ICS 35.030
CCS L 80

GM

中华人民共和国密码行业标准

GM/T 0010—2023
代替 GM/T 0010—2012

SM2 密码算法加密签名消息语法规范

SM2 cryptography message syntax specification

2023-12-04 发布　　2024-06-01 实施

国家密码管理局　　发布

目　　次

前言 …… Ⅲ
1　范围 …… 1
2　规范性引用文件 …… 1
3　术语和定义 …… 1
4　缩略语 …… 1
5　OID 定义 …… 1
6　基本类型定义 …… 2
6.1　CertificateRevocationLists …… 2
6.2　ContentEncryptionAlgorithmIdentifier …… 2
6.3　DigestAlgorithmIdentifier …… 2
6.4　DigestEncryptionAlgorithmIdentifier …… 2
6.5　Certificate …… 2
6.6　IssuerAndSerialNumber …… 2
6.7　KeyEncryptionAlgorithmIdentifier …… 3
6.8　Version …… 3
6.9　ContentInfo …… 3
7　数据类型 Data …… 3
8　签名数据类型 signedData …… 3
8.1　signedData 类型 …… 3
8.2　signerInfo 类型 …… 4
9　数字信封数据类型 envelopedData …… 5
9.1　envelopedData 类型 …… 5
9.2　recipientInfo 类型 …… 5
10　签名及数字信封数据类型 signedAndEnvelopedData …… 6
11　加密数据类型 encryptedData …… 7
12　密钥协商类型 keyAgreementInfo …… 7
附录 A（资料性）　示例 …… 8
附录 B（规范性）　SM2 隐式证书的加密签名消息语法 …… 20
附录 C（规范性）　SM2 密钥格式 …… 24

前　言

本文件按照 GB/T 1.1—2020《标准化工作导则　第 1 部分：标准化文件的结构和起草规则》的规定起草。

本文件代替 GM/T 0010—2012《SM2 密码算法加密签名消息语法规范》。与 GM/T 0010—2012 相比，除结构调整和编辑性改动外，主要技术变化如下：

a) 增加了规范性引用文件 GB/T 25069、GM/T 0015 和 GM/Z 4001（见第 3 章，6.5），删除了 PKCS# 6 的引用（见 2012 年版的 6.5）；

b) 删除了术语“算法标识”和“SM2 密码算法”（见 2012 年版的 3.1 和 3.2），增加了术语“隐式证书”（见 3.1）；

c) 增加了隐式证书相关的 3 个 OID（见第 5 章）；

d) 删除了对扩展证书类型的描述（见 2012 年版的 6.5）；

e) 更改了“contentInfo SM2Signature”（见第 8 章，2012 年版的第 8 章）；

f) 更改了 userCertificate（见第 12 章，2012 年版的第 12 章）；

g) 增加了规范性的附录 B（见附录 B）。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由密码行业标准化技术委员会提出并归口。

本文件起草单位：格尔软件股份有限公司、北京海泰方圆科技有限公司、北京数字认证股份有限公司、广东省电子商务认证有限公司、无锡江南信息安全工程技术中心、中电科网络安全科技股份有限公司、北京信安世纪科技股份有限公司、北京国脉信安科技有限公司、上海市数字证书认证中心有限公司、兴唐通信科技有限公司、山东得安信息技术有限公司、山东大学、深圳奥联信息安全技术有限公司、智巡密码（上海）检测技术有限公司、上海颐东网络信息有限公司、国家信息安全工程技术研究中心。

本文件主要起草人：刘平、郑强、谭武征、柳增寿、李述胜、赵永省、赵敏、梁宇宁、汪宗斌、袁锋、孔凡玉、程朝辉、韩玮、马洪富、蒋红宇、王妮娜、刘伟、徐强、李元正、刘承、夏东山。

本文件及其所代替文件的历次版本发布情况为：

——2012 年首次发布为 GM/T 0010—2012；

——本次为第一次修订。

SM2 密码算法加密签名消息语法规范

1 范围

本文件规定了使用 SM2 密码算法的加密签名消息语法。

本文件适用于使用 SM2 密码算法进行加密和签名操作时对操作结果的标准化封装。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本(包括所有的修改单)适用于本文件。

GB/T 25069 信息安全技术 术语

GM/T 0006 密码应用标识规范

GM/T 0009 SM2 密码算法使用规范

GM/T 0015 数字证书格式

GM/Z 4001 密码术语

3 术语和定义

GB/T 25069 和 GM/Z 4001 界定的术语和定义适用于本文件。

3.1

隐式证书 implicit certificate

包含用户标识、公钥还原数据和签发者标识信息但不显式包含证书认证机构(CA)的数字签名的数字证书。

注：用户关联的实际公钥由椭圆曲线系统参数、CA 的公钥、隐式证书中相关的用户信息和公钥还原数据计算得出。

4 缩略语

下列缩略语适用于本文件。

CA：证书认证机构(Certification Authority)

OID：对象标识符(Object Identity)

5 OID 定义

本文件对 9 个对象 data，signedData，envelopedData，signedAndEnvelopedData，encryptedData，keyAgreementInfo，imcSignedData，imcEnvelopedData 和 imcSignedAndEnvelopedData 的标识符进行了定义，详见表 1。

表 1　对象标识符

对象标识符 OID	对象标识符定义
1.2.156.10197.6.1.4.2	SM2 密码算法加密签名消息语法规范
1.2.156.10197.6.1.4.2.1	数据类型
1.2.156.10197.6.1.4.2.2	签名数据类型
1.2.156.10197.6.1.4.2.3	数字信封数据类型
1.2.156.10197.6.1.4.2.4	签名及数字信封数据类型
1.2.156.10197.6.1.4.2.5	加密数据类型
1.2.156.10197.6.1.4.2.6	密钥协商类型
1.2.156.10197.6.1.4.2.7	隐式证书签名数据类型
1.2.156.10197.6.1.4.2.8	隐式证书数字信封数据类型
1.2.156.10197.6.1.4.2.9	隐式证书签名及数字信封数据类型

6　基本类型定义

6.1　CertificateRevocationLists

CertificateRevocationLists 类型标明一个证书撤销列表的集合。

CertificateRevocationLists ::= SET OF CertificateRevocationList

6.2　ContentEncryptionAlgorithmIdentifier

ContentEncryptionAlgorithmIdentifier 类型标明一个数据加密算法。其 OID 应符合 GM/T 0006。

ContentEncryptionAlgorithmIdentifier ::= AlgorithmIdentifier

6.3　DigestAlgorithmIdentifier

DigestAlgorithmIdentifier 类型标明一个消息摘要算法，本文件为 SM3 密码算法，其 OID 应符合 GM/T 0006。

DigestAlgorithmIdentifier ::= AlgorithmIdentifier

6.4　DigestEncryptionAlgorithmIdentifier

DigestEncryptionAlgorithmIdentifier 类型标明一个签名算法，本文件为 SM2 签名算法，其 OID 应符合 GM/T 0006。

DigestEncryptionAlgorithmIdentifier ::= AlgorithmIdentifier

6.5　Certificate

Certificate 类型指定一个符合 GM/T 0015 格式的证书。它表示集合足以包含从可识别的“根”或“顶级 CA”到所有签名者的证书链。

Certificates ::= SET OF Certificate

6.6　IssuerAndSerialNumber

IssuerAndSerialNumber 类型标明一个证书颁发者可识别名和颁发者确定的证书序列号，可据此

确定一份证书和与此证书对应的实体及公钥。

```
IssuerAndSerialNumber ::= SEQUENCE {
        issuer Name,
        serialNumber CertificateSerialNumber
}
```

6.7 KeyEncryptionAlgorithmIdentifier

KeyEncryptionAlgorithmIdentifier 类型标明加密对称密钥的加密算法。

```
KeyEncryptionAlgorithmIdentifier ::= AlgorithmIdentifier
```

6.8 Version

Version 类型标明语法版本号。

```
Version ::= INTEGER(1)
```

6.9 ContentInfo

ContentInfo 类型标明内容交换通用语法结构，内容交换的通用语法结构定义如下：

```
ContentInfo ::= SEQUENCE {
        contentType ContentType,
        content[0] EXPLICIT ANY DEFINED BY contentType OPTIONAL
}
ContentType ::= OBJECT IDENTIFIER
```

其中：

ContentType 内容类型是一个对象标识符，其定义见第 5 章。

content 内容，可选。

7 数据类型 Data

数据类型 Data 结构定义如下：

```
Data ::= OCTET STRING
```

数据类型 Data 表示任意的字节串。

8 签名数据类型 signedData

8.1 signedData 类型

signedData 数据类型由任意类型的数据和至少一个签名者的签名值组成。任意类型的数据能同时被任意数量的签名者签名。签名数据类型示例见附录 A 中的 A.1，隐式证书的签名数据类型应符合附录 B 中的 B.1。

signedData 数据类型结构定义如下：

```
SignedData ::= SEQUENCE {
        version Version,
        digestAlgorithms DigestAlgorithmIdentifiers,
        contentInfo ContentInfo,
        certificates[0] IMPLICIT Certificates OPTIONAL,
        crls[1] IMPLICIT CertificateRevocationLists OPTIONAL,
```

```
        signerInfos SignerInfos
}
DigestAlgorithmIdentifiers ::= SET OF DigestAlgorithmIdentifier
SignerInfos::= SET OF SignerInfo
```

结构中各项含义见表 2。

表 2 signedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 1
digestAlgorithms	DigestAlgorithmIdentifiers	消息摘要算法标识符的集合
contentInfo	ContentInfo	数据内容
certificates	Certificates	证书的集合
crls	CertificateRevocationLists	证书撤销列表的集合
signInfos	SignerInfos	签名者信息的集合

8.2 signerInfo 类型

signerInfo 类型结构定义如下：

```
SignerInfo ::= SEQUENCE {
        version Version,
        issuerAndSerialNumber IssuerAndSerialNumber,
        digestAlgorithm DigestAlgorithmIdentifier,
        authenticatedAttributes[0] IMPLICIT Attributes OPTIONAL,
        digestEncryptionAlgorithm  DigestEncryptionAlgorithmIdentifier,
        encryptedDigest EncryptedDigest,
        unauthenticatedAttributes[1] IMPLICIT Attributes OPTIONAL
}
EncryptedDigest ::=OCTET STRING
```

结构中各项含义见表 3。

表 3 SignerInfo 数据类型

字段名称	数据类型	含义
Version	Version	版本号，此处取值为 1
issuerAndSerialNumber	IssuerAndSerialNumber	一个证书颁发者可识别名和颁发者确定的证书序列号，可据此确定一份证书和与此证书对应的实体及公钥
digestAlgorithm	DigestAlgorithmidentifier	对内容进行摘要计算的消息摘要算法，本文件采用 SM3 密码算法
authenticatedAttributes	Attributes	是经由签名者签名的属性的集合，该域可选。如果该域存在，该域中摘要的计算方法是对原文进行摘要计算结果
digestEncryptionAlgorithm	DigestEncryptionAlgorithmIdentifier	SM2 椭圆曲线数字签名算法标识符
encryptedDigest	OCTET STRING	值是 SM2Signature，用签名者私钥进行签名的结果，其定义应遵循 GM/T 0009

9 数字信封数据类型 envelopedData

9.1 envelopedData 类型

数字信封 envelopedData 数据类型由加密数据和至少一个接收者的数据加密密钥的密文组成。其中，加密数据是用数据加密密钥加密的，数据加密密钥是用接收者的公钥加密的。

数字信封数据类型示例见 A.2，隐式证书数字信封数据类型应符合 B.2。

该类型用于为接收者的 data、digestedData 或 signedData 三种类型的数据做数字信封。

envelopedData 数据类型结构定义如下：

```
EnvelopedData ::= SEQUENCE {
    version Version,
        recipientInfos RecipientInfos,
        encryptedContentInfo EncryptedContentInfo
}
RecipientInfos ::= SET OF RecipientInfo
```

结构中各项含义见表 4。

表 4 EnvelopedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 1
recipientInfos	RecipientInfos	接收者信息的集合，至少要有一个接收者
encryptedContentInfo	EncryptedContentInfo	被加密的内容

```
EncryptedContentInfo ::= SEQUENCE {
contentType ContentType,
contentEncryptionAlgorithm ContentEncryptionAlgorithmIdentifier,
encryptedContent[0] IMPLICIT EncryptedContent OPTIONAL,
sharedInfo1 [1] IMPLICIT OCTET STRING OPTIONAL,
sharedInfo2 [2] IMPLICIT OCTET STRING OPTIONAL
    }
EncryptedContent ::= OCTET STRING
```

结构中各项含义见表 5。

表 5 EncryptedContentInfo 数据类型

字段名称	数据类型	含义
contenType	ContentType	内容的类型
contentEncryptionAlgorithm	ContentEncryptionAlgorithmIdentifier	内容加密算法(和相应的参数)
encryptedContent	EncryptedContent	内容加密的结果，可选
sharedInfo1	OCTET STRING	协商好的共享信息，可选
sharedInfo2	OCTET STRING	协商好的共享信息，可选

9.2 recipientInfo 类型

接收者信息用 recipientInfo 类型表示。

recipientInfo 类型结构定义如下：

```
RecipientInfo ::= SEQUENCE{
        version Version,
        issuerAndSerialNumber IssuerAndSerialNumber,
        keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
        encryptedKey OCTET STRING
}
```

结构中各项含义见表 6。

表 6　RecipientInfo 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 1
issuerAndSerialNumber	IssuerAndSerial Number	颁发者可辨别名和颁发序列号
keyEncryptionAlgorithm	KeyEncryptionAlgo-rithmIdentifier	用接收者公钥加密数据加密密钥的算法，为 SM2 椭圆曲线加密算法或 SM2 隐式证书公钥加密机制
encryptedKey	OCTET STRING	数据加密密钥密文 SM2cipher，其定义应遵循 GM/T 0009

10　签名及数字信封数据类型 signedAndEnvelopedData

signedAndEnvelopedData 数据类型由任意类型的加密数据、至少一个接收者的数据加密密钥和至少一个签名者的签名组成，隐式证书签名及数字信封数据类型应符合 B.3。

signedAndEnvelopedData 数据类型结构定义如下：

```
SignedAndEnvelopedData ::= SEQUENCE {
        version Version,
        recipientInfos RecipientInfos,
        digestAlgorithms DigestAlgorithmIdentifiers,
        encryptedContentInfo EncryptedContentInfo,
        certificates[0] IMPLICIT Certificates OPTIONAL,
        crls[1] IMPLICIT CertificateRevocationLists OPTIONAL,
        signerInfos SignerInfos
}
```

结构中各项含义见表 7。

表 7　SignedAndEnvelopedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 1
recipientInfos	RecipientInfos	接受者信息的集合，至少一个元素
digestAlgorithms	DigestAlgorithmIdentifiers	消息摘要算法标识符的集合
encryptedContentInfo	EncryptedContentInfo	被加密的内容，可是任何定义的数据类型
certificates	Certificates	证书或隐式证书的集合，可选
crls	CertificateRevocationLists	证书撤销列表的集合
signerInfos	SignerInfos	签名者的集合，至少要有一个元素

11 加密数据类型 encryptedData

encryptedData 数据类型由任意类型的加密后的数据组成，数据类型既没有接收者也没有加密的数据加密密钥。加密数据类型示例见 A.3。

encryptedData 数据类型定义如下：

```
EncryptedData ::= SEQUENCE {
        version Version,
        encryptedContentInfo EncryptedContentInfo
}
```

结构中各项含义见表 8。

表 8 encryptedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 1
encryptedContentInfo	EncryptedContentInfo	已加密的内容信息

12 密钥协商类型 keyAgreementInfo

密钥协商 keyAgreementInfo 数据类型标明两个用户之间建立一个共享秘密密钥的结构，通过这种方式能确定一个共享秘密密钥的值。密钥协商类型示例见 A.4。

该类型用于两个用户为产生共享秘密密钥进行的公共参数交换。

```
KeyAgreementInfo::= SEQUENCE{
    version    Version,
    tempPublicKeyR    SM2PublicKey,
    userCertificate    Certificate, OPTIONAL
    userID    OCTET STRING
}
```

结构中各项含义见表 9。

表 9 keyAgreementInfo 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 1
tempPublicKeyR	SM2PublicKey	临时公钥，SM2 密钥格式应遵循附录 C，结构定义应遵循 GM/T 0006
userCertificate	Certificate	用户证书，可选
userID	OCTET STRING	用户标识

附 录 A
（资料性）
示 例

A.1 签名数据类型示例

```
  0  1083: SEQUENCE {
  4    10:   OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 2'
 16  1067:   [0] {
 20  1063:     SEQUENCE {
 24     1:       INTEGER 1
 27    15:       SET {
 29    13:         SEQUENCE {
 31     9:           OBJECT IDENTIFIER '1 2 156 10197 1 401 1'
 42     0:           NULL
        :           }
        :         }
 44    40:       SEQUENCE {
 46    10:         OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 1'
 58    26:         [0] {
 60    24:           OCTET STRING 'qewrqwer1234123qwerqasdf'
        :           }
        :         }
 86   764:       [0] {
 90   760:         SEQUENCE {
 94   668:           SEQUENCE {
 98     3:             [0] {
100     1:               INTEGER 2
        :               }
103    18:             INTEGER 21 05 70 34 0F 6D 32 45 17 8F 0C 10 E5 61 91 88 6D BF
123    12:             SEQUENCE {
125     8:               OBJECT IDENTIFIER '1 2 156 10197 1 501'
135     0:               NULL
        :               }
137    98:             SEQUENCE {
139    11:               SET {
141     9:                 SEQUENCE {
143     3:                   OBJECT IDENTIFIER countryName (2 5 4 6)
148     2:                   UTF8String 'CN'
        :                   }
        :                 }
152    13:               SET {
154    11:                 SEQUENCE {
```

```
156   3:                OBJECT IDENTIFIER organizationName (2 5 4 10)
161   4:                UTF8String 'Test'
        :                }
        :              }
167   27:            SET {
169   25:              SEQUENCE {
171   3:                 OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
176   18:                UTF8String '12140000405703345Q'
        :                }
        :              }
196   39:            SET {
198   37:              SEQUENCE {
200   3:                 OBJECT IDENTIFIER commonName (2 5 4 3)
205   30:                UTF8String '.....'
        :                }
        :              }
        :            }
237   30:          SEQUENCE {
239   13:            UTCTime 27/03/2019 05:52:45 GMT
254   13:            UTCTime 13/10/2019 05:52:45 GMT
        :          }
269   52:          SEQUENCE {
271   11:            SET {
273   9:               SEQUENCE {
275   3:                 OBJECT IDENTIFIER countryName (2 5 4 6)
280   2:                 UTF8String 'CN'
        :                }
        :              }
284   10:            SET {
286   8:               SEQUENCE {
288   3:                 OBJECT IDENTIFIER organizationName (2 5 4 10)
293   1:                 UTF8String 'O'
        :                }
        :              }
296   12:            SET {
298   10:              SEQUENCE {
300   3:                 OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
305   3:                 UTF8String '001'
        :                }
        :              }
310   11:            SET {
312   9:               SEQUENCE {
314   3:                 OBJECT IDENTIFIER commonName (2 5 4 3)
319   2:                 UTF8String '12'
```

```
        :                 }
        :               }
        :             }
 323   89:           SEQUENCE {
 325   19:             SEQUENCE {
 327    7:               OBJECT IDENTIFIER ecPublicKey (1 2 840 10045 2 1)
 336    8:               OBJECT IDENTIFIER '1 2 156 10197 1 301'
        :               }
 346   66:             BIT STRING
        :               04 07 77 FC 92 42 CF 15 46 06 D3 7E 1D 30 54 CD
        :               10 6E 33 F7 B4 3A 58 9F B9 0C 33 9F C0 CE FE 1E
        :               FA 6A CC 87 53 9B C9 A1 3D 7C 0F 43 63 C0 80 D9
        :               E3 5A 11 9E A7 13 48 A1 D1 A2 F2 86 A6 FA 8C C1
        :               A1
        :             }
 414  348:           [3] {
 418  344:             SEQUENCE {
 422   11:               SEQUENCE {
 424    3:                 OBJECT IDENTIFIER keyUsage (2 5 29 15)
 429    4:                 OCTET STRING, encapsulates {
 431    2:                   BIT STRING 5 unused bits
        :                     '111'B
        :                   }
        :                 }
 435    9:               SEQUENCE {
 437    3:                 OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 442    2:                 OCTET STRING, encapsulates {
 444    0:                   SEQUENCE {}
        :                   }
        :                 }
 446   19:               SEQUENCE {
 448    3:                 OBJECT IDENTIFIER extKeyUsage (2 5 29 37)
 453   12:                 OCTET STRING, encapsulates {
 455   10:                   SEQUENCE {
 457    8:                     OBJECT IDENTIFIER
        :                       codeSigning (1 3 6 1 5 5 7 3 3)
        :                     }
        :                   }
        :                 }
 467  145:               SEQUENCE {
 470    3:                 OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
 475  137:                 OCTET STRING, encapsulates {
 478  134:                   SEQUENCE {
 481   61:                     SEQUENCE {
```

```
483   59:                    [0] {
485   57:                      [0] {
487   55:                        [6]
      :                    'http://127.0.0.1:7070/crl/UtrustSecCA/UtrustSecC'
      :                    'A_0.crl'
      :                        }
      :                      }
      :                    }
544   69:                  SEQUENCE {
546   67:                    [0] {
548   65:                      [0] {
550   63:                        [6]
      :                    'http://127.0.0.1:7070/crl/UtrustSecCA/inc/Utrust'
      :                    'SecCA_inc_0.crl'
      :                        }
      :                      }
      :                    }
      :                  }
      :                }
615   29:              SEQUENCE {
617    3:                OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
622   22:                OCTET STRING, encapsulates {
624   20:                  OCTET STRING
      :                  0A 2E AB D9 2D EC A9 B3 2C E4 03 38 07 63 D2 4E
      :                  58 E2 4B 63
      :                  }
      :                }
646   31:              SEQUENCE {
648    3:                OBJECT IDENTIFIER
      :                  authorityKeyIdentifier (2 5 29 35)
653   24:                OCTET STRING, encapsulates {
655   22:                  SEQUENCE {
657   20:                    [0]
      :                    CC 67 24 42 1A C3 F8 AD 27 AC D2 F2 E5 84 81 1F
      :                    E9 C5 2F AE
      :                    }
      :                  }
      :                }
679   85:              SEQUENCE {
681    3:                OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
686   78:                OCTET STRING, encapsulates {
688   76:                  SEQUENCE {
690   74:                    SEQUENCE {
```

```
692   10:                        OBJECT IDENTIFIER '1 2 156 112562 6 4 1 1'
704   60:                        SEQUENCE {
706   58:                          SEQUENCE {
708    8:                            OBJECT IDENTIFIER cps (1 3 6 1 5 5 7 2 1)
718   46:                            IA5String
       :                      'http://127.0.0.1:7070/cps/UtrustSecCA/cps.html'
       :                            }
       :                          }
       :                        }
       :                      }
       :                    }
       :                  }
       :                }
       :              }
       :            }
766   12:          SEQUENCE {
768    8:            OBJECT IDENTIFIER '1 2 156 10197 1 501'
778    0:            NULL
       :            }
780   72:          BIT STRING, encapsulates {
783   69:            SEQUENCE {
785   33:              INTEGER
       :                00 87 AB 1D E6 51 06 E9 8E A4 EE D8 1B 90 C2 0E
       :                87 10 15 39 DC B7 B4 49 39 D2 79 A2 79 F4 A8 F5
       :                91
820   32:              INTEGER
       :                74 13 72 97 AB A4 CC 38 02 06 52 97 16 62 B2 21
       :                6B 4A 81 D9 6E 8A D0 EA D9 72 F1 7C 91 0C 52 CF
       :              }
       :            }
       :          }
       :        }
854  230:      SET {
857  227:        SEQUENCE {
860    1:          INTEGER 1
863  120:          SEQUENCE {
865   98:            SEQUENCE {
867   11:              SET {
869    9:                SEQUENCE {
871    3:                  OBJECT IDENTIFIER countryName (2 5 4 6)
876    2:                  UTF8String 'CN'
       :                  }
       :                }
880   13:              SET {
882   11:                SEQUENCE {
884    3:                  OBJECT IDENTIFIER organizationName (2 5 4 10)
```

```
 889    4:                     UTF8String 'Test'
        :                     }
        :                   }
 895   27:                 SET {
 897   25:                   SEQUENCE {
 899    3:                     OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
 904   18:                     UTF8String '12140000405703345Q'
        :                     }
        :                   }
 924   39:                 SET {
 926   37:                   SEQUENCE {
 928    3:                     OBJECT IDENTIFIER commonName (2 5 4 3)
 933   30:                     UTF8String '.............................'
        :                     }
        :                   }
        :                 }
 965   18:               INTEGER 21 05 70 34 0F 6D 32 45 17 8F 0C 10 E5 61 91 88 6D BF
        :               }
 985   13:             SEQUENCE {
 987    9:               OBJECT IDENTIFIER '1 2 156 10197 1 401 1'
 998    0:               NULL
        :               }
1000   13:             SEQUENCE {
1002    9:               OBJECT IDENTIFIER '1 2 156 10197 1 301 1'
1013    0:               NULL
        :              }
1015   70:             OCTET STRING, encapsulates {
1017   68:               SEQUENCE {
1019   32:                 INTEGER
        :                   71 1A B6 79 5F E7 97 A8 C6 66 81 13 BC 55 10 78
        :                   94 4A 7E B9 0E 54 EB 6D F4 51 74 0A 87 A6 A1 30
1053   32:                 INTEGER
        :                   78 44 5E FB 53 B6 AB D7 CD 27 BA 61 EB C1 F5 46
        :                   D6 1C B9 48 9C 0C 8F E2 AD B7 D7 4E 18 E5 5A 3A
        :                 }
        :               }
        :             }
        :           }
        :         }
        :       }
        :     }
        :   }
```

A.2 数字信封类型示例

```
  0  317: SEQUENCE {
  4   10:   OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 3'
```

```
 16  301:  [0] {
 20  297:    SEQUENCE {
 24    1:      INTEGER 1
 27  229:      SET {
 30  226:        SEQUENCE {
 33    1:          INTEGER 1
 36   82:          SEQUENCE {
 38   68:            SEQUENCE {
 40   11:              SET {
 42    9:                SEQUENCE {
 44    3:                  OBJECT IDENTIFIER countryName (2 5 4 6)
 49    2:                  PrintableString 'CN'
       :                  }
       :                }
 53   13:              SET {
 55   11:                SEQUENCE {
 57    3:                  OBJECT IDENTIFIER organizationName (2 5 4 10)
 62    4:                  UTF8String 'Test'
       :                  }
       :                }
 68   13:              SET {
 70   11:                SEQUENCE {
 72    3:                  OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
 77    4:                  UTF8String 'Test'
       :                  }
       :                }
 83   23:              SET {
 85   21:                SEQUENCE {
 87    3:                  OBJECT IDENTIFIER commonName (2 5 4 3)
 92   14:                  UTF8String 'Beijing SM2 CA'
       :                  }
       :                }
       :              }
108   10:            INTEGER 1A 10 00 00 00 00 00 05 B3 2B
       :            }
120   13:          SEQUENCE {
122    9:            OBJECT IDENTIFIER '1 2 156 10197 1 301 3'
133    0:            NULL
       :            }
135  122:          OCTET STRING, encapsulates {
137  120:            SEQUENCE {
139   32:              INTEGER
       :                37 90 1C FE B5 A4 16 1D 2C 34 7B 01 A8 FD 99 49
       :                BA CA 2E 46 5C 34 DD 37 59 20 6B 79 0E 09 3D 4F
```

```
173  32:                INTEGER
       :                  6F 2F CD B0 B5 80 08 E4 17 A0 62 E8 08 1E A1 46
       :                  2D 69 57 E7 54 DD 98 69 7E 93 92 85 F5 F0 05 7F
207  32:                OCTET STRING
       :                  7B AF A0 52 77 94 7F 6E 71 CD 76 F8 1D B3 E3 CA
       :                  20 F9 05 36 F5 43 B7 F9 07 4D 78 79 C8 25 C2 3D
241  16:                OCTET STRING CB 54 9E C6 13 C7 B7 B3 1D 1B ED 9E BC 9A 11 9D
       :                }
       :              }
       :            }
       :          }
259  60:        SEQUENCE {
261  10:          OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 1'
273  28:          SEQUENCE {
275   8:            OBJECT IDENTIFIER '1 2 156 10197 1 104 2'
285  16:            OCTET STRING 5B 10 A3 84 00 15 BD 11 F6 2E E3 7F 5C 79 B4 1B
       :            }
303  16:          [0] 9D 13 CE E5 01 B7 90 87 89 4B 64 0B C6 F9 C0 3F
       :          }
       :        }
       :      }
       :    }
```

A.3 加密数据类型示例

```
  0  97: SEQUENCE {
  2  10:   OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 5'
 14  83:   [0] {
 16  81:     SEQUENCE {
 18   1:       INTEGER 1
 21  76:       SEQUENCE {
 23  10:         OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 1'
 35  28:         SEQUENCE {
 37   8:           OBJECT IDENTIFIER '1 2 156 10197 1 104 2'
 47  16:           OCTET STRING 5D 6F 43 0A 9F AF D4 7C DA D6 3D 4D 56 1C 15 7E
       :           }
 65  32:         [0]
       :           B6 E6 61 AC 2C 6E 10 91 7D 93 9C 9B 3B E4 7E 94
       :           05 4F 30 C6 13 ED B6 35 B4 57 1A 7A 59 F2 8A C8
       :         }
       :       }
       :     }
       :   }
```

A.4 密钥协商类型示例

```
0 608: SEQUENCE {
```

```
  4   10:  OBJECT IDENTIFIER '1 2 156 10197 6 1 4 2 6'
 16  592:  [0] {
 20  588:    SEQUENCE {
 24    1:      INTEGER 1
 27   66:      BIT STRING
        :        04 E2 B0 3C 32 60 5A 6A 92 23 07 68 5D 55 0A C0
        :        D9 15 32 D2 72 39 43 62 B6 AF 40 17 1C C5 67 2A
        :        AF EF 1F 88 C3 5C DC 51 62 6A F5 60 06 4D A5 68
        :        A1 28 12 19 CB 9D 6A A4 A9 7F 56 DC 62 E0 46 C6
        :        51
 95  495:      [0] {
 99  404:        SEQUENCE {
103    3:          [0] {
105    1:            INTEGER 2
        :            }
108   11:          INTEGER 20 00 02 02 00 32 41 71 65 94 54
121   12:          SEQUENCE {
123    8:            OBJECT IDENTIFIER '1 2 156 10197 1 501'
133    0:            NULL
        :            }
135   71:          SEQUENCE {
137   11:            SET {
139    9:              SEQUENCE {
141    3:                OBJECT IDENTIFIER countryName (2 5 4 6)
146    2:                PrintableString 'CN'
        :                }
        :              }
150   13:            SET {
152   11:              SEQUENCE {
154    3:                OBJECT IDENTIFIER organizationName (2 5 4 10)
159    4:                UTF8String 'ROOT'
        :                }
        :              }
165   13:            SET {
167   11:              SEQUENCE {
169    3:                OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
174    4:                UTF8String 'ROOT'
        :                }
        :              }
180   26:            SET {
182   24:              SEQUENCE {
184    3:                OBJECT IDENTIFIER commonName (2 5 4 3)
189   17:                UTF8String 'PublicTrustCA0001'
        :                }
```

```
        :             }
        :           }
208   30:         SEQUENCE {
210   13:           UTCTime 24/03/2020 09:16:59 GMT
225   13:           UTCTime 24/03/2021 09:16:59 GMT
        :           }
240   48:         SEQUENCE {
242   11:           SET {
244    9:             SEQUENCE {
246    3:               OBJECT IDENTIFIER countryName (2 5 4 6)
251    2:               PrintableString 'CN'
        :               }
        :             }
255   16:           SET {
257   14:             SEQUENCE {
259    3:               OBJECT IDENTIFIER stateOrProvinceName (2 5 4 8)
264    7:               UTF8String 'BeiJing'
        :               }
        :             }
273   15:           SET {
275   13:             SEQUENCE {
277    3:               OBJECT IDENTIFIER commonName (2 5 4 3)
282    6:               UTF8String 'usrcrt'
        :               }
        :             }
        :           }
290   89:         SEQUENCE {
292   19:           SEQUENCE {
294    7:             OBJECT IDENTIFIER ecPublicKey (1 2 840 10045 2 1)
303    8:             OBJECT IDENTIFIER '1 2 156 10197 1 301'
        :             }
313   66:           BIT STRING
        :             04 BA 6C B3 83 FC 5F 05 25 EE BC D2 37 FD 0B 47
        :             DA 82 00 2E 5B 4C A8 27 6A 2B 0A 04 52 19 E0 6E
        :             35 E0 1C 73 D8 FE A0 11 3F B7 67 D7 D2 86 39 50
        :             F4 F7 54 C5 62 A5 2F CA 54 DB 4C D1 96 6E FE 7D
        :             A8
        :           }
381  124:         [3] {
383  122:           SEQUENCE {
385   29:             SEQUENCE {
387    3:               OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
392   22:               OCTET STRING, encapsulates {
394   20:                 OCTET STRING
```

```
        :                     50 88 E8 30 36 E0 67 8A 25 E1 5C C5 50 A8 57 70
        :                     5D 92 65 8F
        :                   }
        :                 }
416     9:                SEQUENCE {
418     3:                  OBJECT IDENTIFIER basicConstraints (2 5 29 19)
423     2:                  OCTET STRING, encapsulates {
425     0:                    SEQUENCE {}
        :                     }
        :                   }
427    11:                SEQUENCE {
429     3:                  OBJECT IDENTIFIER keyUsage (2 5 29 15)
434     4:                  OCTET STRING, encapsulates {
436     2:                    BIT STRING
        :                       '10011100'B
        :                     }
        :                   }
440    17:                SEQUENCE {
442     9:                  OBJECT IDENTIFIER
        :                     netscape-cert-type (2 16 840 1 113730 1 1)
453     4:                  OCTET STRING, encapsulates {
455     2:                    BIT STRING
        :                       '11111111'B
        :                     }
        :                   }
459    29:                SEQUENCE {
461     3:                  OBJECT IDENTIFIER extKeyUsage (2 5 29 37)
466    22:                  OCTET STRING, encapsulates {
468    20:                    SEQUENCE {
470     8:                      OBJECT IDENTIFIER serverAuth (1 3 6 1 5 5 7 3 1)
480     8:                      OBJECT IDENTIFIER clientAuth (1 3 6 1 5 5 7 3 2)
        :                       }
        :                     }
        :                   }
490    15:                SEQUENCE {
492     8:                  OBJECT IDENTIFIER '2 16 840 1 113732 2'
502     3:                  OCTET STRING, encapsulates {
504     1:                    UTF8String '1'
        :                     }
        :                   }
        :                 }
        :               }
        :             }
507    12:           SEQUENCE {
```

```
509    8:           OBJECT IDENTIFIER '1 2 156 10197 1 501'
519    0:           NULL
       :           }
521   71:         BIT STRING, encapsulates {
524   68:           SEQUENCE {
526   32:             INTEGER
       :               34 09 D4 F0 54 E8 E6 A8 2C C1 1A 5E 46 5A 20 86
       :               A5 11 A2 E1 38 C4 E9 6C D1 3F A8 FF EF 91 BD 5E
560   32:             INTEGER
       :               2B 9E F3 35 0A EE 57 3C 7A 30 64 89 9D E8 19 A1
       :               10 35 38 70 3C C4 FE 28 50 4F A5 F9 01 13 48 4B
       :             }
       :           }
       :         }
594   16:       OCTET STRING '1234567812345678'
       :       }
       :     }
       :   }
```

附 录 B
（规范性）
SM2 隐式证书的加密签名消息语法

B.1 隐式证书签名数据类型 imcSignedData

B.1.1 ImcSignedData 类型

imcSignedData 数据类型由任意类型的数据和至少一个签名者的签名值组成。任意类型的数据能同时被任意数量的签名者签名。

imcSignedData 数据类型结构定义如下：

```
ImcSignedData ::= SEQUENCE {
        version Version,
        digestAlgorithms DigestAlgorithmIdentifiers,
        contentInfo ContentInfo,
        certificates[0] IMPLICIT ImcCertificates OPTIONAL,
        crls[1] IMPLICIT ImcCertificateRevocationLists OPTIONAL,
        signerInfos SignerInfos
}
ImcCertificates::= SET OF OCTET STRING  --OCTET STRING 包含一个隐式证书的编码
ImcCertificateRevocationLists::= SET OF OCTET STRING  --OCTET STRING 包含一个隐式证书的 CRL 的编码
ImcSignerInfos::= SET OF ImcSignerInfo
```

结构中各项含义见表 B.1。

表 B.1 ImcSignedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 2
digestAlgorithms	DigestAlgorithmIdentifiers	消息摘要算法标识符的集合
contentInfo	ContentInfo	数据内容
certificates	ImcCertificates	隐式证书的集合，可选
crls	ImcCertificateRevocationLists	隐式证书撤销列表的集合，可选
signInfos	ImcSignerInfos	签名者信息的集合

B.1.2 ImcSignerInfo 类型

ImcSignerInfo 类型结构定义如下：

```
ImcSignerInfo ::= SEQUENCE {
        version Version,
        sid CertificateId,
        digestAlgorithm DigestAlgorithmIdentifier,
```

```
        authenticatedAttributes[0] IMPLICIT Attributes OPTIONAL,
        digestEncryptionAlgorithm DigestEncryptionAlgorithmIdentifier,
        encryptedDigest EncryptedDigest,
        unauthenticatedAttributes [1] IMPLICIT Attributes OPTIONAL
}
EncryptedDigest ::=OCTET STRING
CertificateId::= SEQUENCE {
  hashAlgorithm                DigestAlgorithmIdentifier,
  truncatedCertHash            OCTET STRING,
}
```

结构中各项含义见表 B.2。

表 B.2　ImcSignerInfo 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 2
sid	CertificateId	隐式证书的 Hash 值
digestAlgorithm	DigestAlgorithmIdentifier	对内容进行摘要计算的消息摘要算法，本文件采用 SM3 密码算法
authenticatedAttributes	Attributes	是经由签名者签名的属性的集合，该域可选。如果该域存在，该域中摘要的计算方法是对原文进行摘要计算结果
digestEncryptionAlgorithm	DigestEncryptionAlgorithmIdentifier	SM2 隐式证书公钥签名机制标识符
encryptedDigest	OCTET STRING	值是 SM2Signature，用签名者私钥进行签名的结果，其定义应遵循 GM/T0009
unauthenticatedAttributes	Attributes	不进行签名的属性的集合，该域可选

B.2　隐式证书数字信封数据类型 imcEnvelopedData

B.2.1　ImcEnvelopedData 类型

数字信封 imcEnvelopedData 数据类型由加密数据和至少一个接收者的数据加密密钥的密文组成。其中，加密数据是用数据加密密钥加密的，数据加密密钥是用接收者的公钥加密的。

该类型用于为接收者的 data、digestedData 或 imcSignedData 三种类型的数据做数字信封。

imcEnvelopedData 数据类型结构定义如下：

```
ImcEnvelopedData ::= SEQUENCE {
        version Version,
        recipientInfos imcRecipientInfos,
        encryptedContentInfo EncryptedContentInfo
}
imcRecipientInfos ::= SET OF imcRecipientInfo
```

结构中各项含义见表 B.3。

表 B.3 ImcEnvelopedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 2
recipientInfos	imcRecipientInfos	接收者信息的集合，至少要有一个接收者
encryptedContentInfo	EncryptedContentInfo	被加密的内容信息

B.2.2 ImcRecipientInfo 类型

接收者信息用 ImcRecipientInfo 类型表示。

ImcRecipientInfo 类型结构定义如下：

```
ImcRecipientInfo::= SEQUENCE{
        version Version,
        rid CertificateId,
        keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
        encryptedKey OCTET STRING
}
```

结构中各项含义见表 B.4。

表 B.4 RecipientInfo 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 2
rid	CertificateId	隐式证书的 Hash 值
keyEncryptionAlgorithm	KeyEncryptionAlgorithmIdentifier	用接收者公钥加密数据加密密钥的算法，为 SM2 隐式证书公钥加密机制
encryptedKey	OCTET STRING	数据加密密钥密文 SM2cipher，其定义应遵循 GM/T 0009

B.3 隐式证书签名及数字信封数据类型 imcSignedAndEnvelopedData

imcSignedAndEnvelopedData 数据类型由任意类型的加密数据、至少一个接收者的数据加密密钥和至少一个签名者的签名组成。

imcSignedAndEnvelopedData 数据类型结构定义如下：

```
ImcSignedAndEnvelopedData ::= SEQUENCE {
        version Version,
        recipientInfos ImcRecipientInfos,
        digestAlgorithms DigestAlgorithmIdentifiers,
        encryptedContentInfo EncryptedContentInfo,
    certificates[0] IMPLICIT ImcCertificates OPTIONAL,
    crls[1] IMPLICIT ImcCertificateRevocationLists OPTIONAL,
          signerInfos ImcSignerInfos
}
```

结构中各项含义见表 B.5。

表 B.5　ImcSignedAndEnvelopedData 数据类型

字段名称	数据类型	含义
version	Version	版本号，此处取值为 2
recipientInfos	RecipientInfos	接受者信息的集合，至少一个元素
digestAlgorithms	DigestAlgorithmIdentifiers	消息摘要算法标识符的集合
encryptedContentInfo	EncryptedContentInfo	被加密的内容，可是任何定义的数据类型
certificates	ImcCertificates	隐式证书的集合，可选
crls	ImcCertificateRevocationLists	隐式证书撤销列表的集合，可选
signerInfos	SignerInfos	签名者的集合，至少要有一个元素

附 录 C
（规范性）
SM2 密钥格式

C.1 椭圆曲线参数语法

定义如下：

```
Parameters ::= CHOICE {
        ecParameters ECParameters,
        namedCurve ObjectIdentifier,
        implicitlyCA NULL }
```

在用于 SM2 密码算法表达时，只使用 namedCurve 这一种表达方法，SM2 密码算法曲线定义的 OID 应符合 GM/T 0006。

C.2 公钥语法

定义如下：

```
SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm AlgorithmIdentifier {{ECPKAlgorithms}},
    subjectPublicKey SM2PublicKey
}
```

其中：

algorithm 定义了公钥的类型

subjectPublicKey 定义了公钥的实际值

AlgorithmIdentifier 是对象标识与参数的绑定，其定义如下：

```
AlgorithmIdentifier ::= SEQUENCE {
    algorithm OBJECT IDENTIFIER,
    parameters ANY DEFINED BY algorithm OPTIONAL
}
```

对于 SM2 密码算法和 SM2 隐式证书公钥机制，其 OID(algorithm)定义应符合 GM/T 0006。

C.3 私钥语法

定义如下：

```
ECPrivateKey{CURVES:IOSet} ::= SEQUENCE {
    version INTEGER { ecPrivkeyVer1(1) } (ecPrivkeyVer1),
    privateKey SM2PrivateKey,
    parameters [0] Parameters{{IOSet}} OPTIONAL,
    publicKey [1] SM2PublicKey
}
```

其中：

version 指定了私钥的版本号，这里使用整数 1 来表示 SM2 私钥的版本号。