NIST
Building 820, Room 426
Gaithersburg, MD 20899
USA
mél : wpolk@nist.gov