For Publication on the ICAO Website

MACHINE READABLE TRAVEL DOCUMENTS

Digital Travel Credentials (DTC)

Virtual Component Data Structure and PKI Mechanisms

TECHNICAL REPORT

Version – 1.2
October 2020

DISCLAIMER: All reasonable precautions have been taken by ICAO to verify the information contained in this publication. However, the published material is being distributed without warranty of any kind, either expressed or implied; nor does it necessarily represent the decisions or policies of ICAO. The responsibility for the interpretation and use of the material contained or referred to in this publication lies with the reader and in no event shall ICAO be liable for damages arising from reliance upon or use of the same. This publication shall not be considered as a substitute for the government policies or decisions relating to information contained in it. This publication contains the collective views of an international group of experts, believed to be reliable and accurately reproduced at the time of printing. Nevertheless, ICAO does not assume any legal liability or responsibility for the accuracy or completeness of the views expressed by the international group of experts.

File	: WG3TF5_N0290 TR - Digital Travel Credentials v1.2.doc
Author	: ISO/IEC JTC1 SC17 WG3/TF5

Release Control

Release	Date	Description
0.01	May 2018	Initial Draft capturing the discussions in WG3 meeting, Port Douglas
0.02	May 2018	Added ASN specifications
0.03	May 2018	Minor edits
0.04	June 2018	First published draft
0.05	August 2018	Incorporating changes from Policy Group Discussions in Singapore and comment resolution of CH, JP and NZ comments
0.06	March 2019	Rewrite of the document based on DTC Policy Paper 1.0 approved by NTWG in November 2018 and WG3 discussions in Sunnyvale in March 2019
0.07	August 2019	Changes based on Comment Resolution, NTWG Munich feedback and August meeting of DTC editors
0.08	February 2020	Changes based on Outcomes of meeting of DTC editors in Singapore and NTWG Wellington discussions
0.91	May 2020	Comment resolution for Japan and Germany comments
1.0	May 2020	Final disposition of comments for version 0.91
1.1	June 2020	Renaming of the TR
1.2	July 2020	Allow multiple type of Interface Types for DTC-PC in the ASN.1 (based on comments from Germany). Corrected ASN.1 for DTCDOE length.

Table of contents

1. Introduction

1.1 Background

This document specifies the file structure of the Digital Travel Credentials (DTC) and the associated PKI to support the DTC.

1.2 DTC Approach

The DTC consists of a DTC – VC (Virtual Component) and an optional DTC-PC (Physical Component).

The DTC-VC is a file that can be stored on any medium and does not have any inherent access protection on its contents. In case, an associated DTC-PC is issued, there will be a cryptographic link between the DTC-VC and the DTC-PC.

There are two possibilities for the contents of the DTC-VC:

- The contents of the DTC-VC may be identical to the LDS and SOD of an existing eMRTD.
- The DTC-VC may be issued without any relation to an existing eMRTD.

Based on the above, the following three types of DTC are defined:

- eMRTD bound –no additional DTC-PC other than an underlying eMRTD.
- eMRTD-PC bound – there exists an additional DTC-PC apart from the underlying eMRTD.
- PC bound –only a DTC-PC exists and there is no underlying eMRTD.

The current version of the specification does not yet define the DTC-PC. This will be done in a separate document that will be released later. However, the structure of the DTC-VC will remain identical to the structure defined in this document and will not be affected by the choice of technology of the DTC-PC.

1.3 Interaction between DTC-PC and Inspection System (IS)

Inspection Systems should be able to differentiate between an eMRTD and a DTC-PC, when presented with either of them. For eMRTDs, the first file that is read by the IS is the EF.CardAccess (It is expected that BAC will be deprecated sometime in the future and hence this specification assumes that all eMRTDs presented to an IS support PACE). The DTC-PC will also present EF.CardAccess. The EF.CardAccess presented by the DTC-PC will contain an additional field called DTCCapabilitiesInfo, which is defined in this specification. If the EF.CardAccess does not contain DTCCapabilitiesInfo, then the presented token is an eMRTD. If it contains DTCCapabilitiesInfo, then it is a DTC-PC.

The cryptographic link between the DTC-VC and the DTC-PC will reuse anti cloning mechanisms already established in current eMRTDs viz. Chip Authentication or Active Authentication or other mechanism that may be defined by ICAO.

1.4 Scope of DTC specifications

This scope of the current specifications is as follows:

- Define a structure that is common to all flavours of DTC that may be issued.
- Define a mechanism to differentiate whether the DTC data is based on an existing eMRTD or is independent of an eMRTD
- Define a mechanism to recognize if a DTC-PC is associated with the DTC-VC.
- Define the requirements of the DTC-PC.
- In case the DTC-VC has a link with a DTC-PC, the capability to revoke the DTC.
- When the DTC-VC has the same information as an existing eMRTD, the ability to revoke the DTC independent of the eMRTD.

1.5 Terminology

1.5.1 Technical report terminology

The key words "MUST", “MUST NOT”, "SHALL", “SHALL NOT”, "REQUIRED", "SHOULD", “SHOULD NOT”, "RECOMMENDED", "MAY", and “OPTIONAL” in this document are to be interpreted as described in [RFC 2119].

In case OPTIONAL features are implemented, they MUST be implemented as described in this Technical Report.

1.5.2 Terms and Definitions

Term	Definition
DTC	Digital Travel Credential – Travel Credentials issued in a Digital Format
DTC-VC	DTC Virtual Component is a digital representation of the holder’s identity
DTC-PC	DTC Physical Component which has a cryptographic link to the DTC-VC
eMRTD bound DTC	A DTC no additional DTC-PC other than an underlying eMRTD
eMRTD-PC bound DTC	A DTC with an additional DTC-PC apart from the underlying eMRTD
PC bound DTC	A DTC where only a DTC-PC exists and there is no underlying eMRTD
DTC Signer	An entity that digitally signs the DTC-VC

2. Technical Specifications

2.1 DTC-VC Specification

2.1.1 DTCCapabilitiesInfo

The DTCCapabilitiesInfo is present for eMRTD-PC bound and PC bound DTC types to indicate the capabilities of the DTC-PC.

It has the following information:

2.1.1.1 DataTransferInterfaceType

This indicates the interface (NFC/BLE/etc.) that will be used by the DTC-PC to communicate with the Inspection System. The DTC-PC may support more than one interface.

2.1.1.2 UserConsentInfo

User Consent is necessary before the DTC-PC presents any identifiable information to the Inspection System. The UserConsentinfo indicates whether the device hosting the DTC-PC has a capability to assert User Consent on the device itself (Device Local) or will require a Proof of Possession (POP) to be negotiated between the DTC-PC and the Inspection System. The details of the POP will be specified in a separate document.

2.1.1.3 VirtualComponentPresence

The device hosting the DTC-PC MAY contain the DTC-VC as well, though this is NOT REQUIRED. This flag indicates to the Inspection System whether it can receive the DTC-VC from the same device.

2.1.1.4 SecurityAssuranceLevelIndicator

An indicator to identify the security/trust level associated with the DTC-PC device itself. [ISO/IEC 23220] is working on defining the semantics for this indicator and the same will be used here. The data structure defined by ISO/IEC 23220 is encapsulated in an OctetString.

2.1.2 DTCSecurityInfo

The DTCSecurityInfo provides the cryptographic link between the DTC-VC and the DTC-PC. It contains the DTCIdentifier, DTCDOE, SecurityInfos and ActiveAuthenticationPublicKeyInfo which are defined below. The DTCSecurityInfo MUST be present for eMRTD-PC bound and PC Bound DTC. It MUST NOT be present if DTC is eMRTD bound.

2.1.2.1 DTCIdentifier

The DTC-VC will have a unique identifier only in the case of an eMRTD-PC bound and PC bound DTC. In the case case of eMRTD-PC bound DTC, the DTC Identifier MUST be different from the Document Number in DG1. This allows the DTC to be revoked independently of the eMRTD. In the case of a PC bound DTC, the DTC Identifier MUST be the same as the Document Number in DG1. The restrictions for document number in DG1 as specified in [Doc 9303-10] apply to DTCIdentifier as well.

2.1.2.2 DTCDOE

The date of expiry of the DTC is represented by DTCDOE. It is present only in the case of an eMRTD-PC bound and PC bound DTC. In the case of the eMRTD-PC bound DTC, it

SHALL NOT be greater than the date of expiry of the eMRTD (as encoded in DG1). In the case of the PC Bound DTC, the value encoded in DTCDOE MUST exactly match the date of expiry encoded in DG1.

The date of expiry of the DTC MUST be encoded as an eight-digit string consisting of four digits for the year (YYYY) immediately followed by two digits for the number of the month (MM) and by two digits for the day (DD). The structure SHALL be as follows: YYYYMMDD. Following this format, 13 September 2019 will be shown as: 20190913.

2.1.2.3 SecurityInfos

The DTC-PC has an EF.CardAccess which contains the Security Infos for PACE, defined in [Doc 9303]-11 Section 9.2, along with an additional Security Info, DTCCapabilitiesInfo (See Section 2.1.1). The additional DTCCapabilitiesInfo MUST be present in EF.CardAccess of the DTC-PC and it can be used to determine that the presented document is a DTC-PC. If it does not contain the additional field, it is an eMRTD.

To indicate support for the protocols and supported parameters, the DTC-VC SHALL provide the following Security Infos in this field:

- All the Security Infos contained in EF.CardAccess (See above) including the DTCCapabilitiesInfo
- At least one of the following Security Infos
    - Security Infos for Active Authentication (See [Doc 9303]-11 Section 9.2)
    - Security Infos for Chip Authentication (See [Doc 9303]-11 Section 9.2)

2.1.2.4 ActiveAuthenticationPublicKeyInfo

The ActiveAuthenticationPublicKeyInfo contains the Active Authentication Public Key and is REQUIRED when implementing the OPTIONAL Active Authentication protocol as described in [Doc 9303]-11.

2.1.3 DTCContentInfo

The DTC-VC is defined in a DER encoded structure called DTCContentInfo.

The processing rules below apply.

m	mandatory – the field MUST be present
r	recommended - the field SHOULD be present
x	do not use – the field MUST NOT be populated
o	optional – the field MAY be present
c	conditional – the field is REQUIRED under certain conditions

Value		Comments
DTCContentInfo		
version	m	Value = v1
DTCData	m	
dtcSOD	c	MUST be present if DTC is eMRTD Bound or eMRTD-PC Bound. This field MUST NOT be present if DTC is PC Bound.
dtcDG1	m	
dtcDG2	m	
dtcDG3 – dtcDG16	o	
dtcSecurityInfo	c	See Section 2.1.2

Value		Comments
DTCIdentifier	m	See Section 2.1.2.1
DTCDOE	m	See Section 2.1.2.2
SecurityInfos	m	See Section 2.1.2.3
ActiveAuthenticationPublicKeyInfo	c	See Section 2.1.2.4
dtcOtherInfos	o	The dtcOtherInfos is for internal State or organization use.
DTCTBS	c	Contains the hash value of each data value in DTCData. MUST be present if DTC is eMRTD-PC Bound or PC Bound. This field MUST NOT be present if DTC is eMRTD Bound.
DTCSignerInfo	c	MUST be present if DTC is eMRTD-PC Bound or PC Bound. This field MUST NOT be present if DTC is eMRTD Bound.
certificate	m	Defined in [RFC 5280]. MUST contain the DTC Signer certificate
digestAlgorithm	m	The algorithm identifier of the algorithm used to produce the hash value of DTCTBS, each data value of DTCTBS and SignedAttrs.
signedAttrs	m	MUST include signing-time and message-digest. `MessageDigest` contains the computed hash value of ASN.1 DER encoded DTCTBS (Note: ASN.1 DER encoded DTCTBS does not include the EXPLICIT [0] tag)
signatureAlgorithm	m	The algorithm identifier of the algorithm used to produce the signature value, and any associated parameters.
dtcSignature	m	The result of the signature generation process. Message Digest Calculation Process: The result is the message digest of the complete DER encoding of the SignedAttributes value contained in the signedAttrs field. A separate encoding the signedAttrs field is performed for message digest calculation. The IMPLICIT [0] tag in the signedAttrs is not used for the DER encoding, rather an EXPLICIT SET OF tag is used. That is, the DER encoding of the EXPLICIT SET OF tag, rather than of the IMPLICIT [0] tag, MUST be included in the message digest calculation along with the length and content octets of the SignedAttributes value. dtcSignature Generation Process: The input to the dtcSignature generation process includes the result of the message

Value		Comments
		digest calculation process and the DTC signer's private key.

2.1.4 ASN.1 Specification

```
DTCContentInfo  ::=  SEQUENCE {
  version         Version,
  dtcData         DTCData,
  dtcTBS          [0] EXPLICIT DTCTBSValues OPTIONAL,
                     -- MUST be present if DTC is eMRTD-PC Bound or PC
                     -- Bound. This field MUST NOT be present if DTC is
                     -- eMRTD Bound.
  dtcSignerInfo   [1] EXPLICIT DTCSignerInfo OPTIONAL
                     -- MUST be present if DTC is eMRTD-PC Bound or PC
                     -- Bound. This field MUST NOT be present if DTC is
                     -- eMRTD Bound.
}

DTCTBSValues  ::=  SEQUENCE SIZE (3..ub-DTCData) OF DTCTBSValue

Version  ::=  INTEGER  {  v1(1)  }

ub-DTCData INTEGER  ::=  31

DTCData  ::=  SEQUENCE {
  dtcSOD                 [0]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of SOD defined
                           -- in [Doc 9303]-10.
                           -- MUST be present if DTC is eMRTD Bound or
                           -- eMRTD-PC Bound. This field MUST NOT be present
                           -- if DTC is PC Bound.
  dtcDG1                 [1]   IMPLICIT OCTET STRING,
                           -- Contains the encoding of Data Group 1 defined
                           -- in [Doc 9303]-10.
  dtcDG2                 [2]   IMPLICIT OCTET STRING,
                           -- Contains the encoding of Data Group 2 defined
                           -- in [Doc 9303]-10.
  dtcDG3                 [3]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 3 defined
                           -- in [Doc 9303]-10.
  dtcDG4                 [4]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 4 defined
                           -- in [Doc 9303]-10.
  dtcDG5                 [5]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 5 defined
                           -- in [Doc 9303]-10.
  dtcDG6                 [6]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 6 defined
                           -- in [Doc 9303]-10.
  dtcDG7                 [7]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 7 defined
                           -- in [Doc 9303]-10.
  dtcDG8                 [8]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 8 defined
                           -- in [Doc 9303]-10.
  dtcDG9                 [9]   IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 9 defined
                           -- in [Doc 9303]-10.
  dtcDG10                [10]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 10 defined
                           -- in [Doc 9303]-10.
  dtcDG11                  [11]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 11 defined
                           -- in [Doc 9303]-10.
  dtcDG12                  [12]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 12 defined
                           -- in [Doc 9303]-10.
  dtcDG13                  [13]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 13 defined
                           -- in [Doc 9303]-10.
  dtcDG14                  [14]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 14 defined
                           -- in [Doc 9303]-10.
  dtcDG15                  [15]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 15 defined
                           -- in [Doc 9303]-10.
  dtcDG16                  [16]  IMPLICIT OCTET STRING OPTIONAL,
                           -- Contains the encoding of Data Group 16 defined
                           -- in [Doc 9303]-10.
  ...,
  dtcSecurityInfo  [22]  EXPLICIT DTCSecurityInfo OPTIONAL,
                           -- MUST be present if DTC is eMRTD-PC Bound or PC
                           -- Bound. This field MUST NOT be present if DTC
                           -- is eMRTD Bound.

  dtcOtherInfos    [23]  EXPLICIT DTCOtherInfos OPTIONAL,
                           -- The dtcOtherInfos is for internal State use.
                           -- MAY be present if DTC is eMRTD-PC Bound or PC
                           -- Bound. This field MUST NOT be present if DTC
                           -- is eMRTD Bound as it is not part of signed
                          -- data.


  ...
}

DTCTBSValue  ::=  SEQUENCE {
  dtcHashNumber    DTCHashNumber,
  dtcHashValue     OCTET STRING
}

DTCHashNumber  ::=  INTEGER {
  dtcSODNum                  (0),
  dtcDG1Num                  (1),
  dtcDG2Num                  (2),
  dtcDG3Num                  (3),
  dtcDG4Num                  (4),
  dtcDG5Num                  (5),
  dtcDG6Num                  (6),
  dtcDG7Num                  (7),
  dtcDG8Num                  (8),
  dtcDG9Num                  (9),
  dtcDG10Num       (10),
  dtcDG11Num       (11),
  dtcDG12Num       (12),
  dtcDG13Num       (13),
  dtcDG14Num       (14),
  dtcDG15Num       (15),
  dtcDG16Num       (16),
  ...,
  dtcSecurityInfos (22),
  dtcOtherInfos              (23)
  ...
```

```
}

DTCSecurityInfo  ::= SEQUENCE {
  dtcIdentifier                       DTCIdentifier,
  dtcDOE                              DTCDOE,
  securityInfos                       SecurityInfos,
  activeAuthenticationPublicKeyInfo   ActiveAuthenticationPublicKeyInfo
                                       OPTIONAL,
  ...
}

DTCIdentifier  ::=  UTF8String (SIZE (1..9))
                         -- The restrictions for document number in DG1 as
                         -- specified in [Doc 9303-10] apply to
                         -- DTCIdentifier as well.

DTCDOE ::= UTF8String (SIZE (8))

SecurityInfos  ::=  SET OF SecurityInfo

SecurityInfo  ::=  SEQUENCE { protocol
              OBJECT IDENTIFIER,
  requiredData     ANY DEFINED BY protocol,
  optionalData     ANY DEFINED BY protocol OPTIONAL
}
-- Formed using data structure of SecurityInfo
DTCCapabilitiesInfo  ::=  SEQUENCE {
  protocol         OBJECT IDENTIFIER(id-icao-dtcCapabilitiesInfo),
  capabilities     Capabilities
}

-- DTC OIDs
id-icao OBJECT IDENTIFIER ::= {joint-iso-itu-t(2) international(23)
icao(136)}
id-icao-mrtd OBJECT IDENTIFIER ::= {id-icao 1}
id-icao-mrtd-security OBJECT IDENTIFIER ::= {id-icao-mrtd 1}
id-icao-dtc OBJECT IDENTIFIER ::= { id-icao-mrtd-security 12}
id-icao-dtcSigner OBJECT IDENTIFIER ::= {id-icao-dtc 1}
id-icao-dtcAttributes OBJECT IDENTIFIER ::= {id-icao-dtc 2}
id-icao-dtcCapabilitiesInfo OBJECT IDENTIFIER ::= {id-icao-dtcAttributes 1}


Capabilities  ::=  SEQUENCE {
  dataTransferInterfaceType          DataTransferInterfaceTypes,
  userConsentInfo                    BOOLEAN, -- True if proof of
                                              -- possession required
  virtualComponentPresence           BOOLEAN, -- True if the device hosting
                                              -- the DTC-PC hosts the
                                              -- DTC-VC, false otherwise

  securityAssuranceLevelIndicator    OCTET STRING, -- Value of this field to
                                                   -- be defined by ISO/IEC
                                                   -- 23220 in future
  ...
}

DataTransferInterfaceTypes ::= SET SIZE (1..MAX) OF DataTransferInterfaceType

DataTransferInterfaceType  ::=  ENUMERATED {
  nfc       (0),
  ble       (1),
  iso7816   (2),
  ...
```

```
}

ActiveAuthenticationPublicKeyInfo  ::=  SubjectPublicKeyInfo

SubjectPublicKeyInfo  ::=  SEQUENCE {
  algorithm       AlgorithmIdentifier,
                        --Defined in RFC 5280
  subjectPublicKey      BIT STRING
}

DTCSignerInfo  ::=  SEQUENCE {

  certificate                 Certificate,
                                -- Defined in RFC 5280
  digestAlgorithm       DigestAlgorithmIdentifier,
  signedAttrs                 [0] IMPLICIT SignedAttributes,
  signatureAlgorithm          SignatureAlgorithmIdentifier,
  dtcSignature                DTCSignature

    -- Message Digest Calculation Process:
    -- The result is the message digest of the complete DER encoding of the
    -- SignedAttributes value contained in the signedAttrs field.
    -- A separate encoding the signedAttrs field is performed for message
    -- digest calculation. The IMPLICIT [0] tag in the signedAttrs is not
    -- used for the DER encoding, rather an EXPLICIT SET OF tag is used.
    -- That is, the DER encoding of the EXPLICIT SET OF tag, rather than of
    -- the IMPLICIT [0] tag, MUST be included in the message digest
    -- calculation along with the length and content octets of the
    -- SignedAttributes value.
    --
    -- DTCSignature Generation Process:
    -- The input to the signature generation process includes the result of
    -- the message digest calculation process and the signer's private key.
}

DigestAlgorithmIdentifier  ::= AlgorithmIdentifier

SignatureAlgorithmIdentifier  ::= AlgorithmIdentifier

DTCOtherInfos  ::=  SEQUENCE SIZE (1..MAX) OF DTCOtherInfo

DTCOtherInfo  ::=  SEQUENCE  {
  infoDesc        UTF8String (SIZE (1..MAX)),
  infoValue       OCTET STRING
}


SignedAttributes  ::=  SET SIZE (2..MAX) OF Attribute

Attribute  ::=  SEQUENCE { attrType
  OBJECT IDENTIFIER, attrValues SET
  OF AttributeValue }

AttributeValue::=  ANY

DTCSignature  ::=  OCTET STRING


-- Message-digest attribute values
id-messageDigest OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)
rsadsi(113549) pkcs(1) pkcs9(9) 4 }

MessageDigest  ::=  OCTET STRING
  -- Contains the computed hash value of ASN.1 DER encoded DTCTBS (Note:
```

```
  -- ASN.1 DER encoded DTCTBS does not include the EXPLICIT [0] tag)


-- Signing-time attribute values
id-signingTime OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840)
rsadsi(113549) pkcs(1) pkcs9(9) 5 }

SigningTime  ::=  Time

Time ::= CHOICE {
  utcTime              UTCTime,
  generalizedTime      GeneralizedTime }
```

2.2 DTC PKI Certificate Profiles

Issuing States or organizations MUST issue certificates that conform to the profiles specified below if the DTC is linked to a physical device.

The CSCA that issues DTC Signer certificates SHALL be the same CSCA that issues Document Signer certificates.

Note: The country code for the issuer MUST be identical between the CSCA, the Document Signer Certificate used to sign the contained SOD (if present) and the DTC Signer Certificate.

Profiles for the certificate types are defined below.

The profiles use the following terminology for presence requirements of each of the components/extensions:

m mandatory – the field MUST be present
x do not use – the field MUST NOT be populated
o optional – the field MAY be present

The profiles use the following terminology for criticality requirements of extensions that may/must be included:

c critical – receiving applications MUST be able to process this extension.
nc non-critical - receiving applications that do not understand this extension MAY ignore it.

2.2.1 CSCA Certificate Profile

There is no change to the CSCA certificate profile.

2.2.2 DTC Signer Certificate Profile

DTC Signer certificates MUST comply with the Document Signer certificate profile defined in [Doc 9303]-12, with the following exceptions:

Subject Field:
The “subject” field of DTC Signer certificates MUST be populated as follows:

countryName: MUST be present. The value contains a country code that MUST follow the format of two letter country codes, specified in [DOC 9303]-3.

commonName: MUST be present. The value in this attribute MUST NOT exceed 9 characters in length.

Other attributes MUST NOT be included.

Certificate extensions:
DTC Signer certificates MUST contain the certificate extensions identified in the table below. All other certificate extensions MUST NOT be included.

Mandatory Certificate Extensions

Extension name	DTC Signer		Comments
	Presence	**Criticality**	
AuthorityKeyIdentifier	**m**	**nc**	
keyIdentifier	m		
authorityCertIssuer	o		
authorityCertSerialNumber	o		
SubjectKeyIdentifier	**m**	**nc**	
subjectKeyIdentifier	m		
KeyUsage	**m**	**c**	
digitalSignature	m		
nonrepudiation	x		
keyEncipherment	x		
dataEncipherment	x		
keyAgreement	x		
keyCertSign	x		
cRLSign	x		
encipherOnly	x		
decipherOnly	x		
ExtKeyUsage	**m**	**c**	See note 1

Note 1: The Object Identifier (OID) that MUST be included in the extendedKeyUsage extension for DTC Signer certificates is 2.23.136.1.1.12.1

3. PKI Trust and Validation

Validation of DTC Signer follows the same basic validation procedure as already specified in [Doc 9303]-12.

Extended Key Usage (EKU) MUST be included in all DTC Signer Certificates. The validation algorithm MUST ensure that the particular EKU as defined in this document is present in the DTC Signer Certificate.

The primary distribution mechanism for the DTC Signer Certificates is the DTC-VC.

3.1 DTC Trust Anchor

The Trust Anchor for validating DTCs is the CSCA of the Issuing State or Organization and Trust Anchor Management follows the process specified in [Doc 9303]-12.

3.2 Validation

The validation of a DTC involves the validation of the DTC-VC structure itself, the validation of the contents of the LDS contained within the DTC and optionally a validation of the link between the DTC-VC and the DTC-PC. The mechanism to establish the cryptographic link between the DTC-VC and the DTC-PC will be defined along with the specifications of the DTC-PC.

3.2.1 DTC-VC Validation

The validation of the DTC-VC is a three-step process.

1. The DTC Signer `Certificate` contained within the `DTCSignerInfo` can be used to verify the `DTCSignature` value and `SignedAttributes` value contained in the signedAttrs field.
2. The hash value of `DTCTBSValues` MUST match the value of `MessageDigest` in `SignedAttributes`. The hash algorithm identifier can be retrieved from the `digestAlgorithm` field.
3. The hash value of each data contained in the `DTCData` MUST be equal to the hash value contained in `DTCTBSValues`. The hash algorithm identifier can be retrieved from the `digestAlgorithm` field.

4. Reference documentation

The following documentation served as reference for this Technical Report:

[RFC 2119] RFC 2119, S. Bradner, "Key Words for Use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997

[Doc 9303] ICAO Doc 9303, 7th Edition, “Machine Readable Travel Documents”

[RFC 5280] RFC 5280, D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, W. Polk, , “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”, May 2008

[ISO/IEC 23220] ISO/IEC 23220 Cards and security devices for personal identification — Building blocks for identity management via mobile devices (working draft)

Annex A Abbreviations

Abbreviation	
ASN.1	Abstract Syntax Notation One
CA	Certification Authority
CRL	Certificate Revocation List
CSCA	Country Signing Certification Authority
DER	Distinguished Encoding Rule
DS	Document Signer
DTC	Digital Travel Credentials
DTC-PC	DTC Physical Component
DTC-VC	DTC Virtual Component
eMRTD	electronic MRTD
IC	Integrated Circuit
ICAO	International Civil Aviation Organization
LDS	Logical Data Structure
MRTD	Machine Readable Travel Document
PKI	Public Key Infrastructure
PKD	Public Key Directory
SO_D	Document Security Object
DTCDOE	DTC Date of Expiry
DTCTBS	DTC To Be Signed

- END -