**Memory Stucture and Security Mechanisms for Machine Readable Travel Documents**

*Abstract: Machine Readable Travel Documents (MRTDs), including eID and ePassPort/eVisa, are standard specifications developed by International Civil Aviation Organization (ICAO) which aim at facilitating the process of personal identity verification during border clearance as well as other official transactions with high level of security requirements. This paper will provide general information about Memory Stucture and Security Mechanisms of electronic Machine Readable Travel Document (eMRTD) application, implementing on a JavaCard.*

**I. Memory Structure:**

A summary of memory structure of multiple and single eMRTD application on a JavaCard can be seen on Fig. 1 and Fig. 2 respectively. In general, only LDS1 eMRTD application is mandatory for all types of electronic travel document such as eID or ePassPort. All other LDS2, such as Travel Records application, Visa Records application and Additional Biometricss application; are optional which might be installed to provide additional functions.

A picture containing diagram

Description automatically generated

Figure 1. Memory structure of Multi-Application eMRTD

***(Reproduced from ICAO Doc 9303 Machine Readable Travel Documents Part 10)***

Diagram

Description automatically generated

Figure 2. eMRTD File Structure Summary

***(Reproduced from ICAO Doc 9303 Machine Readable Travel Documents Part 10)***

In the scope of this paper, we will focus on memory structure of LDS1 eMRTD application (Fig. 1). There are two groups of elementary files (EFs), including common EFs, which available are conditional; and LDS1 eMRTD EFs. A summary information about some common and LDS1 eMRTD EFs, which require for security mechanisms, can be seen on Table 1 below:

|  |  |  |
| --- | --- | --- |
| Common EFs | Available | Notes |
| EF.ATR/INFO | Conditional | - Contain in the master file, REQUIRED if the optional LDS2 application is present. This EF is optional if only LDS1 application is present.  - Indicate operating characteristics of the card. |
| EF.DIR | Conditional | - Contain in the master file, REQUIRED if optional LDS2 applications are present (EF.DIR MUST be included in SecurityInfos present in EF.CardSecurity).  - Indicate a list of applications supported by the eMRTD. |
| EF.CardAccess | Conditional | - Contain in the master file, REQUIRED if the optional PACE access control as defined in Doc 9303-11 is invoked.  - Include SecurityInfos that are required for PACE: PACEInfo and PACEDomainParameterInfo. |
| EF.CardSecurity | Conditional | - Contained in the master file, REQUIRED if :  • PACE with Chip Authentication Mapping is supported by the IC; (1)  • Terminal Authentication in the MF is supported by the IC; (2) or  • Chip Authentication in the MF is supported by the IC. (3)  - and MUST include:  • ChipAuthenticationInfo as required by Chip Authentication; if (1)  • ChipAuthenticationPublicKeyInfo as required by PACE-CAM/Chip Authentication; if (2)  • TerminalAuthenticationInfo as required by Terminal Authentication; if (3)  • the SecurityInfos contained in EF.CardAccess. if (2) |
| eMRTD EFs |  | Located in the LDS1 eMRTD application |
| EF.COM | Mandatory | - LDS1 eMRTD application MUST have only one file EF.COM that contains LDS version information, Unicode version information and a list of the Data Groups that are present for the application. |
| EF.SOD | Mandatory | - Contain a Document Security Object, is digitally signed by the issuing State and contains hash values of the LDS contents. |
| EF.DG14 | Conditional | - Contain security options for additional security mechanisms, is REQUIRED if Chip Authentication or PACE-GM/-IM is supported by the eMRTD chip. |
| EF.DG15 | Conditional | - Contain the Active Authentication Public Key, REQUIRED when implementing the optional Active Authentication chip authentication as described in Doc 9303-11. |
| *Please refer for ICAO DOC 9303 Part 10 for detail information regarding other Mandatory and Conditional EFs of* *LDS1 eMRTD application.* | | |

**II. Security Mechanisms for LDS1 eMRTD application**

Accessing to an eMRTD application should follow a security mechanisms:

+ Gain access to the contactless IC of the eMRTD: BAC/PACE Authentication.

+ Authentication of data: Passive Authentication.

+ Authentication of the chip: Active Authentication, Chip Authentication.

+ Additional access control mechanisms: Terminal Authentication, Data Encryption.

+ Reading data.

As can be seen from Table 1, security mechanisms of LDS1 eMRTD application should require security information (SecurityInfo) from different EFs, depending on the authentication mode supported by card (NFC chip) and terminal.

It should be noted that, according to ICAO from January 2018, all eMRTD chips will only implement PACE authentication so EF.CardAccess should be available on all new chip. As a result, older chips (made before 2018) which only support BAC and do not contain EF.CardAccess might become unreadable by the new terminal that only supports PACE. Therefore, in case of an unreadable card, some steps could be performed to verify as follow:

+ Send command APDU (CAPDU) to select eMRTD application to check if the application exists, a negative response was returned means the card might not be an eMRTD document.

+ Send command APDU (CAPDU) to select EF.CardAccess file to check if it exists, a negative response was returned means the card not support PACE and a terminal that support BAC could be used to check.

In addition, within terminal software code, an “exception trap” should be implemented during the process of getting SecurityInfo from EF.CardAccess and EF.CardSecurity files for PACE, which can give a warning if any error occurs, for the purpose of debugging later.

In rare cases, the possibility of card is unreadable due to unaccessible of EF.CardAccess file might be caused by unexpected errors of the card memory. As with any other electronic device, the memory of the card also has a limited lifespan, depending on a lot of factors such as operating environment (Moisture, Temperature, Chemical exposure (solvents, acids, cleaning agents, oils…) or UV light exposure), read/write cycle (threshold is about 100000 cycles) as well as physical interaction (such as Abrasion). So, to avoid that, it is recommended to avoid operating the card in an extreme environment, or overuse when continuously accessing the card for a long period of time.