INTERNATIONAL STANDARD ISO/IEC 24727-3:2008
TECHNICAL CORRIGENDUM 1

Published 2010-09-15

Identification cards — Integrated circuit card programming interfaces —

Part 3: **Application interface**

TECHNICAL CORRIGENDUM 1

Cartes d'identification — Interfaces programmables de cartes à puce —

Partie 3: Interface d'application

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to ISO/IEC 24727-3:2008 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 17, *Cards and personal identification*.

The machine-readable listings of the ISO24727-3-ALGO and ISO24727-3-API ASN.1 modules are available as separate files in the same file directory as this document. The files are named “ISO24727-3-ALGO.asn” and “ISO24727-3-API.asn”.

Page 13, 5.4.2

In the third paragraph, second sentence, delete “within the card-application”.

Page 14, 5.4.3

In the first paragraph, replace the first sentence with the following:

“The Authentication Protocol shall be the mechanism used by the client-application to set the authentication state for the differential-identity indicated by the DIDName.”

ICS 35.240.15 **Ref. No. ISO/IEC 24727-3:2008/Cor.1:2010(E)**

Published in Switzerland

Page 15, 5.4.3

In the fourth paragraph, replace the first sentence with the following:

"Local differential-identity authentication states are valid for the connection handle specified during their authentication. Local authentication states shall be set to FALSE when the CardApplicationDisconnect action is requested for that connection handle or the connection handle otherwise becomes invalid."

Page 15, 5.4.3

In the fourth paragraph, replace the second sentence with the following:

"Global authentication states shall be valid for all connections to card-applications managed by the alpha card-application. Global authentication states remain in effect until the connection handle that was used to establish the global authentication state is disconnected or otherwise becomes invalid."

Page 22, 7.4.1

Add the following sentence at the end of the first paragraph:

"Only one session may be established per connection handle."

Page 22, 7.4.5

Add "`API_ACTIVE_SESSION`" immediately below "`API_SECURITY_CONDITION_NOT_SATISFIED`".

Page 145, B.1

Add the following text at the end of B.1:

An ASN.1 module, "ISO24727-3-ALGO" is defined as the following:

```
ISO24727-3-ALGO {iso(1) standard(0) iso24727(24727) part3(3) annexB (13) }
-- Version 1.3, 03-Mar-2010
--
-- IF-PROFILE value '01'
--
-- *According to ISO/IEC 24727-2, the optional IF-PROFILE field in the CCD is
-- used to indicate that a card provides an implementation of ISO/IEC 24727-3.

-- © ISO/IEC 2008-2010
-- All rights reserved. Unless otherwise specified, no part of this publication
-- may be reproduced or utilized in any form or by any means, electronic or
-- mechanical, including photocopying and microfilm, without permission in
-- writing from either ISO at the address below or ISO's member body in the
-- country of the requester.
--
--    ISO copyright office
--      Case postale 56 • CH-1211 Geneva 20
--      Tel. + 41 22 749 01 11
--      Fax + 41 22 749 09 47
--      E-mail copyright@iso.org
--      Web www.iso.org

DEFINITIONS AUTOMATIC TAGS EXTENSIBILITY IMPLIED ::=
BEGIN
-- EXPORTS; Exports all
IMPORTS;

-- Major and Minor Revision values for this ASN.1 Module
revMajISO24727-3-ALGO INTEGER ::= 1
revMinISO24727-3-ALGO INTEGER ::= 3
```

```
AlgorithmIDParameters ::= SEQUENCE {
  algorithm
    ALGORITHMIDENTIFIERPARAMETERS.&id({SupportedAlgorithms}),
  parameters
    ALGORITHMIDENTIFIERPARAMETERS.&Type
      ({SupportedAlgorithms}{@algorithm}) OPTIONAL}

ALGORITHMIDENTIFIERPARAMETERS ::= CLASS {&id    OBJECT IDENTIFIER,
                                         &Type OPTIONAL }
WITH SYNTAX {ID &id
             [TYPE &Type]}

-- Algorithm OIDs

-- Add an Unknown
id-unknownAlgorithmIdentifier OBJECT IDENTIFIER ::=
     {iso(1) standard(0) iso24727(24727) part3(3) annexB(13)
      algorithmIdentifiers(1) unknown(0)}

unknownAlgorithmIdentifier ALGORITHMIDENTIFIERPARAMETERS ::=
{ID    id-unknownAlgorithmIdentifier}
```

Page 146

Replace B.2.2 to B.7 with the following:

```
-- B.2.2 Symmetric Algorithms from 18033-3

id-is18033-3        OBJECT IDENTIFIER ::= { 1 0 18033 3 }
id-bc64             OBJECT IDENTIFIER ::= { 1 0 18033 3 1 }
id-bc128            OBJECT IDENTIFIER ::= { 1 0 18033 3 2 }
id-bc64-tdea        OBJECT IDENTIFIER ::= { 1 0 18033 3 1 1 }
id-bc64-misty1      OBJECT IDENTIFIER ::= { 1 0 18033 3 1 2 }
id-bc64-cast128     OBJECT IDENTIFIER ::= { 1 0 18033 3 1 3 }
id-bc128-aes        OBJECT IDENTIFIER ::= { 1 0 18033 3 2 1 }
id-bc128-camellia   OBJECT IDENTIFIER ::= { 1 0 18033 3 2 2 }
id-bc128-seed       OBJECT IDENTIFIER ::= { 1 0 18033 3 2 3 }

is18033-3 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-is18033-3 }
bc64 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc64 }
bc128 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc128 }
bc64-tdea ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc64-tdea }
bc64-misty1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc64-misty1 }
bc64-cast128 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc64-cast128 }
bc128-aes ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc128-aes }
bc128-camellia ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc128-camellia }
bc128-seed ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-bc128-seed }


--B 2.3 AES (128-bit key)
id-aes128-ECB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 1 }
id-aes128-CBC   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 2 }
id-aes128-OFB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 3 }
```

```
id-aes128-CFB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 4 }

aes128-ECB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes128-ECB }
aes128-CBC ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes128-CBC }
aes128-OFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes128-OFB }
aes128-CFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes128-CFB }


-- B.2.4 AES (192-bit key)
id-aes192-ECB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 21 }
id-aes192-CBC   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 22 }
id-aes192-OFB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 23 }
id-aes192-CFB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 24 }

aes192-ECB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes192-ECB }
aes192-CBC ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes192-CBC }
aes192-OFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes192-OFB }
aes192-CFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes192-CFB }


-- B.2.5 AES (256 bit key)
id-aes256-ECB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 41 }
id-aes256-CBC   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 42 }
id-aes256-OFB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 43 }
id-aes256-CFB   OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 1 44 }

aes256-ECB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes256-ECB }
aes256-CBC ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes256-CBC }
aes256-OFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes256-OFB }
aes256-CFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-aes256-CFB }


-- B.2.6 DES
id-des-ECB         OBJECT IDENTIFIER ::= { 1 3 14 3 2 6 }
id-des-CBC         OBJECT IDENTIFIER ::= { 1 3 14 3 2 7 }

-- Carries an IV as a parameter.

id-des-OFB         OBJECT IDENTIFIER ::= { 1 3 14 3 2 8 }

-- Carries an FBParameter as a parameter.
-- where,
--    FBParameter ::= SEQUENCE {
--        iv IV,
--        numberOfBits NumberOfBits
--    }
-- and,
--    IV ::= OCTET STRING
-- NumberOfBits ::= INTEGER
-- number of feedback bits (1-64)

des-ECB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-des-ECB }
```

```
des-CBC ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-des-CBC }
des-OFB ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-des-OFB }


-- B.2.7 Others
id-des-EDE3-CBC    OBJECT IDENTIFIER ::= { 1 2 840 113549 3 7}
id-rc2CBC          OBJECT IDENTIFIER ::= { 1 2 840 113549 3 2}
id-rc5-CBC-PAD     OBJECT IDENTIFIER ::= { 1 2 840 113549 3 9}

des-EDE3-CBC ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-des-EDE3-CBC }
rc2CBC ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-rc2CBC }
rc5-CBC-PAD ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-rc5-CBC-PAD }
```

B.3 Asymmetric Algorithms

```
-- B.3.1 Public Keys
id-rsa-public-key  OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 1 }
id-dsa-public-key  OBJECT IDENTIFIER ::= { 1 2 840 10040 4 1 }
id-ec-public-key   OBJECT IDENTIFIER ::= { 1 2 840 10045 2 1 }

rsa-public-key ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-rsa-public-key }
dsa-public-key ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dsa-public-key }
ec-public-key ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ec-public-key }


-- B.3.2 Asymmetric Encryption
id-rsa-oaep        OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 7 }

rsa-oaep ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-rsa-oaep }


-- B.3.3 Signatures
id-dsa-with-SHA-1          OBJECT IDENTIFIER ::= { 1 2 840 10040 4 3 }
id-md2WithRSAEncryption    OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 2 }
id-md5WithRSAEncryption    OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 4 }
id-sha-1WithRSAEncryption  OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 5 }
id-sha224WithRSAEncryption OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 14 }
id-sha256WithRSAEncryption OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 11 }
id-sha384WithRSAEncryption OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 12 }
id-sha512WithRSAEncryption OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 13 }
id-rsaSSA-PSS              OBJECT IDENTIFIER ::= { 1 2 840 113549 1 1 10 }
-- same OID for all hash algorithms (hash algorithm is specified as a parameter).

dsa-with-SHA-1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dsa-with-SHA-1 }
md2WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-md2WithRSAEncryption }
md5WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-md5WithRSAEncryption }
sha-1WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-sha-1WithRSAEncryption }
sha224WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-sha224WithRSAEncryption }
sha256WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-sha256WithRSAEncryption }
```

```
sha384WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-sha384WithRSAEncryption }
sha512WithRSAEncryption ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-sha512WithRSAEncryption }
rsaSSA-PSS ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-rsaSSA-PSS }
```

B.4 Elliptic Curve Algorithms

```
-- B.4.1 Elliptic Curves
id-ansiX9p192r1 OBJECT IDENTIFIER ::= { 1 2 840 10045 3 1 1 }
id-ansiX9t163k1 OBJECT IDENTIFIER ::= { 1 3 132 0 1 }
id-ansiX9t163r2 OBJECT IDENTIFIER ::= { 1 3 132 0 15 }
id-ansiX9p224r1 OBJECT IDENTIFIER ::= { 1 3 132 0 33 }
id-ansiX9t233k1 OBJECT IDENTIFIER ::= { 1 3 132 0 26 }
id-ansiX9t233r1 OBJECT IDENTIFIER ::= { 1 3 132 0 27 }
id-ansiX9p256r1 OBJECT IDENTIFIER ::= { 1 2 840 10045 3 1 7 }
id-ansiX9t283k1 OBJECT IDENTIFIER ::= { 1 3 132 0 16 }
id-ansiX9t283r1 OBJECT IDENTIFIER ::= { 1 3 132 0 17 }
id-ansiX9p384r1 OBJECT IDENTIFIER ::= { 1 3 132 0 34 }
id-ansiX9t409k1 OBJECT IDENTIFIER ::= { 1 3 132 0 36 }
id-ansiX9t409r1 OBJECT IDENTIFIER ::= { 1 3 132 0 37 }
id-ansiX9p521r1 OBJECT IDENTIFIER ::= { 1 3 132 0 35 }
id-ansiX9t571k1 OBJECT IDENTIFIER ::= { 1 3 132 0 38 }
id-ansiX9t571r1 OBJECT IDENTIFIER ::= { 1 3 132 0 39 }

ansiX9p192r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9p192r1 }
ansiX9t163k1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t163k1 }
ansiX9t163r2 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t163r2 }
ansiX9p224r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9p224r1 }
ansiX9t233k1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t233k1 }
ansiX9t233r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t233r1 }
ansiX9p256r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9p256r1 }
ansiX9t283k1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t283k1 }
ansiX9t283r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t283r1 }
ansiX9p384r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9p384r1 }
ansiX9t409k1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t409k1 }
ansiX9t409r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t409r1 }
ansiX9p521r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9p521r1 }
ansiX9t571k1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t571k1 }
ansiX9t571r1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ansiX9t571r1 }


-- B.4.2 Signatures
id-ecDSA-with-SHA-1    OBJECT IDENTIFIER ::= { 1 2 840 10045 4 1 }
id-ecDSA-with-SHA-224  OBJECT IDENTIFIER ::= { 1 2 840 10045 4 3 1 }
id-ecDSA-with-SHA-256  OBJECT IDENTIFIER ::= { 1 2 840 10045 4 3 2 }
id-ecDSA-with-SHA-384  OBJECT IDENTIFIER ::= { 1 2 840 10045 4 3 3 }
id-ecDSA-with-SHA-512  OBJECT IDENTIFIER ::= { 1 2 840 10045 4 3 4 }
```

```
ecDSA-with-SHA-1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ecDSA-with-SHA-1 }
ecDSA-with-SHA-224 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ecDSA-with-SHA-224 }
ecDSA-with-SHA-256 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ecDSA-with-SHA-256 }
ecDSA-with-SHA-384 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ecDSA-with-SHA-384 }
ecDSA-with-SHA-512 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-ecDSA-with-SHA-512 }
```

B.5 Hash Functions

```
id-md2     OBJECT IDENTIFIER ::= { 1 2 840 113549 2 2 }
id-md5     OBJECT IDENTIFIER ::= { 1 2 840 113549 2 5 }
id-sha1    OBJECT IDENTIFIER ::= { 1 3 14 3 2 26 }
id-sha224  OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 2 4 }
id-sha256  OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 2 1 }
id-sha384  OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 2 2 }
id-sha512  OBJECT IDENTIFIER ::= { 2 16 840 1 101 3 4 2 3 }

md2 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-md2 }
md5 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-md5 }
sha1 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-sha1 }
sha224 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-sha224 }
sha256 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-sha256 }
sha384 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-sha384 }
sha512 ALGORITHMIDENTIFIERPARAMETERS ::= {ID id-sha512 }
```

B.6 Message Authentication Codes

```
id-hmacWithSHA1     OBJECT IDENTIFIER ::= { 1 2 840 113549 2 7}
id-hmacWithSHA224   OBJECT IDENTIFIER ::= { 1 2 840 113549 2 8}
id-hmacWithSHA256   OBJECT IDENTIFIER ::= { 1 2 840 113549 2 9}
id-hmacWithSHA384   OBJECT IDENTIFIER ::= { 1 2 840 113549 2 10}
id-hmacWithSHA512   OBJECT IDENTIFIER ::= { 1 2 840 113549 2 11}
id-hmac-MD5         OBJECT IDENTIFIER ::= { 1 3 6 1 5 5 8 1 1 }

hmacWithSHA1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-hmacWithSHA1 }
hmacWithSHA224 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-hmacWithSHA224 }
hmacWithSHA256 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-hmacWithSHA256 }
hmacWithSHA384 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-hmacWithSHA384 }
hmacWithSHA512 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-hmacWithSHA512 }
hmac-MD5 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-hmac-MD5 }
```

B.7 Key Establishment

```
id-dhSinglePass-stdDH-sha1kdf-scheme       OBJECT IDENTIFIER ::=
                                                       { 1 3 133 16 840 63 0 2 }
id-dhSinglePass-cofactorDH-sha1kdf-scheme  OBJECT IDENTIFIER ::=
                                                       { 1 3 133 16 840 63 0 3 }
id-mqvSinglePass-sha1kdf-scheme            OBJECT IDENTIFIER ::=
                                                      { 1 3 133 16 840 63 0 16 }
```

```
id-x9-63-scheme                           OBJECT IDENTIFIER ::=
                                                 { 1 2 840 63 0 }
id-secg-scheme                            OBJECT IDENTIFIER ::= { 1 3 132 1 }
id-dhSinglePass-stdDH-sha1kdf             OBJECT IDENTIFIER ::=
                                               { 1 2 840 63 0 2 }
id-dhSinglePass-cofactorDH-sha1kdf        OBJECT IDENTIFIER ::=
                                               { 1 2 840 63 0 3 }
id-dhSinglePass-cofactorDH-recommendedKDF  OBJECT IDENTIFIER ::= { 1 3 132 1 1 }
id-dhSinglePass-cofactorDH-specifiedKDF    OBJECT IDENTIFIER ::= { 1 3 132 1 2 }
id-iso-kdf1                               OBJECT IDENTIFIER ::=
                                              { 1 0 18033 2 5 1 }

dhSinglePass-stdDH-sha1kdf-scheme ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dhSinglePass-stdDH-sha1kdf-scheme }
dhSinglePass-cofactorDH-sha1kdf-scheme ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dhSinglePass-cofactorDH-sha1kdf-scheme }
mqvSinglePass-sha1kdf-scheme ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-mqvSinglePass-sha1kdf-scheme }
x9-63-scheme ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-x9-63-scheme }
secg-scheme ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-secg-scheme }
dhSinglePass-stdDH-sha1kdf ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dhSinglePass-stdDH-sha1kdf }
dhSinglePass-cofactorDH-sha1kdf ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dhSinglePass-cofactorDH-sha1kdf }
dhSinglePass-cofactorDH-recommendedKDF ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dhSinglePass-cofactorDH-recommendedKDF }
dhSinglePass-cofactorDH-specifiedKDF ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-dhSinglePass-cofactorDH-specifiedKDF }
iso-kdf1 ALGORITHMIDENTIFIERPARAMETERS ::=
  {ID id-iso-kdf1 TYPE OBJECT IDENTIFIER}
```

Insert the following after B.7:

B.8 Enumeration of Supported Algorithms

```
SupportedAlgorithms ALGORITHMIDENTIFIERPARAMETERS ::=
  { unknownAlgorithmIdentifier |

-- B.2 Symmetric Algorithms
   is18033-3 |
   bc64 |
   bc128 |
   bc64-tdea |
   bc64-misty1 |
   bc64-cast128 |
   bc128-aes |
   bc128-camellia |
   bc128-seed |

-- B 2.3 AES (128-bit key)
   aes128-ECB |
   aes128-CBC |
   aes128-OFB |
   aes128-CFB |

-- B.2.4 AES (192-bit key)
   aes192-ECB |
   aes192-CBC |
   aes192-OFB |
   aes192-CFB |

-- B.2.5 AES (256 bit key)
   aes256-ECB |
   aes256-CBC |
   aes256-OFB |
   aes256-CFB |

-- B.2.6 DES
   des-ECB |
   des-CBC |
   des-OFB |

-- B.2.7 Others
   des-EDE3-CBC |
   rc2CBC |
   rc5-CBC-PAD |

-- B.3 Asymmetric Algorithms
-- B.3.1 Public Keys
   rsa-public-key |
   dsa-public-key |
   ec-public-key |

-- B.3.2 Asymmetric Encryption
   rsa-oaep |

-- B.3.3 Signatures
   dsa-with-SHA-1 |
   md2WithRSAEncryption |
   md5WithRSAEncryption |
   sha-1WithRSAEncryption |
   sha224WithRSAEncryption |
   sha256WithRSAEncryption |
   sha384WithRSAEncryption |
   sha512WithRSAEncryption |
   rsaSSA-PSS |

-- B.4 Elliptic Curve Algorithms
-- B.4.1 Elliptic Curves
   ansiX9p192r1 |
   ansiX9t163k1 |
   ansiX9t163r2 |
   ansiX9p224r1 |
   ansiX9t233k1 |
   ansiX9t233r1 |
   ansiX9p256r1 |
   ansiX9t283k1 |
   ansiX9t283r1 |
   ansiX9p384r1 |
   ansiX9t409k1 |
   ansiX9t409r1 |
   ansiX9p521r1 |
   ansiX9t571k1 |
   ansiX9t571r1 |

-- B.4.2 Signatures
   ecDSA-with-SHA-1 |
   ecDSA-with-SHA-224 |
   ecDSA-with-SHA-256 |
   ecDSA-with-SHA-384 |
   ecDSA-with-SHA-512 |

-- B.5 Hash Functions
   md2 |
   md5 |
   sha1 |
   sha224 |
   sha256 |
   sha384 |
   sha512 |

-- B.6 Message Authentication Codes
   hmacWithSHA1 |
   hmacWithSHA224 |
   hmacWithSHA256 |
   hmacWithSHA384 |
   hmacWithSHA512 |
   hmac-MD5 |

-- B.7 Key Establishment
   dhSinglePass-stdDH-sha1kdf-scheme |
   dhSinglePass-cofactorDH-sha1kdf-scheme |
   mqvSinglePass-sha1kdf-scheme |
   x9-63-scheme |
   secg-scheme |
   dhSinglePass-stdDH-sha1kdf |
   dhSinglePass-cofactorDH-sha1kdf |
   dhSinglePass-cofactorDH-recommendedKDF |
   dhSinglePass-cofactorDH-specifiedKDF |
   iso-kdf1 }


END
```

Page 154, C.1

Replace the content of C.1 with the following:

```
ISO24727-3-API {iso(1) standard(0) iso24727(24727) part3(3) annexC(14) }
-- Version 1.74, 05-Mar-2010
--
-- IF-PROFILE value '01'
--
-- *According to ISO/IEC 24727-2, the optional IF-PROFILE field in the CCD is
-- used to indicate that a card provides an implementation of ISO/IEC 24727-3.

-- © ISO/IEC 2008-2010
-- All rights reserved. Unless otherwise specified, no part of this publication
-- may be reproduced or utilized in any form or by any means, electronic or
-- mechanical, including photocopying and microfilm, without permission in
-- writing from either ISO at the address below or ISO's member body in the
-- country of the requester.
--
--    ISO copyright office
--      Case postale 56 • CH-1211 Geneva 20
--      Tel. + 41 22 749 01 11
--      Fax + 41 22 749 09 47
--      E-mail copyright@iso.org
--      Web www.iso.org

DEFINITIONS AUTOMATIC TAGS EXTENSIBILITY IMPLIED ::=
BEGIN
--EXPORTS (all)
IMPORTS size-max-NameLength, size-max-NodePathLength, size-max-Padding,
        size-max-SecurityCondition,
        ByteValue, URIType,
        ApplicationIdentifier, ObjectIdentifier, Name, TransactionIdentifier,
        IFDAction, IFDName, GenericHandleType, GenericIdentifierType
```

```
          FROM ISO24727-COMMON { iso(1) standard(0) iso24727(24727) }
      AlgorithmIDParameters
          FROM ISO24727-3-ALGO {iso(1) standard(0) iso24727(24727)
                                part3(3) annexB (13) };

-- Major and Minor Revision values for this ASN.1 Module
revMajISO24727-3-API INTEGER ::= 1
revMinISO24727-3-API INTEGER ::= 74
```

Pages 154–177, C.1.2 to C.2.40

Modify C.1.2 to C.2.40 as follows:

```
-- C.1.2 The Data Types
~~ByteValue ::= INTEGER (FROM (0..255) )~~
ConnectionHandle ::= INTEGER
~~ApplicationIdentifier ::= [APPLICATION 15] OCTET STRING~~
~~ObjectIdentifier ::= OBJECT IDENTIFIER~~
~~Name ::= VisibleString (SIZE(1..size-max-NameLength))~~
CardApplicationName ::= ApplicationIdentifier
CardApplicationNameList ::= SET OF CardApplicationName
DSIName ::= Name
DSINameList ::= SET OF DSIName
DSIContent ::= OCTET STRING
DataSetName ::= Name
DataSetNameList ::= SET OF DataSetName
~~ProtocolTerminationPoint ::= OCTET STRING~~
~~TransactionIdentifier ::= OCTET STRING~~
DIDScope ::= CHOICE {
local ~~[0]~~ NULL,
global ~~[1]~~ NULL
}
CardApplicationServiceLoadPackage ::= OCTET STRING
ExecuteActionRequest ::= OCTET STRING
ExecuteActionConfirmation ::= OCTET STRING
CipherBuffer ::= OCTET STRING
MessageBuffer ::= OCTET STRING
HashBuffer ::= OCTET STRING
SignatureBuffer ::= OCTET STRING
RandomDataBuffer ::= OCTET STRING

-- C.1.3 The Data Structures
DifferentialIdentityAuthenticationState ::= SEQUENCE {
dIDName DIDName,
dIDScope DIDScope,
dIDState BOOLEAN
}
SecurityCondition ::= CHOICE {
didAuthentication ~~[1]~~ DifferentialIdentityAuthenticationState,
always ~~[2]~~ BOOLEAN (~~FROM (TRUE)~~ TRUE),
never ~~[3]~~ BOOLEAN (~~FROM (FALSE)~~FALSE),
not ~~[4]~~ SecurityCondition,
and ~~[5]~~ SEQUENCE SIZE (1..size-max-SecurityCondition) OF SecurityCondition,
or ~~[6]~~ SEQUENCE SIZE (1..size-max-SecurityCondition) OF SecurityCondition
}
AccessRule ::= SEQUENCE {
cardApplicationService CardApplicationServiceName,
action ActionName,
securityCondition SecurityCondition
}
AccessControlList ::= SET OF AccessRule
~~ReaderAction ::= ENUMERATED { reset(0), unpower(1), eject(2), confiscate(3)~~
~~}~~
CardApplicationPathInfo ::= SEQUENCE {
pathSecurity PathSecurityType OPTIONAL,
-- The pathSecurity element specifies the protection
-- between the local dispatcher and the remote dispatcher
-- which is located at channelHandle.protocolTerminationPoint
channelHandle ChannelHandleType OPTIONAL,
contextHandle ContextHandleType OPTIONAL,
iFDName ~~UTF8String~~IFDName OPTIONAL,
slotIndex INTEGER OPTIONAL,
cardApplication ApplicationIdentifier
}
PathSecurityType ::= SEQUENCE {
protocol PATHSECURITYPARAMETERS.&id({SupportedPathSecurityProtocols}),
parameters PATHSECURITYPARAMETERS.&Type({SupportedPathSecurityProtocols}
{@protocol}) OPTIONAL
}
unknownPathSecurityProtocolOID OBJECT IDENTIFIER ::= { iso(1) standard(0)
iso24727(24727) part3(3)
annex-c(2) pathSecurityProtocols(0) unknown(0) }
unknownPathSecurityProtocol PATHSECURITYPARAMETERS ::= {
ID unknownPathSecurityProtocolOID}
~~TYPE NULL~~
~~}~~

SupportedPathSecurityProtocols PATHSECURITYPARAMETERS ::=
{unknownPathSecurityProtocol}
PATHSECURITYPARAMETERS ::= CLASS {
&id OBJECT IDENTIFIER,
&Type OPTIONAL
} WITH SYNTAX {ID &id
[TYPE &Type]}

ProtocolTerminationPoint ::= URIType
~~URIType ::= UTF8String~~
~~-- The URIType is a string, which represents a UIR according to RFC 3986.~~
~~-- A URI may be~~
~~-- * a URL according to RFC 1738 (e.g. http://www.example.com or~~
~~http://192.168.1.1 )~~
~~-- * a URN according to RFC 2141, which may be used to referece for example~~
~~-- * OIDs according to RFC 3061 (e.g. oid:1.0.24727.3.0.1)~~
~~-- * IETF documents according to RFC 2648, which in turn may define~~
~~-- protocols (e.g. ietf:rfc:4346 may indicate that TLS v1.1~~
~~-- shall be used to protect a channel)~~
~~-- * phone numbers according to RFC 3966~~
~~-- The list of registered name is maintained by IANA~~
~~-- (see http://www.iana.org/assignments/urn-namespaces ).~~
ChannelHandleType ::= SEQUENCE {
protocolTerminationPoint ProtocolTerminationPoint OPTIONAL,
sessionIdentifier ~~UTF8String~~GenericIdentifierType OPTIONAL,
binding URIType OPTIONAL
}
~~ContextHandleType ::= OCTET STRING~~
~~CardApplicationPathRequest ::= CardApplicationPath~~
ContextHandleType ::= GenericHandleType
CardApplicationPathSet ::= SET OF CardApplicationPathInfo
```

```
CardApplicationServiceDescription ::= ~~[APPLICATION 1101]~~ CHOICE {
serviceDescriptionText VisibleString,
serviceDescriptionURL URL
}
CertificateInfo ::= CHOICE {
efidOrPath ~~[0]~~ OCTET STRING,
certificateContent ~~[1]~~ OCTET STRING
}
CertificateType ::= CHOICE {
typeIndex INTEGER,
typeID OBJECT IDENTIFIER
}
DSI ::= SEQUENCE {
dsiName DSIName,
dsiContent DSIContent
}
DSIReference ::= SEQUENCE {
aID ApplicationIdentifier,
dataSetName DataSetName,
dsiName ~~DataSetName,~~DSIName
}
DataSet ::= SEQUENCE {
dsName DataSetName,
dsContent SEQUENCE OF DSI,
dsACL AccessControlList
}

DIDQualifier ::= CHOICE {
applicationIdentifier ApplicationIdentifier,
objectIdentifier OBJECT IDENTIFIER,
applicationFunction BIT STRING
}
DIDUpdateData ::= SEQUENCE {
marker OCTET STRING,
qualifier DIDQualifier OPTIONAL
}
DIDStructure ::= SEQUENCE {
name DIDName,
protocol OBJECT IDENTIFIER,
scope DIDScope,
authenticated BOOLEAN,
marker OCTET STRING,
qualifier DIDQualifier OPTIONAL
}
DIDAuthenticationData ::= OCTET STRING
DIDName ::= ~~CHOICE {~~Name
~~plainName [0] Name,~~
~~qualifiedName [1] VisibleString (FROM~~
~~("a".."z"|"A".."Z"|"0".."9"|"./-_~%#"))~~
~~}~~
DIDReference ::= SEQUENCE {
scope DIDScope,
dIDName DIDName
}
DIDNameList ::= SET OF DIDName
~~MarkerList ::= SEQUENCE OF Marker~~

TargetName ::= CHOICE {
datasetName ~~[0]~~ DataSetName,
didName ~~[1]~~ DIDName,
cardApplicationName ~~[2] ~~CardApplicationName
}
TargetType ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "DATA_SET"~~|~~
~~"~~--  "DIFFERENTIAL-IDENTITY"~~|~~
--  "CARD-APPLICATION"
~~)~~})
URL ::= CHOICE {
printable PrintableString,
ia5 IA5String
}
-- Secure Transport Syntax, an adaptation of PKCS#7 envelopedData content
type - RFC 2315
ContentInfo ::= SEQUENCE {
contentType ContentType,
content EnvelopedData
}

ContentType ::= OBJECT IDENTIFIER
EnvelopeDataVersion ::= INTEGER (0)
CertificateSerialNumber ::= INTEGER
KeyEncryptionAlgorithmIdentifier ::= AlgorithmI~~dentifier~~DParameters
ContentEncryptionAlgorithmIdentifier ::= AlgorithmI~~dentifier~~DParameters
~~AlgorithmIdentifier ::= SEQUENCE {~~
~~algorithm OBJECT IDENTIFIER,~~
~~parameters ANY DEFINED BY algorithm OPTIONAL~~
~~}~~

EnvelopedData ::= SEQUENCE {
version EnvelopeDataVersion,
recipientInfos RecipientInfos,
encryptedContentInfo EncryptedContentInfo
}
RecipientInfos ::= SET OF RecipientInfo
RecipientInfo ::= SEQUENCE {
version EnvelopeDataVersion,
issuerAndSerialNumber IssuerAndSerialNumber,
keyEncryptionAlgorithm KeyEncryptionAlgorithmIdentifier,
encryptedKey EncryptedKey
}
IssuerAndSerialNumber ::= SEQUENCE {
issuer Name,
serialNumber CertificateSerialNumber
}
EncryptedKey ::= OCTET STRING
EncryptedContentInfo ::= SEQUENCE {
contentType ContentType,
contentEncryptionAlgorithm ContentEncryptionAlgorithmIdentifier,
encryptedContent ~~[0] IMPLICIT ~~EncryptedContent OPTIONAL
}
EncryptedContent ::= OCTET STRING
-- C.1.4 The Card-Application Services
CardApplicationServiceName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "Connection"~~|~~
~~"~~--  "CardApplication"~~|~~
~~"~~--  "NamedData"~~|~~
~~"~~--  "Cryptographic"~~|~~
~~"~~--  "DifferentialIdentity"~~|~~
--  "Authorization"
```

```
~~)~~})
CardApplicationServiceNameList ::= SET OF CardApplicationServiceName
ActionName ::= CHOICE {
apiAccessEntryPoint ~~[0] ~~APIAccessEntryPointName,
connectionServiceAction ~~[1] ~~ConnectionServiceActionName,
cardApplicationServiceAction ~~[2] ~~CardApplicationServiceActionName,
namedDataServiceAction ~~[3] ~~NamedDataServiceActionName,
cryptographicServiceAction ~~[4] ~~CryptographicServiceActionName,
differentialIdentityServiceAction ~~[5] ~~DifferentialIdentityServiceActionName,
authorizationServiceAction ~~[6] ~~AuthorizationServiceActionName
}
-- C.1.5 The Actions/Entry Points
APIAccessEntryPointName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
"--  "Initialize"~~ | ~~
"--  "Terminate"~~ | ~~
--  "CardApplicationPath"
~~)~~})
ConnectionServiceActionName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
"--  "CardApplicationConnect"~~ | ~~
"--  "CardApplicationDisconnect"~~ | ~~
"--  "CardApplicationStartSession"~~ | ~~
--  "CardApplicationEndSession"
~~)~~})
CardApplicationServiceActionName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
"--  "CardApplicationList"~~ | ~~
"--  "CardApplicationCreate"~~ | ~~
"--  "CardApplicationDelete"~~ | ~~
"--  "CardApplicationServiceList"~~ | ~~
"--  "CardApplicationServiceCreate"~~ | ~~
"--  "CardApplicationServiceLoad"~~ | ~~
"--  "CardApplicationServiceDelete"~~ | ~~
"--  "CardApplicationServiceDescribe"~~ | ~~
--  "ExecuteAction"
~~)~~})
NamedDataServiceActionName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
"--  "DataSetList"~~ | ~~
"--  "DataSetCreate"~~ | ~~
"--  "DataSetSelect"~~ | ~~
"--  "DataSetDelete"~~ | ~~
"--  "DSICreate"~~ | ~~
"--  "DSIDelete"~~ | ~~
"--  "DSIWrite"~~ | ~~
"--  "DSIRead"
~~)~~})
CryptographicServiceActionName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
"--  "Encipher"~~ | ~~
"--  "Decipher"~~ | ~~
"--  "GetRandom"~~ | ~~
"--  "Sign"~~ | ~~
"--  "VerifySignature"~~ | ~~
"--  "VerifyCertificate"
~~)~~})
DifferentialIdentityServiceActionName ::=
                      VisibleString (~~FROM (~~CONSTRAINED BY {
"--  "DIDList"~~ | ~~
"--  "DIDCreate"~~ | ~~
"--  "DIDGet"~~ | ~~
```

```
~~"~~--   "DIDUpdate"~~|~~
~~"~~--   "DIDDelete"~~|~~
~~"~~--   "DIDAuthenticate"
~~)~~})
AuthorizationServiceActionName ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "ACLList"~~|~~
~~"~~--   "ACLModify"
~~)~~})
-- C.1.6 The Return Codes
InitializeReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"
~~)~~})
TerminateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_WARNING_CONNECTION_DISCONNECTED"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationPathReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_TOO_MANY_RESULTS"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationConnectReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_EXCLUSIVE_NOT_AVAILABLE"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationDisconnectReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY
{
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_WARNING_SESSION_ENDED"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationStartSessionReturnCode ::=
                           VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_NEXT_REQUEST"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~|~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~|~~
~~"~~--   "API_DID_ALREADY_AUTHENTICATED"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
```

```
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationEndSessionReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY
{
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NO_ACTIVE_SESSION"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationListReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationCreateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAME_EXISTS"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationDeleteReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_WARNING_CONNECTION_DISCONNECTED"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationServiceListReturnCode ::=
                         VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationServiceCreateReturnCode ::=
                          VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAME_EXISTS"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
```

```
CardApplicationServiceLoadReturnCode ::=
                              VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--  "API_NOT_INITIALIZED"~~|~~
~~"~~--  "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--  "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--  "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationServiceDeleteReturnCode ::=
                           VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--  "API_NOT_INITIALIZED"~~|~~
~~"~~--  "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--  "API_COMMUNICATION_FAILURE"
~~)~~})
CardApplicationServiceDescribeReturnCode ::=
                           VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--  "API_NOT_INITIALIZED"~~|~~
~~"~~--  "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--  "API_COMMUNICATION_FAILURE"
~~)~~})
ExecuteActionReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--  "API_NOT_INITIALIZED"~~|~~
~~"~~--  "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--  "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--  "API_COMMUNICATION_FAILURE"
~~)~~})
DataSetListReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NOT_INITIALIZED"~~|~~
~~"~~--  "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--  "API_COMMUNICATION_FAILURE"
~~)~~})
DataSetCreateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NAME_EXISTS"~~|~~
~~"~~--  "API_NOT_INITIALIZED"~~|~~
~~"~~--  "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--  "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--  "API_COMMUNICATION_FAILURE"
~~)~~})
DataSetSelectReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--  "API_OK"~~|~~
~~"~~--  "API_INCORRECT_PARAMETER"~~|~~
~~"~~--  "API_NAMED_ENTITY_NOT_FOUND"~~|~~
```

```
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DataSetDeleteReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DSIListReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DSICreateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAME_EXISTS"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DSIDeleteReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DSIWriteReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DSIReadReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_PREREQUISITE_NOT_SATISFIED"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
```

```
~~)~~})
EncipherReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~|~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DecipherReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~|~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
GetRandomReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~|~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
HashReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~|~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
SignReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~|~~
~~"~~--   "API_INCORRECT_PARAMETER"~~|~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~|~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~|~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~|~~
~~"~~--   "API_NOT_INITIALIZED"~~|~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~|~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~|~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
VerifySignatureReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
```

```
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~,~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
VerifyCertificateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_INVALID_KEY"~~,~~
~~"~~--   "API_INVALID_SIGNATURE"~~,~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~,~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DIDListReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DIDCreateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAME_EXISTS"~~,~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DIDGetReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DIDUpdateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~,~~
```

```
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DIDDeleteReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
DIDAuthenticateReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_NEXT_REQUEST"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_PROTOCOL_NOT_RECOGNIZED"~~,~~
~~"~~--   "API_INAPPROPRIATE_PROTOCOL_FOR_ACTION"~~,~~
~~"~~--   "API_DID_ALREADY_AUTHENTICATED"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
ACLListReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
ACLModifyReturnCode ::= VisibleString (~~FROM (~~CONSTRAINED BY {
~~"~~--   "API_OK"~~,~~
~~"~~--   "API_INCORRECT_PARAMETER"~~,~~
~~"~~--   "API_NAMED_ENTITY_NOT_FOUND"~~,~~
~~"~~--   "API_NOT_INITIALIZED"~~,~~
~~"~~--   "API_SECURITY_CONDITION_NOT_SATISFIED"~~,~~
~~"~~--   "API_INSUFFICIENT_RESOURCES"~~,~~
~~"~~--   "API_COMMUNICATION_FAILURE"
~~)~~})
```

C.2 The Application Programming Interface

```
-- C.2.1 Initialize
Initialize ::= SEQUENCE {
argument NULL OPTIONAL,
result NULL OPTIONAL,
return InitializeReturnCode
}
-- C.2.2 Terminate
Terminate ::= SEQUENCE {
argument NULL OPTIONAL,
result NULL OPTIONAL,
return TerminateReturnCode
}
```

```
-- C.2.3 CardApplicationPath
CardApplicationPathArgument ::= ~~[APPLICATION 1024]~~ SEQUENCE {
cardAppPathRequest CardApplicationPathInfo
}
CardApplicationPathResult ::= ~~[APPLICATION 1025]~~ SEQUENCE {
cardAppPathResultSet CardApplicationPathSet
}
CardApplicationPath ::= SEQUENCE {
argument CardApplicationPathArgument,
result CardApplicationPathResult OPTIONAL,
return CardApplicationPathReturnCode
}
-- C.2.4 CardApplicationConnect
CardApplicationConnectArgument ::= ~~[APPLICATION 1026]~~ SEQUENCE {
cardApplicationPath CardApplicationPathInfo,
exclusiveUse BOOLEAN
}
CardApplicationConnectResult ::= ~~[APPLICATION 1027]~~ SEQUENCE {
connectionHandle ConnectionHandle
}
CardApplicationConnect ::= SEQUENCE {
argument CardApplicationConnectArgument,
result CardApplicationConnectResult OPTIONAL,
return CardApplicationConnectReturnCode
}
-- C.2.5 CardApplicationDisconnect
CardApplicationDisconnectArgument ::= ~~[APPLICATION 1028]~~ SEQUENCE {
connectionHandle ConnectionHandle,
action ~~ReaderAction~~IFDAction OPTIONAL
}
CardApplicationDisconnect ::= SEQUENCE {
argument CardApplicationDisconnectArgument,
result NULL OPTIONAL,
return CardApplicationDisconnectReturnCode
}
-- C.2.6 CardApplicationStartSession
CardApplicationStartSessionArgument ::= ~~[APPLICATION 1030]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
authenticationProtocolData DIDAuthenticationData,
samConnectionHandle ConnectionHandle OPTIONAL
}
CardApplicationStartSessionResult ::= SEQUENCE {
authenticationProtocolData DIDAuthenticationData
}
CardApplicationStartSession ::= [APPLICATION 1031] SEQUENCE {
argument CardApplicationStartSessionArgument,
result CardApplicationStartSessionResult OPTIONAL,
return CardApplicationStartSessionReturnCode
}
-- C.2.7 CardApplicationEndSession
CardApplicationEndSessionArgument ::= ~~[APPLICATION 1032]~~ SEQUENCE {
connectionHandle ConnectionHandle
}
CardApplicationEndSession ::= [APPLICATION 1033] SEQUENCE {
argument CardApplicationEndSessionArgument,
result NULL OPTIONAL,
return CardApplicationEndSessionReturnCode
}
```

```
-- C.2.8 CardApplicationList
CardApplicationListArgument ::= ~~[APPLICATION 1034]~~ SEQUENCE {
connectionHandle ConnectionHandle
}
CardApplicationListResult ::= ~~[APPLICATION 1035]~~ SEQUENCE {
cardApplicationNameList CardApplicationNameList
}
CardApplicationList ::= SEQUENCE {
argument CardApplicationListArgument,
result CardApplicationListResult OPTIONAL,
return CardApplicationListReturnCode
}
-- C.2.9 CardApplicationCreate
CardApplicationCreateArgument ::= ~~[APPLICATION 1036]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationName CardApplicationName,
cardApplicationACL AccessControlList
}

CardApplicationCreate ::= SEQUENCE {
argument CardApplicationCreateArgument,
result NULL OPTIONAL,
return CardApplicationCreateReturnCode
}
-- C.2.10 CardApplicationDelete
CardApplicationDeleteArgument ::= ~~[APPLICATION 1038]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationName CardApplicationName
}
CardApplicationDelete ::= SEQUENCE {
argument CardApplicationDeleteArgument,
result NULL OPTIONAL,
return CardApplicationDeleteReturnCode
}
-- C.2.11 CardApplicationServiceList
CardApplicationServiceListArgument ::= ~~[APPLICATION 1040]~~ SEQUENCE {
connectionHandle ConnectionHandle
}
CardApplicationServiceListResult ::= ~~[APPLICATION 1041]~~ SEQUENCE {
cardApplicationServiceNameList CardApplicationServiceNameList
}
CardApplicationServiceList ::= SEQUENCE {
argument CardApplicationServiceListArgument,
result CardApplicationServiceListResult OPTIONAL,
return CardApplicationServiceListReturnCode
}
-- C.2.12 CardApplicationServiceCreate
CardApplicationServiceCreateArgument ::= ~~[APPLICATION 1042]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationServiceName CardApplicationServiceName
}
CardApplicationServiceCreate ::= SEQUENCE {
argument CardApplicationServiceCreateArgument,
result NULL OPTIONAL,
return CardApplicationServiceCreateReturnCode
}
-- C.2.13 CardApplicationSeviceLoad
CardApplicationServiceLoadArgument ::= ~~[APPLICATION 1044]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationServiceName CardApplicationServiceName,
code CardApplicationServiceLoadPackage
}
CardApplicationSeviceLoad ::= SEQUENCE {
argument CardApplicationServiceLoadArgument,
result NULL OPTIONAL,
return CardApplicationServiceLoadReturnCode
}
-- C.2.14 CardApplicationServiceDelete
CardApplicationServiceDeleteArgument ::= ~~[APPLICATION 1046]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationServiceName CardApplicationServiceName
}
CardApplicationServiceDelete ::= SEQUENCE {
argument CardApplicationServiceDeleteArgument,
result NULL OPTIONAL,
return CardApplicationServiceDeleteReturnCode
}
-- C.2.15 CardApplicationServiceDescribe
CardApplicationServiceDescribeArgument ::= ~~[APPLICATION 1048]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationServiceName CardApplicationServiceName
}
CardApplicationServiceDescribeResult ::= ~~[APPLICATION 1049]~~ SEQUENCE {
serviceDescription CardApplicationServiceDescription
}
CardApplicationServiceDescribe ::= SEQUENCE {
argument CardApplicationServiceDescribeArgument,
result CardApplicationServiceDescribeResult OPTIONAL,
return CardApplicationServiceDescribeReturnCode
}
-- C.2.16 ExecuteAction
ExecuteActionArgument ::= ~~[APPLICATION 1050]~~ SEQUENCE {
connectionHandle ConnectionHandle,
cardApplicationServiceName CardApplicationServiceName,
actionName ActionName,
request ExecuteActionRequest
}
ExecuteActionResult ::= ~~[APPLICATION 1051]~~ SEQUENCE {
confirmation ExecuteActionConfirmation
}
ExecuteAction ::= SEQUENCE {
argument ExecuteActionArgument,
result ExecuteActionResult OPTIONAL,
return ExecuteActionReturnCode
}
-- C.2.17 DataSetList
DataSetListArgument ::= ~~[APPLICATION 1052]~~ SEQUENCE {
connectionHandle ConnectionHandle
}

DataSetListResult ::= ~~[APPLICATION 1053]~~ SEQUENCE {
dataSetNameList DataSetNameList
}
DataSetList ::= SEQUENCE {
argument DataSetListArgument,
result DataSetListResult OPTIONAL,
return DataSetListReturnCode
}
```

```
-- C.2.18 DataSetCreate
DataSetCreateArgument ::= ~~[APPLICATION 1054] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dataSetName DataSetName,
dataSetACL AccessControlList
}
DataSetCreate ::= SEQUENCE {
argument DataSetCreateArgument,
result NULL OPTIONAL,
return DataSetCreateReturnCode
}
-- C.2.19 DataSetSelect
DataSetSelectArgument ::= ~~[APPLICATION 1056] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dataSetName DataSetName
}
DataSetSelect ::= SEQUENCE {
argument DataSetSelectArgument,
result NULL OPTIONAL,
return DataSetSelectReturnCode
}
-- C.2.20 DataSetDelete
DataSetDeleteArgument ::= ~~[APPLICATION 1058] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dataSetName DataSetName
}
DataSetDelete ::= SEQUENCE {
argument DataSetDeleteArgument,
result NULL OPTIONAL,
return DataSetDeleteReturnCode
}
-- C.2.21 DSIList
DSIListArgument ::= ~~[APPLICATION 1060] ~~SEQUENCE {
connectionHandle ConnectionHandle
}
DSIListResult ::= ~~[APPLICATION 1061] ~~SEQUENCE {
dsiNameList DSINameList
}

DSIList ::= SEQUENCE {
argument DSIListArgument,
result DSIListResult OPTIONAL,
return DSIListReturnCode
}
-- C.2.22 DSICreate
DSICreateArgument ::= ~~[APPLICATION 1062] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dsiName DSIName,
dsiContent DSIContent
}
DSICreate ::= SEQUENCE {
argument DSICreateArgument,
result NULL OPTIONAL,
return DSICreateReturnCode
}
-- C.2.23 DSIDelete
DSIDeleteArgument ::= ~~[APPLICATION 1064] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dsiName DSIName
}
```

```
DSIDelete ::= SEQUENCE {
argument DSIDeleteArgument,
result NULL OPTIONAL,
return DSIDeleteReturnCode
}
-- C.2.24 DSIWrite
DSIWriteArgument ::= ~~[APPLICATION 1066] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dsiName DSIName,
dsiContent DSIContent
}
DSIWrite ::= SEQUENCE {
argument DSIWriteArgument,
result NULL OPTIONAL,
return DSIWriteReturnCode
}
-- C.2.25 DSIRead
DSIReadArgument ::= ~~[APPLICATION 1068] ~~SEQUENCE {
connectionHandle ConnectionHandle,
dsiName DSIName
}
DSIReadResult ::= ~~[APPLICATION 1069] ~~SEQUENCE {
dsiContent DSIContent
}
DSIRead ::= SEQUENCE {
argument DSIReadArgument,
result DSIReadResult OPTIONAL,
return DSIReadReturnCode
}
-- C.2.26 Encipher
EncipherArgument ::= ~~[APPLICATION 1070] ~~SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
plainText CipherBuffer
}
EncipherResult ::= ~~[APPLICATION 1071] ~~SEQUENCE {
cipherText CipherBuffer
}
Encipher ::= SEQUENCE {
argument EncipherArgument,
result EncipherResult OPTIONAL,
return EncipherReturnCode
}
-- C.2.27 Decipher
DecipherArgument ::= ~~[APPLICATION 1072] ~~SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
cipherText CipherBuffer
}
DecipherResult ::= ~~[APPLICATION 1073] ~~SEQUENCE {
plainText CipherBuffer
}
Decipher ::= SEQUENCE {
argument DecipherArgument,
result DecipherResult OPTIONAL,
return DecipherReturnCode
}
```

```
-- C.2.28 GetRandom
GetRandomArgument ::= ~~[APPLICATION 1074]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName
}
GetRandomResult ::= ~~[APPLICATION 1075]~~ SEQUENCE {
random RandomDataBuffer
}
GetRandom ::= SEQUENCE {
argument GetRandomArgument,
result GetRandomResult OPTIONAL,
return GetRandomReturnCode
}

-- C.2.29 Hash
HashArgument ::= ~~[APPLICATION 1076]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
message MessageBuffer
}
HashResult ::= ~~[APPLICATION 1077]~~ SEQUENCE {
hash HashBuffer
}
Hash ::= SEQUENCE {
argument HashArgument,
result HashResult OPTIONAL,
return HashReturnCode
}
-- C.2.30 Sign
SignArgument ::= ~~[APPLICATION 1078]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
message MessageBuffer
}
SignResult ::= ~~[APPLICATION 1079]~~ SEQUENCE {
signature SignatureBuffer
}
Sign ::= SEQUENCE {
argument SignArgument,
result SignResult OPTIONAL,
return SignReturnCode
}
-- C.2.31 VerifySignature
VerifySignatureArgument ::= ~~[APPLICATION 1080]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
signature SignatureBuffer,
message MessageBuffer
}
VerifySignature ::= SEQUENCE {
argument VerifySignatureArgument,
result NULL OPTIONAL,
return VerifySignatureReturnCode
}
```

```
-- C.2.32 VerifyCertificate
VerifyCertificateArgument ::= ~~[APPLICATION 1082]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
rootCert DIDName,
certificateType CertificateType,
certificate CertificateInfo
}
VerifyCertificate ::= SEQUENCE {
argument VerifyCertificateArgument,
result NULL OPTIONAL,
return VerifyCertificateReturnCode
}
-- C.2.33 DIDList
DIDListArgument ::= ~~[APPLICATION 1084]~~ SEQUENCE {
connectionHandle ConnectionHandle,
filter DIDQualifier OPTIONAL
}
DIDListResult ::= ~~[APPLICATION 1085]~~ SEQUENCE {
didNameList DIDNameList
}
DIDList ::= SEQUENCE {
argument DIDListArgument,
result DIDListResult OPTIONAL,
return DIDListReturnCode
}
-- C.2.34 DIDCreate
DIDCreateArgument ::= ~~[APPLICATION 1086]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didName DIDName,
authProtocolOID ObjectIdentifier,
didUpdateData DIDUpdateData,
didACL AccessControlList
}
DIDCreate ::= SEQUENCE {
argument DIDCreateArgument,
result NULL OPTIONAL,
return DIDCreateReturnCode
}
-- C.2.35 DIDGet
DIDGetArgument ::= ~~[APPLICATION 1088]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName
}
DIDGetResult ::= ~~[APPLICATION 1089]~~ SEQUENCE {
didStructure DIDStructure
}
DIDGet ::= SEQUENCE {
argument DIDGetArgument,
result DIDGetResult OPTIONAL,
return DIDGetReturnCode
}

-- C.2.36 DIDUpdate
DIDUpdateArgument ::= ~~[APPLICATION 1090]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didName DIDName,
didUpdateData DIDUpdateData
}
```

```
DIDUpdate ::= SEQUENCE {
argument DIDUpdateArgument,
result NULL OPTIONAL,
return DIDUpdateReturnCode
}
-- C.2.37 DIDDelete
DIDDeleteArgument ::= ~~[APPLICATION 1092]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didName DIDName
}
DIDDelete ::= SEQUENCE {
argument DIDDeleteArgument,
result NULL OPTIONAL,
return DIDDeleteReturnCode
}
-- C.2.38 DIDAuthenticate
DIDAuthenticateArgument ::= ~~[APPLICATION 1094]~~ SEQUENCE {
connectionHandle ConnectionHandle,
didScope DIDScope,
didName DIDName,
authenticationProtocolData DIDAuthenticationData,
samConnectionHandle ConnectionHandle OPTIONAL
}
DIDAuthenticateResult ::= ~~[APPLICATION 1095]~~ SEQUENCE {
authenticationProtocolData DIDAuthenticationData
}
DIDAuthenticate ::= SEQUENCE {
argument DIDAuthenticateArgument,
result DIDAuthenticateResult OPTIONAL,
return DIDAuthenticateReturnCode
}
-- C.2.39 ACLList
ACLListArgument ::= ~~[APPLICATION 1096]~~ SEQUENCE {
connectionHandle ConnectionHandle,
targetType TargetType,
targetName TargetName
}
ACLListResult ::= ~~[APPLICATION 1097]~~ SEQUENCE {
targetACL AccessControlList
}

ACLList ::= SEQUENCE {
argument ACLListArgument,
result ACLListResult OPTIONAL,
return ACLListReturnCode
}
-- C.2.40 ACLModify
ACLAccessRuleModifyArgument ::= ~~[APPLICATION 1098]~~ SEQUENCE {
connectionHandle ConnectionHandle,
targetType TargetType,
targetName TargetName,
cardApplicationServiceName CardApplicationServiceName,
actionName ActionName,
securityCondition SecurityCondition
}
ACLAccessRuleModify ::= SEQUENCE {
argument ACLAccessRuleModifyArgument,
result NULL OPTIONAL,
return ACLModifyReturnCode
}
```

Pages 177–178, C.3

Modify C.3 as follows:

C.3 The Remote Call (Marshalling/Unmarshalling) Interface

```
ServiceChoice ::= [APPLICATION 2101] CHOICE {
apiAccessChoice ~~[0]~~ APIAccessChoice,
connectionServiceChoice ~~[1]~~ ConnectionServiceChoice,
cardApplicationServiceChoice ~~[2]~~ CardApplicationServiceChoice,
namedDataServiceChoice ~~[3]~~ NamedDataServiceChoice,
cryptographicServiceChoice ~~[4]~~ CryptographicServiceChoice,
differentialIdentityServiceChoice ~~[5]~~ DifferentialIdentityServiceChoice,
authorizationServiceChoice ~~[6]~~ AuthorizationServiceChoice
}
APIAccessChoice ::= [APPLICATION 2102] CHOICE {
initialize ~~[0]~~ InitializeCall,
terminate ~~[1]~~ TerminateCall,
cardApplicationPath ~~[2]~~ CardApplicationPathCall
}
ConnectionServiceChoice ::= [APPLICATION 2103] CHOICE {
cardApplicationConnect ~~[0]~~ CardApplicationConnectCall,
cardApplicationDisconnect ~~[1]~~ CardApplicationDisconnectCall,
cardApplicationStartSession ~~[2]~~ CardApplicationStartSessionCall,
cardApplicationEndSession ~~[3]~~ CardApplicationEndSessionCall
}
CardApplicationServiceChoice ::= [APPLICATION 2104] CHOICE {
cardApplicationList ~~[0]~~ CardApplicationListCall,
cardApplicationCreate ~~[1]~~ CardApplicationCreateCall,
cardApplicationDelete ~~[2]~~ CardApplicationDeleteCall,
cardApplicationServiceList ~~[3]~~ CardApplicationServiceListCall,
cardApplicationServiceCreate ~~[4]~~ CardApplicationServiceCreateCall,
cardApplicationServiceLoad ~~[5]~~ CardApplicationServiceLoadCall,
cardApplicationServiceDelete ~~[6]~~ CardApplicationServiceDeleteCall,
cardApplicationServiceDescribe ~~[7]~~ CardApplicationServiceDescribeCall,
executeAction ~~[8]~~ ExecuteActionCall
}

NamedDataServiceChoice ::= [APPLICATION 2105] CHOICE {
dataSetList ~~[0]~~ DataSetListCall,
dataSetCreate ~~[1]~~ DataSetCreateCall,
dataSetSelect ~~[2]~~ DataSetSelectCall,
dataSetDelete ~~[3]~~ DataSetDeleteCall,
dSIList ~~[4]~~ DSIListCall,
dSICreate ~~[5]~~ DSICreateCall,
dSIDelete ~~[6]~~ DSIDeleteCall,
dSIWrite ~~[7]~~ DSIWriteCall,
dSIRead ~~[8]~~ DSIReadCall
}
CryptographicServiceChoice ::= [APPLICATION 2106] CHOICE {
encipher ~~[0]~~ EncipherCall,
decipher ~~[1]~~ DecipherCall,
getRandom ~~[2]~~ GetRandomCall,
hash ~~[3]~~ HashCall,
sign ~~[4]~~ SignCall,
verifySignature ~~[5]~~ VerifySignatureCall,
verifyCertificate ~~[6]~~ VerifyCertificateCall
}
DifferentialIdentityServiceChoice ::= [APPLICATION 2107] CHOICE {
didList ~~[0]~~ DIDListCall,
didCreate ~~[1]~~ DIDCreateCall,
didGet ~~[2]~~ DIDGetCall,
didUpdate ~~[3]~~ DIDUpdateCall,
didDelete ~~[4]~~ DIDDeleteCall,
didAuthenticate ~~[5]~~ DIDAuthenticateCall
}
AuthorizationServiceChoice ::= [APPLICATION 2108] CHOICE {
aclList ~~[0]~~ ACLListCall,
aclModify ~~[1]~~ ACLModifyCall
}
```

Pages 178–187, C.4 to C.9.2

Modify C.4 to C.9.2 as follows:

C.4 Connection Service

```
-- C.4.1 Initialize
InitializeCall ::= [APPLICATION 2001] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument NULL OPTIONAL
}
InitializeReturn ::= [APPLICATION 2002] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode InitializeReturnCode
}
-- C.4.2 Terminate
TerminateCall ::= [APPLICATION 2003] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument NULL OPTIONAL
}

TerminateReturn ::= [APPLICATION 2004] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode TerminateReturnCode
}
-- C.4.3 CardApplicationPath
CardApplicationPathCall ::= [APPLICATION 2005] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationPathArgument
}
CardApplicationPathReturn ::= [APPLICATION 2006] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result CardApplicationPathResult OPTIONAL,
returnCode CardApplicationPathReturnCode
}
-- C.4.4 CardApplicationConnect
CardApplicationConnectCall ::= [APPLICATION 2007] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationConnectArgument
}
```

```
CardApplicationConnectReturn ::= [APPLICATION 2008] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result CardApplicationConnectResult OPTIONAL,
returnCode CardApplicationConnectReturnCode
}
-- C.4.5 CardApplicationDisconnect
CardApplicationDisconnectCall ::= [APPLICATION 2009] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationDisconnectArgument
}
CardApplicationDisconnectReturn ::= [APPLICATION 2010] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationDisconnectReturnCode
}
-- C.4.6 CardApplicationStartSession
CardApplicationStartSessionCall ::= [APPLICATION 2011] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationStartSessionArgument
}
CardApplicationStartSessionReturn ::= [APPLICATION 2012] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result CardApplicationStartSessionResult OPTIONAL,
returnCode CardApplicationStartSessionReturnCode
}

-- C.4.7 CardApplicationEndSession
CardApplicationEndSessionCall ::= [APPLICATION 2013] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationEndSessionArgument
}
CardApplicationEndSessionReturn ::= [APPLICATION 2014] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationEndSessionReturnCode
}
```

C.5 Card Application Service

```
-- C.5.1 CardApplicationList
CardApplicationListCall ::= [APPLICATION 2015] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationListArgument
}
CardApplicationListReturn ::= [APPLICATION 2016] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result CardApplicationListResult OPTIONAL,
returnCode CardApplicationListReturnCode
}
-- C.5.2 CardApplicationCreate
CardApplicationCreateCall ::= [APPLICATION 2017] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationCreateArgument
}
CardApplicationCreateReturn ::= [APPLICATION 2018] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationCreateReturnCode
}
-- C.5.3 CardApplicationDelete
```

```
CardApplicationDeleteCall ::= [APPLICATION 2019] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationDeleteArgument
}
CardApplicationDeleteReturn ::= [APPLICATION 2020] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationDeleteReturnCode
}
-- C.5.4 CardApplicationServiceList
CardApplicationServiceListCall ::= [APPLICATION 2021] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationServiceListArgument
}

CardApplicationServiceListReturn ::= [APPLICATION 2022] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result CardApplicationServiceListResult OPTIONAL,
returnCode CardApplicationServiceListReturnCode
}
-- C.5.5 CardApplicationServiceCreate
CardApplicationServiceCreateCall ::= [APPLICATION 2023] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationServiceCreateArgument
}
CardApplicationServiceCreateReturn ::= [APPLICATION 2024] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationServiceCreateReturnCode
}
-- C.5.6 CardApplicationServiceLoad
CardApplicationServiceLoadCall ::= [APPLICATION 2025] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationServiceLoadArgument
}
CardApplicationServiceLoadReturn ::= [APPLICATION 2026] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationServiceLoadReturnCode
}
-- C.5.7 CardApplicationServiceDelete
CardApplicationServiceDeleteCall ::= [APPLICATION 2027] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument CardApplicationServiceDeleteArgument
}
CardApplicationServiceDeleteReturn ::= [APPLICATION 2028] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode CardApplicationServiceDeleteReturnCode
}
-- C.5.8 CardApplicationServiceDescribe
CardApplicationServiceDescribeCall ::= [APPLICATION 2029] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
~~argument CardApplicationServiceDeleteArgument~~
argument CardApplicationServiceDescribeArgument
}
CardApplicationServiceDescribeReturn ::= [APPLICATION 2030] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result CardApplicationServiceDescribeResult OPTIONAL,
returnCode CardApplicationServiceDescribeReturnCode
}

-- C.5.9 ExecuteAction
ExecuteActionCall ::= [APPLICATION 2031] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument ExecuteActionArgument
}
ExecuteActionReturn ::= [APPLICATION 2032] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result ExecuteActionResult OPTIONAL,
returnCode ExecuteActionReturnCode
}
```

C.6 Named Data Service

```
-- C.6.1 DataSetList
DataSetListCall ::= [APPLICATION 2033] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DataSetListArgument
}
DataSetListReturn ::= [APPLICATION 2034] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DataSetListResult OPTIONAL,
returnCode DataSetListReturnCode
}
-- C.6.2 DataSetCreate
DataSetCreateCall ::= [APPLICATION 2035] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DataSetCreateArgument
}
DataSetCreateReturn ::= [APPLICATION 2036] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DataSetCreateReturnCode
}
-- C.6.3 DataSetSelect
DataSetSelectCall ::= [APPLICATION 2037] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DataSetSelectArgument
}
DataSetSelectReturn ::= [APPLICATION 2038] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DataSetSelectReturnCode
}
-- C.6.4 DataSetDelete
DataSetDeleteCall ::= [APPLICATION 2039] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DataSetDeleteArgument
}

DataSetDeleteReturn ::= [APPLICATION 2040] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DataSetDeleteReturnCode
}
-- C.6.5 DSIList
DSIListCall ::= [APPLICATION 2041] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DSIListArgument
}
DSIListReturn ::= [APPLICATION 2042] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DSIListResult OPTIONAL,
returnCode DSIListReturnCode
}
-- C.6.6 DSICreate
DSICreateCall ::= [APPLICATION 2043] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DSICreateArgument
}
DSICreateReturn ::= [APPLICATION 2044] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DSICreateReturnCode
}
-- C.6.7 DSIDelete
DSIDeleteCall ::= [APPLICATION 2045] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DSIDeleteArgument
}
DSIDeleteReturn ::= [APPLICATION 2046] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DSIDeleteReturnCode
}
-- C.6.8 DSI Write
DSIWriteCall ::= [APPLICATION 2047] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DSIWriteArgument
}
DSIWriteReturn ::= [APPLICATION 2048] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DSIWriteReturnCode
}

-- C.6.9 DSIRead
DSIReadCall ::= [APPLICATION 2049] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DSIReadArgument
}
DSIReadReturn ::= [APPLICATION 2050] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DSIReadResult OPTIONAL,
returnCode DSIReadReturnCode
}
```

C.7 Cryptographic Service

```
-- C.7.1 Encipher
EncipherCall ::= [APPLICATION 2051] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument EncipherArgument
}
EncipherReturn ::= [APPLICATION 2052] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result EncipherResult OPTIONAL,
returnCode EncipherReturnCode
}
```

```
-- C.7.2 Decipher
DecipherCall ::= [APPLICATION 2053] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DecipherArgument
}
DecipherReturn ::= [APPLICATION 2054] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DecipherResult OPTIONAL,
returnCode DecipherReturnCode
}
-- C.7.3 Get Random
GetRandomCall ::= [APPLICATION 2055] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument GetRandomArgument
}
GetRandomReturn ::= [APPLICATION 2056] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result GetRandomResult OPTIONAL,
returnCode GetRandomReturnCode
}
-- C.7.4 Hash
HashCall ::= [APPLICATION 2057] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument HashArgument
}

HashReturn ::= [APPLICATION 2058] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result HashResult OPTIONAL,
returnCode HashReturnCode
}
-- C.7.5 Sign
SignCall ::= [APPLICATION 2059] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument SignArgument
}
SignReturn ::= [APPLICATION 2060] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result SignResult OPTIONAL,
returnCode SignReturnCode
}
-- C.7.6 Verify Signature
VerifySignatureCall ::= [APPLICATION 2061] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument VerifySignatureArgument
}
VerifySignatureReturn ::= [APPLICATION 2062] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode VerifySignatureReturnCode
}
-- C.7.7 Verify Certificate
VerifyCertificateCall ::= [APPLICATION 2063] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument VerifyCertificateArgument
}
VerifyCertificateReturn ::= [APPLICATION 2064] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode VerifyCertificateReturnCode
}
```

C.8 Differential-Identity Service

```
-- C.8.1 DIDList
DIDListCall ::= [APPLICATION 2065] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DIDListArgument
}
DIDListReturn ::= [APPLICATION 2066] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DIDListResult OPTIONAL,
returnCode DIDListReturnCode
}

-- C.8.2 DIDCreate
DIDCreateCall ::= [APPLICATION 2067] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DIDCreateArgument
}
DIDCreateReturn ::= [APPLICATION 2068] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DIDCreateReturnCode
}
-- C.8.3 DIDGet
DIDGetCall ::= [APPLICATION 2069] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DIDGetArgument
}
DIDGetReturn ::= [APPLICATION 2070] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DIDGetResult OPTIONAL,
returnCode DIDGetReturnCode
}
-- C.8.4 DIDUpdate
DIDUpdateCall ::= [APPLICATION 2071] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DIDUpdateArgument
}
DIDUpdateReturn ::= [APPLICATION 2072] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DIDUpdateReturnCode
}
-- C.8.5 DIDDelete
DIDDeleteCall ::= [APPLICATION 2073] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DIDDeleteArgument
}
DIDDeleteReturn ::= [APPLICATION 2074] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode DIDDeleteReturnCode
}
-- C.8.6 DIDAuthenticate
DIDAuthenticateCall ::= [APPLICATION 2075] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument DIDAuthenticateArgument
}
```

```
DIDAuthenticateReturn ::= [APPLICATION 2076] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result DIDAuthenticateResult OPTIONALreturnCode DIDAuthenticateReturnCode
}
```

C.9 Authorization Service

```
-- C.9.1 ACLList
ACLListCall ::= [APPLICATION 2077] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument ACLListArgument
}
ACLList~~ArgumentReturn~~Return ::= [APPLICATION 2078] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result ACLListResult OPTIONAL,
returnCode ACLListReturnCode
}
-- C.9.2 ACLModify
ACLModifyCall ::= [APPLICATION 2079] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
argument ACLAccessRuleModifyArgument
}
ACLModifyReturn ::= [APPLICATION 2080] SEQUENCE {
transactionIdentifier TransactionIdentifier OPTIONAL,
result NULL OPTIONAL,
returnCode ACLModifyReturnCode
}
```

Pages 187–192, C.10 to C.10.24

Modify C.10 to C.10.24 as follows:

C.10 Authentication Protocol Structures

```
-- Note, sections C.10.1 thru C.10.3 are intentionally skipped so that
-- the reader may easily reference between A.n and C.10.n.
-- C.10.4 Asymmetric Internal Authenticate
MarkerAP004 ::= SEQUENCE {
signatureAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keySize INTEGER,
keyPair CHOICE {
keyPairInline SEQUENCE {
publicKeyMaterial OCTET STRING,
privateKey OCTET STRING
},
generateFlag NULL
},
nonceSize INTEGER
}

-- C.10.5 Asymmetric External Authenticate
MarkerAP005 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keySize INTEGER,
publicKeyMaterial OCTET STRING,
nonceSize INTEGER
}
-- C.10.6 Symmetric Internal Authenticate
MarkerAP006 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keySize INTEGER,
secretKey OCTET STRING,
nonceSize INTEGER
}
-- C.10.7 Symmetric External Authenticate
MarkerAP007 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keySize INTEGER,
secretKey OCTET STRING,
nonceSize INTEGER
}
-- C.10.8 Compare
MarkerAP008 ::= SEQUENCE {
minDataLength INTEGER,
maxDataLength INTEGER,
paddingCharacter OCTET STRING,
markerTemplate OCTET STRING
}
-- C.10.9 PIN Compare
MarkerAP009 ::= SEQUENCE {
minDataLength INTEGER,
maxDataLength INTEGER,
storedLength INTEGER,
paddingCharacter OCTET STRING,
markerTemplate OCTET STRING,
maxAttempts INTEGER,
attemptsCounter INTEGER,
pinRef OCTET STRING,
pinValue VisibleString
}
-- C.10.10 Biometric Compare
MarkerAP010 ::= SEQUENCE {
bit OCTET STRING,
markerTemplate OCTET STRING
}

-- C.10.11 Mutual Authentication with Key Establishment
MarkerAP011 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
macAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
~~derivationAlgorithmK_enc OBJECT IDENTIFIER,~~
~~derivationAlgorithmK_mac OBJECT IDENTIFIER,~~
~~derivationAlgorithmK_IFD OBJECT IDENTIFIER,~~
~~derivationAlgorithmSessionKeysAndCounters OBJECT IDENTIFIER~~
derivationAlgorithmK-enc AlgorithmIDParameters,
derivationAlgorithmK-mac AlgorithmIDParameters,
derivationAlgorithmK-IFD AlgorithmIDParameters,
derivationAlgorithmSessionKeysAndCounters AlgorithmIDParameters
}
-- C.10.12 Client-Application Mutual Authentication with Key Establishment
MarkerAP012 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
macAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
encryptionAlgorithmForSessionKey ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
macAlgorithmForSessionKey ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
derivationAlgorithmSessionKeysAndCounter ~~OBJECT
IDENTIFIER~~AlgorithmIDParameters
}
-- C.10.13 Client-Application Asymmetric External Authenticate
MarkerAP013 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keySize INTEGER,
publicKeyMaterial OCTET STRING,
nonceSize INTEGER
}
-- C.10.14 Modular Extended Access Control Protocol (M-EAC)
MarkerAP014 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
encryptionAlgorithmForSessionKey ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
macAlgorithmForSessionKey ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
~~k_enc~~k-enc OCTET STRING,
~~k_mac~~k-mac OCTET STRING,
~~derivationAlgorithmK_enc OBJECT IDENTIFIER,~~
~~derivationAlgorithmK_mac OBJECT IDENTIFIER,~~
derivationAlgorithmK-enc AlgorithmIDParameters,
derivationAlgorithmK-mac AlgorithmIDParameters,
keySize INTEGER,
keyPair CHOICE {
keyPairInline SEQUENCE {
publicKeyMaterial OCTET STRING,
privateKey OCTET STRING
},
generateFlag NULL
},
nonceSize INTEGER
}
-- C.10.15 Key Transport with mutual authentication based on RSA
MarkerAP015 ::= SEQUENCE {
encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keySize INTEGER,
keyPair CHOICE {
keyPairInline SEQUENCE {
publicKeyMaterial OCTET STRING,
privateKey OCTET STRING
},
generateFlag NULL
},
nonceSize INTEGER
}
-- C.10.16 Age Attainment
MarkerAP016 ::= SEQUENCE {
dateOfBirthReference DSIReference,
attainedDate INTEGER
}
-- C.10.17 Asymmetric Session Key Establishment
MarkerAP017 ::= SEQUENCE {
~~encryptionAlgorithm OBJECT IDENTIFIER,~~
~~keySize INTEGER,~~
```

```
~~keyTransportProtectionType INTEGER,~~
~~privateTransKeyReference DIDReference,~~
~~CHOICE {~~
    encryptionAlgorithm AlgorithmIDParameters,
    keySize INTEGER,
    keyTransportProtectionType INTEGER,
    privateTransKeyReference DIDReference,
    keyPair CHOICE {
        byValue SEQUENCE {
            privateKey OCTET STRING,
            publicKeyMaterial OCTET STRING
        },
        byReference SEQUENCE {
            privateKeyReference DIDReference,
            publicKeyReference DIDReference
        },
        generateFlag ~~BOOLEAN~~NULL
    },
    sessionMACAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
    sessionENCAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters
}
-- C.10.18 Secure PIN Compare
MarkerAP018 ::= SEQUENCE {
    minDataLength INTEGER,
    maxDataLength INTEGER,
    paddingCharacter OCTET STRING,
    markerTemplate OCTET STRING,
    maxAttempts INTEGER,
    attemptCounter INTEGER,
    encryptionAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
    keySize INTEGER,
    keyTransportProtectionType INTEGER,
    nonceSize INTEGER,
~~nonce INTEGER,~~
~~CHOICE {~~
~~SEQUENCE {~~
~~privateKey OCTET STRING,~~
~~publicKeyMaterial OCTET STRING~~
~~},~~
~~SEQUENCE {~~
~~privateKeyReference DIDReference,~~
~~publicKeyMaterialReference DIDReference~~
~~},~~
~~generateFlag NULL~~
~~}~~
    nonce OCTET STRING,
    keyPair CHOICE {
        byValue SEQUENCE {
            privateKey OCTET STRING,
            publicKeyMaterial OCTET STRING
        },
        byReference SEQUENCE {
            privateKeyReference DIDReference,
            publicKeyMaterialReference DIDReference
        },
        generateFlag NULL
    }
}
```

```
-- C.10.19 EC Key Agreement with Card-Application Authentication
MarkerAP019 ::= SEQUENCE {
domainParameters ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keyEstablishmentAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
kDFHashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionMacAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionEncAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
nonceSize INTEGER,
keyPair CHOICE {
keyPairInline SEQUENCE {
iccPublicKey OCTET STRING,
iccPrivateKey OCTET STRING
},
genKeyPa~~ri~~irFlag NULL
},
iccIdentifier OCTET STRING,
iccCert OCTET STRING
}
-- C.10.20 EC Key Agreement with Mutual Authentication
MarkerAP020 ::= SEQUENCE {
domainParameters ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keyEstablishmentAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
kDFHashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionMacAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionEncAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
authAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
nonceSize INTEGER,
keyPair CHOICE {
keyPairInline SEQUENCE {
iccPublicKey OCTET STRING,
iccPrivateKey OCTET STRING
},
genKeyPa~~ri~~irFlag NULL
},
iccIdentifier OCTET STRING,
iccCert OCTET STRING,
rootIdentifier OCTET STRING,
rootPublicKey OCTET STRING,
iccCertKnown BOOLEAN,
verifyClientCert BOOLEAN,
enforcePrivacy BOOLEAN
}
-- C.10.21 Simple EC-DH Key Agreement
MarkerAP021 ::= SEQUENCE {
domainParameters ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keyEstablishmentAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
kDFHashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionMacAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionEncAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters
}

-- C.10.22 GP Symmetric Authentication (Explicit Mode)
MarkerAP022 ::= SEQUENCE {
sdPublicKey OCTET STRING,
sdPrivateKey OCTET STRING,
sdCertificate OCTET STRING,
exPublicKey OCTET STRING, -- PK.TP EX.AUT as defined in GP 2.2 (F.1.2.1
Overview)
ocePublicKey OCTET STRING, -- PK.OCE.AUT as defined in GP 2.2 (F.1.2.1
Overview)
kaExPublicKey OCTET STRING, -- PK.KA EX.AUT as defined in GP 2.2 (F.1.2.1)
kaInCertificate OCTET STRING, -- CERT.KA IN.AUT as defined in GP 2.2
(F.1.2.1 Overview)
sessionMacAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionEncAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keyEncryptionAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
hashAlgorithm ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
nonceSize INTEGER
}
-- C.10.23 GP Symmetric Authentication (Explicit Mode)
MarkerAP023 ::= SEQUENCE {
sdStaticEncKey OCTET STRING,
sdStaticMacKey OCTET STRING,
sdStaticDekKey OCTET STRING,
sdSequenceCounter OCTET STRING,
sessionMacAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionEncAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keyDerivationAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
nonceSize INTEGER
}
-- C.10.24 GP Symmetric Authentication (Implicit Mode)
MarkerAP024 ::= SEQUENCE {
sdStaticBaseKey OCTET STRING,
sdSequenceCounter OCTET STRING,
sessionMacAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
sessionEncAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters,
keyDerivationAlgo ~~OBJECT IDENTIFIER~~AlgorithmIDParameters
}

END
```

Page 192

Insert the following new annex after Annex C:

Annex D
(normative)

ISO24727-COMMON module

```
ISO24727-COMMON { iso(1) standard(0) iso24727(24727) }
-- Version 1.5, 03-Mar-2010
--
-- IF-PROFILE value '01'
--
-- *According to ISO/IEC 24727-2, the optional IF-PROFILE field in the CCD is
-- used to indicate that a card provides an implementation of ISO/IEC 24727-3.

-- © ISO/IEC 2008-2010
-- All rights reserved. Unless otherwise specified, no part of this publication
-- may be reproduced or utilized in any form or by any means, electronic or
-- mechanical, including photocopying and microfilm, without permission in
-- writing from either ISO at the address below or ISO's member body in the
-- country of the requester.
--
--    ISO copyright office
--      Case postale 56 • CH-1211 Geneva 20
--      Tel. + 41 22 749 01 11
--      Fax + 41 22 749 09 47
--      E-mail copyright@iso.org
--      Web www.iso.org

DEFINITIONS AUTOMATIC TAGS EXTENSIBILITY IMPLIED ::=
BEGIN
-- EXPORTS (all)
IMPORTS;

-- Major and Minor Revision values for this ASN.1 Module
revMajISO24727-COMMON INTEGER ::= 1
revMinISO24727-COMMON INTEGER ::= 5

NonNegativeInt    ::= INTEGER (0..32767)
PositiveInt       ::= INTEGER (1..32767)

-- C.1.1 Constants
size-max-NameLength INTEGER ::= 255
size-max-NodePathLength INTEGER ::= 255
size-max-Padding INTEGER ::= 16
size-max-SecurityCondition INTEGER ::= 255

ByteValue ::= INTEGER (0..255)

ApplicationIdentifier ::= OCTET STRING
ObjectIdentifier ::= OBJECT IDENTIFIER
Name ::= VisibleString (SIZE(1..size-max-NameLength))
IFDName     ::= Name

GenericHandleType ::= OCTET STRING
GenericIdentifierType ::= OCTET STRING

TransactionIdentifier ::= GenericIdentifierType
```

```
URIType ::= UTF8String
-- The URIType is a string, which represents a URI according to RFC 3986.
-- A URI may be
-- * a URL according to RFC 1738 (e.g. http://www.example.com or
--   http://192.168.1.1 )
-- * a URN according to RFC 2141, which may be used to referece for example
-- * OIDs according to RFC 3061 (e.g. oid:1.0.24727.3.0.1)
-- * IETF documents according to RFC 2648, which in turn may define
--   protocols (e.g. ietf:rfc:4346 may indicate that TLS v1.1
--   shall be used to protect a channel)
-- * phone numbers according to RFC 3966
-- The list of registered names is maintained by IANA
-- (see http://www.iana.org/assignments/urn-namespaces ).

IFDAction ::= CHOICE {

        reset NULL,
        unpower NULL,
        eject NULL,
        confiscate NULL
}

IFDSessionIdentifier ::= GenericHandleType

IFDChannelHandle  ::= SEQUENCE {
    ref                                     OCTET STRING,
    protocolTerminationPoint  URIType OPTIONAL,
    sessionID                               IFDSessionIdentifier OPTIONAL,
    binding                           URIType OPTIONAL
}


END
```