ICAO

Doc 9303

Machine Readable Travel Documents

Eighth Edition, 2021

Part 10: Logical Data Structure (LDS) for Storage of Biometrics and Other Data in the Contactless Integrated Circuit (IC)

Approved by and published under the authority of the Secretary General

INTERNATIONAL CIVIL AVIATION ORGANIZATION

ICAO

Doc 9303

Machine Readable Travel Documents

Eighth Edition, 2021

Part 10: Logical Data Structure (LDS) for Storage of Biometrics and Other Data in the Contactless Integrated Circuit (IC)

Approved by and published under the authority of the Secretary General

INTERNATIONAL CIVIL AVIATION ORGANIZATION

Published in separate English, Arabic, Chinese, French, Russian
and Spanish editions by the
INTERNATIONAL CIVIL AVIATION ORGANIZATION
999 Robert-Bourassa Boulevard, Montréal, Quebec, Canada H3C 5H7

Downloads and additional information are available at www.icao.int/Security/FAL/TRIP

Doc 9303, *Machine Readable Travel Documents*
Part 10 — *Logical Data Structure (LDS) for Storage of Biometrics and Other Data in the Contactless Integrated Circuit (IC)*
Order No.: 9303P10
ISBN 978-92-9265-394-1 (print version)
ISBN 978-92-9275-420-4 (electronic version)

AMENDMENTS

Amendments are announced in the supplements to the *Products and Services Catalogue;* the Catalogue and its supplements are available on the ICAO website at www.icao.int. The space below is provided to keep a record of such amendments.

RECORD OF AMENDMENTS AND CORRIGENDA

AMENDMENTS		
No.	Date	Entered by
1	14/6/2024	ICAO

CORRIGENDA		
No.	Date	Entered by

TABLE OF CONTENTS

Page

1. SCOPE

Part 10 of Doc 9303 defines the Logical Data Structure (LDS) for eMRTDs required for global interoperability and defines the specifications for the organization of data on the contactless IC (Integrated Circuit). This requires the identification of all mandatory and optional Data Elements and a prescriptive ordering and/or grouping of Data Elements that MUST be followed to achieve global interoperability for electronic reading of the electronic passport.

Doc 9303-10 provides specifications to enable States and integrators to implement a contactless IC into an electronic travel document. This part defines all mandatory and optional data elements, file structures, and application profiles for the contactless IC.

The Eighth Edition of Doc 9303 incorporates the specifications for the optional Travel Records, Visa Records, and Additional Biometrics applications (known as LDS2 applications) as an extension of the mandatory eMRTD application (known as LDS1).

Part 10 shall be read in conjunction with:

- Part 1 — *Introduction*;
- Part 3 — *Specifications Common to all MRTDs*;
- Part 4 — *Specifications for Machine Readable Passports (MRPs) and other TD3 Size MRTDs*;
- Part 5 — *Specifications for TD1 Size Machine Readable Official Travel Documents (MROTDs)*;
- Part 6 — *Specifications for TD2 Size Machine Readable Official Travel Documents (MROTDs)*;

and the relevant contactless IC parts:

- Part 9 — *Deployment of Biometric Identification and Electronic Storage of Data in MRTDs;*
- Part 11 — *Security Mechanisms for MRTDs*;
- Part 12 — *Public Key Infrastructure for MRTDs*.

2. STRUCTURE OF DOC 9303-10

Doc 9303, Part 10 is organized into sections to include:

Specifications common to both LDS1 and LDS2 applications:

- Common attributes;
- All commands for both LDS1 and LDS2; and
- Common Elementary Files (EFs) for both LDS1 and LDS2;

Specifications for the LDS1 eMRTD application;

Specifications for the LDS2 applications:

- Travel Records;
- Visa Records;
- Additional Biometrics; and
- Specifications for LDS2 file access conditions.

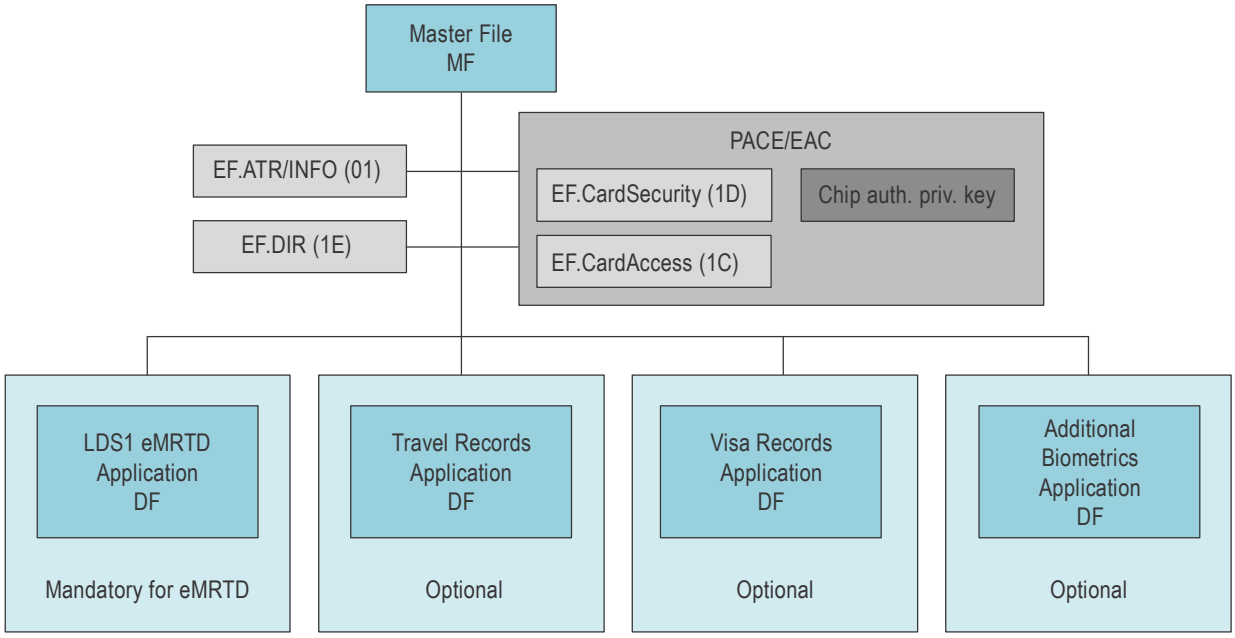


Figure 1. Applications for both LDS1 and LDS2

The eMRTD may support one, several or all of these:

- LDS1 eMRTD application MANDATORY;
- LDS2 Travel Records application OPTIONAL;
- LDS2 Visa Records application OPTIONAL;
- LDS2 Additional Biometrics application OPTIONAL.

3. SPECIFICATIONS COMMON TO LDS1 AND LDS2

3.1 Minimum Requirements for Interoperability

The following SHALL be the minimum requirements for interoperability of proximity contactless IC-based electronic passport:

- [ISO/IEC 14443-1], [ISO/IEC 14443-2], [ISO/IEC 14443-3], [ISO/IEC 14443-4] including all associated amendments and corrigenda;
- [ISO/IEC 10373-6] test specification compliant including all associated amendments and corrigendum;
- Type A or Type B signal interface;
- Support for a file structure as defined by [ISO/IEC 7816-4] for variable length transparent files;
- Support for one or more applications and appropriate [ISO/IEC 7816-4] commands as specified in Doc 9303.

3.2 Electrical Characteristics

The radio frequency power and signal interface SHALL be as defined in [ISO/IEC14443-2]. A minimum of 424 kilobits per second transmission speed is advised. Use of the EMD features specified in [ISO/IEC 14443-2] is OPTIONAL.

3.3 Physical Characteristics

It is recommended that the size of the coupling antenna area be in accordance with [ISO/IEC 14443-1] Class 1 (ID-1 antenna size) only.

3.4 Transmission Protocol

The eMRTD SHALL support half-duplex transmission protocol defined in [ISO/IEC14443-4]. The eMRTD SHALL support either Type A or Type B transmission protocols, and initialization, anticollision and transmission protocols according to ISO/IEC 14443.

3.4.1 Request Command and Answer to Request

The contactless IC SHALL respond to Request Command Type A (REQA) or Request Command Type B (REQB) with Answer to Request Type A (ATQA) or Answer to Request Type B (ATQB), as appropriate.

3.4.2 Random vs Fixed Identifier for the Contactless IC

The eMRTD may serve as a “beacon” in which the contactless IC emits a Unique IDentifier (UID) for Type A, and PUPI for Type B when initially activated. This might allow identification of the issuing authority. [ISO/IEC 14443] allows the choice of the option whether the eMRTD presents a fixed identifier, assigned uniquely for only that eMRTD, or a random number, which is different at each start of the communication dialogue. Some issuing States prefer to implement a unique number for security reasons or any other reason. Other issuers give greater preference to concerns about data privacy and the possibility to track persons due to fixed IC identifiers.

Choosing the one or the other option does not decrease interoperability since a reader terminal when compliant with ISO/IEC 14443 will understand both methods. The use of random IC identifiers is RECOMMENDED, but States MAY choose to apply unique UIDs for Type A or unique PUPIs for Type B.

3.5 Command Set

All commands, formats, and their status bytes are defined in [ISO/IEC 7816-4] and [ISO/IEC 7816-8] with the exception of the FILE AND MEMORY MANAGEMENT command. The minimum set of commands to be supported by the LDS1 eMRTD MUST be as follows:

SELECT;
READ BINARY.

It is recognized that additional commands will be required to establish the correct security environment and implement the optional security provisions identified in Doc 9303-11. Implementation of the mechanisms specified in Doc 9303-11 requires support of the following additional commands:

GET CHALLENGE;
EXTERNAL AUTHENTICATE/ MUTUAL AUTHENTICATE;
INTERNAL AUTHENTICATE;
MANAGE SECURITY ENVIRONMENT;
GENERAL AUTHENTICATE.

If optional LDS2 applications are present, the eMRTD SHALL additionally support the following commands:

For the Travel Records Application:

READ RECORD;
APPEND RECORD;
SEARCH RECORD;
FILE AND MEMORY MANAGEMENT;
PERFORM SECURITY OPERATION (PSO).

For the Visa Records Application:

READ RECORD;
APPEND RECORD;
SEARCH RECORD;
FILE AND MEMORY MANAGEMENT;
PERFORM SECURITY OPERATION (PSO).

For the Additional Biometrics Application:

UPDATE BINARY;
READ RECORD;
APPEND RECORD;
SEARCH RECORD;
ACTIVATE;
FILE AND MEMORY MANAGEMENT;
PERFORM SECURITY OPERATION (PSO).

Further details on command protocols can be found in Doc 9303-11.

3.5.1 SELECT

The LDS1 eMRTD supports two structure selection methods that are file identifier and short EF identifier. Readers support at least one of the two methods. The file identifier and Short EF Identifier is MANDATORY for the contactless IC operating system, but OPTIONAL for the reader.

3.5.2 READ BINARY

The support of the READ BINARY command with an odd INS byte by an eMRTD is CONDITIONAL. The eMRTD SHALL support this command variant if it supports data groups with 32 768 bytes or more.

3.6 Command Formats and Parameter Options (LDS1 and LDS2)

3.6.1 Application DF Selection Using the SELECT Command

Applications have to be selected by their DF name indicating the application identifier (AID). After the selection of an application, the file within this application can be accessed.

Note.— DF names have to be unique. Therefore, the selection of an application using the DF name can be done from wherever needed.

3.6.1.1 Selection of Master File

Table 1. SELECT Command for MF Selection

CLA	‘00’
INS	‘A4’
P1	‘00’
P2	‘0C’
Lc field	Absent
Data field	Absent
Le field	Absent

SELECT Command Response

Data field	Absent
SW1-SW2	‘9000’ Normal processing Other values to indicate checking or execution errors

Note.— It is RECOMMENDED that the SELECT MF command not be used.

3.6.1.2 Selection of Application DF

An application DF SHALL be selected by using SELECT command with DF name indicating application identifier (AID). The parameters for the Application Protocol Data Unit (APDU) command are shown below:

Table 2. SELECT Command with AID for Application DF Selection

CLA	'00'
INS	'A4'
P1	'04'
P2	'0C'
Lc field	Length of the command data field
Data field	DF name (AID)
Le field	Absent

SELECT Command Response

Data field	Absent
SW1-SW2	'9000' Normal processing Other values to indicate checking or execution errors

3.6.2 EF Selection Using the SELECT Command

EF is selected by the SELECT command with EF identifier. When the EF is selected, it has to be assured that the application DF storing the EF has previously been selected.

Table 3. SELECT Command with File Identifier for EF Selection

CLA	'00' / '0C'
INS	'A4'
P1	'02'
P2	'0C'
Lc field	'02'
Data field	File Identifier
Le field	Absent

SELECT Command Response

Data field	Absent
SW1-SW2	'9000' Normal processing Other values to indicate checking or execution errors

The eMRTD SHALL support the SELECT command with file identifier as specified in Table 3. The inspection system SHALL support at least one of the following methods:

- The SELECT command with file identifier as specified in Table 3;
- The READ BINARY command with even INS code and short EF identifier as specified in Table 5.

3.6.3 Reading Data from EF (READ BINARY)

There are two methods to read data from the eMRTD: by selecting EF then reading the data of the selected EF, or by reading the data directly using the short EF identifier. Support for short EF identifier is MANDATORY for the eMRTD. It is therefore RECOMMENDED that the inspection system use short EF identifier.

3.6.3.1 Reading Data from Selected EF (Transparent File)

Table 4. READ BINARY Command for Selected EF

CLA	'00' / '0C'
INS	'B0'
P1	Offset
P2	
Lc field	Absent
Data field	Absent
Le field	Present for encoding Ne > 0

READ BINARY Command Response

Data field	Data read
SW1-SW2	'9000' Normal processing Other values to indicate checking or execution errors

3.6.3.2 Reading Data Using EF Identifier (Transparent File)

Table 5. READ BINARY Command with Short EF Identifier

CLA	'00' / '0C'
INS	'B0'
P1	Short EF Identifier
P2	Offset
Lc field	Absent
Data field	Absent
Le field	Present for encoding Ne > 0. Maximum number of bytes expected in the response data field

READ BINARY Command Response

Data field	Data read
SW1-SW2	'9000' Normal processing Other values to indicate checking or execution errors

3.6.4 Extended Lc/Le Support

Depending on the size of the cryptographic objects (e.g. public keys, signatures), APDUs with extended length fields MUST be used to send this data to the eMRTD chip. For details on extended length field, see [ISO/IEC 7816-4].

3.6.4.1 Extended Length and eMRTD Chips

For eMRTD chips, support of extended length field is CONDITIONAL. If the cryptographic algorithms and key sizes selected by the issuing State require the use of extended length field, the eMRTD chips SHALL support extended length field. If the eMRTD chip supports extended length field this MUST be indicated in the ATS or in EF.ATR/INFO as specified in [ISO/IEC 7816-4].

3.6.4.2 Terminals

For terminals, support of extended length field is MANDATORY. A terminal SHOULD examine whether or not support for extended length field is indicated in the eMRTD chip's ATR/ATS or in EF.ATR/INFO before using this option. The terminal MUST NOT use extended length field for APDUs other than the following commands unless the exact input and output buffer sizes of the eMRTD chip are explicitly stated in the ATS or in EF.ATR/INFO.

- MSE:Set KAT;
- GENERAL AUTHENTICATE.

3.6.5 Command Chaining

Command chaining MUST be used for the GENERAL AUTHENTICATE command to link the sequence of commands to the execution of the protocol. Command chaining MUST NOT be used for other purposes unless clearly indicated by the chip. For details on command chaining, see [ISO/IEC 7816-4].

3.6.6 EFs Greater than 32 767 Bytes

The maximum size of an EF is normally 32 767 bytes, but some contactless ICs support larger files. A different READ BINARY parameter option and command format is required to access the data area when the offset is greater than 32 767. This format of command SHOULD be used after the length of the template has been determined and the need to access the data in the extended data area has been determined. For example, if the data area contains multiple biometric data objects, it may not be necessary to read the entire data area. Once the offset for the data area is greater than 32 767, this command format SHALL be used. The offset is placed in the command field rather than in the parameters P1 and P2.

Table 6. READ BINARY Command Format When Offset is Greater than 32 767 Bytes

CLA	'00' / '0C'
INS	'B1'
P1	See Table 7
P2	
Lc field	Length of the command data field
Data field	Offset DO'54'
Le field	Present for encoding Ne > 0. Maximum number of bytes expected in the response data field

READ BINARY Command Response

Data field	Discretionary DO'53'
SW1-SW2	'9000' Normal processing Other values to indicate checking or execution errors

Table 7. P1-P2 Coding of READ BINARY Command with INS = B1

P1								P2								Meaning
b8	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	**b8**	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	Selected EF
0	0	0	0	0	0	0	0	0	0	0	Not all equal					Short EF identifier
Not all zero											X	X	X	X	X	EF identifier

Both Length and Value fields of BER-TLV data object are variable length and can be encoded in different ways (see [ISO/IEC 7816-4]: “BER-TLV length fields”).

For performance reasons, communication between the eMRTD and the terminal SHOULD be kept as short as possible. Therefore, Length and Value fields in the BER-TLV data object SHOULD be as short as possible. This applies not only for Offset data objects in odd INS READ BINARY commands but also for all other BER-TLV data objects exchanged between the eMRTD and the terminal.

Examples for encoded Offset in Data-field:

- Offset: ‘0001’ is encoded as Tag = ’54’ Length = ’01’ Value = ’01’;
- Offset: ‘FFFF’ is encoded as Tag = ’54’ Length = ’02’ Value = ’FFFF’.

The subsequent READ BINARY commands SHALL specify the offset in the Data field. The final READ BINARY command SHOULD request the remaining data area.

With respect to [ISO/IEC 7816-4], there are no constraints specified on the offset value when bit 1 of INS is set to 1 to allow a broader use.

Note 1.— In some instances, there are eMRTDs where B1 and the traditional B0 READ Binary commands could not overlap. In other words, B0 only should be used to read the first 32 767 bytes and B1 from 32 K upward. For others there could be a small overlap of 256 bytes around the 32 767 threshold to allow a smoother transition between B0 and B1. For this latter group, B1 could be used right from the beginning of the file, i.e. with an offset starting from 0 to allow the same command to be used to read the full content.

Note 2.— The odd INS byte is not to be used by the inspection system if the size of an EF is 32 767 bytes or less.

3.7 Records Handling and Commands (LDS2)

Travel Records, Visa Records and Certificates MUST be stored in EF under the respective applications and have a linear structure with records of variable size. See Figures 4 and 5.

Records within each EF MUST be referenced by a record number. Each record number MUST be unique and sequential (zero referencing the selected record is out of the scope of this document).

Within each EF supporting a linear structure, the record numbers MUST be sequentially assigned when appending, such as in the order of creation; the first record (number one) is the first created record.

The following [ISO/IEC 7816-4] commands MUST be used for records access:

- APPEND RECORD Appending of Travel Records, Visas, Certificates;
- READ RECORD(S) Reading of one or more Travel Records, Visas, Certificates;
- SEARCH RECORD Searching of one or more Travel Records, Visas, Certificates.

Note.— Acronyms used in this sub-section are defined in [ISO/IEC 7816-4].

3.7.1 APPEND RECORD Command

The command initiates the appending of a new record at the end of a linear structure.

Table 8. APPEND RECORD Command

CLA	'0C'
INS	'E2'
P1	'00' (any other value is invalid)
P2	See Table 10
Lc field	Length of the command data field
Data field	Record to be appended
Le field	Absent

Table 9. APPEND RECORD Response

Data field	Absent
SW1-SW2	'9000' Normal processing; '6A84' Not enough memory space in the file; '6700' Wrong length (the record to be appended is longer than the specified maximum length); Other values to indicate checking or execution errors

Table 10. Coding of P2 in the APPEND RECORD Command

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x	x	-	-	-	Short EF identifier
-	-	-	-	-	0	0	0	Any other value is RFU

3.7.2 READ RECORD Command

The command returns the full or partial content of one or more addressed record(s) of the selected EF. Depending on the record size and the content of the Le field, the response data field contains one of the following:

- the first part of the addressed record;
- one (or more) full addressed record(s);

- one (or more) full addressed record(s) followed by the first part of the next record.

See [ISO/IEC 7816-4] for details and Appendix H for an example of reading a travel record.

Figure 2 illustrates the response data field. The comparison of N_r with the TLV structure indicates whether the unique record (read one record) or the last record (read all records) is incomplete, complete or padded.

Table 11. READ RECORD Command

CLA	'0C'	
INS	'B2'	
P1	Record number ('00' references the current record)	
P2	See Table 13	
Lc field	Absent	
Data field	INS = 'B2'	Absent
Le field	Maximum number of bytes to be read encoded as extended length field; Le = '00 00 00' (any other value is out of scope of the specification)	

Table 12. READ RECORD Response

Data field	Data read
SW1-SW2	'9000' Normal processing; '6A83' (Record not found); Other values to indicate checking or execution errors

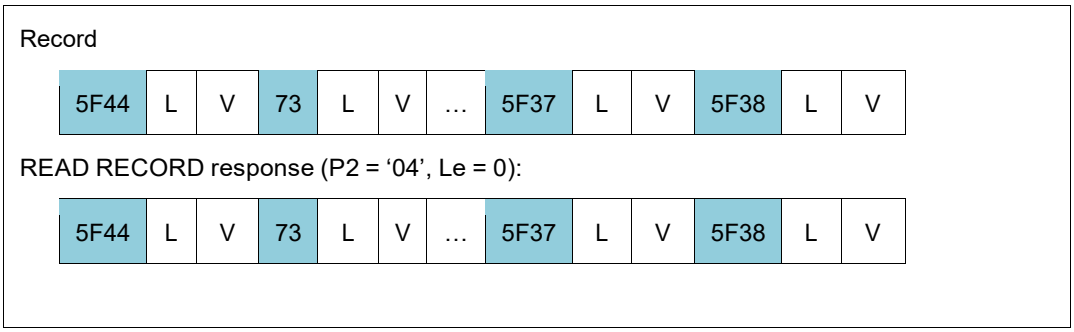
Table 13. Coding of P2 with the READ RECORD Command

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
x	x	x	x	x	-	-	-	Short EF identifier
-	-	-	-	-	1	x	x	**Record number in P1**
-	-	-	-	-	1	0	0	— Read record P1
-	-	-	-	-	1	0	1	— Read all records from P1 up to the last

Note 1.— Other bits combinations are out of scope of this specification. If the Le field contains only bytes set to '00', then the command should read completely either the single requested record, or the requested sequence of records, depending on bits 3, 2 and 1 of P2 and within the limit of the maximum supported length for the extended Le field.

Note 2.— The READ RECORD command with short length fields is out of scope of this specification.

Case a — Complete read of one record (the Le field contains only bytes set to ‘00’)



Case b — Read several records up to the end of the file (the Le field contains only bytes set to ‘00’)

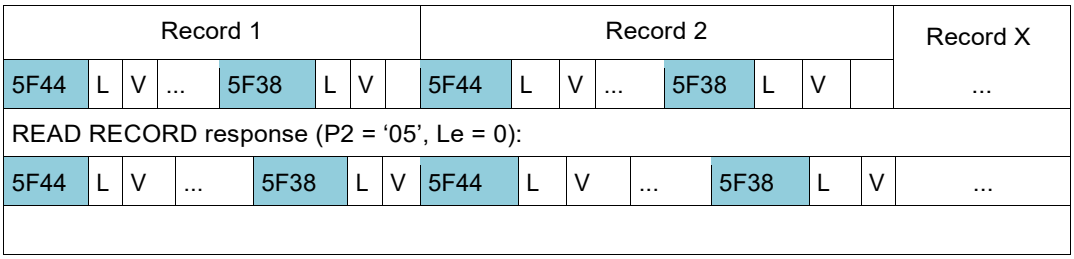


Figure 2. Response Data Fields

3.7.3 SEARCH RECORD Command

The command initiates a search on records stored within the respective EF. The command data field contains Record handling DO’7F76’ defining file reference, search configuration and search string (see Table 17). The response data field returns the Record handling DO’7F76’ containing one or more DO’02’ and containing the record number matching the search criteria within the addressed EF.

In an EF supporting records of variable size with a linear structure, the search MAY NOT take into account the records with a search window shorter than the search string.

Table 14. SEARCH RECORD Command

CLA	‘0C’
INS	‘A2’
P1	‘00’
P2	See Table 16
Lc field	Length of command data field
Data field	Record handling DO’7F76’ (See Table 17)
Le field	‘00’ (short length) or ’00 00’ (extended length)

Table 15. SEARCH RECORD Response

Data field	Record handling template DO‘7F76‘ containing one file reference DO‘51’ with one or more integer DO‘02‘ and containing a record number matching the search criteria
SW1-SW2	‘9000’ Normal processing; ‘6282’ Warning: Unsuccessful search Other values to indicate checking or execution errors

Note.— The response data field may be absent if no match is found.

Table 16. Coding of P2 for the SEARCH RECORD Command

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	1	1	1	1	0	0	0	Search record through multiple EFs
Any other value is RFU.								

Table 17. Record Handling Template for Enhanced Multiple Record Search

Tag	Value			Notes
‘7F76’				Record handling DO
	Tag	**Value**		
	‘51’	File identifier or short EF identifier		File reference DO
	‘A1’			Search configuration template
	Tag	**Value**		
	‘80’	‘00’ / ’30’		Search configuration parameter: - search in record number ascending order - step-width for the search: byte-wise - search termination: ‘00’ - Search all addressed records ‘30’ - Terminate search after the first match
	‘B0’			Search window template
	Tag	**Value**		
	‘02’	Offset		
	‘02’	Number of bytes		
	Tag	**Value**		
	‘A3’			Search string template

Tag	Value				Notes
		Tag	**Value**		
		‘B1’			
			Tag	**Value**	
			‘81’	Search string	

Note 1.— An empty offset DO in the search window template is not supported.

Note 2.— If the search window template makes use of the value ‘00’ for the number of bytes, the LDS2 eMRTD chip MUST search all bytes from the offset in the records.

Note 3.— The SEARCH RECORD command supports only the DOs specified in Table 17. This implies that the SEARCH RECORD command supports exactly one file reference DO in the record handling DO and exactly one search string in the search string template. The command MAY ignore additional DOs or answer with an error code if additional DOs are used.

3.8 Transparent Files Handling and Other (LDS2)

The Additional Biometrics transparent EFs are created by the LDS2 eMRTD issuer in the Operational Deactivated state (creation mechanism is out of scope of this specification). In the Deactivated state, the EF can be selected, written, updated and read with appropriate authorizations.

The following [ISO/IEC 7816-4] commands MUST be used for writing and reading Additional Biometrics transparent EFs:

- UPDATE BINARY Writing of Additional Biometrics;
- READ BINARY Reading of Additional Biometrics.

The following [ISO/IEC 7816-9] command MUST be used for activating the transparent EF after LSD2 read and write access conditions are successfully satisfied:

- ACTIVATE Activating of Additional Biometrics EF.

Note.— Acronyms used in this sub-section are defined in [ISO/IEC 7816-4].

In the Activated state, the EF can be selected and read with appropriate authorizations (related to the Activated state), and no authorization of any kind allows for writing or appending the transparent EF.

The FILE AND MEMORY MANAGEMENT (FMM) command MUST be used before writing to determine if there is enough available memory space in the EF.

The IS MUST use the following writing sequence for the EF.Biometrics:

- The first UPDATE BINARY (odd INS) command MUST contain the following DOs in the data field:

  - DO'54' containing the offset '00';
  - DO'53' which MAY contain the first block of the data to be stored. This DO MAY be empty ('53 00'); and

  - Proprietary DO'C0' indicating the total EF size (memory size to allocate) is optional.

  Note 1.— The LDS2 eMRTD MAY use the EF size information in DO'C0' for the internal memory allocation (e.g. for explicit dynamic memory allocation). If the LDS2 eMRTD does not support the EF size information DO (e.g., memory has been allocated statically by the issuer, or LDS2 eMRTD supports implicit dynamic EF memory reallocation), then the LDS2 eMRTD MAY ignore the DO'C0', proceed with writing of the first block of the EF and return '9000', or it MAY return the '6A80' error for incorrect parameter in the command data field.

  Note 2.— If the LDS2 eMRTD returns any error in response to UPDATE BINARY with the proprietary DO'C0', then the IS MUST send the standard [ISO/IEC 7816-4] UPDATE BINARY (odd INS) command with zero offset DO'54' and DO'53', without the DO'C0'.

- Subsequent UPDATE BINARY (odd INS, without DO'C0') commands SHOULD use the offset n+1 where n denotes the number of bytes written so far to the EF.Biometrics, i.e. the terminal SHOULD sequentially write the EF data without a gap or overlap between the two consecutive UPDATE BINARY commands.

- READ BINARY command MAY be used after any UPDATE BINARY command to verify the data written to the EF.

- The ACTIVATE command MUST finalize EF.Biometrics personalization by permanently disabling writing into the EF.

3.8.1 UPDATE BINARY Command

A contactless IC which supports the Additional Biometrics Application MUST support the UPDATE BINARY command with odd INS byte 'D7' according to Table 18.

The value of the BER-TLV Offset Data Object in the command data field specifies the offset; the value of the BER-TLV Discretionary Data Object in the command data field specifies the data to be written; the value of the optional BER-TLV File Size Data Object in the command data field specifies the total EF size. The length fields of these BER-TLV data objects should be encoded as shortly as possible.

When the command data field of UPDATE BINARY command has proprietary DO'C0', the bit 8 of CLA byte of command APDU MUST be set to 1 (CLA = '8C').

Table 18. UPDATE BINARY Command with odd INS

CLA	’0C’ / ‘8C’
INS	‘D7’
P1	File identifier ‘00 00’ identifies the current EF
P2	
Lc	Length of the command data field
Data field	Offset Data Object (tag ‘54’) || Discretionary Data Object (tag ‘53’) || File Size Data Object (tag ‘C0’) (optional)
Le	Absent

Table 19. UPDATE BINARY Response

Data field	Absent
SW1-SW2	‘9000’ Normal processing; ‘6A84’ (Not enough memory space in the file) ‘6A80’ Incorrect parameters in the command data field (e.g., DO’C0 not supported) ‘6982’ Security status not satisfied: The EF.Biometrics is in EF Activated state Other values to indicate checking or execution errors

If the IS does not follow the UPDATE BINARY sequence as specified in Section 3.8 (i.e. the first UPDATE BINARY does not start at offset 0), the LDS2 eMRTD chip MAY terminate the UPDATE BINARY command with an error.

3.8.2 ACTIVATE Command

The ACTIVATE command initiates the transition of the currently selected Additional Biometrics EF from the Deactivated state to the Activated state.

Table 20. ACTIVATE Command

CLA	‘0C’
INS	‘44’
P1	‘00’
P2	‘00’
Lc	Absent
Data field	Absent
Le	Absent

Table 21. ACTIVATE Response

Data field	Absent
SW1-SW2	'9000' Normal processing; Other values to indicate checking or execution errors *Note 1.— SW1-SW2 = '61XX' (normal processing) and SW1-SW2 = '62XX' or '63XX' (warning processing) are out of scope of this document.*

After successful execution of this command, the currently selected EF.Biometrics MUST be switched to the Activated state. In case an error occurs (SW different from '9000'), the currently selected EF.Biometrics MUST remain in the Deactivated state.

Immediately after successful execution of this command (SW1-SW2 = '9000'), the effective authorization required to perform an action on the EF.Biometrics MUST be the one corresponding to the Activated state (according to Table 98). The effective authorization corresponding to the Deactivated state MUST NOT raise any access rights for the EF.Biometrics.

3.8.3 FILE AND MEMORY MANAGEMENT Command

FILE AND MEMORY MANAGEMENT (FMM) command initiates a query of the used or free memory size for the addressed EF. This command is provided for LDS2 eMRTD as proprietary. This command may be used for checking the available free space for the addressed EF before writing or appending. This command may also be used for getting the last appended record number for reading. P1 indicates the EF addressing method, the current EF or file reference DO'51' can be used. P2 indicates the content of the query. The total number of bytes in the addressed EF with transparent or record structure and the number of existing or remaining records for the addressed record EF are provided. The total number of bytes comprises bytes available in the EF without any structural information. This number excludes any structural information that may be required by the LDS2 eMRTD chip. The assumption for the number of remaining records is that the size of all remaining records is at maximum. After a successful FMM command, the referenced EF becomes the current EF.

Table 22. FILE AND MEMORY MANAGEMENT (FMM) Command

CLA	'8C'	
INS	'5F'	
P1	See Table 23	
P2	See Table 24	
Lc	Absent for encoding Nc = 0, present for encoding Nc > 0	
Data field	P1 = '00'	Absent
	P1 = '01'	File reference DO'51' (See [ISO/IEC 7816-4])
Le	'00'	

P1 specifies the EF selection method. P2 contains a bit map specifying which information MUST be included in the response.

Table 23. Coding of P1 in the FFM Command

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0	0	0	0	0	Current EF
0	0	0	0	0	0	0	1	File reference DO’51’ in the command data field
Any other value is RFU.								

Table 24. Coding of P2 in the FFM Command

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
-	-	-	-	-	-	-	1	Total number of bytes in the addressed EF
-	-	-	-	-	-	1	-	Number of remaining records in the addressed record EF
-	-	-	-	-	1	-	-	Number of existing records in the addressed record EF
x	x	x	x	x	-	-	-	00000 (any other value is RFU)
Any other value is RFU.								

Table 25. Coding of DO’51’ in the FMM Command Data Field

Tag	Length	Value
‘51’	1	Short EF identifier (bits b8 to b4 encode a number from 1 to 30; bits b3 to b1 are set to 000)
	2	File identifier

The FMM command response contains a set of DOs representing the requested file and memory size information.

Table 26. FMM Command Response

Data field	Absent or control information according to P2. See Table 27.
SW1-SW2	‘9000’, checking or execution errors as per [ISO/IEC 7816-4]

Table 27. File and Memory Management

Tag	Length	Value		
'7F78'	Var	File and memory management DOs		
		Tag	**Len**	**Value**
		'81'	Var	Total number of bytes in the addressed EF
		'82'	Var	Number of remaining records in the addressed record EF
		'83'	Var	Number of existing records in the addressed record EF

Note 1.— The LDS2 eMRTD chip MUST return only the Data objects in the FMM DO that are requested by means of P2.

Note 2.— The FMM response data is valid only for the specified EF. FMM response data from different EFs may not be independent, e.g. if different EFs share the available memory. The IS should take this into account if combining FMM response data of different EFs.

Note 3.— When secure messaging is applied to the FMM command, Secure Messaging (SM) DO'85' MUST be used for encapsulating encrypted command data.

3.9 File Structure Specifications

Information in an LDS2 eMRTD is stored in a file system defined in [ISO/IEC 7816-4]. The file system is organized hierarchically into dedicated files (DFs) and elementary files (EFs). DFs contain EFs or other dedicated files. An optional master file (MF) may be the root of the file system.

Note.— The need for a master file is determined by the choice of operating systems, LDS1 or LDS2 applications, and optional access conditions.

3.9.1 Encoding of Data

The following types of coding are permitted for the Data Elements:

A = Alpha character [a-z, A-Z];
N = Numeric character [0-9];
S = Special character ['<'];
B = Binary data;
U = UTF-8 encoded UNICODE characters.

UTF-8 encoding of UNICODE characters:

- For any character equal to or below 127 (hex '7F'), the UTF-8 encoding uses one byte which is the same as the ASCII value;

- For characters equal to or below 2 047 (hex ‘07FF’), the UTF-8 encoding uses two bytes;
    - The first byte has two high bits set and the third bit clear (i.e. hex ‘C2’ to ‘DF’);
    - The second byte has the high bit set and the second bit clear (i.e. ‘80’ to ‘BF’);
- For all characters equal to or greater than 2 048 and less that 65 535 (hex ‘FFFF’), the UTF-8 encoding uses three bytes.

3.10 Application Selection — DF

The eMRTDs SHALL support at least one application as follows:

- The LDS1 eMRTD application is MANDATORY;
    - The LDS1 eMRTD application SHALL consist of data recorded by the Issuing State or organization, Data Groups 1 through 16 together with the Document Security Object (EF.SOD);
    - The Document Security Object (EF.SOD) within the LDS1 eMRTD application consists of the hash values as defined in Doc 9303-11 and Doc 9303-12 for the Data Groups in use, and is needed to validate the integrity of data created by the issuer and stored in the LDS1 eMRTD application.
- The LDS1 eMRTD application MAY optionally support the additional LDS2 applications described in Doc 9303 as:
    - Travel Records Application;
    - Visa Records Application; and
    - Additional Biometrics Application.

In addition, issuing States or organizations may wish to add other applications. The file structure SHALL accommodate such additional applications, but the specifics of such applications are out of scope of Doc 9303.

The LDS1 and LDS2 applications SHALL be selected by use of the Application Identification (AID) as a reserved DF name. The AID SHALL consist of the Registered Application Identifier assigned by ISO according to [ISO/IEC 7816-5] and a Proprietary Application Identifier Extension (PIX) as specified within this document:

The context of LDS1 eMRTD application uses two different tag allocation schemes for application class tag, as defined in Doc 9303-10 (LDS tag) and [ISO/IEC 7816-6] (Interindustry tag):

- EF.ATR/INFO and EF.DIR use Interindustry tag allocation scheme;
- DFs and their EFs use LDS tag allocation scheme.

Interindustry tags specified in this document are used in LDS context, so coexistent tag allocation scheme is not required.

3.11 Common Elementary Files (EFs)

The following common EFs for LDS1 and LDS2 applications MAY be existed under the MF:

- EF.ATR/INFO;
- EF.DIR;
- EF.CardAccess; and
- EF.CardSecurity.

3.11.1 EF.ATR/INFO (CONDITIONAL)

EF.ATR/INFO is a transparent EF contained in the master file and is conditionally REQUIRED if the optional LDS2 application is present. This EF is optional if only LDS1 application is present. The short EF identifier at the MF level is ’01’.

Table 28. EF.ATR/INFO

File Name	EF.ATR/INFO
File ID	‘2F01’
Short EF Identifier	‘01’
Select Access	ALWAYS
Read Access	ALWAYS
Write/Update/Erase Access	NEVER
File structure	Transparent
Size	Variable

The contents of the EF.ATR/INFO can be retrieved by using the SELECT command followed by the READ BINARY command. The READ BINARY command response data field contains the content of the EF.ATR/INFO.

Table 29. Data Elements of EF.ATR/INFO for LDS2

<table>
<tr><th>Tag</th><th>Length</th><th colspan="3">Value</th><th>Notes</th></tr>
<tr><td rowspan="4">‘47’</td><td rowspan="4">‘03’</td><td colspan="4">Card capabilities</td></tr>
<tr><td colspan="3">byte 1 - first software function</td><td>b8 = 1: DF selection by full DF name
b7 to b4 and b1 are out of scope of Doc 9303 b3 = 1: short EF identifier supported
b2 = 1: record number supported</td></tr>
<tr><td colspan="3">byte 2 - second software function</td><td>b8, b7, b6 and b5 are out of scope of Doc 9303 b4 to b1 = 0001: one byte data unit size</td></tr>
<tr><td colspan="3">byte 3 - third software function</td><td>b8 = 1: command chaining supported
b7 = 1: Extended Lc and Le fields supported b6 = 1: Extended length information in EF.ATR/INFO
b5 to b1 are out of scope of Doc 9303</td></tr>
<tr><td rowspan="4">‘7F66’</td><td rowspan="4">Var</td><td colspan="4">Extended length information</td></tr>
<tr><td>Tag</td><td>Length</td><td>Value</td><td>Notes</td></tr>
<tr><td>‘02’</td><td>Var</td><td>Positive integer - the maximum number of bytes in a command APDU</td><td>MUST be at least 1 000 (decimal) for LDS2</td></tr>
<tr><td>‘02’</td><td>Var</td><td>Positive integer - the maximum number of bytes expected in the response APDU</td><td>MUST be at least 1 000 (decimal) for LDS2</td></tr>
</table>

Note 1.— Further data objects MAY be present in EF.ATR/INFO.

Note 2.— EF.ATR/INFO uses Interindustry tag allocation scheme as defined in [ISO/IEC 7816-4].

3.11.2 EF.DIR (CONDITIONAL)

EF.DIR is a transparent EF contained in the master file defined by [ISO/IEC 7816-4]. EF.DIR is conditionally REQUIRED if any optional LDS2 applications are present. If any optional LDS2 applications are present EF.DIR MUST be included in SecurityInfos present in EF.CardSecurity. A full description of SecurityInfo for EF.DIR can be found in Doc 9303-11. The short EF identifier at the MF level is ’1E’.

Table 30. EF.DIR

File Name	EF.DIR
File ID	‘2F00’
Short EF Identifier	‘1E’
Select Access	ALWAYS
Read Access	ALWAYS
Write/Update/Erase Access	NEVER
File structure	Transparent
Size	Variable

EF.DIR is RECOMMENDED to be present in the MF. EF.DIR MUST be present if more than the mandatory LDS1 application is present and indicate a list of applications supported by the eMRTD. It MUST contain a set of application templates containing application identifier DO in any order.

Table 31. EF.DIR Format

Tag	L	Value			Description
‘61’	‘09’				LDS1 eMRTD Application Template
		Tag	**L**	**Value**	LDS1 eMRTD Application International AID: ‘A0 00 00 02 47 10 01’
		‘4F’	‘07’	‘A0 00 00 02 47 10 01’	
‘61’	‘09’				Travel Records Application Template
		Tag	**L**	**Value**	Travel Records International AID: ‘A0 00 00 02 47 20 01’
		‘4F’	‘07’	‘A0 00 00 02 47 20 01’	
‘61’	‘09’				Visa Records Application Template
		Tag	**L**	**Value**	Visa Records International AID: ‘A0 00 00 02 47 20 02’
		‘4F’	‘07’	‘A0 00 00 02 47 20 02’	
‘61’	‘09’				Additional Biometrics Application Template
		Tag	**L**	**Value**	Additional Biometrics International AID: ‘A0 00 00 02 47 20 03’
		‘4F’	‘07’	‘A0 00 00 02 47 20 03’	

Note.— EF.DIR uses standard tag allocation scheme as defined in [ISO/IEC 7816-4].

3.11.3 EF.CardAccess (CONDITIONAL)

EF.CardAccess is a transparent EF contained in the master file and is conditionally REQUIRED if the optional PACE access control as defined in Doc 9303-11 is invoked. A full description of SecurityInfos for PACE can be found in Doc 9303-11.

The short EF identifier at the MF level is '1C'.

Table 32. EF.CardAccess

File Name	EF.CardAccess
File ID	'011C'
Short EF Identifier	'1C'
Select Access	ALWAYS
Read Access	ALWAYS
Write/Update/Erase Access	NEVER
File structure	Transparent
Size	Variable

The file CardAccess contained in the master file is REQUIRED if PACE is supported by the eMRTD chip and SHALL contain the following SecurityInfos that are required for PACE:

- `PACEInfo`;
- `PACEDomainParameterInfo`.

Table 33. EF.CardAccess Storage on the IC

File Name	EF.CardAccess
File ID	'011C'
Short EF ID	'1C'
Read Access	ALWAYS
Write Access	NEVER
Size	Variable
Content	DER encoded SecurityInfos. See Doc 9303-11.

3.11.4 EF.CardSecurity (CONDITIONAL)

EF.CardSecurity is a transparent EF contained in the master file and is conditionally REQUIRED if the optional PACE with Chip Authentication Mapping as defined in Doc 9303-11 is invoked. A full description of SecurityInfos for PACE with Chip Authentication Mapping can be found in Doc 9303-11.

The short EF identifier at the MF level is ‘1D’.

EF.CardSecurity contained in the MF is REQUIRED if:

- PACE with Chip Authentication Mapping is supported by the IC;
- Terminal Authentication in the MF is supported by the IC; or
- Chip Authentication in the MF is supported by the IC.

and MUST contain:

- ChipAuthenticationInfo as required by Chip Authentication;
- ChipAuthenticationPublicKeyInfo as required by PACE-CAM/Chip Authentication;
- TerminalAuthenticationInfo as required by Terminal Authentication;
- the SecurityInfos contained in EF.CardAccess.

The file EF.CardSecurity contained in the master file is REQUIRED if PACE with Chip Authentication Mapping is supported by the eMRTD chip and SHALL contain the following SecurityInfos:

- ChipAuthenticationPublicKeyInfo as required for PACE-CAM;
- The SecurityInfos contained in CardAccess.

Table 34. EF.CardSecurity Storage on the IC

File Name	EF.CardSecurity
File ID	‘011D’
Short EF ID	‘1D’
Read Access	PACE
Write Access	NEVER
Size	Variable

The file CardSecurity SHALL be implemented as SignedData according to [RFC 3369] with content type id-SecurityObject within the field encapContentInfo. The Security Objects SHALL be signed by the Document Signer. The Document Signer Certificate MUST be included in SignedData. The following Object Identifier SHALL be used to identify the content type:

```
bsi-de OBJECT IDENTIFIER ::= {
    itu-t(0) identified-organization(4) etsi(0)
    reserved(127) etsi-identified-organization(0) 7
}
id-SecurityObject OBJECT IDENTIFIER ::= {
    bsi-de applications(3) eID(2) 1
}
```

The data structure SignedData is defined as follows:

```
SignedData ::= SEQUENCE{
    version CMSVersion,
    digestAlgorithms DigestAlgorithmIdentifiers,
    encapContentInfo EncapsulatedContentInfo,
    certificates [0] IMPLICIT CertificateSet OPTIONAL,
    crls [1] IMPLICIT RevocationInfoChoices OPTIONAL,
    signerInfos SignerInfos
}

DigestAlgorithmIdentifiers ::= SET OF DigestAlgorithmIdentifier

EncapsulatedContentInfo ::= SEQUENCE {
    eContentType ContentType,
    eContent [0] EXPLICIT OCTET STRING OPTIONAL
}

ContentType ::= OBJECT IDENTIFIER

SignerInfos ::= SET OF SignerInfo

SignerInfo ::= SEQUENCE {
    version CMSVersion,
    sid SignerIdentifier,
    digestAlgorithm DigestAlgorithmIdentifier,
    signedAttrs [0] IMPLICIT SignedAttributes OPTIONAL,
    signatureAlgoritm SignatureAlgorithmIdentifier,
    signature SignatureValue,
    unsignedAttrs [1] IMPLICIT UnsignedAttributes OPTIONAL
}
```

```
SignerIdentifier ::= CHOICE {
    issuerAndSerialNumber IssuerAndSerialNumber,
    subjectKeyIdentifier [0] SubjectKeyIdentifier
}

SignatureValue ::= OCTET STRING
```

4. LDS1 eMRTD APPLICATION (MANDATORY)

The LDS1 eMRTD structure provides space to store and digitally sign mandatory and optional data elements that can be used to link the holder to the document. The information stored in the LDS1 eMRTD portion of the eMRTD becomes static at the time of issuance, and cannot be modified in any possible way. This feature is necessary to ensure that personal information is protected, and that document tampering may be more easily detected. While the LDS1 version of eMRTD includes optional data fields that could be used to expand the use of the eMRTD (i.e. additional biometrics, automated border clearance, etc.), the requirement of write-protecting the LDS1 eMRTD chip application at the time of issuance is MANDATORY.

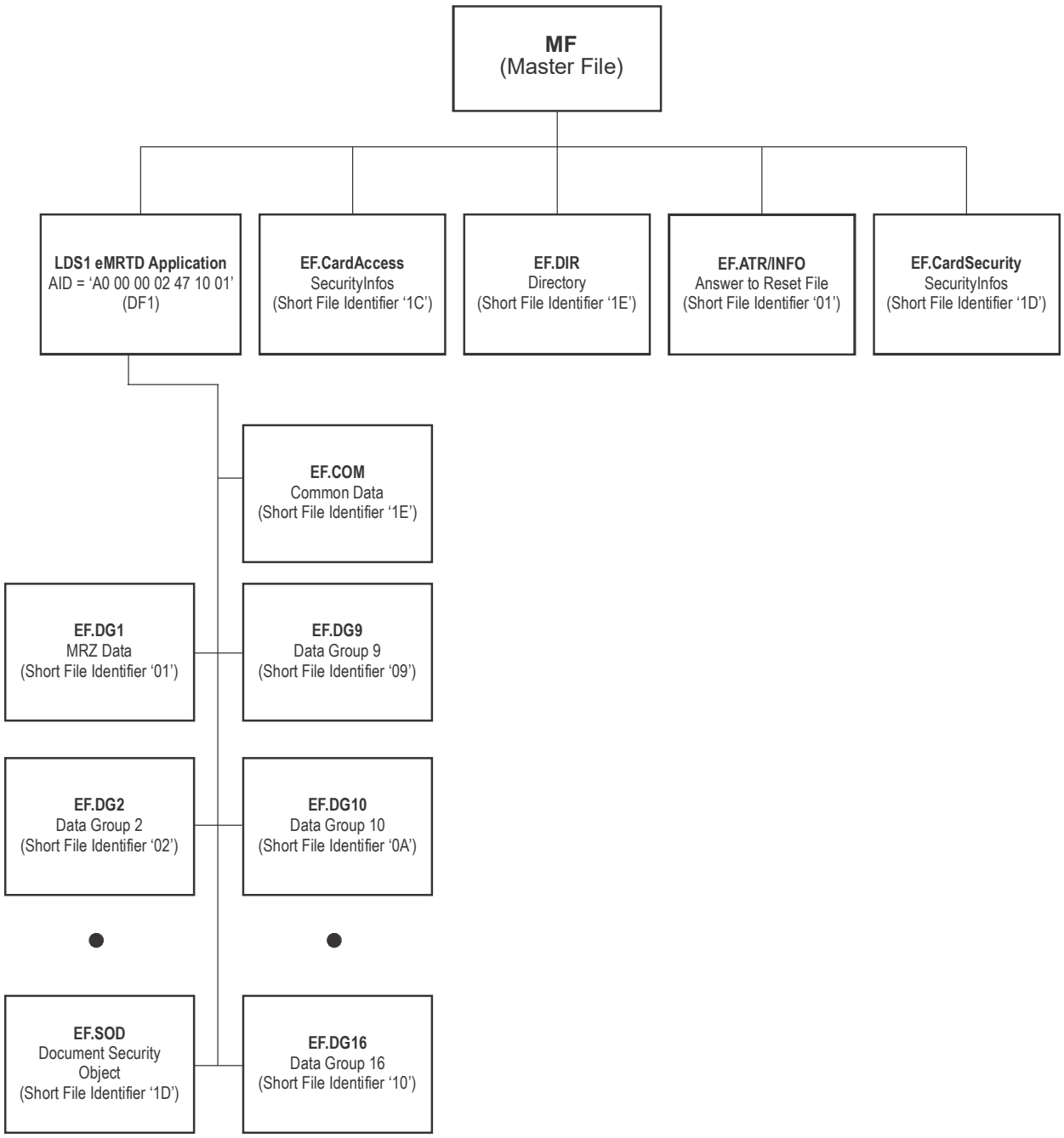


Figure 3. LDS1 eMRTD File Structure Summary

4.1 Application Selection — DF

The LDS1 eMRTD application SHALL be selected by use of the Application Identification (AID) as a reserved DF name. The AID SHALL consist of the Registered Application Identifier assigned by ISO according to [ISO/IEC 7816-5] and a Proprietary Application Identifier Extension (PIX) as specified within this document:

- The Registered Application Identifier is ’A000000247’;
- The issuer stored data application SHALL use PIX = ’1001’;
- The full AID of the LDS1 eMRTD application is ‘A0 00 00 02 47 10 01’.

The IC MUST reject the selection of an application if the extension for this application is absent.

4.2 Random Ordering Scheme

The Random Ordering Scheme allows Data Groups and Data Elements to be recorded following a random ordering which is consistent with the ability of the optional capacity expansion technology to allow direct retrieval of specific Data Elements even if they are recorded out of order. Variable length Data Elements are encoded as TLV data objects specified in ASN.1.

4.3 Random Access File Representation

The Random Access File Representation has been defined with the following considerations and assumptions.

To support a wide variety of implementations, the LDS includes a wide variety of optional Data Elements. These Data Elements are included to facilitate LDS1 eMRTD authentication, rightful holder authentication, and to expedite processing at document/person points.

The data structure must support:

- a limited or extensive set of Data Elements;
- multiple occurrences of specific Data Elements;
- continuing evolution of specific implementations.
- support at least one application data set;
- allow for other national specific applications;
- support optional Active Authentication of the document using a stored asymmetrical key pair;
- support rapid access of selected Data Elements to facilitate rapid document processing;
- immediate access to necessary Data Elements; and
- direct access to data templates and biometric data.

4.4 Grouping of Data Elements

Groupings of Data Elements added by issuing States or approved receiving organizations may or may not be present in an LDS. More than one recording of grouped Data Elements added by receiving States or approved receiving organizations can be present in the LDS.

The ability for a receiving State or approved receiving organization to add data to the LDS is not supported in this edition of Doc 9303.

The LDS is considered to be a single cohesive entity containing the number of groupings of Data Elements recorded in the optional capacity expansion technology at the time of machine reading.

The LDS has been designed with sufficient flexibility that it can be applied to all types of eMRTDs. Within the figures and tables which follow, some data items are only applicable to machine readable visas and to machine readable passports or require a different presentation in relation to these documents.

Within the LDS, logical groupings of related Data Elements have been established. These logical groupings are referred to as Data Groups.

4.5 Requirements of the Logical Data Structure

The contactless IC capacity expansion technology contained in an LDS1 eMRTD selected by an issuing State or organization must allow data to be accessible by receiving States.

ICAO has determined that the predefined, standardized Logical Data Structure (LDS) SHALL meet a number of mandatory requirements:

- ensure efficient and optimum facilitation of the rightful holder;
- ensure protection of details recorded in the optional capacity expansion technology;
- allow global interoperability of capacity expanded data based on the use of a single LDS common to all eMRTDs;
- address the diverse optional capacity expansion needs of issuing States and organizations;
- provide expansion capacity as user needs and available technology evolve;
- support a variety of data protection options;
- utilize existing international specifications to the maximum extent possible, in particular the emerging international specifications for globally interoperable biometrics.

4.5.1 Security

Only the issuing State or organization SHALL have write access to these Data Groups. Therefore, there are no interchange requirements and the methods to achieve write protection are not part of this specification. Once the chip has been locked (after personalization and before issuance) no LDS1 Application data can be written, modified, or deleted to/at/from the chip. After issuance a locked chip cannot be unlocked.

4.5.2 Authenticity and Integrity of Data

To allow confirmation of the authenticity and integrity of recorded details, an authenticity/integrity object is included. Each Data Group MUST be represented in this authenticity/integrity object, which is recorded within a separate EF (EF.SOD). Using the Common Biometric Exchange Formats Framework (CBEFF) structure utilized for Encoded Identification Feature Data Groups 2-4 and optional “additional biometric security” features defined in Doc 9303-12, identity confirmation details (e.g. biometric templates) MAY also be individually protected at the discretion of the issuing State or organization.

4.5.3 Ordering of LDS

The Random Ordering Scheme SHALL only be used for international interoperability.

4.5.4 Data Storage Capacity of the Contactless IC

The data storage capacity of the contactless IC is at the discretion of the issuing State but SHALL be a minimum of 32 kB. This minimum capacity is necessary to store the mandatory stored facial image the MRZ data, and the necessary elements for securing the data. The storage of additional facial, fingerprint and/or iris images may require a significant increase in data storage capacity. There is no maximum contactless IC data capacity specified.

In the event that a State’s PKI infrastructure is not available to sign LDS1 eMRTD data as part of personalization, and the issuance of the document(s) cannot be delayed, it is RECOMMENDED that the LDS1 eMRTD contactless IC be left blank and be locked. The LDS1 eMRTD SHOULD contain an appropriate endorsement on this. This is expected to be an exceptional circumstance.

4.5.5 Storage of Other Data

A State MAY use the storage capacity of the contactless IC in an eMRTD to expand the machine readable data capacity of the LDS1 eMRTD beyond that defined for global interoperability. This can be for such purposes as providing machine readable access to identity document information (e.g. birth certificate details), stored personal identity confirmation (biometrics) and/or document authenticity verification details.

4.5.6 International Standard for Encoding Biometrics

ISO/IEC 39794 succeeded [ISO/IEC 19794:2005] as international standard for encoding biometrics. The following transition time table has been defined:

- inspection systems MUST be able to handle ISO/IEC 39794 data by 1-1-2026 after a six-year preparation period starting on 1-1-2020;

- between 2026 and 2030, issuing states and organizations can use the data formats specified in ISO/IEC 19794-X:2005 or in ISO/IEC 39794-X during a four-year transition period. During this transition period, interoperability and conformity testing will be essential; and

- from 1-1-2030 on, passport issuers MUST use ISO/IEC 39794-X for encoding biometric data.

The ICAO Technical Report, ISO/IEC 39794-5 Application Profile for eMRTDs[1], provides guidance on the transition from [ISO/IEC 19794:2005] to ISO/IEC 39794.

Logical Data Structure of eMRTDs includes DG2 for face (mandatory), DG3 for fingerprint (optional) and DG4 for iris (optional). Each Data Group contains biometric data encoded in accordance with the international standards in order to keep international interoperability.

All of the above DGs (DG2, DG3 and DG4) MUST use the Biometric Information Template (BIT) group template with nested BITs (see Doc 9303-10). The nested BIT structure contains biometric data that can be encoded using one of two types of standards, ISO/IEC 19794 series first edition or ISO/IEC 39794 series.

Biometric data encoded in ISO/IEC 19794 series first edition is stored in the data object identified by tag ‘5F2E’. Biometric data encoded in ISO/IEC 39794 series is stored in the data object identified by tag ‘7F2E’.

Table 1: Tags for biometric data

Tag	Standard No.
5F2E	ISO/IEC 19794 series first edition
7F2E	ISO/IEC 39794 series

Biometric data encoded in the data object identified by tag ‘7F2E’ MUST use the data structure in the table below.

Table 2: Data Structure under DO’7F2E’

<table>
<tr><th>Tag</th><th>L</th><th colspan="5">Value</th></tr>
<tr><td>7F2E</td><td>Var.</td><td colspan="5">Biometric data template defined in ISO/IEC 7816-11.</td></tr>
<tr><td></td><td></td><td>Tag</td><td>L</td><td colspan="3">Value</td></tr>
<tr><td></td><td></td><td>A1</td><td>Var.</td><td colspan="3">Biometric data in standardized format (Constructed)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>Tag</td><td>L</td><td>Value</td></tr>
<tr><td></td><td></td><td></td><td></td><td>64, 65 or 66</td><td>Var</td><td>DO defined in the ISO/IEC 39794 series Table 3.</td></tr>
</table>

[1] Reference can be found on www.icao.int/security/fal/trip

Table 3: Tags for DOs defined in ISO/IEC 39794

Standard No.	Tag
ISO/IEC 39794-4	64
ISO/IEC 39794-5	65
ISO/IEC 39794-6	66

4.6 LDS1 eMRTD Elementary Files (EFs)

4.6.1 Header and Data Group Presence Information EF.COM (MANDATORY)

EF.COM is located in the LDS1 eMRTD application (Short File Identifier = '1E') and contains LDS version information, Unicode version information and a list of the Data Groups that are present for the application. The LDS1 eMRTD application MUST have only one file EF.COM that contains the common information for the application.

The Data Elements that may occur in this template are as follows:

Table 35. EF.COM Normative Tags

Tag	L	Value		
'60'	Var	Application level information		
		Tag	**L**	**Value**
		'5F01'	'04'	LDS Version number with format aabb, where aa defines the version of the LDS and bb defines the update level.
		'5F36'	'06'	Unicode Version number with format aabbcc, where aa defines the major version, bb defines the minor version and cc defines the release level.
		'5C'	Var	Tag list. List of all Data Groups present.

A Header and Data Group Presence Map SHALL be included. The header SHALL contain the following information which enables a receiving State or approved receiving organization to locate and decode the various Data Groups and Data Elements contained within the block of data recorded by the issuing State or organization.

It is RECOMMENDED that inspection systems that rely on the EF.COM be modified to use the SO_D described in the LDS version 1.8 as soon as possible.

4.6.1.1 LDS version number

The LDS version number defines the format version of the LDS. The exact format to be used for storing this value is defined in Section 4.6 of this document. Standardized format for an LDS Version Number is "aabb", where:

- "aa" = number (01-99) identifying the major version of the LDS (i.e. significant additions to the LDS);
- "bb" = number (01-99) identifying the minor version of the LDS.

4.6.1.2 UNICODE version number

The Unicode version number identifies the coding method used when recording alphabetic, numeric and special characters, including national characters. The exact format to be used for storing this value is defined in Section 4.7.1 of this document. The standardized format for a Unicode version number is “aabbcc”, where:

- “aa” = number identifying the major version of the Unicode specification (i.e. significant additions to the specification, published as a book);
- “bb” = number identifying the minor version of the Unicode specification (i.e. character additions or more significant normative changes, published as a technical report); and
- “cc” = number identifying the update version of the Unicode specification (i.e. any other changes to normative or important informative portions of the specification that could change programme behaviour. These changes are reflected in new Unicode character database files and an update page). For historical reasons, the numbering within each of the fields (i.e. a, b, c) is not necessarily consecutive.

The Universal Character Set (UCS) MUST comply with [ISO/IEC 10646].

4.6.2 Document Security Object EF.SOD (MANDATORY)

In addition to the LDS Data Groups, the contactless IC also contains a Document Security Object stored in EF.SOD. This object is digitally signed by the issuing State and contains hash values of the LDS contents.

Table 36. EF.SOD Tags

Tag	L	Value
‘77’	Var	Document Security Object

There are two versions of the Document Security Object EF.SOD which have been deployed. There is the legacy EF.SOD V0 which can be found in Appendix D and the RECOMMENDED EF.SOD V1 in this section. Only one EF.SOD is REQUIRED and allowed.

4.6.2.1 Document Security Object EF.SOD V1 LDS v1.8

The Document Security Object V1 for the LDS v1.8 has been extended with a signed attribute, containing the LDS and Unicode version information:

```
LDSSecurityObject ::= SEQUENCE {
   version LDSSecurityObjectVersion,
   hashAlgorithm DigestAlgorithmIdentifier,
   dataGroupHashValues SEQUENCE SIZE (2..ub-DataGroups) OF
DataGroupHash,
   ldsVersionInfo LDSVersionInfo OPTIONAL
   -- If present, version MUST be V1
}
LDSVersionInfo ::= SEQUENCE {
   ldsVersion PrintableString,
   unicodeVersion PrintableString }
```

4.6.2.2 SignedData Type for SO_D V1

The Document Security Object is implemented as a SignedData Type, as specified in [RFC 3369], Cryptographic Message Syntax (CMS), August 2002. All security objects MUST be produced in Distinguished Encoding Rule (DER) format to preserve the integrity of the signatures within them.

Note 1.— m = REQUIRED — the field SHALL be present.

Note 2.— x= do not use — the field SHOULD NOT be populated.

Note 3.— o = optional — the field MAY be present.

Note 4.— c = choice — the field content is a choice from alternatives.

Table 37. Signed Data Type for SO_D V1

Value		Comments
SignedData		
Version	m	Value = v3
digestAlgorithms	m	
encapContentInfo	m	
eContentType	m	id-icao-mrtd-security-ldsSecurityObject
eContent	m	The encoded contents of an ldsSecurityObject.
Certificates	m	States are REQUIRED to include the Document Signer Certificate (C_{DS}) which can be used to verify the signature in the signerInfos field.
Crls	x	It is recommended that States do not use this field.
signerInfos	m	It is recommended that States provide only one signerInfo within this field.
SignerInfo	m	
Version	m	The value of this field is dictated by the sid field. See RFC 3369 Doc 9303-12 for rules regarding this field.
Sid	m	
issuerandSerialNumber	c	It is recommended that States support this field over subjectKeyIdentifier.
subjectKeyIdentifier	c	
digestAlgorithm	m	The algorithm identifier of the algorithm used to produce the hash value over encapsulatedContent and SignedAttrs.
signedAttrs	m	Producing States may wish to include additional attributes for inclusion in the signature, however these do not have to be processed by receiving States except to verify the signature value.
signatureAlgorithm	m	The algorithm identifier of the algorithm used to produce the signature value and any associated parameters.
Signature	m	The result of the signature generation process.

Value		Comments
unsignedAttrs	o	Producing States may wish to use this field, but it is not recommended and receiving States may choose to ignore them.

4.6.2.3 ASN.1 Profile LDS Document Security Object for SO_D V1

```
LDSSecurityObjectV1 { joint-iso-itu-t(2) international(23) icao(136)
mrtd(1) security(1) ldsSecurityObject(1)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN

IMPORTS

-- Imports from RFC 3280 [PROFILE]
AlgorithmIdentifier FROM
PKIX1Explicit88 { iso(1) identified-organization(3) dod(6)
internet(1) security(5) mechanisms(5) pkix(7)
id-mod(0) id-pkix1-explicit(18) };

-- Constants

ub-DataGroups INTEGER ::= 16

-- Object Identifiers

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-mrtd-security-ldsSecurityObject  OBJECT  IDENTIFIER  ::=  {id-icao-  mrtd-
security 1}

-- LDS Security Object

LDSSecurityObjectVersion ::= INTEGER {v0(0), v1(1)
-- If LDSSecurityObjectVersion is V1, ldsVersionInfo MUST be present
}

DigestAlgorithmIdentifier ::= AlgorithmIdentifier

LDSSecurityObject ::= SEQUENCE {
   version LDSSecurityObjectVersion,
   hashAlgorithm DigestAlgorithmIdentifier,
   dataGroupHashValues SEQUENCE SIZE (2..ub-DataGroups) OF
   DataGroupHash,
   ldsVersionInfo LDSVersionInfo OPTIONAL
   -- If present, version MUST be V1
}
```

```
DataGroupHash ::= SEQUENCE {
   dataGroupNumber DataGroupNumber,
   dataGroupHashValue OCTET STRING }

DataGroupNumber ::= INTEGER {
   dataGroup1              (1),
   dataGroup2              (2),
   dataGroup3              (3),
   dataGroup4              (4),
   dataGroup5              (5),
   dataGroup6              (6),
   dataGroup7              (7),
   dataGroup8              (8),
   dataGroup9              (9),
   dataGroup10            (10),
   dataGroup11            (11),
   dataGroup12            (12),
   dataGroup13            (13),
   dataGroup14            (14),
   dataGroup15            (15),
   dataGroup16            (16)}

LDSVersionInfo ::= SEQUENCE {
   ldsVersion PrintableString
   unicodeVersion Printable String }
END
```

Note 1.— The field `dataGroupHashValue` contains the calculated hash over the complete contents of the Data Group EF, specified by `dataGroupNumber`.

Note 2.— `DigestAlgorithmIdentifiers` MUST omit `NULL` parameters, while the `SignatureAlgorithmIdentifier` (as defined in RFC 3447) MUST include `NULL` as the parameter if no parameters are present, even when using SHA2 Algorithms in accordance with RFC 5754. Inspection system MUST accept the field `DigestAlgorithmIdentifiers` with both conditions, i.e. absent parameters and `NULL` parameters.

4.7 Data Elements Forming Data Groups 1 through 16

Data Groups 1 (DG1) through 16 (DG16) individually consist of a number of mandatory, optional, and conditional Data Elements. The specified order of Data Elements within the Data Group SHALL be followed. Each Data Group SHALL be stored in one transparent EF. Addressing EFs SHALL be by Short EF Identifier as shown in Table 38. The EFs SHALL have file names for these files that SHALL be according to the number n, EF.DGn, where n is the Data Group number.

Table 38. Mandatory and Optional Data Elements that Combine to Form the Structure of Data Groups 1 (DG1) through 16 (DG16)

Data Group	EF Name	Short EF Identifier	EF Identifier	Tag
Common	EF.COM	‘1E’	‘01 1E’	‘60’
DG1	EF.DG1	‘01’	‘01 01’	‘61’
DG2	EF.DG2	‘02’	‘01 02’	‘75’
DG3	EF.DG3	‘03’	‘01 03’	‘63’
DG4	EF.DG4	‘04’	‘01 04’	‘76’
DG5	EF.DG5	‘05’	‘01 05’	‘65’
DG6	EF.DG6	‘06’	‘01 06’	‘66’
DG7	EF.DG7	‘07’	‘01 07’	‘67’
DG8	EF.DG8	‘08’	‘01 08’	‘68’
DG9	EF.DG9	‘09’	‘01 09’	‘69’
DG10	EF.DG10	‘0A’	‘01 0A’	‘6A’
DG11	EF.DG11	‘0B’	‘01 0B’	‘6B’
DG12	EF.DG12	‘0C’	‘01 0C’	‘6C’
DG13	EF.DG13	‘0D’	‘01 0D’	‘6D’
DG14	EF.DG14	‘0E’	‘01 0E’	‘6E’
DG15	EF.DG15	‘0F’	‘01 0F’	‘6F’
DG16	EF.DG16	‘10’	‘01 10’	‘70’
Document Security Object	EF.SOD	‘1D’	‘01 1D’	‘77’
Common	EF.CARDACCESS	‘1C’	‘01 1C’	
Common	EF.ATR/INFO	‘01’	‘2F 01’	
Common	EF.CardSecurity	‘1D’	‘01 1D’	

4.7.1 DATA GROUP 1 — Machine Readable Zone Information (MANDATORY)

The Data Elements of Data Group 1 (DG1) are intended to reflect the entire contents of the MRZ whether it contains actual data or filler characters. Details on the implementation of the MRZ are dependent on the type of LDS1 eMRTD (TD1, TD2 or TD3 formats).

This Data Element contains the REQUIRED machine readable zone (MRZ) information for the document in template ‘61’. The template contains one data object, the MRZ in data object ‘5F1F’. The MRZ data object is a composite Data Element, identical to the OCR-B MRZ information printed on the document.

Table 39. Data Group 1 Tags

Tag	L	Value		
'61'	Var			
		Tag	**L**	**Value**
		'5F1F'	Var	The MRZ data object as a composite Data Element. (REQUIRED) (The Data Element contains all mandatory fields from Document Type through to Composite check digit.)

4.7.1.1 DATA GROUP 1 — EF.DG1 Data Elements for TD1 Size LDS1 eMRTD

This section describes the Data Elements that may be present in Data Group 1 (DG1). Storage, ordering and coding requirements of DG1 are intended to be exactly the same as found in the printed MRZ and described in Doc 9303, Parts 3 and 5. Data Elements and their format within each Data Group area for TD1 SHALL be as in the following table:

Note.— A = Alpha character [A-Z], N = Numeric character [0-9], S = Special character ['<'], F = fixed-length field.

Table 40. Data Elements for TD1 Format

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding
01	M	Document code	2	F	A,S
02	M	Issuing State or organization	3	F	A,S
03	M	Document number (Nine most significant characters)	9	F	A,N,S
04	M	Check digit — Document number or filler character (<) indicating document number exceeds nine characters	1	F	N,S
05	M	Optional data and/or in the case of a Document Number exceeding nine characters, least significant characters of document number plus document number check digit plus filler character	15	F	A,N,S
06	M	Date of birth	6	F	N,S

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding
07	M	Check digit — Date of birth	1	F	N
08	M	Sex	1	F	A,S
09	M	Date of Expiry	6	F	N
10	M	Check digit — Date of expiry	1	F	N
11	M	Nationality	3	F	A,S
12	M	Optional data	11	F	A,N,S
13	M	Composite check digit	1	F	N
14	M	Name of holder	30	F	A,N,S

4.7.1.2 DATA GROUP 1 — EF.DG1 Data Elements for TD2 Size eMRTD

This section describes the Data Elements that may be present in Data Group 1 (DG1). Storage, ordering and coding requirements of DG1 are intended to be exactly the same as found in the printed MRZ and described in Doc 9303, Parts 3 and 6. Data Elements and their format within each Data Group area for TD2 SHALL be as in the following table:

Note.— A = Alpha character [A-Z], N = Numeric character [0-9], S = Special character ['<'], F = fixed-length field.

Table 41. Data Elements for TD2 Format

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding
01	M	Document code	2	F	A,S
02	M	Issuing State or organization	3	F	A,S
03	M	Name of holder	31	F	A,N,S
04	M	Document number (Nine principal characters)	9	F	A,N,S
05	M	Check digit	1	F	N,S
06	M	Nationality	3	F	A,S
07	M	Date of birth	6	F	N,S
08	M	Check digit	1	F	N
09	M	Sex	1	F	A,S
10	M	Date of expiry	6	F	N
11	M	Check digit	1	F	N
12	M	Optional data plus filler character	7	F	A,N,S
13	M	Composite Check Digit - MRZ line 2	1	F	N

4.7.1.3 DATA GROUP 1 — EF.DG1 Data Elements for TD3 Size LDS1 eMRTD

This section describes the Data Elements that may be present in Data Group 1 (DG1). Storage, ordering and coding requirements of DG1 are intended to be exactly the same as found in the printed MRZ and described in Doc 9303, Parts 3 and 4. Data Elements and their format within each Data Group area for TD3 SHALL be as in the following table:

Note.— A = Alpha character [A-Z], N = Numeric character [0-9], S = Special character ['<'], F = fixed-length field.

Table 42. Data Elements for TD3 Format

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding
01	M	Document code	2	F	A,S
02	M	Issuing State or organization	3	F	A,S
03	M	Name of holder	39	F	A,S
04	M	Document number	9	F	A,N,S
05	M	Check digit — Document number	1	F	N,S
06	M	Nationality	3	F	A,S
07	M	Date of birth	6	F	N,S
08	M	Check digit — Date of birth	1	F	N
09	M	Sex	1	F	A,S
10	M	Date of expiry	6	F	N
11	M	Check digit — Date of expiry or valid until date	1	F	N
12	M	Optional data	14	F	A,N,S
13	M	Check digit	1	F	N
14	M	Composite check digit	1	F	N

4.7.2 DATA GROUP 2 — Encoded Identification Features — Face (MANDATORY)

Data Group 2 (DG2) represents the globally interoperable biometric for machine assisted identity confirmation with machine readable travel documents, which SHALL be an image of the face of the holder as an input to a face recognition system. If there is more than one recording, the most recent internationally interoperable encoding SHALL be the first entry.

Table 43. Data Group 2 Tags

Tag	L	Value
‘75’	Var	See Biometric encoding of EF.DG2

4.7.2.1 Biometric encoding of EF.DG2

DG2 MUST use the Biometric Information Template (BIT) group template with nested BITs specified in [ISO/IEC 7816-11], which allows the possibility to store multiple biometric templates and is in harmony with the CBEFF. The biometric sub-header defines the type of biometric that is present and the specific biometric feature. The nested option of ISO/IEC [7816-11] is always to be used, even for encodings of a single biometric template. The latter case is indicated by numbering with n = 1.

Each nested template has the following structure:

Table 44. Data Group 2 — Biometric Encoding Tags

Tag	**L**	**Value**				
'7F61'	Var	Biometric Information Template Group Template				
		Tag	**L**	**Value**		
		'02'	'01'	Integer — Number of instances of this type of biometric		
		'7F60'	Var	1st Biometric Information Template		
			Tag	**L**		
			'A1'	Var	Biometric Header Template (BHT)	
				Tag	**L**	**Value**
				'80'	'02'	ICAO header version 0101 (Optional) — Version of the CBEFF patron header format
				'81'	'01-03'	Biometric type (Optional)
				'82'	'01'	Biometric sub-type Optional for DG2
				'83'	'07'	Creation date and time (Optional)
				'85'	'08'	Validity period (from through) (Optional)
				'86'	'04'	Creator of the biometric reference data (PID) (Optional)
				'87'	'02'	Format owner (REQUIRED)
				'88'	'02'	Format type (REQUIRED)
			'5F2E' or '7F2E'	Var	Biometric data (encoded according to Format Owner) also called the biometric data block (BDB). See 4.5.6.	

The default OID of CBEFF is used. The OID data object (Tag '06') just under Biometric Information Template (BIT, Tag '7F60') specified in [ISO/IEC 7816-11] is not included in this structure. Likewise, the Tag allocation authority is not specified in the structure.

To facilitate interoperability, the first biometric recorded in each Data Group SHALL be encoded as per [ISO/IEC19794-5].

Note.— ISO/IEC 39794 will succeed ISO/IEC 19794:2005 as the international standard for encoding biometrics. See Section 4.5.6.

4.7.2.2 DATA GROUP 2 — EF.DG2 Data Elements

This section describes the Data Elements that may be present in Data Group 2 (DG2): Data Elements and their format within each Data Group area SHALL be as in the following tables:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 45. Data Elements for DG2

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M	Number of face biometric encodings recorded	1	F	N	1 to 9 identifying number of unique encodings of data on the face.
02	M	Header		Var	A,N	Data Element may recur as defined by Data element 01.
03	M	Face biometric data encoding(s)		Var	B	Data Element may recur as defined by Data element 01.

4.7.3 DATA GROUP 3 — Additional Identification Feature — Finger(s) (OPTIONAL)

ICAO recognizes that Member States may elect to use fingerprint recognition as additional biometric technologies in support of machine assisted identity confirmation, which SHALL be encoded as Data Group 3 (DG3).

Table 46. Data Group 3 Tags

Tag	L	Value
'63'	Var	See Biometric encoding of EF.DG3

4.7.3.1 Biometric Encoding of EF.DG3

DG3 MUST use the Biometric Information Template (BIT) group template with nested BITs specified in [ISO/IEC 7816-11], which allows the possibility to store multiple biometric templates and is in harmony with the CBEFF. The biometric sub-header defines the type of biometric that is present and the specific biometric feature. The nested option of [ISO/IEC 7816-11] MUST be used, even for encodings of a single biometric template. The latter case is indicated by numbering with n = 1. The number of instances in DG3 can be '0...n'.

Each nested template has the following structure:

Table 47. Data Group 3 Nested Tags

Tag	L	Value				
‘7F61’	Var	Biometric Information Template Group Template				
		Tag	**L**	**Value**		
		‘02’	‘01’	Integer — Number of instances of this type of biometric		
		‘7F60’	Var	1st Biometric Information Template		
			Tag	**L**		
			‘A1’	Var	Biometric Header Template (BHT)	
				Tag	**L**	**Value**
				‘80’	‘02’	ICAO header version ‘0101’ (Optional) — Version of the CBEFF patron header format
				‘81’	‘01-03’	Biometric type (Optional)
				‘82’	‘01’	Biometric sub-type REQUIRED for DG3
				‘83’	‘07’	Creation date and time (Optional)
				‘85’	‘08’	Validity period (from through) (Optional)
				‘86’	‘04’	Creator of the biometric reference data (PID) (Optional)
				‘87’	‘02’	Format owner (REQUIRED)
				‘88’	‘02’	Format type (REQUIRED)
			‘5F2E’ or ‘7F2E’	Var	Biometric data (encoded according to Format Owner) also called the biometric data block (BDB). See 4.5.6.	
		Tag	**L**			
		‘7F60’	Var	2nd Biometric Information Template		
			Tag	**L**		
			‘A1’	Var	Biometric Header Template (BHT)	
				Tag	**L**	**Value**
				‘80’	‘02’	ICAO header version ‘0101’ (Optional) — Version of the CBEFF patron header format
				‘81’	‘01-03’	Biometric type (Optional)
				‘82’	‘01’	Biometric sub-type REQUIRED for DG3
				‘83’	‘07’	Creation date and time (Optional)
				‘85’	‘08’	Validity period (from through) (Optional)
				‘86’	‘04’	Creator of the biometric reference data (PID)

Tag	L	Value				
						(Optional)
				‘87’	‘02’	Format owner (REQUIRED)
				‘88’	‘02’	Format type (REQUIRED)
			‘5F2E’ or ‘7F2E’	Var	Biometric data (encoded according to Format Owner) also called the biometric data block (BDB). See 4.5.6.	

The default OID of CBEFF is used. The OID data object (Tag ‘06’) just under Biometric Information Template (BIT, Tag ‘7F60’) specified in [ISO/IEC 7816-11] is not included in this structure. Likewise the Tag allocation Authority is not specified in the structure.

To facilitate interoperability, the first biometric recorded in each Data Group SHALL be encoded as per [ISO/IEC19794-4].

Note.— ISO/IEC 39794 will succeed ISO/IEC 19794:2005 as the international standard for encoding biometrics. See Section 4.5.6.

4.7.3.2 DATA GROUP 3 — EF.DG3 Data Elements

This section describes the Data Elements that may be present in Data Group 3 (DG3). Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character [‘<’], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 48. Data Elements for DG3

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M (If encoded finger(s) feature recorded)	Number of finger(s) biometric encodings recorded	1	F	N	0 to n identifying number of unique encodings of data on the finger(s).
02	M (If encoded finger(s) feature recorded)	Header		Var	B	Data Element may recur as defined by Data element 01.
03	M (If encoded finger(s) feature recorded)	Finger biometric data encoding(s)		Var	B	Data Element may recur as defined by Data element 01.

4.7.3.2.1 Biometric sub-type encoding

The biometric header template Tags and their assigned values are the minimum each implementation SHALL support as shown in the following table. Each single biometric information template has the following structure:

Table 49. Encoding of Sub-Features Scheme for the Encoding of Sub-Features: CBEFF

b8	b7	b6	b5	b4	b3	b2	b1	Biometric Sub-type
0	0	0	0	0	0	0	0	No information given
						0	1	Right
						1	0	Left
			0	0	0			No meaning
			0	0	1			Thumb
			0	1	0			Pointer
			0	1	1			Middle
			1	0	0			Ring
			1	0	1			Little
X	X	X						Reserved for future use

4.7.3.2.2 Encoding of Zero Instance

States not issuing LDS1 eMRTDs with fingerprints SHOULD NOT populate DG3. DG3 of this structure has the drawback that it will result in a static DG3 hash in the SO_D for all LDS1 eMRTDs where the biometric features are not present and populated at the time of LDS1 eMRTD issuance, but the DG3 is declared. For interoperability purposes States supporting fingerprints in their LDS1 eMRTDs MUST store an empty Biometric Information Group Template in cases where no fingerprints are available at the time of LDS1 eMRTD issuance. The template counter denotes a value of ‘00’ in this case.

It is RECOMMENDED to add Tag ‘53’ with issuer defined content (e.g. a random number).

Table 50. Encoding Zero Instances

Tag	**L**	**Value**				
‘63’	Var	LDS element				
		Tag	**L**	**Value**		
		‘7F 61’	‘03’	Biometric Information Group Template		
			‘02’	‘01’	‘00’	Defines that there are no Biometric Information Templates stored in this Data Group.
		‘53’	Var	Issuer defined content (e.g. a random number).		

4.7.3.2.3 Encoding of One Instance

In cases where only one fingerprint is available, the single instance MUST be encoded in the following manner (example for DG3 – fingerprint):

Table 51. Encoding One Instance

Tag	**L**	**Value**						
‘63’	Var	LDS element where aa is the total length of the entire LDS data content						
		Tag	**L**	**Value**				
		‘7F 61’	Var	Biometric Information Group Template.				
			‘02’	‘01’	‘01’	Defines the total number of fingerprints stored as Biometric Information Templates that follow.		
			‘7F 60’	Var	First biometric information template where cc is the total length of the entire BIT			
				‘A1’	Var	Biometric Header Template.		
					‘81’	‘01’	‘08’	Biometric type “Fingerprint”
					‘82’	‘01’	‘0A’	Biometric sub-type “left pointer finger”
					‘87’	‘02’	‘01 01’	Format Owner JTC 1 SC 37
					‘88’	‘02’	‘00 07’	Format Type [ISO/IEC 19794-4]
					Note that the BHT may contain additional optional elements. Of course, this fingerprint can either be a left or right finger depending on the available image.			
				‘5F 2E’	Var	Biometric Data. The Biometric Data Block MUST contain exactly one fingerprint image.		

Note.— ISO/IEC 39794 will succeed ISO/IEC 19794:2005 as international standard for encoding biometrics. See Section 4.5.6.

4.7.3.2.4 Encoding of More than One Instance

To achieve interoperability each feature MUST be stored in an individual Biometric Information Template. The feature position MUST be specified within the CBEFF biometric sub-type if this information is available. The following table contains a worked example for the CBEFF encoding of an interoperable DG3 element with two fingerprint images.

Table 52. Encoding Greater than One Instance

Tag	**L**	**Value**						
'63'	Var	LDS element where *aa* is the total length of the entire LDS data content						
		Tag	**L**	**Value**				
		'7F 61'	Var	Biometric Information Template Group Template.				
			'02'	'01'	'02'	Defines the total number of fingerprints stored as Biometric Information Templates that follow.		
			'7F 60'	Var	First biometric information template.			
				'A1'	Var	Biometric Header Template.		
					'81'	'01'	'08'	Biometric type "Fingerprint"
					'82'	'01'	'0A'	Biometric sub-type "left pointer finger"
					'87'	'02'	'01 01'	Format Owner JTC 1 SC 37
					'88'	'02'	'00 07'	Format Type [ISO/IEC 19794-4]
					Note that the BHT may contain additional optional elements. It is also possible that the order of fingerprints (left/right) is different.			
				'5F 2E'	Var	Biometric Data Block. The Biometric Data Block MUST contain exactly one fingerprint image.		
			'7F 60'	Var	Second biometric information template.			
				'A1'	Var	Biometric Header Template.		
					'81'	'01'	'08'	Biometric type "Fingerprint"
					'82'	'01'	'09'	Biometric sub-type "right pointer finger"
					'87'	'02'	'01 01'	Format Owner JTC 1 SC 37
					'88'	'02'	'00 07'	Format Type [ISO/IEC 19794-4]
					Note that the BHT may contain additional optional elements. It is also possible that the order of fingerprints (left/right) is different.			
				'5F 2E'	Var	Biometric Data Block. The Biometric Data Block MUST contain exactly one fingerprint image.		

Note.— ISO/IEC 39794 will succeed ISO/IEC 19794:2005 as the international standard for encoding biometrics. See Section 4.5.6.

4.7.4 DATA GROUP 4 — Additional Identification Feature — Iris(es) (OPTIONAL)

ICAO recognizes that member States may elect to use iris recognition as additional biometric technologies in support of machine assisted identity confirmation, which SHALL be encoded as Data Group 4 (DG4).

Table 53. Data Group 4 Tags

Tag	L	Value
'76'	Var	See Biometric encoding of EF.DG4

4.7.4.1 Biometric Encoding of EF.DG4

DG4 MUST use the Biometric Information Template (BIT) group template with nested BITs specified in [ISO/IEC 7816-11], which allows the possibility to store multiple biometric templates and is in harmony with the CBEFF. The biometric sub-header defines the type of biometric that is present and the specific biometric feature. The nested option of ISO/IEC [7816-11] MUST be used, even for encodings of a single biometric template. The latter case is indicated by numbering with n = 1. The number of instances in DG4 can be '0...n'.

Each nested template has the following structure:

Table 54. Data Group 4 Nested Tags

Tag	L	Value				
'7F61'	Var	Biometric Information Template Group Template				
		Tag	**L**	**Value**		
		'02'	'1'	Integer — Number of instances of this type of biometric		
		'7F60'	Var	1st Biometric Information Template		
			Tag	**L**	**Value**	
			'A1'	Var	Biometric Header Template (BHT)	
				Tag	**L**	**Value**
				'80'	'02'	ICAO header version '0101' (Optional) — Version of the CBEFF patron header format
				'81'	'01-03'	Biometric type (Optional)
				'82'	'01'	Biometric sub-type, REQUIRED for DG4
				'83'	'07'	Creation date and time (Optional)
				'85'	'08'	Validity period (from through) (Optional)
				'86'	'04'	Creator of the biometric reference data (PID) (Optional)
				'87'	'02'	Format owner (REQUIRED)
				'88'	'02'	Format type (REQUIRED)

<table>
<tr><th>Tag</th><th>L</th><th colspan="5">Value</th></tr>
<tr><td></td><td></td><td></td><td>‘5F2E’ or ‘7F2E’</td><td>Var</td><td colspan="2">Biometric data (encoded according to Format Owner) also called the biometric data block (BDB). See 4.5.6.</td></tr>
<tr><td></td><td></td><td>Tag</td><td>L</td><td colspan="3">Value</td></tr>
<tr><td></td><td></td><td>‘7F60’</td><td>Var</td><td colspan="3">2nd Biometric Information Template</td></tr>
<tr><td></td><td></td><td></td><td>Tag</td><td>L</td><td colspan="2">Value</td></tr>
<tr><td></td><td></td><td></td><td>‘A1’</td><td>Var</td><td colspan="2">Biometric Header Template (BHT)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>Tag</td><td>L</td><td>Value</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘80’</td><td>‘02’</td><td>ICAO header version ‘0101’ (Optional) — Version of the CBEFF patron header format</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘81’</td><td>‘01-03’</td><td>Biometric type (Optional)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘82’</td><td>‘01’</td><td>Biometric sub-type REQUIRED for DG4</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘83’</td><td>‘07’</td><td>Creation date and time (Optional)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘85’</td><td>‘08’</td><td>Validity period (from through) (Optional)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘86’</td><td>‘04’</td><td>Creator of the biometric reference data (PID) (Optional)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘87’</td><td>‘02’</td><td>Format owner (REQUIRED)</td></tr>
<tr><td></td><td></td><td></td><td></td><td>‘88’</td><td>‘02’</td><td>Format type (REQUIRED)</td></tr>
<tr><td></td><td></td><td></td><td>‘5F2E’ or ‘7F2E’</td><td>Var</td><td colspan="2">Biometric data (encoded according to Format Owner) also called the biometric data block (BDB). See 4.5.6.</td></tr>
</table>

The default OID of CBEFF is used. The OID data object (Tag ‘06’) just under Biometric Information Template (BIT, Tag ‘7F60’) specified in [ISO/IEC 7816-11] is not included in this structure. Likewise the Tag allocation authority is not specified in the structure.

To facilitate interoperability, the first biometric recorded in each Data Group SHALL be encoded as per [ISO/IEC19794-6].

Note.— ISO/IEC 39794 will succeed ISO/IEC 19794:2005 as the international standard for encoding biometrics. See Section 4.5.6.

4.7.4.2 DATA GROUP 4 — EF.DG4 Data Elements

This section describes the Data Elements that may be present in Data Group (DG4). Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character [‘<’], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 55. Data Elements for DG4

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M, if encoded eye(s) feature included	Number of eye biometric encodings recorded	1	F	N	1 to 9 identifying number of unique encodings of data on the eye(s).
02	M, if encoded eye(s) feature included	Header		Var	B	Data Element may recur as defined by Data element 01.
03	M, if encoded eye(s) feature included	Eye biometric data encoding(s)		Var	B	Data Element may recur as defined by Data element 01.

4.7.4.2.1 Biometric Sub-Type Encoding

The biometric header template Tags and their assigned values are the minimum each implementation SHALL support as shown in the following table. Each single biometric information template has the following structure:

Table 56. Encoding of sub-features scheme for the encoding of sub-features: CBEFF

b8	b7	b6	b5	b4	b3	b2	b1	Biometric Sub-type
0	0	0	0	0	0	0	0	No information given
						0	1	Right
						1	0	Left
			0	0	0			Reserved for future use
			0	0	1			Reserved for future use
			0	1	0			Reserved for future use
			0	1	1			Reserved for future use
			1	0	0			Reserved for future use
			1	0	1			Reserved for future use
X	X	X						Reserved for future use

4.7.4.2.2 Encoding of Zero Instance

States not issuing LDS1 eMRTDs with irises SHOULD NOT populate DG4. DG4 of this structure has the drawback that it will result in a static DG4 hash in the SO_D for all LDS1 eMRTDs where the biometric features are not present and populated at the time of LDS1 eMRTD issuance but the DG4 is declared. For interoperability purposes States supporting irises in their LDS1 eMRTDs MUST store an empty Biometric Information Group Template in cases where no irises are available at the time of LDS1 eMRTD issuance. The template counter denotes a value of '00' in this case.

It is RECOMMENDED to add Tag '53' with issuer defined content (e.g. a random number).

Table 57. Encoding Zero Instances

Tag	L	Value				
'76'	Var	LDS element				
		Tag	**L**	**Value**		
		'7F 61'	'03'	Biometric Information Template Group Template		
			'02'	'01'	'00'	Defines that there are no Biometric Information Templates stored in this Data Group.
		'53'	Var	Issuer defined content (e.g. a random number).		

4.7.4.2.3 Encoding of One Instance

In cases where only one iris is available, the single instance MUST be encoded.

4.7.4.2.4 Encoding of More than One Instance

To achieve interoperability each feature MUST be stored in an individual Biometric Information Template. The feature position MUST be specified within the CBEFF biometric sub-type if this information is available.

4.7.5 DATA GROUP 5 — Displayed Portrait (OPTIONAL)

Data Elements assigned to Data Group 5 (DG5) SHALL be as follows:

Table 58. Data Group 5 Tags

Tag	L	Value		
'65'	Var			
		Tag	**L**	**Value**
		'02'	Var	Number of instances of this type of displayed image (REQUIRED in first template. Not used in succeeding templates.)
		'5F40'	Var	Displayed portrait

The following format owners are recognized for the specified type of displayed image.

Table 59. DG5 Formats

Displayed Image	Format Owner
Displayed Facial Image	[ISO/IEC 10918], JFIF option

4.7.5.1 DATA GROUP 5 — EF.DG5 Data Elements (Optional)

This section describes the Data Elements that may be present in Data Group 5 (DG5). Data Elements and their format within DG5 SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 60. Data Elements for DG5

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M (If displayed portrait recorded)	Number of displayed portraits recorded	1	F	N	1 to 9 identifying number of unique recordings of displayed portrait.
02	M (If displayed portrait recorded)	Displayed portrait representation(s)		Var	A,N	Data Element may recur as defined by Data element 01.
03	M (If displayed portrait recorded)	Number of bytes in representation of displayed portrait	5	F	N	00001 to X9, identifying number of bytes in representation of displayed portrait immediately following.
04	M (If displayed portrait recorded)	Representation of displayed portrait		Var	B	Formatted as per [ISO/IEC 10918-1] or [ISO/IEC 15444].

Note.— Data Element 02 SHALL be encoded as defined in [ISO/IEC 10918], using the JFIF option or [ISO/IEC 15444] using JPEG 2000 image coding system.

4.7.6 DATA GROUP 6 — Reserved for Future Use

Data Elements assigned to Data Group 6 (DG6) SHALL be as follows:

Table 61. Data Group 6 Tags

Tag	L	Value
‘66’	Var	

4.7.6.1 DATA GROUP 6 — EF.DG6 Data Elements

The data elements for DG6 are reserved for future use.

4.7.7 DATA GROUP 7 — Displayed Signature or Usual Mark (OPTIONAL)

Data Elements assigned to Data Group 7 (DG7) SHALL be as follows:

Table 62. Data Group 7 Tags

Tag	L	Value		
‘67’	Var			
		Tag	**L**	**Value**
		‘02’	Var	Number of instances of this type of displayed image (REQUIRED in first template. Not used in succeeding templates.)
		‘5F43’	Var	Displayed Signature

The following format owners are recognized for the specified type of displayed image:

Table 63. DG7 Formats

Displayed Image	Format Owner
Displayed Signature/usual mark	[ISO/IEC 10918], JFIF option

4.7.7.1 DATA GROUP 7 — EF.DG7 Data Elements (OPTIONAL)

This section describes the Data Elements that may be present in Data Group 7 (DG7). Data Elements and their format within each DG7 SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character [‘<’], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 64. Data Elements for DG7

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M (If displayed signature or usual mark recorded)	Number of displayed signature or usual marks	1	F	N	1 to 9 identifying number of unique recordings of displayed signature or usual mark.
02	M (If displayed signature or usual mark recorded)	Displayed signature or usual mark representation		Var	B	Data Element may recur as defined by DE 01. Formatted as per [ISO/IEC 10918-1] or [ISO/IEC 15444].

Note.— Data Element 02 SHALL be encoded as defined in [ISO/IEC 10918], using the JFIF option, or [ISO/IEC 15444] using JPEG 2000 image coding system.

4.7.8 DATA GROUP 8 — Data Feature(s) (OPTIONAL)

This Data Group has yet to be defined. Until then, they are available for temporary proprietary usage. This Data Element could use a structure similar to that for biometric templates, machine assisted security feature verification and encoded detail(s). Data Elements combining to form Data Group 8 (DG8) SHALL be as follows:

Table 65. Data Group 8 Tags

Tag	L	Value		
'68'	Var	To Be Defined		
		Tag	**L**	**Value**
		'02'	'1'	Integer — Number of instances of this type of template (REQUIRED in first template. Not used in succeeding templates.)
			Var	Header Template. Details to be defined.

4.7.8.1 DATA GROUP 8 — EF.DG8 Data Elements

This section describes the Data Elements that may be present in DG8. Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 66. Data Elements for DG8

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M (If this encoded feature is used)	Number of data feature(s)	1	F	N	1 to 9, identifying number of unique encodings of data feature(s) (embraces Data element 02 through 03).
02	M (If this encoded feature is used)	Header (to be defined)	1			Header details to be defined.
03	M (If this encoded feature is used)	Data feature(s) data	999 Max	Var	A,N,S, U,B	Format defined at the discretion of issuing State or organization.

4.7.9 DATA GROUP 9 — Structure Feature(s) (OPTIONAL)

This Data Group has yet to be defined. Until then, it is available for temporary proprietary use. These Data Elements could use a structure similar to that for biometric templates. Data Elements combining to form Data Group 9 (DG9) SHALL be as follows:

Table 67. Data Group 9 Tags

Tag	L	Value		
'69'	Var	To Be Defined		
		Tag	**L**	**Value**
		'02'	'01'	Integer — Number of instances of this type of template (REQUIRED in first template. Not used in succeeding templates.)
			X	Header Template. Details to be defined.

4.7.9.1 DATA GROUP 9 — EF.DG9 Data Elements

DG9 Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 68. Data Elements for DG9

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M (If this encoded feature is used)	Number of structure feature(s)	1	F	N	1 to 9, identifying number of unique encodings of structure feature(s) (embraces Data element 02 through 03).
02	M (If this encoded feature is used)	Header (to be defined)			N	Header details to be defined
03	M (If this encoded feature is used)	Structure feature(s) data		Var	B	

4.7.10 DATA GROUP 10 — Substance Feature(s) (OPTIONAL)

This Data Group has yet to be defined. Until then, it is available for temporary proprietary usage. These Data Elements could use a structure similar to that for biometric templates. Data Elements combining to form Data Group 10 (DG10) SHALL be as follows:

Table 69. Data Group 10 Tags

Tag	L	Value		
'6A'	Var			
		Tag	**L**	**Value**
		'02'	'01'	Integer — Number of instances of this type of template (REQUIRED in first template. Not used in succeeding templates.)
			Var	To Be Defined.

4.7.10.1 DATA GROUP 10 — EF.DG10 Data Elements

This section describes the Data Elements that may be present in DG10. Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 70. Data Elements for DG10

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M (If this encoded feature is used)	Number of substance feature(s) recorded	1	F	N	1 to 9, identifying number of unique encodings of substance feature(s) (embraces Data elements 02 and 03).
02	M (If this encoded feature is used)	Header (to be defined)	TBD	TBD	N	Details to be defined.
03	M (If this encoded feature is used)	Substance feature(s) data	999 Max	Var	A,N,S, U,B	Format defined at the discretion of issuing State or organization.

4.7.11 DATA GROUP 11 — Additional Personal Detail(s) (OPTIONAL)

This Data Group is used for additional details about the document holder. Since all of the Data Elements within this group are optional, a Tag list is used to define those present. Data Elements combining to form Data Group 11 (DG11) SHALL be as follows:

Note.— This template may contain non-Latin characters.

Table 71. Data Group 11 Tags

Tag	**L**	**Value**				
'6B'	Var					
		Tag	**L**	**Value**		
		'5C'	Var			Tag list with list of Data Elements in the template.
		'5F0E'	Var			Full name of document holder in national characters. Encoded per Doc 9303 rules.
		'A0'	Var			Content-specific class
				Tag	**L**	**Value**
				'02'	'01'	Number of other names
				'5F0F'	Var	Other name formatted per Doc 9303. The data object repeats as many times as indicated in number of other names (data object with Tag'02')
		Tag	**L**	**Value**		
		'5F10'	Var			Personal number
		'5F2B'	'08'			Full date of birth yyyymmdd
		'5F11'	Var			Place of birth. Fields separated by '<'
		'5F42'	Var			Permanent address. Fields separated by '<'
		'5F12'	Var			Telephone
		'5F13'	Var			Profession
		'5F14'	Var			Title
		'5F15'	Var			Personal summary
		'5F16'	Var			Proof of citizenship. Compressed image per [ISO/IEC 10918]
		'5F17'	Var			Other valid TD numbers. Separated by '<'
		'5F18'	Var			Custody information

4.7.11.1 DATA GROUP 11 — EF.DG11 Data Elements

This section describes the Data Elements that may be present in DG11. Data Elements and their format within each Data Group area SHALL be as in the following table:

Note 1.— DG11 SHALL be encoded as defined in [ISO/IEC 10918], using the JFIF option or [ISO/IEC 15444] using JPEG 2000 image coding system.

Note 2.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 72. Data Elements for DG11

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	O	Name of holder (in full)	99 Max	Var	B	Filler characters (<) inserted as per MRZ. No fillers inserted at end of line. Truncation not permitted.
02	O	Other name(s)	99 Max	Var	B	Filler characters (<) inserted as per MRZ. No fillers inserted at end of line. Truncation not permitted.
03	O	Personal number	99 Max	Var	U	Free-form text.
04	O	Full date of birth	8	F	N	YYYYMMDD
05	O	Place of birth	99 Max	Var	U	Free-form text.
06	O	Address	99 Max	Var	U	Free-form text.
07	O	Telephone	99 Max	Var	N,S	Free-form text. Encoding per ITU-T E.164 recommended
08	O	Profession	99 Max	Var	U	Free-form text.
09	M, if Data element 08 included	Title	99 Max	Var	U	Free-form text.
10	M, if Data element 09 included	Personal summary	99 Max	Var	U	Free-form text.
11	M, if Data element 10 included	Proof of citizenship		Var	B	Image of citizenship document formatted as per [ISO/IEC 10918-1]
12	O	Other valid travel document(s) Travel document number	99 Max	Var	U	Free-form text, separated by <.
13	O	Custody information	999 Max	Var	U	Free-form text.

Note.— In case the month (MM) or the day (DD) are unknown, the interoperable way to indicate this in DG11 is to set the respective characters to '00'. In case the century and the year (CCYY) are unknown, the interoperable way to indicate this in DG11 is to set the respective characters to '0000'. Issuer-assigned dates MUST always be used consistently.

4.7.12 DATA GROUP 12 — Additional Document Detail(s) (OPTIONAL)

This Data Group is used for additional information about the document. All Data Elements within this group are optional.

Table 73. Data Group 12 Tags

Tag	**L**	**Value**				
'6C'	Var					
		Tag	**L**	**Value**		
		'5C'	Var			Tag list with list of Data Elements in the template
		'5F19'	Var			Issuing Authority
		'5F26'	'08'			Date of issue. yyyymmdd
		'A0'	Var			Content-specific class
				Tag	**L**	**Value**
				'02'	'01'	Number of other persons
				'5F1A'	Var	Name of other person formatted per Doc 9303 rules. The data object repeats as many times as indicated in number of other names Data element 02 (data object with Tag'02').
		'5F1B'	Var			Endorsements, observations
		'5F1C'	Var			Tax/Exit requirements
		'5F1D'	Var			Image of front of document. Image per ISO/IEC 10918.
		'5F1E'	Var			Image of rear of document. Image per ISO/IEC 10918.
		'5F55'	'0E'			Date and time of document personalization yyyymmddhhmmss
		'5F56'	Var			Serial number of personalization system

It is RECOMMENDED that Inspection Systems support both 8 bytes ASCII and BCD date/time encoding.

4.7.12.1 DATA GROUP 12 — EF.DG12 Data Elements

This section describes the Data Elements that may be present in Data Group 12 (DG12). Data Elements and their format within each Data Group SHALL be as in the following table:

Note 1.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Note 2.— Data Elements 07 and 08 SHALL be encoded as defined in [ISO/IEC 10918], using the JFIF option or [ISO/IEC 15444] using JPEG 2000 image coding system.

Table 74. Data Elements for DG12

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	O	Issuing Authority	99 Max	Var	U	Free-form text.
02	O	Date of issue	8	F	N	Date of issue of document; i.e. YYYYMMDD.
03	O	Other person(s) details	99 Max	Var	U	Free-form text
04	O	Endorsement(s)/ Observation(s)	99 Max	Var	U	Free-form text.
05	O	Tax/Exit requirements	99 Max	Var	U	Free-form text.
06	O	Image of front of eMRTD		Var	B	Formatted as per [ISO/IEC 10918-1]
07	O	Image of rear of MRTD		Var	B	Formatted as per [ISO/IEC 10918-1]
08	O	Personalization time	14	F	N	yyyymmddhhmmss
09	O	Personalization device serial number	99 max	Var	U	Free format.

4.7.13 DATA GROUP 13 — Optional Details(s) (OPTIONAL)

Data Elements combining to form Data Group 13 (DG13) are at the discretion of the issuing State or organization and SHALL be as follows:

Table 75. Data Group 13 Tags

Tag	L	Value
‘6D’	Var	

4.7.14 DATA GROUP 14 — Security Options (CONDITIONAL)

Data Group 14 (DG14) contains security options for additional security mechanisms. For details see Doc 9303-11. The file DG14 contained in the eMRTD Application is REQUIRED if Chip Authentication or PACE-GM/-IM is supported by the eMRTD chip.

Table 76. Data Group 14 Tags

Tag	L	Value
'6E'	Var	Refer to Doc 9303-10 DG14 SecurityInfos

4.7.14.1 DATA GROUP 14 — EF.DG14 Data Elements

This section describes the Data Elements that may be present in DG14. Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 77. Data Elements for DG14

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
	O	SecurityInfos		Var	B	Refer to Doc 9303-10. DG14 SecurityInfos as defined in 4.7.14.2

4.7.14.2 DATA GROUP 14 SecurityInfos

The following generic ASN.1 data structure SecurityInfos allows various implementations of security options for secondary biometrics. For interoperability reasons, it is RECOMMENDED that this data structure be provided by the eMRTD chip in DG14 to indicate supported security protocols. The data structure is specified as follows:

```
SecurityInfos ::=    SET of SecurityInfo

SecurityInfo  ::=    SEQUENCE {
       protocol            OBJECT IDENTIFIER,
       requiredData  ANY DEFINED BY protocol,
       optionalData  ANY DEFINED BY protocol OPTIONAL
}
```

The elements contained in a SecurityInfo data structure have the following meaning:

- The object identifier protocol identifies the supported protocol;
- The open type requiredData contains protocol specific mandatory data;
- The open type optionalData contains protocol specific optional data.

4.7.15 DATA GROUP 15 — Active Authentication Public Key Info (CONDITIONAL)

This OPTIONAL Data Group contains the Active Authentication Public Key and is REQUIRED when implementing the optional Active Authentication chip authentication as described in Doc 9303-11.

Table 78. Data Group 15 Tags

Tag	L	Value
'6F'	Var	Refer to Doc 9303-11

4.7.15.1 DATA GROUP 15 — EF.DG15 Data Elements

This section describes the Data Elements that may be present in Data Group 15 (DG15). Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 79. Data Elements for DG15

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
	O	ActiveAuthenticationPublicKeyInfo		Var	B	See Doc 9303-11

4.7.16 DATA GROUP 16 — Person(s) to Notify (OPTIONAL)

This Data Group lists emergency notification information. It is encoded as a series of templates using the Tag 'Ax' designation. Data Group 16 (DG16), as all other data groups, SHOULD not be updated after issuance; DG16 is represented by a hash value in the SO_D and the SO_D is only signed once at issuance.

Table 80. Data Group 16 Tags

Tag	L	Value		
'70'	Var			
		Tag	**L**	**Value**
		'02'	'01'	Number of templates (occurs only in first template)
		'Ax'	Var	Start of template, where x (x = 1,2,3…) increments for each occurrence
'5F50'	'08'			Date data recorded
'5F51'	Var			Name of person
'5F52'	Var			Telephone
'5F53'	Var			Address

4.7.16.1 DATA GROUP 16 — EF.DG16 Data Elements

This section describes the Data Elements that may be present in DG16. Data Elements and their format within each Data Group area SHALL be as in the following table:

Note.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, Var = variable-length field.

Table 81. Data Elements for DG16

Data Element	Optional or MANDATORY	Name of Data Element	Number of Bytes	Fixed or Variable	Type of Coding	Coding Requirements
01	M, if DG16 included	Number of persons identified	1	F	N	Identifies number of persons included in the Data Group.
02	M, if DG16 included	Date details recorded	8	F	N	Date notification date recorded; Format = YYYYMMDD.
03	M, if DG16 included	Name of person to notify Primary and secondary identifiers		Var	A,N,S	Filler characters (<) inserted as per MRZ. Truncation not permitted.
04	M, if Data element 03 included	Telephone number of person to notify		Var	N,S	Telephone number in international form (country code and local number). Encoding per ITU-T E.164 recommended.
05	M	Address of person to notify		Var	U	Free-form text.

5. LDS2 APPLICATIONS (OPTIONAL)

Logical Data Structure 2 (LDS2) is an optional and backwards compatible extension to the LDS1 eMRTD chip that would allow for the digital and secure storage of travel information, after the document has been issued. LDS2 extends the use of the eMRTD through the addition of applications that could allow for the digital storage of travel data (visas and travel stamps), and other information that could facilitate the travel of the holder (additional biometrics), over its validity period. Better leveraging the full potential of the eMRTD by "digitizing" the remainder of the data contained in the documents offers a suite of facilitation benefits, while further protecting the document against vulnerabilities such as counterfeiting, copying and unauthorized reading or writing.

The additional and optional applications described as LDS2 are:

- Travel Records (Stamps);
- Electronic Visas; and
- Additional Biometrics.

It is MANDATORY for the LDS1 eMRTD application to be present before any OPTIONAL LDS2 applications may be declared.

5.1 Travel Records Application (CONDITIONAL)

The Travel Records application MAY be implemented by an issuing State or organization. The following is conditionally REQUIRED if the optional Travel Records application has been invoked.

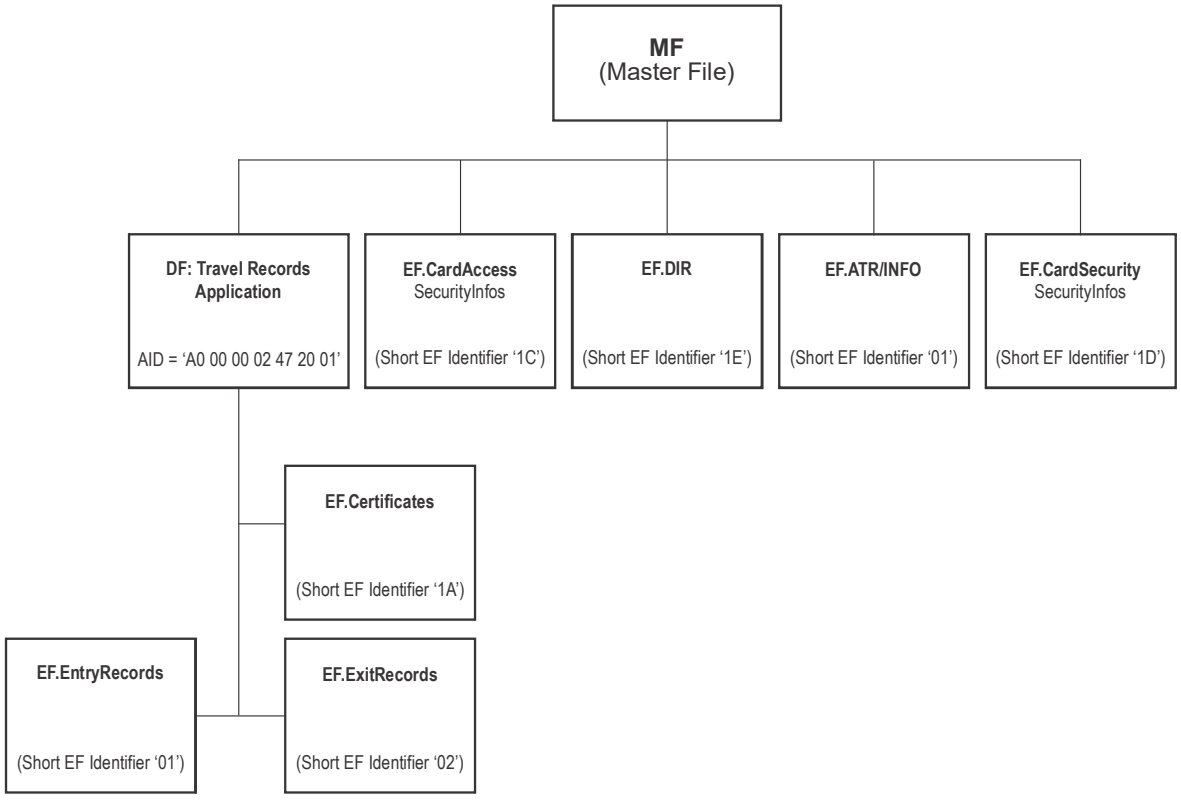


Figure 4. Travel Records Structure

Entry and Exit Travel Records are stored in two separate Elementary Files EF.EntryRecords and EF.ExitRecords, under the Travel Records application DF with both having linear structure with records of variable size as per [ISO/IEC 7816-4]. Travel Records Signer certificates are stored in a separate Elementary File EF.Certificates, having linear structure with records of variable size.

5.1.1 Application Selection — DF

The Travel Records application MUST be selected by use of the Application Identifier (AID) as a reserved DF name. The AID MUST consist of the Registered Application Identifier assigned by ISO according to [ISO/IEC 7816-5] followed by the Proprietary Application Identifier Extension (PIX) of the Travel Records application:

- The Registered Application Identifier is ‘A0 00 00 02 47’;
- The Travel Records application MUST use PIX = ‘20 01’; and
- The full AID of the Travel Records application MUST be ‘A0 00 00 02 47 20 01’.

If the effective authorization does not grant access rights to any data in an LDS2 application, selecting this application MUST be rejected by the IC.

5.1.2 EF.Certificates (MANDATORY)

The Travel Records Signer certificates are stored in an EF inside the application DF and have a linear structure with records of variable size. These certificates are intended to be used by the IS to further offline validation of the digital signatures for each record in both the EF.ExitRecords and EF.EntryRecords files.

Table 82. EF.Certificates

File Name	EF.Certificates
File ID	‘011A’
Short EF Identifier	‘1A’
Select / FMM Access	PACE+TA (Travel record authorization bit b3 according to Table 96)
Read record / Search Record Access	PACE+TA (Travel record authorization bit b3 according to Table 96)
Append Record Access	PACE+TA (Travel record authorization bit b4 according to Table 96)
Write / Update Record Access	NEVER
Erase Record Access	NEVER
File structure	Linear structure with records of variable size
Size	Variable

The certificate record contains a single LDS2-TS Signer X.509 certificate data object. A Certificate record MAY be referenced by one or more entry or exit Travel Record.

Table 83. EF.Certificates Record Format

Tag	Content	Mandatory /Optional	Format	Example
'5F3A'	Certificate serial number	M	V(22)B	'5F3A' 'Len' {Country code || SerialNumber }
'72'	X.509 certificate	M	V (900) B	'72' 'Len' { X.509 Certificate }

Note.— Interindustry tags specified in this table are used in LDS context, so coexistent tag allocation scheme is not required.

DO '5F3A' MUST contain a two-letter country code according to Doc 9303, Part 3 (same encoding and value as the X.509 issuing countryName on the subject's certificate) followed by the certificate serial number.

Each X.509 certificate contains a set of ASN.1 encoded data elements illustrated in Table 84. Detailed requirements for the X.509 Certificate can be found in the certificate profile specification of Doc 9303-12.

Table 84. X.509 Certificate Structure Example

Field			Description	Example value
Certificate				
	version		Must be version 3	2
	serialNumber		Unique positive integer	20 bytes max
	signature		Signature algorithm	ecdsa-with-SHA256
	issuer			
		countryName	Issuing country name	'US'
		commonName	Issuer name (9 characters max.)	'DHSCA0001'
	validity			
		notBefore	Cert. effective date	'131225000000Z'
		notAfter	Cert. expiration date	'230824235959Z'
	subject			
		countryName	IS country name	'US'
		commonName	IS name (9 characters max.)	'SFO000001'
	subjectPublicKeyInfo			

Field			Description	Example value
		Public Key Algorithm	ecPublicKey	
		Subject Public Key	IS public key	ECC256 Public Key
	extensions			
		AuthorityKeyIdentifier		
		ExtKeyUsage		
Signature Algorithm			ecdsa-with-SHA256	
Signature			Issuer's Signature	ECDSA256 signature

Note.— This table is an example for illustration only. Certificate records are written to EF.Certificates located under the Travel Records application DF using the APPEND RECORD command. Certificate records can be read from EF.Certificates using READ RECORD command. Certificate records MUST NOT be updated or erased. The maximum number of records in EF.Certificates under the Travel Records application DF MUST be 254.

5.1.3 EF.ExitRecords (MANDATORY)

Exit Records MUST be appended by an authorized IS upon embarkation.

Table 85. EF.ExitRecords

File Name	EF.ExitRecords
File ID	'0102'
Short EF Identifier	'02'
Select / FMM Access	PACE+TA (Travel record authorization bit b1 according to Table 96)
Read Record / Search Record Access	PACE+TA (Travel record authorization bit b1 according to Table 96)
Append Record Access	PACE+TA (Travel record authorization bit b2 according to Table 96)
Write / Update Record Access	NEVER
Erase Record Access	NEVER
File structure	Linear structure with records of variable size
Size	Variable

The content of an Exit Record is shown in Table 86.

Note.— Interindustry tags specified in the table below are used in LDS context, so coexistent tag allocation scheme is not required.

Table 86. Entry/Exit Record Format

Tag	Tag	Content	Mandatory /OPTIONAL	Format	Example
'5F44'		Embarkation/Debarkation State (copy for SEARCH RECORD)	M	F (3) A	USA
'73'	Entry / Exit Travel Record (signed info)				
	'5F44'	Embarkation/Debarkation State	M	F (3) A	USA
	'5F4C'	Visa approvals, refusals, and revocations	O	V (50) A,N,S,U	Free-form text
	'5F45'	Travel date (Date of entry/exit)	M	F (8) N	20120814 (yyyymmdd)
	'5F4B'	Inspection authority	M	V (10) A,N,S	CBP
	'5F46'	Inspection location (Port of Entry/Exit)	M	V (10) A,N,S	SFO
	'5F4A'	Inspector reference	M	V (20) A,N,S	SFO00001234
	'5F4D'	Result of inspection	O	V (50) A,N,S,U	Free-form text
	'5F49'	Mode of travel	O	F (1) A	A (Air), S (Sea), L (Land)
	'5F48'	Duration of stay (days)	O	V (2) B	'00FF' (255 days)
	'5F4E'	Conditions holder is required to observe while in the issuing State	O	V(50) A,N,S,U	Free-form text
'5F37'	Authenticity token (Signature)		M	V (140) B	'5F' '37' Len {Signature}
'5F38'	Reference (record number) to LDS2-TS Signer certificate in Certificates Store		M	F (1) B	'01' …'FE'

Note 1.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character ['<'], B = Binary data, F = fixed-length field, V = variable-length field.

Note 2.— Since LDS2-TS Signer certificates are likely to be the same in multiple Travel Records (e.g. when entering and exiting a country through the same airport having only one LDS2-TS Signer), before writing/appending a new certificate to the EF.Certificates, the IS should look up the EF.Certificates for a copy of the same certificate, and reference the existing one. This will reduce the size of EF.Certificates and enable faster lookups.

Note 3.— The LDS2 eMRTD does not enforce that an IS writes Entry Records only to the EF.EntryRecords, but not to the EF.ExitRecords, and vice versa.

Note 4.— Embarkation/Debarkation State three-letter code according to Doc 9303-3.

The order of the data objects in a record is fixed. The IS MUST build up the record content using the data objects in the order specified in the table.

Each Record MUST contain a digital signature (Authenticity Token) calculated over the DO'73', including Tag 73 and Length. Signature is generated by the LDS2-TS Signer.

LDS2-TS Signer certificates required to verify the signature of the Travel Record MUST be stored in the EF.Certificates under the Travel Records application DF if not already available in the same file.

Travel Records are written (appended) to EF using APPEND RECORD. Travel Records MUST NOT be altered (updated) or deleted. The maximum number of records in each EF allowed MUST be 254.

5.1.4 EF.EntryRecords (MANDATORY)

Entry Records MUST be appended by an authorized IS upon debarkation.

Table 87. EF.EntryRecords

File Name	EF.EntryRecords
File ID	'0101'
Short EF Identifier	'01'
Select / FMM Access	PACE+TA (Travel record authorization bit b1 according to Table 96)
Read Record / Search Record Access	PACE+TA (Travel record authorization bit b1 according to Table 96)
Append Record Access	PACE+TA (Travel record authorization bit b2 according to Table 96)
Write / Update Record Access	NEVER
Erase Record Access	NEVER
File structure	Linear structure with records of variable size
Size	Variable

The structure of the entry record is identical to the structure of the exit record specified in Table 86.

5.2 Visa Records Application (CONDITIONAL)

The Visa Records application MAY be implemented by an issuing State or organization. The following is conditionally REQUIRED if the optional Visa Records application has been invoked.

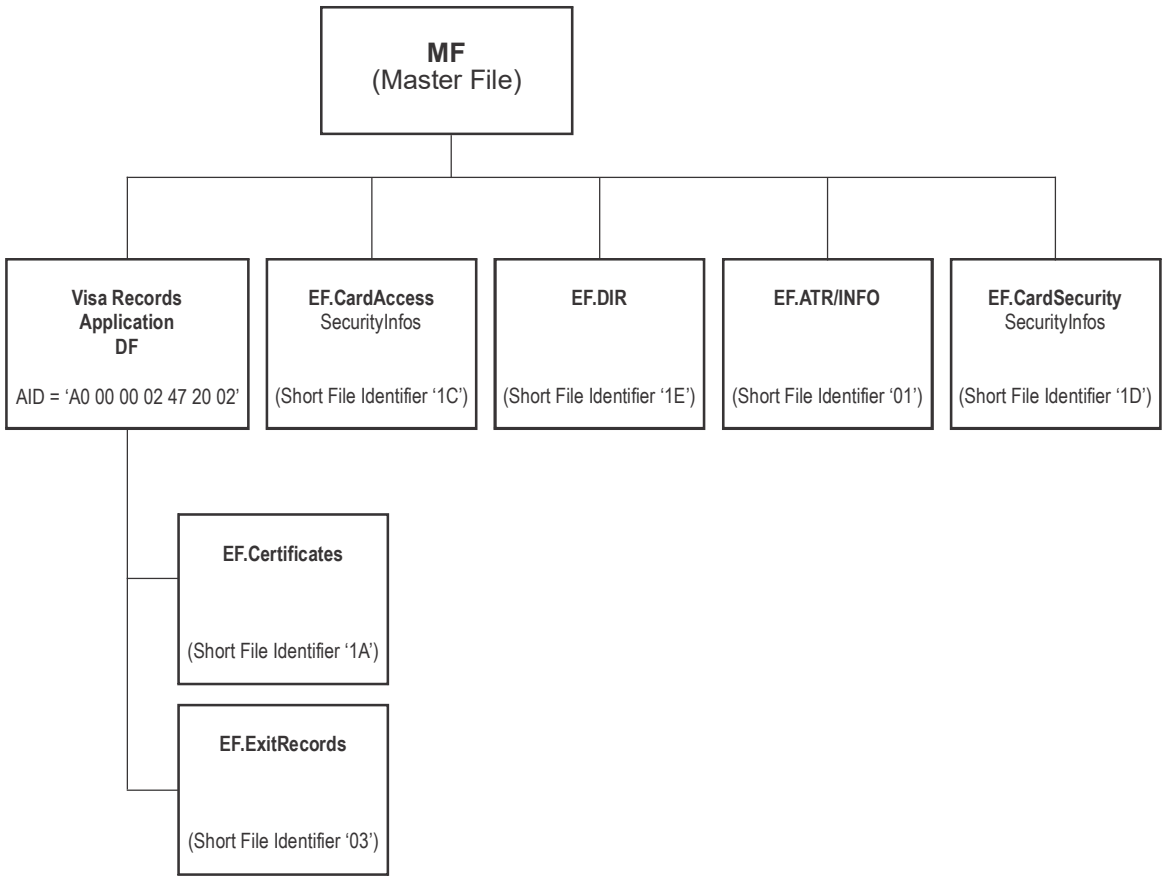


Figure 5. Visa Records Structure

Visa Records are stored in the Elementary File EF.VisaRecords under the Visa Records application DF. EF SHALL have a linear structure with records of variable size as per [ISO/IEC 7816-4]. Visa Records Signer certificates are stored in a separate Elementary File EF.Certificates having a linear structure with records of variable size.

5.2.1 Application Selection — DF

The Visa Records application MUST be selected by use of the Application Identifier (AID) as a reserved DF name. The AID MUST consist of the Registered Application Identifier assigned by ISO according to [ISO/IEC 7816-5] followed by the Proprietary Application Identifier Extension (PIX) of the Visa Records application:

- The Registered Application Identifier is 'A0 00 00 02 47';
- The Visa Records application MUST use PIX = '20 02'; and
- The full AID of the Visa Records application is 'A0 00 00 02 47 20 02'.

If the effective authorization does not grant access rights to any data in an LDS2 application, selecting this application MUST be rejected by the IC.

5.2.2 EF.Certificates (MANDATORY)

The Visa Records Signer certificates are stored in EF.Certificates inside the application DF and have a linear structure with records of variable size. These certificates are intended to be used by the IS to further offline validation of the digital signature for each record in the EF.VisaRecords.

Table 88. EF.Certificates

File Name	EF.Certificates
File ID	'011A'
Short EF Identifier	'1A'
Select / FMM Access	PACE+TA (Visa record authorization bit b3 according to Table 97)
Read Record / Search Record Access	PACE+TA (Visa record authorization bit b3 according to Table 97)
Append Record Access	PACE+TA (Visa record authorization bit b4 according to Table 97)
Write / Update Record Access	NEVER
Erase Record Access	NEVER
File structure	Linear structure with records of variable size
Size	Variable

Certificate record contains a single LDS2-V Signer X.509 certificate data object. A Certificate Record MAY be referenced by one or more Visa Records.

The structure of the Certificate record in Visa Application is identical to the structure of the Certificate record in Travel Record Application specified in Table 83.

Certificate records are written to EF.Certificates located under the Visa Records application DF using the APPEND RECORD command. Certificate records can be read from EF.Certificates using the READ RECORD command. Certificate records MUST NOT be updated or erased. The maximum number of records in EF.Certificates under the Visa Records application DF MUST be 254.

5.2.3 EF.VisaRecords (MANDATORY)

Visa Records MUST be stored in EF.VisaRecords having a linear structure with records of variable size.

Table 89. EF.VisaRecords

File Name	EF.VisaRecords
File ID	‘0103’
Short EF Identifier	‘03’
Select / FMM Access	PACE+TA (Visa record authorization bit b1 according to Table 97)
Read Record / Search Record Access	PACE+TA (Visa record authorization bit b1 according to Table 97)
Append Record Access	PACE+TA (Visa record authorization bit b2 according to Table 97)
Write / Update Record Access	NEVER
Erase Record Access	NEVER
File structure	Linear structure with records of variable size
Size	Variable

Each Visa Record MUST contain a sequence of BER-TLV data objects (DO ‘5F28’ and DO ‘71’), followed by the Authenticity Token (Signature) DO and DO a containing reference to LDS2-V Signer certificate in EF.Certificates. DO ‘71’ contains a set of DOs (fields) listed in the table below.

Note.— Interindustry tags specified in the table below are used in LDS context, so coexistent tag allocation scheme is not required.

Table 90. EF.VisaRecords Format

Tag	Tag	Content	MANDATORY/ OPTIONAL/ CONDITIONAL	Format	Example
‘5F28’		Issuing State or organization (Copy for SEARCH RECORD)	M	F (3) A	NLD
‘71’	Visa Record (signed info)				
	‘5F28’	Issuing State or organization	M	F (3) A	NLD
	‘43’	Document Type	M	F (2) A,N,S	VS
	‘5F71’	Machine Readable Visa of Type A	O	F (48) A,N,S	
	‘5F72’	Machine Readable Visa of Type B	O	F (44) A,N,S	VCD<<DENT<<ARTHUR< PHILIP<<<<<<<<<<<
	‘5F73’	Number of entries	O	V (1) B	‘01’ – ‘FF’

Tag	Tag	Content	MANDATORY/ OPTIONAL/ CONDITIONAL	Format	Example
	‘5F74’	Duration of stay (days, months, years)	O	F (3) B	‘010000’ – ‘FFFFFF’
	‘5F75’	Passport number	O	F (9) A,N,S	XI85935F8
	‘5F76’	Visa Type/class/category	O	V (4) B	
	‘5F77’	Territory Information	O	V (8) B	
	‘49’	Place of issuance (Issuing authority)	M	V (50) A, Sp	NEW YORK
	‘5F25’	Effective Date (Date of issuance)	M	F (8) N	20120826 (yyyymmdd)
	‘5F24’	Expiration Date	M	F (8) N	20130826 (yyyymmdd)
	‘5A’	Document number	M	F (9) A,N,S	XI85935F8
	‘5F32’	Additional information (endorsements: duration, limitations and fees paid)	O	V (50) A,N,S,U	Free-form text
	‘5B’	Name of holder (full name)	M	V (50) A, Sp	VAN DER STEEN MARIANNE LOUISE
	‘5F33’	Primary Identifier (surname)	M	V (50) A, Sp	VAN DER STEEN
	‘5F34’	Secondary Identifier (given name)	M	V (50) A, Sp	MARIANNE LOUISE
	‘5F35’	Sex	M	F (1) A,S	F, M, or <
	‘5F2B’	Date of birth	M	F (8) N,S	19870814 (yyyymmdd)
	‘5F2C’	Nationality	M	F (3) A	NLD
	‘5F1F’	MRZ	M	V (50) A,N,S	VAN<DER<STEEN<< MARIANNE<LOUISE
	‘5F40’	Reference to Additional Biometrics EF	O	F (2) B	’0201’
‘5F37’	Authenticity token (Signature)		M	V (140), B	‘5F’ ‘37’ Len {Signature}
‘5F38’	Reference (record number) to LDS2-V Signer certificate in Certificates Store		M	F (1) B	‘01’ …’FE’

Note 1.— A = Alpha character [a-z, A-Z], N = Numeric character [0-9], S = Special character [‘<’], B = Binary data, F = fixed-length field, V = variable-length field, Sp = Space.

Note 2.— Issuing State three-letter code according to Doc 9303-3.

Note 3.— Optional DO’5F40’, if present, MUST contain the two bytes identifier of the EF within the Additional Biometrics application containing biometric data. This DO may only be used provided the Additional Biometrics application is present on the eMRTD.

The order of the data objects in a record is fixed. The IS MUST build up the record content using the data objects in the order specified in the table.

Each Visa Record MUST contain a digital signature (Authenticity Token) calculated over the DO'71', including Tag 71 and Length. The signature is generated by the LDS2-V Signer.

LDS2-V Signer certificates required to verify the signature of the Visa Record are stored in a separate EF.Certificates store located under the Visa Records application DF.

Each Visa Record MUST be appended to EF.VisaRecords using APPEND RECORD. Visa Records and MUST NOT be altered (updated) or erased. The maximum number of records allowed in EF.VisaRecords MUST be 254.

5.3 Additional Biometrics Application (CONDITIONAL)

The Additional Biometrics application MAY be implemented by an issuing State or organization. The following is conditionally REQUIRED if the optional Additional Biometrics application has been invoked or any visa record has referenced it.

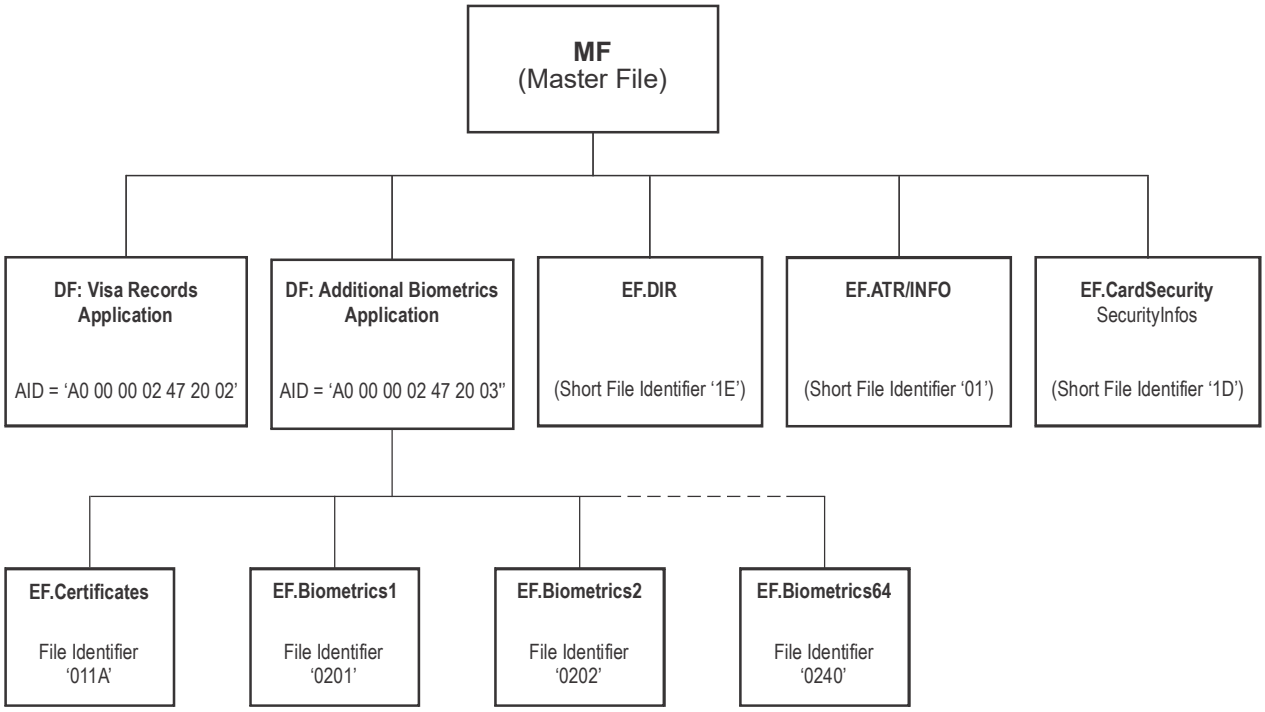


Figure 6. Additional Biometrics Application Structure

5.3.1 Application Selection — DF

The Additional Biometrics application MUST be selected by use of the Application Identifier (AID) as a reserved DF name. The AID MUST consist of the Registered Application Identifier assigned by ISO according to [ISO/IEC 7816-5] followed by the Proprietary Application Identifier Extension (PIX) of the Additional Biometrics application:

- the Registered Application Identifier is ‘A0 00 00 02 47’;
- the Additional Biometrics application MUST use PIX = ‘20 03’; and
- the full AID of the Additional Biometrics application is ‘A0 00 00 02 47 20 03’.

If the effective authorization does not grant access rights to any data in an LDS2 application, selecting this application MUST be rejected by the IC.

5.3.2 EF.Certificates (MANDATORY)

The Additional Biometrics Signer certificates are stored in EF.Certificates inside the application DF and have a linear structure with records of variable size. These certificates are intended to be used by the IS to further offline validation of the digital signature in the EF.Biometrics.

Table 91. EF.Certificates

File Name	EF.Certificates
File ID	‘011A’
Short EF Identifier	‘1A’
Select / FMM Access	PACE+TA (Additional Biometrics authorization byte 1 bit b1 (see Table 98)
Read Record/Search Record access	PACE+TA (Additional Biometrics authorization byte 1 bit b1 (see Table 98)
Append Record Access	PACE+TA (Additional Biometrics authorization byte 1 bit b2 (see Table 98)
Write / Update Record Access	NEVER
Erase Record Access	NEVER
File structure	Linear structure with records of variable size
Size	Variable

The certificate record contains a single Additional Biometrics Signer X.509 certificate data object. A Certificate Record MAY be referenced by one or more Additional Biometrics EF.

The structure of the Certificate record in the Additional Biometrics application is identical to the structure of the Certificate record in the Travel Record application specified in Table 83.

Certificate records are written to EF.Certificates located under the Additional Biometrics application DF using APPEND RECORD command. Certificate records can be read from EF.Certificates using the READ RECORD command. Certificate records MUST NOT be updated or erased. The maximum number of records in EF.Certificates under the Additional Biometrics application DF MUST be 64.

5.3.3 EF.Biometrics

Additional Biometric MUST be stored under Additional Biometrics Application in EFs and have a Transparent Structure as per [ISO/IEC 7816-4].

Each Additional Biometrics EF MAY be linked to one or more records in EF.VisaRecords in the Visa Records application (or other EFs and applications) using Additional Biometrics EF Identifier.

Table 92. EF.Biometrics1 through EF.Biometrics64

File Name	EF.Biometrics1 through EF.Biometrics64
File ID	‘0201’ through ‘0240’
Short EF Identifier	N/A
Select / FMM / Read Access in Deactivated state	PACE+TA (AdditionalBiometrics authorization according to Table 98, bits b2, b4, b6, b8 of byte 2-17)
Write Access in Deactivated state	PACE+TA (AdditionalBiometrics authorization according to Table 98, bits b2, b4, b6, b8 of byte 2-17)
Activate Access in Deactivated state	PACE+TA (AdditionalBiometrics authorization according to Table 98, bits b2, b4, b6, b8 of byte 2-17)
Select / FMM / Read Access in Activated state	PACE+TA (AdditionalBiometrics authorization according to Table 98, bits b1, b3, b5, b7 of byte 2-17)
Write Access in Activated state	NEVER
Activate Access in Activated state	NEVER
Erase Access	NEVER
File structure	Transparent structure
Size	Variable

Each Additional Biometric EF MUST contain a BER-TLV data object DO'7F2E' encapsulating three data objects: the Biometric data DO'5F2E' followed by the Authenticity Token (Signature) DO'5F37' and DO'5F38' containing the reference to an Additional Biometrics Signer certificate in EF.Certificates as shown in the table below.

The content of DO'5F2E' is up to the Additional Biometrics issuer and out of scope of this specification.

The Additional Biometrics EF creation mechanism is out of scope of this specification. The issuer SHOULD pre-create a number of Additional Biometrics EFs.

Note.— Interindustry tags specified in the table below are used in LDS context, so coexistent tag allocation scheme is not required.

Table 93. EF.Biometrics Format

Tag	Tag	Content	MANDATORY/ OPTIONAL/ CONDITIONAL	Format	Example
'7F2E'		Biometric Data Template	M		'7F' '2E' Len {DO'5F2E' || DO'5F37' || DO'5F38'}
	'5F2E'	Additional Biometric data	M	V, B	'5F' '2E' Len {Biometric data}
	'5F37'	Authenticity token (Signature)	M	V (140), B	'5F' '37' Len {Signature}
	'5F38'	Reference (record number) to Additional Biometrics Signer certificate in Certificates Store	M	F (1) B	'01' …'40'

Note.— B = Binary data, F = fixed-length field, V = variable-length field.

The order of the data objects in EF is fixed.

Each Additional Biometrics EF MUST contain a digital signature (Authenticity Token) calculated over the DO'5F2E', including Tag and length. The signature is generated by the Additional Biometrics Signer.

The Additional Biometrics Signer certificate required to verify Additional Biometrics' signature is stored in a separate EF.Certificates store located under the Additional Biometrics application DF.

Each Additional Biometrics EF MUST be written using the UPDATE BINARY command.

Additional Biometrics EF MUST NOT be altered (updated) or erased. The maximum number of Additional Biometrics EFs is 64.

All possible Additional Biometrics EF names, identifiers and short identifiers are listed in Table 94.

Table 94. EF.Biometrics Identifiers

EF name	EF identifier	Short EF identifier
EF.Biometrics1	‘0201’	N/A
EF.Biometrics2	‘0202’	N/A
EF.Biometrics3	‘0203’	N/A
EF.Biometrics4	‘0204’	N/A
EF.Biometrics5	‘0205’	N/A
EF.Biometrics6	‘0206’	N/A
EF.Biometrics7	‘0207’	N/A
EF.Biometrics8	‘0208’	N/A
EF.Biometrics9	‘0209’	N/A
EF.Biometrics10	‘020A’	N/A
EF.Biometrics11	‘020B’	N/A
EF.Biometrics12	‘020C’	N/A
EF.Biometrics13	‘020D’	N/A
EF.Biometrics14	‘020E’	N/A
EF.Biometrics15	‘020F’	N/A
EF.Biometrics16	‘0210’	N/A
EF.Biometrics17	‘0211’	N/A
EF.Biometrics18	‘0212’	N/A
EF.Biometrics19	‘0213’	N/A
EF.Biometrics20	‘0214’	N/A
EF.Biometrics21	‘0215’	N/A
EF.Biometrics22	‘0216’	N/A
EF.Biometrics23	‘0217’	N/A
EF.Biometrics24	‘0218’	N/A
EF.Biometrics25	‘0219’	N/A
EF.Biometrics26	‘021A’	N/A
EF.Biometrics27	‘021B’	N/A
EF.Biometrics28	‘021C’	N/A
EF.Biometrics29	‘021D’	N/A
EF.Biometrics30	‘021E’	N/A
EF.Biometrics31	‘021F’	N/A
EF.Biometrics32	‘0220’	N/A

EF name	EF identifier	Short EF identifier
EF.Biometrics33	‘0221’	N/A
EF.Biometrics34	‘0222’	N/A
EF.Biometrics35	‘0223’	N/A
EF.Biometrics36	‘0224’	N/A
EF.Biometrics37	‘0225’	N/A
EF.Biometrics38	‘0226’	N/A
EF.Biometrics39	‘0227’	N/A
EF.Biometrics40	‘0228’	N/A
EF.Biometrics41	‘0229’	N/A
EF.Biometrics42	‘022A’	N/A
EF.Biometrics43	‘022B’	N/A
EF.Biometrics44	‘022C’	N/A
EF.Biometrics45	‘022D’	N/A
EF.Biometrics46	‘022E’	N/A
EF.Biometrics47	‘022F’	N/A
EF.Biometrics48	‘0230’	N/A
EF.Biometrics49	‘0231’	N/A
EF.Biometrics50	‘0232’	N/A
EF.Biometrics51	‘0233’	N/A
EF.Biometrics52	‘0234’	N/A
EF.Biometrics53	‘0235’	N/A
EF.Biometrics54	‘0236’	N/A
EF.Biometrics55	‘0237’	N/A
EF.Biometrics56	‘0238’	N/A
EF.Biometrics57	‘0239’	N/A
EF.Biometrics58	‘023A’	N/A
EF.Biometrics59	‘023B’	N/A
EF.Biometrics60	‘023C’	N/A
EF.Biometrics61	‘023D’	N/A
EF.Biometrics62	‘023E’	N/A
EF.Biometrics63	‘023F’	N/A
EF.Biometrics64	‘0240’	N/A

5.4 LDS2 Application File Access Conditions (CONDITIONAL)

5.4.1 Roles and Default Authorization Levels (MANDATORY)

Each CV certificate contains a Certificate Holder Authorization Template (CHAT) that identifies the certificate holder role (IS, DV, CVCA) and contains access rights to DG3/DG4 of the REQUIRED LDS2 eMRTD Application (for legacy reasons or other national uses).

CHAT comprises a sequence of two objects:

a) An object identifier specifying the terminal type and the format of the template [TR- 03110]:

```
id-roles OBJECT IDENTIFIER ::= {bsi-de applications(3) mrtd(1) 2}
id-IS       OBJECT IDENTIFIER ::= {id-roles 1}
```

b) An A discretionary data object (tag '53') containing bit-encoded role and read-only access rights of the certificate holder according to the following table:

Table 95. Default CHAT Authorization

	Description	Byte 1							
		b8	**b7**	**b6**	**b5**	**b4**	**b3**	**b2**	**b1**
Role	CVCA	1	1						
	DV (domestic)	1	0						
	DV (foreign)	0	1						
	IS	0	0						
Read Access	RFU								
	RFU								
	RFU								
	RFU								
	DG4 (Iris)							1	
	DG3 (Finger)								1

Note.— The LDS2 eMRTD MUST ignore the value of RFU bits in the Certificate Holder Authorization.

5.4.2 Application Authorization Levels (MANDATORY)

Certificate holder authorizations for each LDS2 application are encoded in CV certificate extensions (one extension per application). Certificate extension is a discretionary template (tag ‘73’) comprising two data objects: an Authorization Object Identifier (tag ‘06’) for a specific application and a discretionary data object (tag ‘53’) containing bit-encoded access rights of the certificate holder to a specified application.

To determine the effective authorization of a certificate holder, the LDS2 eMRTD chip calculates a bitwise Boolean ‘and’ of the access rights contained in the certificate extensions of the IS Certificate, and the referenced DV and CVCA Certificates.

For the Travel Records application, the Authorization Object Identifiers and access rights encodings are:

```
id-icao-lds2-travelRecords        OBJECT IDENTIFIER ::= {id-icao-lds2 1}
id-icao-lds2-travelRecords-access OBJECT IDENTIFIER ::= {id-icao-lds2-
travelRecords 3}
```

Table 96. Authorizations for Travel Records Application

	Description	Byte 1							
		b8	b7	b6	b5	b4	b3	b2	b1
Access rights	RFU								
	RFU								
	RFU								
	RFU								
	Append EF.Certificates					1			
	Read/Search/Select/FMM EF.Certificates						1		
	Append EF.EntryRecords/ExitRecords							1	
	Read/Search/Select/FMM EF.EntryRecords/ExitRecords								1

For the Visa Records application, the Authorization Object Identifiers and access rights encodings are:

```
id-icao-lds2-visaRecords          OBJECT IDENTIFIER ::= {id-icao-lds2 2}

id-icao-lds2-visaRecords-access OBJECT IDENTIFIER  ::= {id-icao-lds2-
visaRecords 3}
```

Table 97. Authorizations for Visa Records Application

	Description	Byte 1							
		b8	b7	b6	b5	b4	b3	b2	b1
Access rights	RFU								
	RFU								
	RFU								
	RFU								
	Append EF.Certificates					1			
	Read/Search/Select/FMM EF.Certificates						1		
	Append EF.VisaRecords							1	
	Read/Search/Select/FMM EF.VisaRecords								1

For the Additional Biometrics application, the Authorization Object Identifiers and access rights encodings are:

```
id-icao-lds2-additionalBiometrics OBJECT IDENTIFIER ::= {id-icao-lds2 3}

id-icao-lds2-additionalBiometrics-access  OBJECT  IDENTIFIER::=  {id-icao-lds2-
additionalBiometrics 3}
```

Table 98. Authorizations for Additional Biometrics Application

	Description	EF Iden-tifier	Authorizations							
			b8	b7	b6	b5	b4	b3	b2	b1
Byte 1	RFU									
	RFU									
	RFU									
	RFU									
	RFU									
	RFU									
	Append EF.Certificates	'011A'							1	
	Select/FMM/Read/Search EF.Certificates	'011A'								1
Byte 2	Select/FMM/Write/Activate/Read EF.Biometrics1 in Deactivated state	'0201'	1							
	Select/FMM/Read EF.Biometrics1 in Activated state	'0201'		1						
	Select/FMM/Write/Activate/Read EF.Biometrics2 in Deactivated state	'0202'			1					
	Select/FMM/Read EF.Biometrics2 in Activated state	'0202'				1				
	Select/FMM/Write/Activate/Read EF.Biometrics3 in Deactivated state	'0203'					1			
	Select/FMM/Read EF.Biometrics3 in Activated state	'0203'						1		
	Select/FMM/Write/Activate/Read EF.Biometrics4 in Deactivated state	'0204'							1	
	Select/FMM/Read EF.Biometrics4 in Activated state	'0204'								1

...

Byte 17	Select/FMM/Write/Activate/Read EF.Biometrics61 in Deactivated state	'023D'	1							
	Select/FMM/Read EF.Biometrics61 in Activated state	'023D'		1						
	Select/FMM/Write/Activate/Read EF.Biometrics62 in Deactivated state	'023E'			1					
	Select/FMM/Read EF.Biometrics62 in Activated state	'023E'				1				
	Select/FMM/Write/Activate/Read EF.Biometrics63 in Deactivated state	'023F'					1			
	Select/FMM/Read EF.Biometrics63 in Activated state	'023F'						1		
	Select/FMM/Write/Activate/Read EF.Biometrics64 in Deactivated state	'0240'							1	
	Select/FMM/Read EF.Biometrics64 in Activated state	'0240'								1

Note 1.— The LDS2 eMRTD MUST ignore the value of RFU bits in the Certificate Holder Authorization.

Note 2.— Issuing States or organizations MUST NOT issue terminal certificates with write/activate authorizations to the IS if they only have read authorizations for Additional Biometrics.

6. OBJECT IDENTIFIERS

6.1 LDS1 and LDS2 Application Object Identifiers Summary

Table 99. LDS1.7, LDS1.8 and LDS2 OIDs

Object Identifier	Value	Comments
id-icao	joint-iso-itu-t(2) international-organizations(23) icao(136)	ICAO OID
id-icao-mrtd	id-icao 1	eMRTD OID
id-icao-mrtd-security	id-icao-mrtd 1	
id-icao-ldsSecurityObject	id-icao-mrtd-security 1	LDS security object
id-icao-mrtd-security-cscaMasterList	id-icao-mrtd-security 2	CSCA master list
id-icao-mrtd-security-cscaMasterListSigningKey	id-icao-mrtd-security 3	
id-icao-mrtd-security-documentTypeList	id-icao-mrtd-security 4	document type list
id-icao-mrtd-security-aaProtocolObject	id-icao-mrtd-security 5	Active Authentication protocol
id-icao-mrtd-security-extensions	id-icao-mrtd-security 6	CSCA name change
id-icao-mrtd-security-extensions-nameChange	id-icao-mrtd-security-extensions 1	
id-icao-mrtd-security-extensions-documentTypeList	id-icao-mrtd-security-extensions 2	DS document type
id-icao-mrtd-security-DeviationList	id-icao-mrtd-security 7	Defect List Base OIDs
id-icao-mrtd-security-DeviationListSigningKey	id-icao-mrtd-security 8	
id-icao-lds2	id-icao-mrtd-security 9	LDS2 Object Identifiers
id-icao-lds2-travelRecords	id-icao-lds2 1	Travel Records application base OID
id-icao-lds2-travelRecords-application	id-icao-lds2-travelRecords 1	Travel Records AID
id-icao-lds2-travelRecords-access	id-icao-lds2-travelRecords 3	Authorization certificate extension
id-icao-lds2-visaRecords	id-icao-lds2 2	Visa Records application base OID
id-icao-lds2-visaRecords-application	id-icao-lds2-visaRecords 1	Visa Records AID
id-icao-lds2-visaRecords-access	id-icao-lds2-visaRecords 3	Authorization certificate extension
id-icao-lds2-additionalBiometrics	id-icao-lds2 3	Additional Biometrics base OID
id-icao-lds2-additionalBiometrics-application	id-icao-lds2-additionalBiometrics 1	Additional Biometrics AID
id-icao-lds2-additionalBiometrics-access	id-icao-lds2-additionalBiometrics 3	Authorization certificate extension
id-icao-lds2Signer	id-icao-lds2 8	LDS2 Signers Object Identifiers
id-icao-tsSigner	id-icao-lds2Signer 1	LDS2 Travel Stamp Signer certificate
id-icao-vSigner	id-icao-lds2Signer 2	LDS2 Visa Signer certificate
id-icao-bSigner	id-icao-lds2Signer 3	LDS2 Biometrics Signer certificate
id-icao-spoc	id-icao-mrtd-security 10	SPOC Object Identifiers
id-icao-spocClient	id-icao-spoc 1	Client
id-icao-spocServer	id-icao-spoc 2	Server

7. ASN.1 SPECIFICATIONS

```
id-icao OBJECT IDENTIFIER ::= {joint-iso-itu-t(2) international-organizations(23)
icao(136) }
id-icao-mrtd OBJECT IDENTIFIER ::= {id-icao 1}
id-icao-mrtd-security OBJECT IDENTIFIER ::= {id-icao-mrtd 1}
id-icao-ldsSecurityObject OBJECT IDENTIFIER ::= {id-icao-mrtd-security 1}

id-icao-mrtd-security-cscaMasterList OBJECT IDENTIFIER ::= {id-icao-mrtd-security 2}
id-icao-mrtd-security-cscaMasterListSigningKey OBJECT IDENTIFIER ::= {id-icao-mrtd-
security 3}
id-icao-mrtd-security-documentTypeList OBJECT IDENTIFIER ::= {id-icao-mrtd-security
4}
id-icao-mrtd-security-aaProtocolObject OBJECT IDENTIFIER ::= {id-icao-mrtd-security
5}
id-icao-mrtd-security-extensions OBJECT IDENTIFIER ::= {id-icao-mrtd-security 6}
id-icao-mrtd-security-extensions-nameChange OBJECT IDENTIFIER ::= {id-icao-mrtd-
security-extensions 1}
id-icao-mrtd-security-extensions-documentTypeList OBJECT IDENTIFIER ::= {id-icao-mrtd-
security-extensions 2}
id-icao-mrtd-security-DeviationList OBJECT IDENTIFIER ::= {id-icao-mrtd-security 7}
id-icao-mrtd-security-DeviationListSigningKey OBJECT IDENTIFIER ::= {id-icao-mrtd-
security 8}

id-icao-lds2 OBJECT IDENTIFIER ::= {id-icao-mrtd-security 9}
```

LDS2 Travel Records application Object Identifiers

```
id-icao-lds2-travelRecords OBJECT IDENTIFIER ::= {id-icao-lds2 1}
id-icao-lds2-travelRecords-application OBJECT IDENTIFIER ::= {id-icao-lds2-
travelRecords 1}
id-icao-lds2-travelRecords-access OBJECT IDENTIFIER ::= {id-icao-lds2-travelRecords
3}
```

LDS2 Visa Records application Object Identifiers

```
id-icao-lds2-visaRecords OBJECT IDENTIFIER ::= {id-icao-lds2 2}
id-icao-lds2-visaRecords-application OBJECT IDENTIFIER ::= {id-icao-lds2-visaRecords
1}
id-icao-lds2-visaRecords-access OBJECT IDENTIFIER ::= {id-icao-lds2-visaRecords 3}
```

LDS2 Additional Biometrics application Object Identifiers

```
id-icao-lds2-additionalBiometrics OBJECT IDENTIFIER ::= {id-icao-lds2 3}
id-icao-lds2-additionalBiometrics-application OBJECT IDENTIFIER ::= {id-icao-lds2-
additionalBiometrics 1}
id-icao-lds2-additionalBiometrics-access OBJECT IDENTIFIER ::= {id-icao-lds2-
additionalBiometrics 3}

id-icao-lds2Signer OBJECT IDENTIFIER ::= {id-icao-lds2 8}
id-icao-tsSigner OBJECT IDENTIFIER ::= {id-icao-lds2Signer 1}
id-icao-vSigner OBJECT IDENTIFIER ::= {id-icao-lds2Signer 2}
id-icao-bSigner OBJECT IDENTIFIER ::= {id-icao-lds2Signer 3}
```

```
id-icao-spoc OBJECT IDENTIFIER ::=  {id-icao-mrtd-security 10}
id-icao-spocClient OBJECT IDENTIFIER ::=  {id-icao-spoc 1}
id-icao-spocServer OBJECT IDENTIFIER ::=  {id-icao-spoc 2}
```

8. REFERENCES (NORMATIVE)

ISO/IEC 14443-1	ISO/IEC 14443-1:2016, *Identification cards — Contactless integrated circuit(s) cards — Proximity cards — Part 1: Physical characteristics*
ISO/IEC 14443-2	ISO/IEC 14443-2:2016, *Identification car*ds *— Contactless integrated circuit(s) cards — Proximity cards — Part 2: Radio frequency power and signal interface*
ISO/IEC 14443-3	ISO/IEC 14443-3:2016, *Identification cards — Contactless integrated circuit(s) cards — Proximity cards — Part 3: Initialization and Anticollision*
ISO/IEC 14443-4	ISO/IEC 14443-4:2016, *Identification cards — Contactless integrated circuit(s) cards — Proximity cards — Part 4: Transmission protocol*
ISO/IEC 10373-6	ISO/IEC 10373-6:2016, *Identification cards — Test methods — Part 6: Proximity cards*
ISO/IEC 18745-2	ISO/IEC 18745-2:2016 *Information technology — Test methods for machine readable travel documents (MRTD) and associated devices — Part 2: Test methods for the contactless interface*
ISO/IEC 7816-2	ISO/IEC 7816-2:2007, *Identification cards — Integrated circuit cards — Part 2: Cards with contacts — Dimensions and location of the contacts*
ISO/IEC 7816-4	ISO/IEC 7816-4:2013, *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*
ISO/IEC 7816-5	ISO/IEC 7816-5:2004, *Identification cards — Integrated circuit cards — Part 5: Registration of application providers*
ISO/IEC 7816-6	ISO/IEC 7816-6:2016, *Identification cards — Integrated circuit cards — Part 6: Interindustry data elements for interchange (Defect report included)*
ISO/IEC 7816-11	ISO/IEC 7816-11:2017, *Identification cards — Integrated circuit cards — Part 11: Personal verification through biometric methods*
ISO/IEC 8825-1	ISO/IEC 8825-1:2008, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*
ISO/IEC 19794-4	ISO/IEC 19794-4:2005, *Information technology — Biometric data interchange formats — Part 4: Finger image data*
ISO/IEC 19794-5	ISO/IEC 19794-5:2005*, Information technology — Biometric data interchange formats — Part 5: Face image data*

ISO/IEC 19794-6	ISO/IEC 19794-6:2011*, Information technology — Biometric data interchange formats — Part 6: IRIS image data*
ISO/IEC 10646	ISO/IEC 10646:2012*, Information technology — Universal Coded Character Set (UCS)*
RFC 3369	Cryptographic Message Syntax 2002
ISO/IEC 10918-1	ISO/IEC 10918-1:1994, *Information technology — Digital compression and coding of continuous-tone still images: Requirements and guidelines*
ISO/IEC 15444	ISO/IEC 15444-n, *JPEG 2000 image coding system*
ISO/IEC 19785	ISO/IEC 19785-n, *Information technology — Common Biometric Exchange Formats Framework*
ISO/IEC 19795-6	ISO/IEC 19795-6:2012, *Information technology — Biometric performance testing and reporting — Part 6: Testing methodologies for operational evaluation*
ISO/IEC 39794-4	ISO/IEC 39794-4:2019, *Information technology — Extensible biometric data interchange formats — Part 4: Finger image data*
ISO/IEC 39794-5	ISO/IEC 39794-5:2019, *Information technology — Extensible biometric data interchange formats — Part 5: Face image data*
ISO/IEC 39794-6	ISO/IEC 39794-6:2021, *Information technology — Extensible biometric data interchange formats — Part 6: Iris image data*

— — — — — — — —

Appendix A to Part 10

LOGICAL DATA STRUCTURE MAPPING EXAMPLES (INFORMATIVE)

The following informative text describes examples of mapping of the Logical Data Structure (LDS v1.7) using a random access representation to a contactless integrated circuit on an eMRTD.

A.1 EF.COM COMMON DATA ELEMENTS

The following example indicates an implementation of LDS Version 1.7 using Unicode Version 4.0.0 having Data Groups 1 (tag '61'), 2 (tag '75'), 4 (tag '76'), and 12 (tag '6C') present.

For this and all other examples, the Tags are printed in **bold**, the Lengths printed *italics*, and the Values are printed in roman. Hexadecimal Tags, lengths and values are in quote marks ('xx').

'60' *'16'*
'5F01' *'04'* '0107'
'5F36' *'06'* '040000'
'5C' *'04'* '6175766C'

The example would read in full hexadecimal representation as:

'60' *'16'*
'5F01' *'04'* '30313037'
'5F36' *'06'* '303430303030'
'5C' *'04'* '6175766C'

A hypothetic LDS Version 15.99 would be encoded as:

'60' *'16'*
'5F01' *'04'* '1599'
'5F36' *'06'* '040000'
'5C' *'04'* '6175766C'

or hexadecimal:

'60' *'16'*
'5F01' *'04'* '31353939'
'5F36' *'06'* '303430303030'
'5C' *'04'* '6175766C'

A.2 EF.DG1 MACHINE READABLE ZONE INFORMATION

A.2.1 TD1 Size eMRTD

An example of the DG1 using this information in a TD1 size LDS1 eMRTD is shown below. The length of the MRZ data element is 90 bytes (‘5A’).

‘61’ *‘5D’* **‘5F1F’** *‘5A’*

```
I<NLDXI85935F86999999990<<<<<<7208148F1108268NLD<<<<<<<<<<<4VAN<DER<STEEN<<MARI
ANNE<LOUISE
```

A.2.2 TD2 Size eMRTD

An example of the DG1 using this information in a TD2 size LDS1 eMRTD is shown below. The length of the MRZ data element is 72 bytes (‘48’).

‘61’ *‘4B’* **‘5F1F’** *‘48’*

```
I<ATASMITH<<JOHN<T<<<<<<<<<<<<<<<<<<<<<<<<<<123456789<HMD7406222M10123130121<<<54
```

A.3 EF.DG2 TO EF.DG4 BIOMETRIC TEMPLATES

DG2 to DG4 use the nested off-card option of [ISO/IEC 7816-11] to have the possibility to store multiple biometric templates of a kind which are in harmony with the Common Biometric Exchange Formats Framework (CBEFF), [NISTR 6529a]. The biometric sub-header defines the type of biometric that is present and the specific biometric feature.

Example: One signed, facial biometric with the biometric data block length of 12 642 bytes (‘3162’ bytes), encoded using a device with a PID of ‘00 01 00 01’, using format type ‘00 04’ owned by template provider ‘00 0A’ was captured on 15 March 2002 (no UTC offset) and is valid from 1 April 2002 through 31 March 2007. ICAO patron template Version 1.0 is being used.

The total length of the template is 12 704 bytes. The template is stored starting at the beginning of EF.DG2 (SFID 02).

‘75’ *‘82319EC’*
- **‘7F61’** *‘823199’*
  - **‘02’** *‘01’* ‘01’
  - **‘7F60’** *‘823191’*
  - **‘A1’** *‘26’*
    - **‘80’** *‘02’* ‘0101’
    - **‘81’** *‘01’* ‘02’
    - **‘83’** *‘07’* ‘20020315133000’
    - **‘85’** *‘08’* ‘2002040120070331’
    - **‘86’** *‘04’* ‘00010001’
    - **‘87’** *‘02’* ‘0101’
    - **‘88’** *‘02’* ‘0008’
  - **‘5F2E’** *‘823162’* ‘… 12 642 bytes of biometric data …’

A.4 EF.DG5 TO EF.DG7 DISPLAYED IMAGE TEMPLATES

Note.— One EF for each DG.

Example: Image template with the displayed image data length of 2 000 bytes. The length of the template is 2 008 bytes (‘07D8’).

‘65’ *‘8207D8’*
‘02’ *‘01’* 1
‘5F40’ *‘8207D0’* ‘….2 000 bytes of image data …’

A.5 EF.DG11 ADDITIONAL PERSONAL DETAILS

The following example shows the following personal details: Full name (John J. Smith), Place of birth (Anytown, MN), Permanent address (123 Maple Rd, Anytown, MN), Telephone number 1-612-555-1212 and Profession (Travel Agent). The length of the template is 99 bytes (*‘63’*).

‘6B’ *‘63’*
‘5C’ *‘0A’* **‘5F0E’ ‘5F11’ ‘5F42’ ‘5F12’ ‘5F13’**
‘5F0E’ *‘0D’* SMITH<<JOHN<J
‘5F11’ *‘0A’* ANYTOWN<MN
‘5F42’ *‘17’* 123 MAPLE RD<ANYTOWN<MN
‘5F12’ *‘0E’* 16125551212
‘5F13’ *‘0C’* TRAVEL<AGENT

A.6 EF.DG16 PERSON(S) TO NOTIFY

Example with two entries: Charles R. Smith of Anytown, MN and Mary J. Brown of Ocean Breeze, CA. The length of the template is 162 bytes (‘A2’).

‘70’ *‘81A2’*

‘02’ *‘01’* 2
‘A1’ ‘4C’
‘5F50’ *‘08’* 20020101
‘5F51’ *‘10’* SMITH<<CHARLES<R
‘5F52’ *‘0B’* 19525551212
‘5F53’ *‘1D’* 123 MAPLE RD<ANYTOWN<MN<55100
‘A2’ ‘4F’
‘5F50’ *‘08’* 20020315
‘5F51’ *‘0D’* BROWN<<MARY<J
‘5F52’ *‘0B’* 14155551212
‘5F53’ *‘23’* 49 REDWOOD LN<OCEAN BREEZE<CA<94000

— — — — — — — —

Appendix B to Part 10

THE CONTACLESS IC IN AN eMRP (INFORMATIVE)

B.1 THE ANTENNA SIZE AND CLASS OF AN eMRTD

The antenna size is at the discretion of the issuing State. With the exception of the antenna size, both the LDS1 and LDS2 eMRTD shall fulfil all tests specified in [ISO/IEC 18745-2] applying the Class 1 specifications.

It is RECOMMENDED for eMRTDs to be also compliant with Class 1 specifications.

There is no mandatory position of the IC, which MAY be placed in an arbitrary position. The location of the contactless antenna is at the discretion of the issuing State as long as it is in one of the following locations:

Data page — IC and antenna within the structure of a data page forming an internal page;
Centre of booklet — Placing the IC and its antenna between the centre pages of the book;
Cover — Placement within the structure or construction of the cover;
Separate sewn-in page — Incorporating the IC and its antenna into a separate page, which MAY be in the form of an ID3 size plastic card, sewn into the book during its manufacture; or
Back cover — Placement within the structure or construction of the back cover.

B.2 BOOTING AND POLLING

An eMRTD brought to an alternate magnetic field of 1.5 A/m as measured in [ISO/IEC 18745-2] shall respond to any REQ/WUP appropriate to its Type after an unmodulated alternate magnetic field of 10 ms. It is RECOMMENDED to be able to respond to any REQ/WUP appropriate to its Type after an unmodulated alternate magnetic field of 5 ms.

B.3 ANTICOLLISION AND TYPE

The eMRTD MAY either declare compliance with Type A or with Type B as defined in [ISO/IEC 14443-2]. It shall not change its Type unless it has been reset by the eMRTD associated Inspection System.

B.4 MANDATORY BIT RATES

The eMRTD shall provide at least the following bit rates, as defined in [ISO/IEC 14443-2], mandatorily: 106 kbit/s and 424 kbit/s in both directions between the eMRTD and the eMRTD associated Inspection System.

The bit rate of 212 kbit/s, and all bit rates from 848 kbit/s up to 6.78 Mbit/s for both directions, and from 10.17 Mbit/s to 27.12 Mbit/s from the eMRTD associated Inspection System to the eMRTD, as defined in [ISO/IEC 14443-2], are optional.

B.5 ELECTROMAGNETIC DISTURBANCE (EMD)

The support of EMD is not mandatory.

Note.— The EMD feature enhances the robustness of the contactless communication between the eMRTD and the eMRTD associated Inspection System against eMRTD generated electromagnetic disturbance. The eMRTD dynamic current consumption during execution of a command may cause an arbitrary load modulation effect (which may not be purely resistive) on the magnetic field. In some cases, the eMRTD associated Inspection System may misinterpret EMD as data sent by the eMRTD, and this may negatively impact proper reception of the eMRTD response.

B.6 (OPTIONAL) SUPPORT OF EXCHANGE OF ADDITIONAL PARAMETERS

The eMRTD MAY support the exchange of additional parameters as defined in [ISO/IEC 14443-4] in order to negotiate bit rates higher than 106 kbit/s. It MAY also use the same additional parameters to negotiate frames with error correction as specified in[ISO/IEC 14443-4].

B.7 SHIELDING

It is RECOMMENDED to not shield any page of the eMRTD.

B.8 (RECOMMENDED) UNIQUE IDENTIFIER (UID) AND PSEUDO-UNIQUE PICC IDENTIFIER (PUPI)

The eMRTD MAY provide a random or fixed UID/PUPI as defined in [ISO/IEC 14443-3].

It is RECOMMENDED to use a random UID/PUPI to enhance the eMRTD holder’s privacy and to reduce the possibility of tracking.

B.9 (RECOMMENDED) RESONANCE FREQUENCY RANGE

There is no requirement on the resonance frequency, of eMRTD Applicants MAY limit the resonance frequency by default to a certain range to increase interoperability.

B.10 (RECOMMENDED) FRAME SIZES

The eMRTD MAY support frame sizes of up to 4 kbyte according to [ISO/IEC 14443]. However, it is RECOMMENDED to support frame sizes of at least 1 kbyte. If supporting frame sizes higher than 1 kbyte, the use of frames with error correction as defined in [ISO/IEC 14443-4] is RECOMMENDED.

Note.— A higher frame size substantially decreases the total processing time of an eMRTD application.

B.11 (RECOMMENDED) FRAME WAITING TIME INTEGER (FWI) AND S-BLOCK REQUEST FOR WAITING TIME EXTENSION [S(WTX)]

It is RECOMMENDED for the eMRTD to set an FWI value of less or equal to 11 in order to enhance performance. It is RECOMMENDED to use S(WTX) commands to extend the Frame Waiting Time for each particular command that requires additional time by using S(WTX) commands of an WTXM no greater than 10.

In case multiple S(WTX) requests are sent by the eMRTD, the total processing time for the current I-Block is RECOMMENDED to not exceed 5 s.

Note.— Lower FWI values as RECOMMENDED herein decrease the loss of time in transmission errors substantially, whereas S(WTX) are the ideal means of providing more time when needed.

— — — — — — — —

Appendix C to Part 10

INSPECTION SYSTEMS (INFORMATIVE)

C.1 OPERATING VOLUME AND TEST POSITIONS

An eMRTD associated Inspection System shall have an operating volume in accordance with one of the defined Inspection System types in [ISO/IEC 18745-2]. The operating volume is the volume in which all requirements of this technical report are fulfilled.

Note.— The test positions for each Inspection System Type are further specified in [ISO/IEC 18745-2] with respect to the (device) 0 mm surface of the eMRTD associated Inspection System.

C.2 PARTICULAR WAVEFORM AND RF REQUIREMENTS

The waveforms of the alternate magnetic field used to communicate shall be fully compliant with [ISO/IEC 14443-2]. In general, there are no exceptions or divergences from the basic standard, except for the field strength.

For eMRTD associated Inspection Systems of Type 1, 2 and 3, the field strength is RECOMMENDED to be at least two A/m at all positions for Class 1. For eMRTD associated Inspection Systems of Type M, the field strength shall be at least 1.5 A/m at all positions for Class 1.

Note.— It may be desirable for eMRTDs to also communicate with other contactless Inspection Systems and mobile devices, e.g. NFC smartphones use 1.5 A/m.

C.3 POLLING SEQUENCES AND eMRTD DETECTION TIME

The polling sequence of the eMRTD associated Inspection System shall provide 10 ms of unmodulated carrier before any REQA/WUPA or REQB/WUPB.

For fast detection and processing, the eMRTD Inspection System:

- Shall poll for Type A and Type B with an equal occurrence of requests for both Types;
- for Inspection System Types 1, 2 and 3, one RF reset should occur in between any REQ/WUP of the same type;
- Shall guarantee at least one polling command for both Type A and Type B within 150 ms for an eMRTD present in the minimum mandatory operating volume according [ISO/IEC 18745-2] at any position.

The eMRTD Inspection System MAY poll for contactless products of any other modulation type on the carrier of 13.56 MHz as long as all the requirements above are fulfilled.

Note.— The unmodulated carrier of 10 ms is required to detect all eMRTDs in the field and is based on former specifications.

C.4 MANDATORY BIT RATES

The eMRTD associated Inspection System shall provide mandatorily: 106 kbit/s and 424 kbit/s in both directions from the eMRTD to the eMRTD associated Inspection System and vice versa.

The bit rate of 212 kbit/s, and all bit rates from 848 kbit/s up to 6.78 Mbit/s for both directions and from 10.17 Mbit/s to 27.12 Mbit/s from eMRTD associated Inspection System to eMRTD as defined in [ISO/IEC 14443-2], are optional.

C.5 ELECTROMAGNETIC DISTURBANCE (EMD)

The support of EMD is not mandatory.

Note.— The EMD feature enhances the robustness of the contactless communication between the eMRTD and the eMRTD associated Inspection System against eMRTD generated electromagnetic disturbance. The eMRTD dynamic current consumption during execution of a command may cause an arbitrary load modulation effect (which may not be purely resistive) on the magnetic field. In some cases, the eMRTD associated Inspection System may misinterpret EMD as data sent by the eMRTD and this may negatively impact proper reception of the eMRTD response.

C.6 SUPPORTED ANTENNA CLASSES

The eMRTD associated Inspection System of Type 1 and Type 2 shall at least support Class 1 eMRTDs in the operating volume.

Class 2 and Class 3 are mandatory in ISO/IEC 14443, but optional for eMRTD Inspection System.

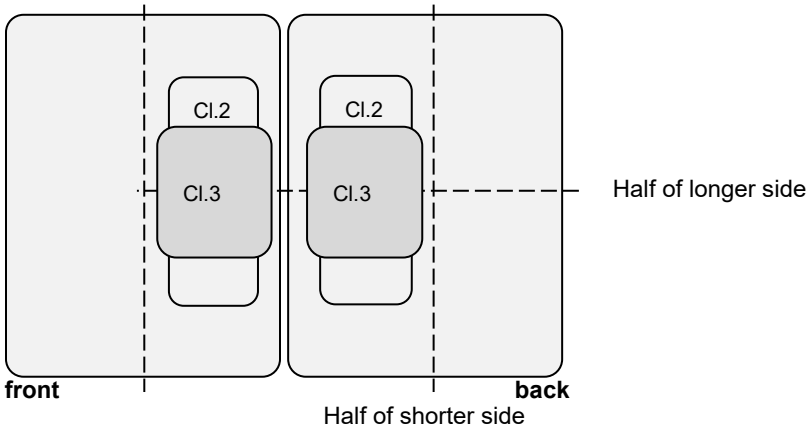


Figure C-1. Mandatory Positions in Each ID-3 Surface in which a Class 2 and Class 3 Antenna Shall be Read by an eMRTD Associated Inspection System of Type 1 and 2.

C.7 (OPTIONAL) FRAME SIZES AND ERROR CORRECTION

The eMRTD associated Inspection System MAY optionally support all frame sizes of up to 4 kbytes as defined in [ISO/IEC 14443-3]. It is RECOMMENDED to use frames with error correction as defined in [ISO/IEC 14443-3] for all supported frame sizes higher than 1 kbyte.

Note.— For eMRTD associated Inspection Systems of Type M, frame sizes higher than 256 bytes are currently not envisaged.

C.8 (OPTIONAL) SUPPORT OF ADDITIONAL CLASSES

eMRTD associated Inspection Systems of all Types MAY in addition support Class 4, Class 5 and Class 6 to be interoperable, for example, with mobile devices providing less coupling to the eMRTD associated Inspection System antenna coil.

C.9 (RECOMMENDED) OPERATING TEMPERATURE

It is RECOMMENDED that the eMRTD associated Inspection System works with temperatures of -10° to 50° Celsius.

C.10 (RECOMMENDED) SUPPORT OF MULTIPLE eMRTDS AND OTHER CARDS OR OBJECTS OR MULTIPLE HOSTS

It is highly RECOMMENDED to design the eMRTD associated Inspection System to handle more than one eMRTD, or one eMRTD and any other card or object compliant with [ISO/IEC 14443].

One of the following rules or a combination MAY be applied, among others:

- apply full anticollision algorithms defined in [ISO/IEC 14443-3];
- check for support of [ISO/IEC 14443-4] and dismiss all non-supporting cards;
- check for an eMRTD application; and
- use Card Identifier (CID) and Node Address (NAD).

Note.— NAD may be also used for mobile devices with multiple hosts.

C.11 (RECOMMENDED) FRAME SIZES

The eMRTD associated Inspection System MAY support frame sizes of up to 4 kbytes according to [ISO/IEC 14443-3]. However, it is RECOMMENDED to support frame sizes of at least 1 kbyte. If supporting frame sizes of 1 kbyte or higher, the use of frames with error correction as defined in [ISO/IEC 14443-4] is RECOMMENDED.

It is RECOMMENDED to perform any splitting of payload from the application layer into a minimum number of frames with an effective length of the maximum supported frame size with the exception of the last frame.

C.12 (RECOMMENDED) ERROR RECOVERY

Subsequent to a transmission error or an unresponsive eMRTD, it is RECOMMENDED for the eMRTD associated Inspection System to send a second R-block containing a negative acknowledgement (R(NAK)) according to the Inspection System rule 4 of [ISO/IEC 14443-4].

C.13 (RECOMMENDED) ERROR DETECTING AND RECOVERY MECHANISM

When using the optional bit rates as well as optional frame sizes higher than 256 bytes, in case of a higher than usual number of transmission errors, it is RECOMMENDED to reduce the bit rate and effective frame size.

— — — — — — — —

Appendix D to Part 10

DOCUMENT SECURITY OBJECT EF.SOD VERSION V0 LDS V1.7 (LEGACY) (INFORMATIVE)

The Document Security Object V0 for the LDS v1.7 does not contain the LDS and Unicode version information:

```
LDSSecurityObject ::= SEQUENCE {
   version LDSSecurityObjectVersion,
   hashAlgorithm DigestAlgorithmIdentifier,
   dataGroupHashValues SEQUENCE SIZE (2..ub-DataGroups) OF
   DataGroupHash}
```

D.1 SIGNEDDATA TYPE FOR SO_D V0

The Document Security Object is implemented as a SignedData Type, as specified in [RFC 3369]. All security objects SHALL be produced in Distinguished Encoding Rule (DER) format to preserve the integrity of the signatures within them.

Note 1.— m = MANDATORY — the field SHALL be present.

Note 2.— x = do not use — the field SHOULD NOT be populated.

Note 3.— o = optional — the field MAY be present.

Note 4.— c = choice — the field content is a choice from alternatives.

Table D-1. Signed Data Type for SO_D V0

Value		Comments
SignedData		
Version	m	Value = v3
digestAlgorithms	m	
encapContentInfo	m	
eContentType	m	id-icao-mrtd-security-ldsSecurityObject
eContent	m	The encoded contents of an ldsSecurityObject.
Certificates	o	States may choose to include the Document Signer Certificate (C_{DS}) which can be used to verify the signature in the signerInfos field.
Crls	x	It is recommended that States do not use this field.
signerInfos	m	It is recommended that States provide only 1 signerInfo within this field.
SignerInfo	m	
Version	m	The value of this field is dictated by the sid field. See RFC3369 Doc 9303-12 for rules regarding this field.
Sid	m	
issuerandSerialNumber	c	It is recommended that States support this field over subjectKeyIdentifier.
subjectKeyIdentifier	c	
digestAlgorithm	m	The algorithm identifier of the algorithm used to produce the hash value over encapsulatedContent and SignedAttrs.
signedAttrs	m	Producing States may wish to include additional attributes for inclusion in the signature, however these do not have to be processed by receiving States except to verify the signature value.
signatureAlgorithm	m	The algorithm identifier of the algorithm used to produce the signature value and any associated parameters.
Signature	m	The result of the signature generation process.
unsignedAttrs	o	Producing States may wish to use this field, but it is not recommended and receiving States may choose to ignore them.

D.2 ASN.1 PROFILE LDS DOCUMENT SECURITY OBJECT FOR SO_D V0

```
LDSSecurityObjectV0 {joint-iso-itu-t (2) international(23) icao(136) mrtd(1)
security(1) ldsSecurityObject(1)}

DEFINITIONS IMPLICIT TAGS ::=
BEGIN
```

```
IMPORTS

-- Imports from RFC 3280 [PROFILE],
AlgorithmIdentifier FROM
PKIX1Explicit88 { iso(1) identified-organization(3) dod(6)
internet(1) security(5) mechanisms(5) pkix(7)
id-mod(0) id-pkix1-explicit(18) };

-- Constants

ub-DataGroups INTEGER ::= 16

-- Object Identifiers
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-mrtd-security-ldsSecurityObject OBJECT   IDENTIFIER   ::=   {id-icao-mrtd-
security 1}

-- LDS Security Object

LDSSecurityObjectVersion ::= INTEGER {v0(0)}

DigestAlgorithmIdentifier ::= AlgorithmIdentifier

LDSSecurityObject ::= SEQUENCE {
   version LDSSecurityObjectVersion,
   hashAlgorithm DigestAlgorithmIdentifier,
   dataGroupHashValues SEQUENCE SIZE (2..ub-DataGroups) OF
   DataGroupHash }

DataGroupHash ::= SEQUENCE {
   dataGroupNumber         DataGroupNumber,
   dataGroupHashValue      OCTET STRING }
DataGroupNumber ::= INTEGER {
   dataGroup1               (1),
   dataGroup2               (2),
   dataGroup3               (3),
   dataGroup4               (4),
   dataGroup5               (5),
   dataGroup6               (6),
   dataGroup7               (7),
   dataGroup8               (8),
   dataGroup9               (9),
   dataGroup10             (10),
   dataGroup11             (11),
   dataGroup12             (12),
   dataGroup13             (13),
   dataGroup14             (14),
   dataGroup15             (15),
   dataGroup16             (16)}
END
```

Note 1.— The field `dataGroupHashValue` *contains the calculated hash over the complete contents of the Data Group EF, specified by* `dataGroupNumber`*.*

Note 2.— `DigestAlgorithmIdentifiers` *MUST omit* `NULL` *parameters, while the* `SignatureAlgorithmIdentifier` *(as defined in RFC 3447) MUST include* `NULL` *as the parameter if no parameters are present, even when using SHA2 Algorithms in accordance with RFC 5754. Inspection system MUST accept the field* `DigestAlgorithmIdentifiers` *with both conditions, i.e. absent parameters and* `NULL` *parameters.*

— — — — — — — —

Appendix E to Part 10

FILE STRUCTURES SUMMARY (INFORMATIVE)

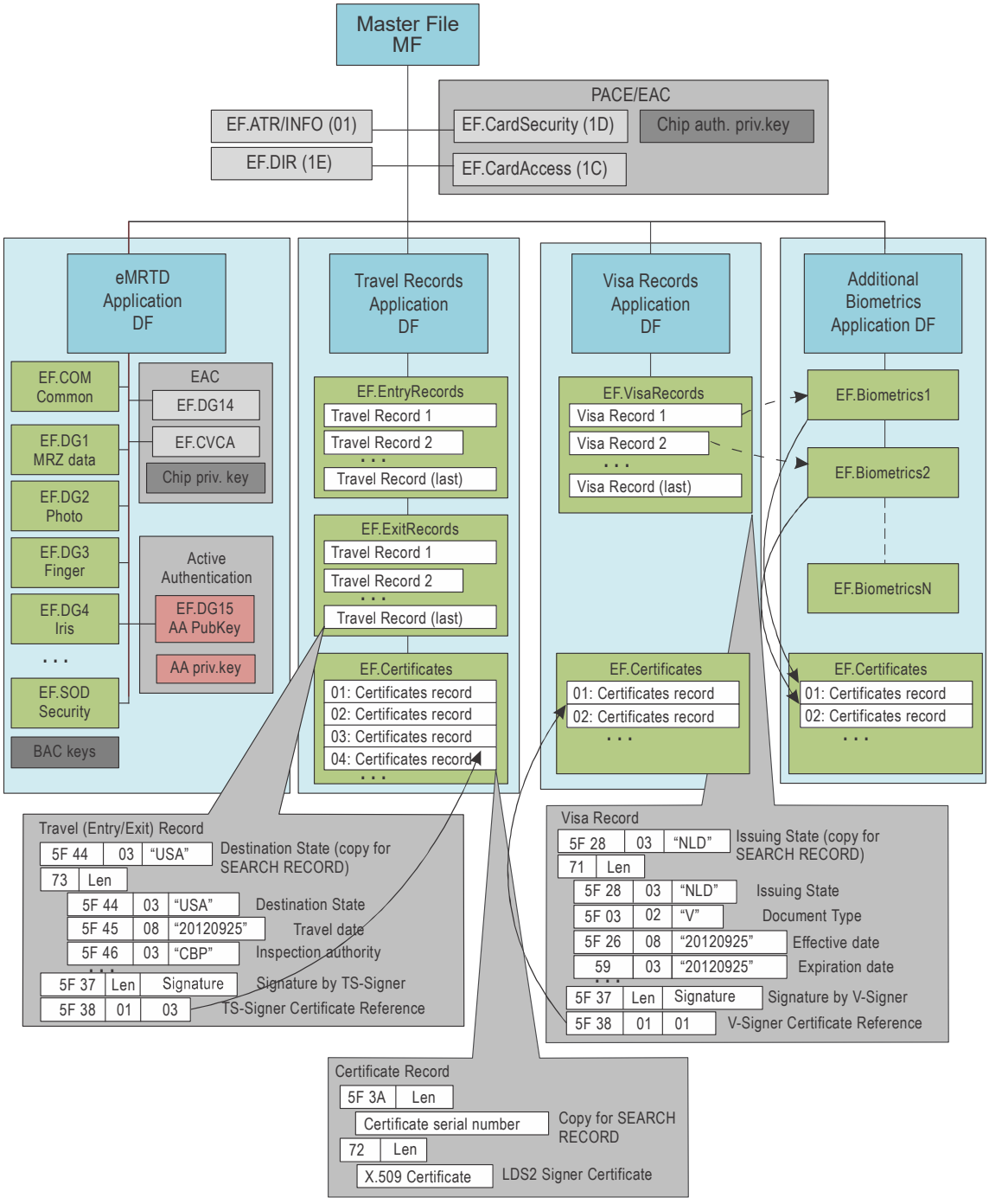


Figure E-1. File Structures Summary

— — — — — — — —

Appendix F to Part 10

LDS AUTHORIZATION SUMMARY (INFORMATIVE)

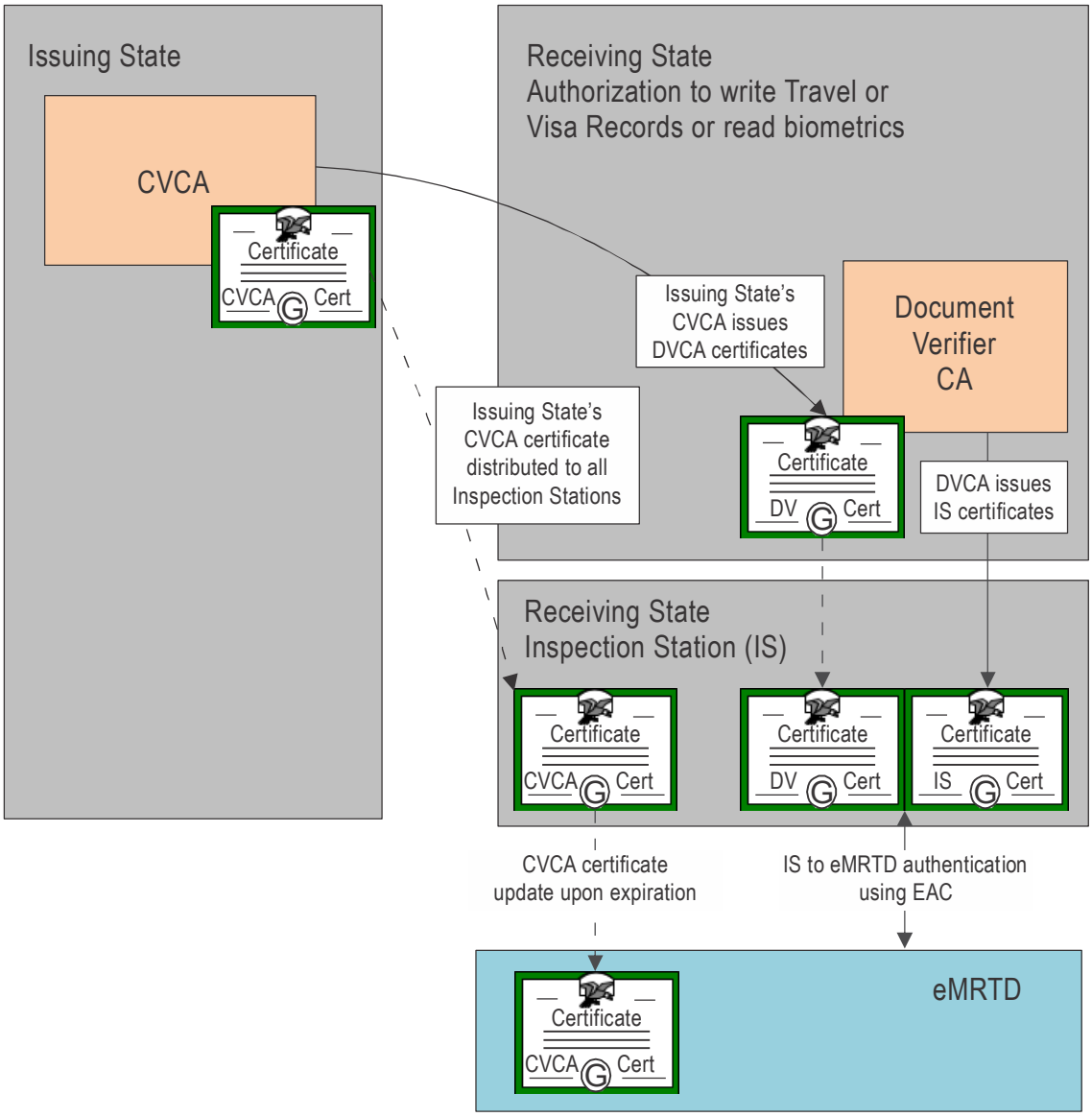


Figure F-1. LDS Authorization Summary

— — — — — — — —

Appendix G to Part 10

LDS DIGITAL SIGNATURE SUMMARY (INFORMATIVE)

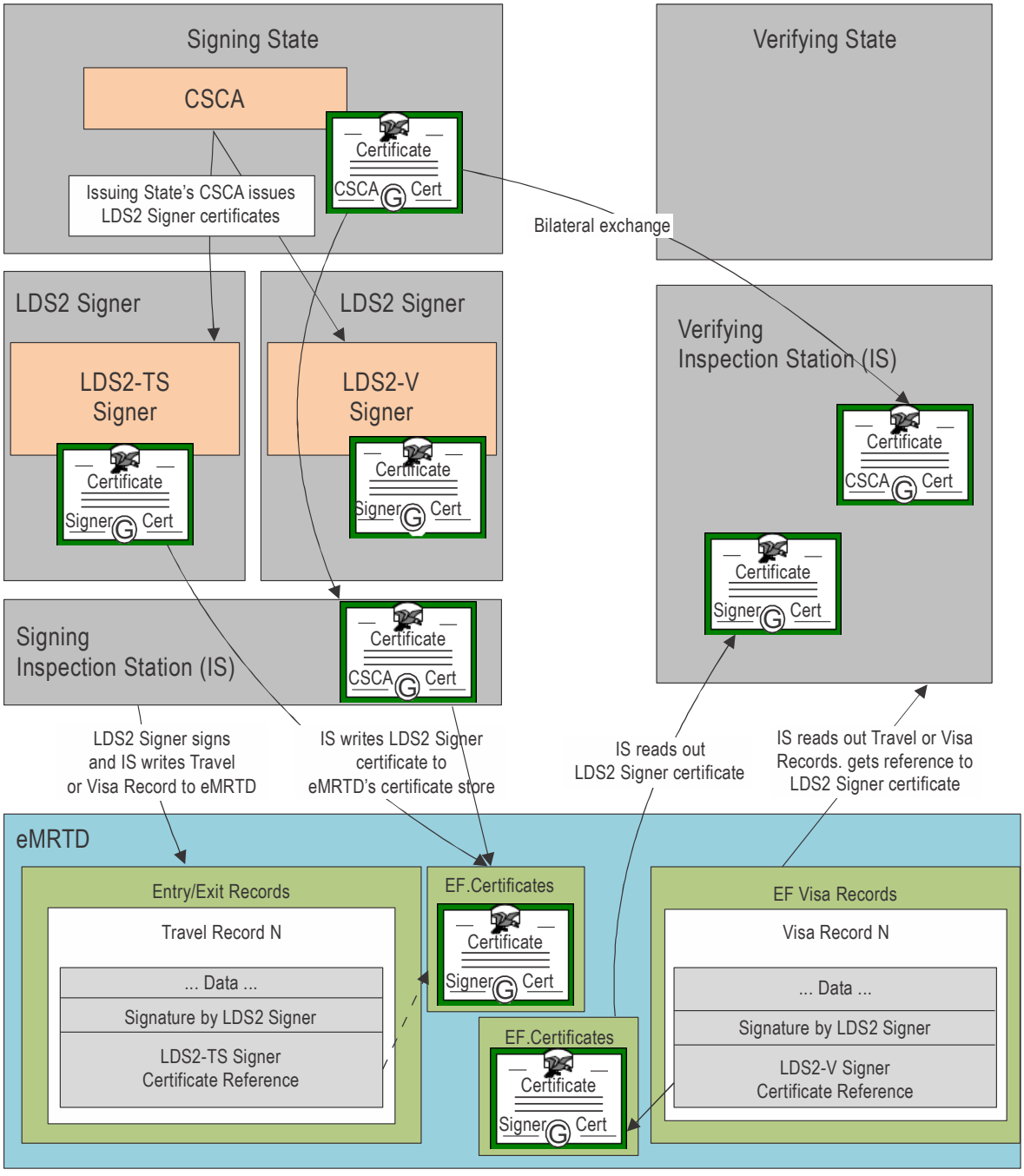


Figure G-1. LDS Digital Signature

— — — — — — — —

Appendix H to Part 10

EXAMPLE READING TRAVEL RECORDS (INFORMATIVE)

H.1 FMM COMMAND RETRIEVING THE NUMBER OF ENTRY RECORDS

CLA	INS	P1	P2	Lc	Data	Le
'80'	'5E'	'01'	'04'	'04'	'51 02 01 01'	'00'

CLA: Proprietary class / no secure messaging
INS: FMM
P1: '01' — EF identifier in command data field
P2: '04' — Return existing number of records in a record EF
Lc: '04'
Data: DO'51' containing Entry Records EF identifier '0101'
Le: '00' (Short Le)

Response: FILE AND MEMORY MANAGEMENT DO representing the number of records in the EF.

Data	SW1-SW2
'7F78 03' '83 01 FD'	'90 00'

The DO in the response data contains the last record number which can be used in the next READ RECORD command (P1).

For example, the last record number '00' means that there are no records in this file, and the response 'FD' means that the number of records is 253 (the maximum number of records is 254).

H.2 READ RECORD COMMAND RETRIEVING THE LAST TRAVEL RECORD FROM THE RETRIEVED LIST

The following command can be used to retrieve a single record using the record number returned by the FMM command:

CLA	INS	P1	P2	Le
'00'	'B2'	'FD'	'04'	'00 00 00'

CLA: Interindustry class / no secure messaging
INS: READ RECORD(S)
P1: Record number from the previous command's response
P2: Record number in P1 / read record P1
Le: '00 00 00' (Extended Le), read entire record

Response: The record number is 253 ('FD').

Data	SW1-SW2
'5F44' 'Len' <Data> || '73' 'Len' <Data> || '5F37' 'Len' <Data> || '5F38' 'Len' <Data>	'90 00'

H.3 READ RECORD COMMAND RETRIEVING THE LAST TWO TRAVEL RECORDS FROM THE RETRIEVED LIST

The following command can be used to retrieve two (or more) records from the list returned by the FMM command. Reading several records in one APDU exchange improves performance. The number of records that can be retrieved by a single command can be determined from extended length information in EF.ATR/INFO and the maximum size of the Travel Record.

CLA	INS	P1	P2	Le
'00'	'B2'	'FC'	'05'	'00 00 00'

CLA: Interindustry class / no secure messaging
INS: READ RECORD(S)
P1: Decremented Record number from the FMM response (253 - 1 = 252 = 'FC')
P2: Record number in P1 / read all records from P1 up to the last record
Le: '00 00 00' (Extended Le), read entire record

Response: The last two records 252 ('FC') and 253 ('FD') are returned.

Data	SW1-SW2
'5F44' 'Len' <Data> || '73' 'Len' <Data> || '5F37' 'Len' <Data> || '5F38' 'Len' <Data> || '5F44' 'Len' <Data> || '73' 'Len' <Data> || '5F37' 'Len' <Data> || '5F38' 'Len' <Data>	'90 00'

— — — — — — — —

Appendix I to Part 10

EXAMPLE SEARCHING RECORDS BY STATE (INFORMATIVE)

I.1 SEARCH RECORD COMMAND SEARCHING TRAVEL RECORD(S) BY DESTINATION STATE

CLA	INS	P1	P2	Lc	Data	Le
'00'	'A2'	'00'	'F8'	Var	'7F 76' 'Len' '51 01 01' 'A1 0B' '80 01 00' 'B0 06' '02 01 03' '02 01 03' 'A3 07' 'B1 05' '81 03' xx xx xx	'00'

CLA: Interindustry class / no secure messaging
INS: SEARCH RECORD(S)
P1: record number = '00'
P2: Search through multiple EFs
Lc: length of command data field
Data: DO'7F76' - Record handling DO
DO'51' - File reference DO (EF.EntryRecords short identifier '01')
DO'A1' - Search configuration template
DO'80' - Search configuration parameter: '00' (search all records)
DO'B0' - Search window template
DO'02' - Offset: '03'
DO'02' - Number of bytes: '03'
DO'A3' - Search string template
DO'B1' - Search string DO
DO'81' - Search string (country code): xx xx xx
Le: '00' (Short Le)

Response: DO'7F76' – Record handling DO
DO'51' - EF.EntryRecords short identifier '01'
One or more DO'02' containing matching record numbers

Data	SW1-SW2
'7F 76' 'Len'' '51 01 01' '02 01 03' '02 01 04'	'90 00'

— — — — — — — —

Appendix J to Part 10

EXAMPLE WRITING TRAVEL RECORD AND CERTIFICATE (INFORMATIVE)

J.1 SEARCH RECORD COMMAND SEARCHING EF.CERTIFICATES BY A CERTIFICATE SERIAL NUMBER

IS checks if LDS2-TS Signer certificate with required serial numbers exists in EF.Certificates. The following command can be used for searching certificates:

CLA	INS	P1	P2	Lc	Data	Le
'00'	'A2'	'00'	'F8'	Var	'7F 76' 'Len' '51 01 1A' 'A1 0B' '80 01 30' 'B0 06' '02 01 03' '02 01' {Search string size} 'A3' 'Len' 'B1' 'Len' '81' 'Len' xx xx .. xx xx	'00'

CLA: Interindustry class / no secure messaging INS: SEARCH RECORD(S)
P1: record number = '00'
P2: Search through multiple EFs
Lc: length of command data field
Data: DO'7F76' - Record handling DO
DO'51' - File reference DO (EF.Certificates short identifier '1A')
DO'A1' - Search configuration template
DO'80' - Search configuration parameter: '30' (stop if record found)
DO'B0' - Search window template
DO'02' - Offset: '03'
DO'02' - Number of bytes: Search string size
DO'A3' - Search string template
DO'B1' - Search string DO
DO'81' - Search concatenation of country code and certificate serial number: xx xx .. xx xx
Le: '00' (Short Le)

Response: DO'7F76' - Record handling DO
DO'51' - EF.Certificates short identifier '1A'
DO'02' - contains matching record number

Data	SW1-SW2
'7F 76 06' '51 01 1A' '02 01 01'	'90 00'

or warning code '62 82' if no record matches the search criteria:

SW1-SW2
'62 82'

If an EF.Certificate record matches the search criteria, the IS can optionally use the returned record number ('01') in a READ RECORD command to check whether the certificate is the correct one. If no EF.Certificate record matches the search criteria, the IS writes the certificate into EF.Certificates using the APPEND RECORD command in Section J.2 and finally writes the entry record using the APPEND RECORD command in Section J.3.

J.2 APPEND RECORD COMMAND WRITING CERTIFICATE

IS writes LDS2-TS Signer certificate into EF.Certificates. The following command can be used for writing certificates:

CLA	INS	P1	P2	Lc	Data	Le
'00'	'E2'	'00'	'D0'	'00' XX XX	'5F3A' 'Len' {certificate serial number} || '72' 'Len' {X.509 certificate}"	Absent

CLA: Interindustry class / no secure messaging
INS: APPEND RECORD
P1: '00' (any other value is invalid)
P2: short EF identifier (= '1A')
Lc: Record length (Extended Lc)
Data: Record data

Response: success or error code

SW1-SW2
'90 00'

J.3 APPEND RECORD COMMAND WRITING TRAVEL RECORD

IS generates Travel Record using reference to LDS2-TS Signer certificate and writes it into EF.EntryRecords using the following command:

CLA	INS	P1	P2	Lc	Data	Le
'00'	'E2'	'00'	'08'	'00' XX XX	'5F44' 'Len' {destination state} || '73' 'Len' {Entry travel record} || '5F37' 'Len' {Signature} || '5F38' 'Len' {Cert Ref}	Absent

CLA: Interindustry class / no secure messaging
INS: APPEND RECORD
P1: '00' (any other value is invalid)
P2: short EF identifier (= '01')
Lc: Record length (Extended Lc)
Data: Record data

Response: success or error code

SW1-SW2
'90 00'

— END —

ISBN 978-92-9275-420-4