



INTERNATIONAL STANDARD ISO/IEC 19785-4:2010
TECHNICAL CORRIGENDUM 1

Published 2013-11-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION
INTERNATIONAL ELECTROTECHNICAL COMMISSION • МЕЖДУНАРОДНАЯ ЭЛЕКТРОТЕХНИЧЕСКАЯ КОМИССИЯ • COMMISSION ÉLECTROTECHNIQUE INTERNATIONALE

Information technology — Common Biometric Exchange Formats Framework —

Part 4: Security block format specifications

TECHNICAL CORRIGENDUM 1

Technologies de l'information — Cadre de formats d'échange biométriques communs —

Partie 4: Spécifications de format de bloc de sécurité

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to ISO/IEC 19785-4:2010 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 37, *Biometrics*.

1. Add the following description at the end of 5.4:

7. (0007 Hex). This has been registered in accordance with ISO/IEC 19785-2 when BER encodings (See ISO/IEC 8825-1) are applied.

2. Add the following subclause after 5.5.3:

5.5.4 The case of BER encodings

{iso registration-authority cbeff(19785) organizations(0) jtc-sc37 (257) sb-formats(3)
general-purpose(0) ber-encoding(7)}

ICS 35.040

Ref. No. ISO/IEC 19785-4:2010/Cor.1:2013(E)

© ISO/IEC 2013 – All rights reserved

Published in Switzerland

or, in XML value notation,

1.1.19785.0.257.3.0.7

3. Replace the definition of type `SubBlockForACBio` in 5.8.1.4 with:

```
SubBlockForACBio ::= SEQUENCE {
    bpuIOIndex INTEGER,
    acbioInstance [0] EXPLICIT ACBioInstance
}
```

4. Replace the definition of type `EnvelopeRelatedData` in 5.8.2.1.2 with:

```
EnvelopeRelatedData ::= SEQUENCE {
    version CBEFFSBVersion DEFAULT v1,
    originatorInfo [0] IMPLICIT OriginatorInfo OPTIONAL,
    recipientInfos RecipientInfos,
    encryptedContentRelatedInfo EncryptedContentRelatedInfo
}
```

5. Replace the definition of type `CBEFFSBVersion` in a) in 5.8.2.1.2 with:

```
CBEFFSBVersion ::= INTEGER { v1(1) } ( v1, ... )
```

6. Replace d) in 5.8.2.1.2 with:

d) The field `encryptedContentRelatedInfo` of type `EncryptedContentRelatedInfo` consists of two fields as follows:

```
EncryptedContentRelatedInfo ::= SEQUENCE {
    contentType IdentifierEncryptedContent,
    contentEncryptionAlgorithm ContentEncryptionAlgorithmIdentifier
}
```

where the field `contentType` identifies the content type of the encrypted content with type `IdentifierEncryptedContent` and the field `contentEncryptionAlgorithm` identifies the content-encryption algorithm with any associated parameters, used to encrypt the biometric data. The same content-encryption algorithm and content-encryption key are used for all recipients.

`IdentifierEncryptedContent` is defined as follows:

```
IdentifierEncryptedContent OBJECT IDENTIFIER ::= { id-cbeffbDB }
```

7. Replace the definition of type `EncryptionRelatedData` in 5.8.2.2.1 with:

```
EncryptionRelatedData ::= SEQUENCE {
    version CBEFFSBVersion DEFAULT v1,
    encryptedContentRelatedInfo EncryptedContentRelatedInfo
}
```

8. Replace the definition of type `SignatureRelatedData` in 5.8.3.1.2 with:

```
SignatureRelatedData ::= SEQUENCE {
```

```

    version CBEFFSBVersion DEFAULT v1,
    digestAlgorithms DigestAlgorithmIdentifiers,
    encapContentRelatedInfo EncapsulatedContentRelatedInfo,
    certificates [0] IMPLICIT CertificateSet OPTIONAL,
    crls [1] IMPLICIT RevocationInfoChoices OPTIONAL,
    signerInfos SignerInfos
}

```

9. Add the following description after e) in 5.8.3.1.2:

f) The field `encapContentRelatedInfo` of type `EncapsulatedContentRelatedInfo` is defined as follows:

```

EncapsulatedContentRelatedInfo ::= SEQUENCE {
    eContentType          IdentifierEncapsulatedContent
}

```

where the field `eContentType` identifies the content type of the content signed with type `IdentifierEncapsulatedContent`, defined as follows:

```

IdentifierEncapsulatedContent OBJECT IDENTIFIER ::= {id-cbeffSBHAndBDB }

```

10. Replace the definition of type `AuthenticationRelatedData` in 5.8.3.2.2 with:

```

AuthenticationRelatedData ::= SEQUENCE {
    version CBEFFSBVersion DEFAULT v1,
    originatorInfo [0] IMPLICIT OriginatorInfo OPTIONAL,
    recipientInfos RecipientInfos,
    macAlgorithm MessageAuthenticationCodeAlgorithm,
    encapContentRelatedInfo EncapsulatedContentRelatedInfo,
    mac MessageAuthenticationCode
}

```

11. Add the following description after e) in 5.8.3.2.2 as follows:

f) The field `encapContentRelatedInfo` is of type `EncapsulatedContentRelatedInfo`, which is defined in 5.8.3.1.2.

12. Add the following description at the end of 6.4:

8 (0008 Hex). This has been registered in accordance with ISO/IEC 19785-2 when BER encodings (See ISO/IEC 8825-1) are applied.

13. Add the following subclause after 6.5.3:

6.5.4 The case of BER encodings

```

{iso registration-authority cbeff(19785) biometric-organization(0) jtc1-sc37(257) sb-
formats(3) signature-only(2) ber-encoding(8)}

```

or, in XML value notation,

1.1.19785.0.257.3.2.8

14. Replace the ASN.1 module in A.1 with:

```

CBEFF-GENERAL-PURPOSE-SECURITY-BLOCK
    {iso(1) standard(0) cbeff(19785) module(0) sb(16) rev(0)}
DEFINITIONS AUTOMATIC TAGS ::= BEGIN
IMPORTS
-- RFC 3852 Cryptographic Message Syntax
    ContentEncryptionAlgorithmIdentifier,
    DigestAlgorithmIdentifiers, SignerInfos, MessageAuthenticationCodeAlgorithm,
    DigestAlgorithmIdentifier, AuthAttributes, MessageAuthenticationCode,
    OriginatorInfo, RecipientInfos
    FROM CryptographicMessageSyntax2004 {
        iso(1) member-body(2) us(840) rsadsi(113549)
        pkcs(1) pkcs-9(9) smime(16) modules(0) cms-2004(24)}
-- ISO/IEC 24761 Authentication context for biometrics
    ACBioInstance, CertificateSet, RevocationInfoChoices
    FROM AuthenticationContextForBiometrics {
        iso(1) standard(0) acbio(24761) module(1) acbio(2) rev(0)} ;

CONTENT-TYPE ::= TYPE-IDENTIFIER

CBEFFSecurityBlock ::= SEQUENCE OF CBEFFSecurityBlockElement

CBEFFSecurityBlockElement ::= CHOICE {
    elementCBEFFSB ContentInfoCBEFFSB,
    subBlockForACBio SubBlockForACBio,
    accumulatedACBioInstances ACBioInstances
}

ContentInfoCBEFFSB ::= SEQUENCE {
    contentType CONTENT-TYPE.&id({ContentTypeCBEFF}),
    content [0] EXPLICIT CONTENT-TYPE.&Type
        ({ContentTypeCBEFF}{@contentType})
}

ContentTypeCBEFF CONTENT-TYPE ::= { envelopeRelatedData | encryptionRelatedData |
    signatureRelatedData | authenticationRelatedData}

EnvelopeRelatedData ::= SEQUENCE {
    version CBEFFSBVersion DEFAULT v1,
    originatorInfo [0] IMPLICIT OriginatorInfo OPTIONAL,
    recipientInfos RecipientInfos,
    encryptedContentRelatedInfo EncryptedContentRelatedInfo
}

CBEFFSBVersion ::= INTEGER { v1(1) } ( v1, ... )

EncryptedContentRelatedInfo ::= SEQUENCE {
    contentType IdentifierEncryptedContent,
    contentEncryptionAlgorithm ContentEncryptionAlgorithmIdentifier
}

IdentifierEncryptedContent OBJECT IDENTIFIER ::= { id-cbeffbDB }

EncryptionRelatedData ::= SEQUENCE {
    version CBEFFSBVersion DEFAULT v1,
    encryptedContentRelatedInfo EncryptedContentRelatedInfo
}

SignatureRelatedData ::= SEQUENCE {
    version CBEFFSBVersion DEFAULT v1,
    digestAlgorithms DigestAlgorithmIdentifiers,
    encapContentRelatedInfo EncapsulatedContentRelatedInfo,
    certificates [0] IMPLICIT CertificateSet OPTIONAL,
    crls [1] IMPLICIT RevocationInfoChoices OPTIONAL,
    signerInfos SignerInfos
}

EncapsulatedContentRelatedInfo ::= SEQUENCE {
    eContentType IdentifierEncapsulatedContent
}

```