# EMV®
# Level 3 (L3) Testing Framework

# Pseudo-function Definitions for Test Card Images

Version 1.4

April 2019

# Legal Notice

This document summarizes EMVCo's present plans for evaluation services and related policies and is subject to change by EMVCo at any time. This document does not create any binding obligations upon EMVCo or any third party regarding the subject matter of this document, which obligations will exist, if at all, only to the extent set forth in separate written agreements executed by EMVCo or such third parties. In the absence of such a written agreement, no product provider, test laboratory or any other third party should rely on this document, and EMVCo shall not be liable for any such reliance.

No product provider, test laboratory or other third party may refer to a product, service or facility as EMVCo approved, in form or in substance, nor otherwise state or imply that EMVCo (or any agent of EMVCo) has in whole or part approved a product provider, test laboratory or other third party or its products, services, or facilities, except to the extent and subject to the terms, conditions and restrictions expressly set forth in a written agreement with EMVCo, or in an approval letter, compliance certificate or similar document issued by EMVCo. All other references to EMVCo approval are strictly prohibited by EMVCo.

Under no circumstances should EMVCo approvals, when granted, be construed to imply any endorsement or warranty regarding the security, functionality, quality, or performance of any particular product or service, and no party shall state or imply anything to the contrary. EMVCo specifically disclaims any and all representations and warranties with respect to products that have received evaluations or approvals, and to the evaluation process generally, including, without limitation, any implied warranties of merchantability, fitness for purpose or non-infringement. All warranties, rights and remedies relating to products and services that have undergone evaluation by EMVCo are provided solely by the parties selling or otherwise providing such products or services, and not by EMVCo, and EMVCo will have no liability whatsoever in connection with such products and services.

This document is provided "AS IS" without warranties of any kind, and EMVCo neither assumes nor accepts any liability for any errors or omissions contained in this document. EMVCO DISCLAIMS ALL REPRESENTATIONS AND WARRANTIES, EXPRESS OR IMPLIED, INCLUDING WITHOUT LIMITATION IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT, AS TO THIS DOCUMENT.

EMVCo makes no representations or warranties with respect to intellectual property rights of any third parties in or in relation to this document. EMVCo undertakes no responsibility to determine whether any implementation of this document may violate, infringe, or otherwise exercise the patent, copyright, trademark, trade secret, know-how, or other intellectual property rights of third parties, and thus any person who implements any part of this document should consult an intellectual property attorney before any such implementation.

Without limiting the foregoing, this document may provide for the use of public key encryption and other technology, which may be the subject matter of patents in several countries. Any party seeking to implement this document is solely responsible for determining whether its activities require a license to any such technology, including for patents on public key encryption technology. EMVCo shall not be liable under any theory for any party's infringement of any intellectual property rights in connection with this document.

# Revision Log

| Version | Date | Description |
|---------|------|-------------|
| V1.0 | March 29th, 2017 | First public release of document |
| V1.1 | March, 08th, 2018 | • Update "Location" information for Pseudo functions that may be located in Format 1 template ID="80"<br>• Add emvcard.msd() pseudo function |
| | May, 03rd, 2018 | • Clarify emvcard.arqc<br>• Use emvcard.cvc3t1() and emvcard.cvc3t2() instead of emvcard.cvc3() twice<br>• Add emvcard.dcvv()<br>• Add emvcard.cpr()<br>• Prescribe the use of an initial value for emvcard.ctq() and emvcard.cpr() |
| V1.2 | June, 12th, 2018 | • Add specification for emvcard.sdad() pseudo function<br>• Locate in the document all the definitions from the MP50 Card Image |
| V1.3 | Nov, 30th, 2018 | • Add emvcard.track2()<br>• Update emvcard.ctq() and emvcard.cpr(). Precise Initial value calculation<br>• Update emvcard.UN() definition<br>• Update emvcard.atc() definition |
| V1.4 | April 11th, 2019 | • Update following TA review |

# Contents

# 1 Executive Summary

EMVCo has defined guidelines and specifications that collectively help to facilitate a standardized approach to the development, qualification and usage of test tools required by financial institution clients and their service providers to perform Level 3 terminal integration testing on EMV contact and contactless acceptance devices. A key component of these deliverables is the *EMVCo Level 3 Testing Framework – Implementation Guidelines [L3FIG]* that provides its targeted audiences with specific implementation details and instructions for creating and supporting various technical components of the *L3 Framework*. These components include:

- Machine-readable L3 Test Card image format - representing the expected card behaviours for each set of Payment System test card images.

- Test Set files for the Test Selection Engine (TSE) component – defining the methodology for test case selection, questions to be asked by the TSE, errors to be reported by the TSE in exception cases, Pass/Fail Criteria definitions, etc.

- Test Session file – generated by the TSE following user entry, and used by the TSE to provide instruction to the Test Tool engine on which cases are to be executed.

- Test Reporting and Logging formats – including the Card to Terminal Log and the Online Message Log

Within the machine-readable L3 Test Card image format, specified in Extensible Mark-up Language (XML), various pseudo-functions are used to address card images behaviours that cannot be easily deduced from the image content alone. This document, the *EMVCo Level 3 Testing Framework – Pseudo-function Definitions for Test Card Images [L3PSEU],* includes the list of currently defined pseudo-functions. It is a companion document to the *EMVCo Level 3 Testing Framework – Implementation Guidelines [L3FIG]*.

# 2    Purpose and Scope

Pseudo-function definitions may be required either to specify card characteristics & behaviours that otherwise cannot be deduced from the card image content alone, or to highlight specific areas for usability reasons. The definitions may, for example, include details of the cryptographic functions used by or within a Payment System's products.

**Note that** in some cases, it may be sufficient just to specify the Cryptogram Version Number (CVN) in the image and not require the use of any pseudo-functions for this capability.

## 2.1    Purpose

This document, the *EMVCo Level 3 Testing Framework – Pseudo-function Definitions for Test Card Images [L3PSEU],* aims at providing the EMVCo L3 stakeholders the appropriate level of information and directives to prepare and implement the EMVCo L3 Card Image machine-readable files. It is intended to be a companion document to the *EMVCo Level 3 Testing Framework – Implementation Guidelines (L3FIG),* and provides its targeted audiences with a list of the currently defined Pseudo-function definitions.

## 2.2    Intended Audiences

The intended audiences of this document and its companion, the *EMVCo Level 3 Testing Framework – Implementation Guidelines (L3FIG),* are:

- Payment Systems – preparing the L3 Test Card Images for delivery to L3 Test Tool suppliers
-  L3 Test Tool supplier – developing compilers and interpreters within their tools to process L3 Test Card Images delivered by Payment Systems
- Financial Institution clients and their service providers – utilizing L3 test tools to perform L3 terminal integration testing.

# 3 Pseudo-function Definitions

The table below includes a list of currently defined pseudo-function definitions for use with the Machine-readable Test Card Images, as specified in Section 3.5 of the *EMVCo Level 3 Testing Framework – Implementation Guidelines [L3FIG]* document.

| Pseudo function Name | TAG | Description | Length | Format | Location | Reference |
|---|---|---|---|---|---|---|
| **simvendor.id()** | 5F 50 | Issuer Uniform resource locator (URL). The URL provides the location of the Issuer's Library Server on the Internet | var | ans | FCI Discretionary Template" ID=BF 0C" | Underlying Payment System's specification |
| **emvcard.arqc()** | 9F 27 | Cryptogram Information Data (CID). GenAC1: Signifies ARQC in case the terminal does not request an AAC and is not offline only TC in case the terminal does not request an AAC and is offline only AAC in case the terminal requests an AAC | 1 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | EMV Book 3 |
| **emvcard.aac()** | 9F 27 | Cryptogram Information Data (CID). GenAC1: Signifies AAC always GenAC2: Signifies AAC always | 1 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | EMV Book 3 |

| Pseudo function Name | TAG | Description | Length | Format | Location | Reference |
|---|---|---|---|---|---|---|
| **emvcard.term()** | 9F 27 | Cryptogram Information Data (CID).<br><br>GenAC1: Signifies follow the terminal request<br><br>GenAC2: Signifies follow the terminal request and don't check issuer auth result | 1 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | EMV Book 3 |
| **emvcard.auth()** | 9F 27 | Cryptogram Information Data (CID).<br><br>GenAC2: Signifies follow the terminal request but decline if the Issuer auth failed (Tag 91 or External auth) | 1 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | EMV Book 3 |
| **emvcard.atc()** | 9F 36 | Application Transaction Counter (ATC) maintained by the L3 CS.<br><br>• emvcard(): Any value<br><br>• emvcard(xx): Fixed value xx | 2 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | Not applicable |
| **emvcard.appcrypto()** | 9F 26 | Application Cryptogram. Cryptogram always returned by the card in response of the GENERATE AC command (even if the terminal required a CDA) | 8 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | Underlying Payment System's specification |

| Pseudo function Name | TAG | Description | Length | Format | Location | Reference |
|---|---|---|---|---|---|---|
| **emvcard.sdad()** | 9F 4B | Signed Dynamic Application Data. Digital signature on critical application parameters for DDA or CDA<br><br>GenAC: emvcard.sdad() is used when CDA supported. There are 3 cases:<br><br>• CDA requested specified by p1 and format 2 is required<br>• P1 is not specified:<br><br>1. 9F4B      if P1=10<br><br>2. 9F4B    if (P1=50 or 90) and (GenAC instance = 1 or CID <> 00??????b)<br>3. 9F26       otherwise | var | Binary | Internal Auth Response Message Template Format 1 ID="80" or Format 2 ID="77"<br><br>GenAC Response Message Template Format 2" ID="77" | Underlying Payment System's specification |
| **emvcard.cvc3t1()** | 9F 61 | CVC3 dynamic value. The CVC3 (Track1) is a 2-byte cryptogram returned by the Card in the response to the COMPUTE CRYPTOGRAPHIC CHECKSUM command | 2 | Binary | ComputeCryptographicChecksum Response Message Template format 2" ID="77" | Underlying Payment System's specification |
| **emvcard.cvc3t2()** | 9F 62 | CVC3 dynamic value. The CVC3 (Track2) is a 2-byte cryptogram returned by the Card in the response to the COMPUTE CRYPTOGRAPHIC CHECKSUM command | 2 | Binary | ComputeCryptographicChecksum Response Message Template format 2" ID="77" | Underlying Payment System's specification |
| **emvcard.iad (*length*)** | 9F 10 | Issuer Application Data. Contains proprietary application data for transmission to the Issuer. *length* indicates the number of bytes | Up to 32 | Binary | GenAC Response Message Template Format 1 ID="80" or Format 2 ID="77" | Underlying Payment System's specification |

| Pseudo function Name | TAG | Description | Length | Format | Location | Reference |
|---|---|---|---|---|---|---|
| **emvcard.ctq(***initv alue***)** | 9F 6C | Card Transaction Qualifiers. Indicate to the device the card CVM requirements, issuer preferences, and card capabilities.<br><br>The emvcard.ctq function calculates the final CTQ based upon the initial value specified as parameter<br><br>•　　At the beginning of the transaction the CTQ is set to the initial value with B1b7,8 reset<br><br>•　　If the terminal supports PIN and requests CVM and B1b8 of the initial value is set then set CTQ B1b8<br><br>•　　Otherwise, if the terminal supports signature and requests CVM<br><br>and B1b7 of the initial value is set then set CTQ B1b7 | 2 | Binary | GPO Response Message Template Format 2" ID="77" | Underlying Payment System's specification |

| Pseudo function Name | TAG | Description | Length | Format | Location | Reference |
|---|---|---|---|---|---|---|
| **emvcard.cpr(*initvalue*)** | | Card Processing Requirement<br><br>The emvcard.cpr function calculates the final CPR based upon the initial value specified as parameter<br><br>• At the beginning of the transaction the CPR is set to the initial value with B1b7,8 reset<br><br>• If the terminal supports PIN and requests CVM and B1b8 of the initial value is set then set CPR B1b8<br><br>• Otherwise, if the terminal supports signature and requests CVM and B1b7 of the initial value is set then set CPR B1b7 | 2 | Binary | GPO Response Message Template Format 2" ID="77" | Underlying Payment System's specification |
| **emvcard.UN(length)** | 9F 7F | Contains the Card challenge (random), obtained in the response to the GET PROCESSING OPTIONS command<br><br>The length parameter indicates the number of bytes of the unpredictable number. | var | Binary | GPO Response Message Template Format 2" ID="77" | Not Applicable |
| **emvcard.msd()** | 57 | Part of the Track2 Equivalent Data | 7 | nibbles | Read Record Response Message Template ID="70" | Underlying Payment System's specification |
| **emvcard.dcvv()** | 9F 7E | Dynamic CVV value in Discover Contactless MS Mode | 3 | Binary | Read Record Response Message Template ID="70" | Underlying Payment System's specification |

| Pseudo function Name | TAG | Description | Length | Format | Location | Reference |
|---|---|---|---|---|---|---|
| **emvcard.track2 (track_2_EMV, track_2_MS)** | 57 | Return the Track 2 Equivalent Data that corresponds to the transaction Mode.<br><br>If the GPO Command indicates EMV Mode (Byte 1 Bit 8 of tag '9F35' is set to '1'), then the value 'track_2_EMV' will be return in tag '57'.<br><br>Otherwise, if the GPO command indicates Mag Stripe Mode (Byte 1 Bit 8 is set to '0'), then the value 'track_2_MS' will be returned in tag '57'. | var | Binary | Read Record Response Message Template ID="70 | Underlying Payment System's specification |

**\*\*\* END OF DOCUMENT \*\*\***