80161508-001

MiniSmart II
Secure Smart Card Reader
Command Interface

August 2, 2019
2018 Rev. C

ID TECH
10721 Walker Street, Cypress, CA90630 Voice: (714) 761-6368 Fax: (714) 761-8880

ID TECH
10721 Walker St.
Cypress, CA 90630

support@idtechproducts.com Visit: http://www.idtechproducts.com

Contents

Introduction

The MiniSmart II is ID TECH's latest EMV-compliant ICC module. The MiniSmart II, certified for EMV Level 1 and Level 2, combines a proven EMV kernel with a compact 33mm x 66mm form factor and supports USB, UART, and RS-232 communication, allowing for smooth integration into a variety of payment environments.

With EMV acceptance now an international payment standard, the MiniSmart II fills a need for a compact, reliable, EMV-ready device that can serve a wide variety of space-limited applications.

This document describes the low-level command API for the MiniSmart II, for integrators who wish to send commands (and receive responses) to/from the device via serial communications. ID TECH also offers (separately) a high-level-language SDK (the "Universal SDK") for programmers who wish to target a Windows platform using C# or Android via Java. The SDK provides a variety of convenience methods and high-level-langauge wrappers around most of the low-level commands described in this document. Contact your ID TECH representative to learn more about the SDK, which is free for all customers. Also, be sure to check the ID TECH public Knowledge Base for additional documentation, utilities, demos, and software updates:
https://atlassian.idtechproducts.com/confluence/display/KB/MiniSmart+II+-+downloads

USB-HID Interface

VID is 0xACD.
PID is 0x3410.
Reports: 64 bytes
Report type: 0
Report[0] = report length
Report[1-63] = data

If more reports to follow, Report[0]=0xBF, data length = 63.

Command/Response format

MiniSmart II uses a simple command/response protocol based on the following packaging of messages:

NGA Protocol Format

```
<STX><CLenL><CLenH><Command_Body/Response_Body><CheckLRC><CheckSum><ETX>
```

Where:
<STX> = 0x02
<CLenL> = length, low (least significant) byte
<CLenH> = length, high byte

<LRC> = Overall LRC (Modulus 2 Exclusive OR) of Command/Response Body; XOR all bytes together (as shown below).
<Checksum> = The overall sum of bytes from Command/Response Body (neglecting overflow); add all bytes and take the low 8 bits.
<ETX> = 0x03

Example:

To get the firmware version, send
```
02 03 00 78 46 01 3F BF 03
```

The command is 78 46 01. The length is 3 bytes (note the little-endian LSB/MSB format, hence the length is given as 03 00). The LRC is 3F; the checksum is BF. This command can be sent in a USB-HID Report 0 of length 64, with the first byte of the report set to the length of the data being sent.

The USB-HID response from the MiniSmart II is:

```
21 02 1b 00 06 49 44 20 54 45 43 48 20 4d 69 6e 69 53 6d 61 72 74 20 49
49 20 56 32 2e 30 30 21 73 03 00 00 00 00 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00
```

The report's first byte (21, in this example) is the length of the actual data.

Calculation of LRC:

```
public static byte[] getLRC(byte[] data)
{
   byte[] lrc = new byte[1];
   lrc[0] = data[0];
   for (int i = 1; i < data.Length; i++)
      lrc[0] ^= data[i];
   return lrc;
}

// Checksum is the same as LRC except it uses addition:
//      lrc[0] += data[i];
// The return value should be constrained to 8 bits (lrc & 255).
```

Commands

The following table lists the commands supported by the device, in alphabetical order (by command name). Documentation of individual commands begins after the table.

Command Summary Table (Alphabetical by Command Name)

Command	Name
72 46 05 02	Authenticate Transaction
72 46 06	Cancel Transaction
72 46 05 03	Complete Transaction
7F 53 00	Default All
72 53 00	Default ICC Group All Settings
70 53 00	Default RS232 Group
78 46 F5	Device Enter into Load Important Data State
78 46 7A 49 52 46 57 00 00 00 00 00 00 00	Enter into Bootloader
78 46 F4	Erase Keys
78 46 C3 49 44 54 45 43 48	Erase Serial Number
72 46 41	Exchange APDU Plaintext
72 52 01 21	Get Ascii Mask Data
70 52 01 41	Get Baud Rate
72 52 01 22	Get BCD Mask Data
78 46 11	Get Bootloader FW Release Version (Internal Command)
72 52 01 04	Get Card Type Option
70 52 01 42	Get Data Bits
78 52 01 50	Get Date and Time
78 46 3E	Get DUKPT Key KSN
72 46 23 01	Get EMV Level One Version Number
72 46 08 01	Get EMV Level Two Version Number
72 52 01 02	Get Encryption Mode for Data encryption Key (ICC DUKPT Key)
78 46 01	Get Firmware Release Version
72 52 01 05	Get ICC L1 Transaction Timeout
72 46 09 13	Get ICC L2 AID List Check Value
72 46 09 14	Get ICC L2 CA Public Key Check Value
72 46 09 02	Get ICC L2 Configuration Check Value
72 46 09 01	Get ICC L2 Kernel Check Value
72 46 24	Get ICC Reader Status
72 52 01 03	Get ICC Voltage Option(Internal Command)
72 46 86 01	Get Interface Device Serial Number
72 46 23 02	Get Internal EMV Level One Version Number

72 46 08 02	Get Internal EMV Level Two Version Number
78 31	Get Internal Firmware Version
78 46 30	Get Key Status
72 52 01 01	Get Key Type for Data encryption Key (ICC DUKPT Key)
78 46 20	Get Model Number
70 52 01 43	Get Parity
72 52 01 20	Get Pre/Post Data Len
78 52 01 01	Get Remote Key Injection Timeout
78 46 02	Get Serial Number
78 46 25	Get Status for Key
70 52 01 45	Get Stop Bits
78 46 50	Implement Self-Test function for Non-PCI device
75 46 31	Load App Firmware Check Value (Internal Command)
75 46 33	Load Bootloader Firmware Check Value (Internal Command)
75 46 17	Load Firmware Key (Internal Command)
78 46 F3	Load Key
78 46 F2	Load LCL-KEK (Internal Command)
75 46 16	Load Manufacture Key (Internal Command)
72 46 4D	Power Off
72 46 6E	Power On (Get ATR)
72 46 01 04	Remove All Application Data
72 46 04 04	Remove All CA Public Key List
72 46 85 04	Remove All Certification Revocation List
72 46 01 02	Remove Application Data
72 46 04 02	Remove CA Public Key
72 46 85 02	Remove Certification Revocation List
72 46 02 02	Remove Terminal Data
72 46 0A 02	Remove Transaction Amount Log
78 46 49	Reset
72 46 03 01	Retrieve AID List
72 46 01 01	Retrieve Application Data
72 46 04 01	Retrieve CA Public Key
72 46 04 05	Retrieve CA Public Key List
72 46 85 01	Retrieve Certification Revocation List
72 46 02 01	Retrieve Terminal Data
72 46 87 01	Retrieve Terminal Identification
72 46 07 01	Retrieve Transaction Result
78 52 00	Review General Group
72 52 00	Review ICC Group All Setting
70 52 00	Review RS232 Group
72 46 01 03	Set Application Data
72 53 01 21 01	Set Ascii Mask Data

70 53 01 41 01	Set Baud Rate
72 53 01 22 01	Set BCD Mask Data
72 46 04 03	Set CA Public Key
72 53 01 04 01	Set Card Type Option
72 46 85 03	Set Certification Revocation List
78 53 01 50 08	Set Date and Time
72 53 01 02 01	Set Encryption Mode for Data encryption Key (ICC DUKPT Key)
72 53 01 05 01	Set ICC L1 Transaction Timeout
72 53 01 03 01	Set ICC Voltage Option (Internal Command)
78 53 01 05 01	Set Idle Waiting Time
72 46 86 03	Set Interface Device Serial Number
72 53 01 01 01	Set Key Type for Data encryption Key (ICC DUKPT Key)
72 53 01 20 02	Set Pre/Post PAN Data Len
78 46 03	Set Serial Number (Internal Command)
70 53 01 45 01	Set Stop Bits
72 46 02 03	Set Terminal Data
72 46 87 03	Set Terminal Identification
72 46 05 01	Start Transaction

Get DUKPT Key KSN

Command Body is 78 46 3E <Length of Data> <KeyNameIndex> <Length of Key Slot> <Key Slot>

Where:
<Length of Data> is 2 bytes, format is Len_L Len_H, is length of <KeyNameIndex> <Length of Key Slot> <Key Slot>
<KeyNameIndex> is 1 byte, please refer to below table.
<Length of Key Slot> is 2 bytes, format is Len_L Len_H
<Key Slot> is any byte (1 byte or 2 bytes), value always 0.

KeyNameIndex	**Key Name**
0x14	LCL-KEK
0x02	Data encryption Key
0x0C	RKI-KEK

Response Body:
15 <Error Code> 78 46 3E – Invalid related Key (90 42, or 90 46, or 90 52) or Key STOP (73 00), Or Do Not Support the Key (90 47), Or
06 78 46 3E <Length of KSN> <KSN>
Where:
<Length of KSN> is 2 bytes, format is Len_L Len_H, value always 0A 00
<KSN> is 10 bytes KSN

Load Key

Command Body is 78 46 F3 <Length of Data> <Length of Encrypted Key ANS.1 Structure> <Encrypted Key ANS.1 Structure> <Length of LCL-KEK KSN> <LCL-KEK KSN>

Where:
<Length of Data> is 2 bytes, format is Len_L Len_H, is length of <Length of Encrypted Key ANS.1 Structure> <Encrypted Key ANS.1 Structure> <Length of LCL-KEK KSN> <LCL-KEK KSN>
<Length of Encrypted Key ANS.1 Structure> is 2 bytes, format is Len_L Len_H
<Encrypted Key ANS.1 Structure> - please refer to <80000405-001 Key Injection for Unattended Devices A.docx> document with F3 command (Sets DUKPT Key).
<Length of LCL-KEK KSN> is 2 bytes, format is Len_L Len_H, value always 0A 00

Note: Device supports loading below Key:

KeyNameIndex	Key Name	Definition	Key Slot	Mode Of Use	Key Usage
0x02	Data encryption Key	Encrypt ICC	0	B or X	B1
0x0C	RKI-KEK	Remote Key Injection	0	X	B1

Response Body:
15 <Error Code> 78 46 F3 - No LCL-KEK (04 00) or LCL-KEK STOP (73 00) or other,
Or
06 78 46 F3 <Length of KCV ANS.1 Structure> <KCV ANS.1 Structure>

Where:
<Length of KCV ANS.1 Structure> is 2 bytes, format is Len_L Len_H
<KCV ANS.1 Structure> - please refer to <80000405-001 Key Injection for Unattended Devices A.docx> document with F3 command (Sets DUKPT Key). KCV is Key Check Value, it is 3 bytes data, algorithm refer to X9.24

Get Firmware Release Version

Command Body is 78 46 01
Response Body is 06 & some bytes ASCII codes

Enter into Bootloader

Command Body is 78 46 7A 49 52 46 57 00 00 00 00 00 00 00 00
Response Body is
06 – Device has the function, or:
15 – Device has not the function.

Get Serial Number

Command Body is 78 46 02
Response Body is
06 + 10 bytes ASCII code Serial Number, or:
15 62 00 – No Serial Number

Get Model Number

Command Body is 78 46 20
Response Body is 06 + Model Number
Where: Model Number is “MINI2-5X” (USB-HID) or “MINI2-2X” (RS232) or “MINI2-0X” (UART)

Reset

Command Body is 78 46 49
Response Body is 06
Note:
Device will Reset (Re-Start) after it responds ACK Response Body.
It is the Highest Priority Command in device except Key Loading State.
If Device implement the function, the Real Date/Time will be started from 2015.01.01 00:00:00.

Implement Self-Test function for Non-PCI device

Command Body is 78 46 50

Response Body is
06 – Self-Test function successful
15 75 00 – Self-Test function failed

Implementation notes:
1. App integrity checking. After failure or error, device will check 2 times again.
2. Bootloader integrity checking. After failure or error, device will check 2 times again.
3. Configuration integrity checking. After failure or error, device will check 2 times again.
4. If Configuration data are large, the checking time increases.

Get Key Status

Command Body is 78 46 30

Response Body is 06 + PIN DUKPT Status + PIN Master Key Status + PIN Session Key Status + Data encryption Key Status + Data encryption Key Status + RKI-KEK
Where:

Key	Status	Note
PIN DUKPT Key	0: None. 1: Exist 0xFF: STOP	Does not support this key. Always 0
PIN Master Key	0: None 1: At least Exist a Master Key	Does not support this key. Always 0
PIN Session Key	0: None. 1: Exist	Does not support this key. Always 0
Data encryption Key / MSR DUKPT Key	0: None. 1: Exist 0xFF: STOP	Does not support this key. Always 0
Data encryption Key / ICC DUKPT Key	0: None. 1: Exist 0xFF: STOP	
RKI-KEK	0: None. 1: Exist 0xFF: STOP	

Get Status for Key

Command Body is 78 46 25

Response Body:
06 <Block Length> <KeyStatusBlock1> <[KeyStatusBlock2]> …<[KeyStatusBlockN]>,
Or
15 <Error Code>
Where:
<Block Length> is 2 bytes, format is Len_L Len_H, is KeyStatusBlock Number
<KeyStatusBlockX> is 4 bytes, format is <Key Index and Key Name> <keyslot> <keystatus>:
<Key Index and Key Name> is 1 byte. Please refer to following table and <80000426-001 KeyNameIndex Database - V51.xls>
<key slot> is 2 bytes. Range is 0 – 9999
<key status> is 1 byte.
0 – Not Exist
1 – Exist
0xFF – (Stop. Only Valid for DUKPT Key)
Support <Key Index and Key Name> Table

KeyNameIndex	Key Name	Definition	Key Slot
0x14	LCL-KEK	Encrypt Other Keys	0
0x02	Data encryption Key	Encrypt ICC	0
0x0C	RKI-KEK	Remote Key Injection	0

Set Remote Key Injection Timeout

Command Body is 78 53 01 01 02 <Timeout_H> <Timeout_L>
Where: <Timeout_H> <Timeout_L> need be 120 seconds ~ 3600 seconds

Response Body is 06

Get Remote Key Injection Timeout

Command Body is 78 52 01 01
Response Body is 06 78 01 01 02 <Timeout_H> <Timeout_L>

Set Date & Time

Command Body is 78 53 01 50 08 <Data/Time Length> <Date Time> <MAC Length>
Where:
<Data/Time Length> is 1 bytes data – Fix is 0x06
<Data Time> is 6 bytes data – Year, Month, Date, Hour, Minute, Second

Item	Value Area (BCD Code)
Year	15~99, 00
Month	01~12
Date	01~31
Hour	00~23
Minute	00~59
Second	00~59

<MAC Length> is 1 byte data – Fix is 0x00

Response Body is 06

Note:
After Power On the device, the Real Time is starting from 2000/01/01 00:00:00, , end to 2100/01/01 00:00:00. User need set the Real Time to implement related transaction.
The command always valid in IDLE State.
If current Date/Time is 2014/08/23 15:24:59, <Date Time> should be 0x14 0x08 0x23 15 24 59 (BCD Code).
Disable Set Date/Time to Before 2012/01/01 00:00:00.
The Default Date/Time is 2015/01/01 00:00:00

Get Date & Time

Command Body is 78 52 01 50

Response Body is 06 78 01 50 06 <Date Time>

Where: <Data Time> is 6 bytes data – Year, Month, Date, Hour, Minute, Second

Set Idle waiting time

Command Body is 78 53 01 05 01 <Time>

Where: <Time> is 0 or 2 seconds ~ 60 seconds (Default is 2 seconds)

Response Body is 06

Note:

- If Waiting time is 0 and if interface is RS232, Disable device work into Low Power Consumption status.
- If Waiting time is Not 0, If Idle status is more than <Time>, if interface is RS232, and if Card is not Seated, device will enter into Low Power Consumption status.
- If device received command in Idle status, the waiting time will be reset.
- If IC card is removed, the waiting time will be reset.

Default General Group (Default All)

Command Body is 78 53 00

Response Body is 06

Below Setting should be reset to default value:

Function Name	Default Value	Note
Remote Key Injection Timeout	120 Seconds	

Get Internal Firmware Version

Command Body is 78 31 – Get Internal Version

Response Body is 06 + Version Number

Review General Group

Command Body is 78 52 00

Response Body is 06 78 02 01 02 <Timeout_H> <Timeout_L>50 06<Date Time>

RS232 Group

Note:
Device will use new stop bits after it responds 0x06.

Set Baud Rate

Command Body is 70 53 01 41 01 ASCIIChar

BaudRate	ASCIIChar
9600	4

19200	6
38400	7
115200	9

Response Body is 06
Note:
Device will use new baud rate after it responds 0x06.

Get Baud Rate

Command Body is 70 52 01 41

Response Body is 06 70 01 41 01 ASCIIChar

Get Data Bits

Command Body is 70 52 01 42

Response Body is 06 70 01 42 01 38 (8 bits Data)

Get Parity

Command Body is 70 52 01 43

Response Body is 06 70 01 43 01 30 (None)

Set Stop Bits

Command Body is 70 53 01 45 01 ASCIIChar

StopBits	ASCIIChar
1	1
2	2

Response Body is 06

Get Stop Bits

Command Body is 70 52 01 45

Response Body is 06 70 01 45 01 ASCIIChar

Default RS232 Group

Command Body is 70 53 00

Response Body is 06
Below Setting should be reset to default value:

Function Name	Default Value
Baud Rate	38400
Stop Bits	1

Review RS232 Group

Command Body is 70 52 00

Response Body is 06 70 04 41 01<Baud Rate> 42 01 38 43 01 30 45<Stop Bits>

Smart Card Group (ICC EMV Level One Task) – Function Command

Get ICC Reader Status

Command Body is 72 46 24
Response Body is 06 + <Reader status> (1 byte)

Bit Position	'0'	'1'
0	ICC Power not ready	ICC Powered
1	Card not seated	Card seated
2~7		

Power On (Get ATR)

Command Body is 72 46 6E

Response Body is 06 + <ATR String>

Note:
This Command is used to power up the currently selected microprocessor card. It follows the ISO7816-3 power up sequence and returns the ATR as its response.

Power Off

Command Body is 72 46 4D

Response Body is 06

Exchange APDU Plaintext

Command Body is 72 46 41 <C-APDU>

Response Body is 06 00 <R-APDU>
Note:
If Data encryption Key (ICC DUKPT Key) was not loaded, command supported.
If Data encryption Key (ICC DUKPT Key) Exist, it is Not Supported and response Error Code (6A 00).

Get KSN

Command Body is 72 46 62 <C-APDU>

Response Body is 06 + <10 bytes KSN> <R-APDU>

Note:
1. If Data encryption Key (ICC DUKPT Key) was not loaded, Unit should response Error Code (04 00) for this command.

Get EMV Level One Version Number

Command Body is 72 46 23 01

Response Body is 06<"IFMVx.yy">.
x.yy is version number. Fist version number is 1.00.

Smart Card Group (ICC EMV Level Two Task) – Function Command

Requires L2 kernel on device.

Setting transaction parameters

1. Send Set Application Data Command

2. Send Set Terminal Data Command

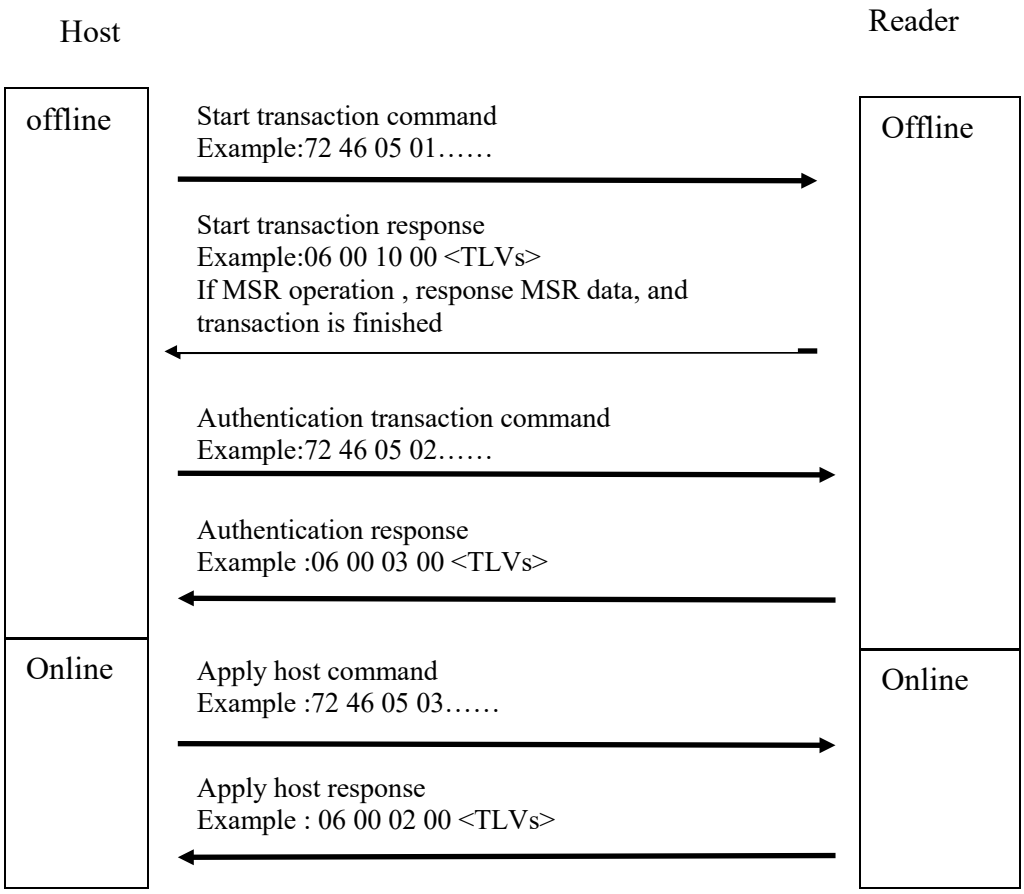
MiniSmart II Support 2C configuration, below are Terminal Data Value for 2C configuration. Manufacture need set it.

Item	Terminal Data (TLV format)	Definition
1	9F 35 01 21	Terminal Type
2	9F 33 03 60 28 C8	Terminal Capability
3	9F 40 05 F0 00 F0 A0 01	Additional Terminal Capability
4	DF 26 01 01	Terminal Supports CRL
5	DF 10 02 65 6E	Language
6	DF 11 01 00	Not Support Transaction Log
7	DF 27 01 00	Not support Exception File
8	DF EE 15 01 01	Terminal Support ASI
9	DF EE 16 01 00	Terminal Encrypt Mode: 00=DUKPT 01=MK/SK
10	DF EE 17 01 07	Terminal Entry Mode for ICC
11	DF EE 18 01 80	Terminal Encrypt Mode for MSR
12	DF EE 1E 08 D0 DC 20 D0 C4 1E 16 00	Contact Terminal Configuration
13	DF EE 1F 01 80	Issuer Script Limit
14	DF EE 1B 08 30 30 30 31 35 31 30 30	ARC
15	DF EE 20 01 3C	ICC Power on detect waiting time
16	DF EE 21 01 0A	ICC L1 waiting time
17	DF EE 22 03 32 3C 3C	Driver time. byte 1 -> Menu. byte 2 -> Get PIN. byte 3 -> MSR

3. Send Set CA Public Key Command;

4. Send Set Certification Revocation List command

Transaction Flow



1. Send Start Transaction Command.
2. Send Authenticate Transaction Command.
3. If received online request from terminal, send Apply Host Response command.
4. When a transaction is completed, interface will output a 06+TLVformat data list, and transaction result (approve, or decline, or…) will display on the LCD.5. If you want to review the transaction result you can send Retrieve Transaction Result Command,. It will output the TLV format data refer to the Option Data list.

Note: To all above commands, if it can return ACK, execute follow steps, otherwise, repeated it till return ACK.

Retrieve EMV Level Two Version Number command can be sent all the time.

Cancel Transaction command can be sent all the time if you want to terminate a transaction.

Retrieve Application Data

Command Body is 72 46 01 01 <LenL> <LenH> <5~16 bytes AID>
Where:
<LenL> <LenH> is the length of AID

Response Body is
06 <TagCounterL> <TagCounterH> <TLV1> <TLV2>…<TLVn> or
15<ErrorCode>.
Where:
<TagCounterL> <TagCounterH> is the number of <TLV>
Note:
If AID List / Application Data does not exist, response 15 F2 00.

Remove Application Data

Command Body is 72 46 01 02 <LenL> <LenH> <5~16 bytes AID>
Where:
<LenL> <LenH> is the length of AID

Response Body is 06 or
15<ErrorCode>
Note:
If AID List does not exist, response 15 F2 00.

Set Application Data

Command Body is 72 46 01 03 <LenL> <LenH> <5~16 bytes AID> <TagCounterL>
<TagCounterH> <TLV1> <TLV2>…<TLVn>.
Where:
<TagCounterL> <TagCounterH>is the Number of <TLV>.
<LenL> <LenH> is the length of AID

Response Body is
06 or
15 <ErrorCode>
Note:
If a <TLV> format was error, response 15 - F2 02 -If AID List has full (MAX is 16), response 15 F2 03

Remove All Application Data

Command Body is 72 46 01 04

Response Body is 06

Retrieve Terminal Data

Command Body is 72 46 02 01

Response Body is
06 <TagCounterL> <TagCounterH> <TLV1> <TLV2>…<TLVn> or
15<ErrorCode>
Where:
<TagCounterL> <TagCounterH> is the number of <TLV>
If Terminal Data does not exist, response 15 F2 01

Remove Terminal Data

Command Body is 72 46 02 02

Response Body is
06

Set Terminal Data

Command Body is 72 46 02 03 <TagCounterL> <TagCounterH> <TLV1>
<TLV2>…<TLVn>.
Where:
<TagCounterL> <TagCounterH>: the Number of <TLV>.
Response Body is
06 or
15 <ErrorCode>
Note:
If a <TLV> format was error, response 15 F2 02

Retrieve AID List

Command Body is 72 46 03 01

Response Body is 06 <NumberL> <NumberH> <AID Block 1> <AID Block 2> … <AID Block N>.
Where:
<NumberL> <NumberH> is Number of AID Blocks.
<AID Block> format is <LenL> <LenH> <Several bytes AID>.
Where:
<LenL> <LenH> is Length of AID

Note:
If AID List does not exist, response 15 F2 00

Retrieve CA Public Key

Command Body is 72 46 04 01<5 bytes RID> <1 byte Index>.

Response Body is 06 <5 bytes RID> <1 byte Index> <1 byte Hash Algorithm> <1 byte Encryption Algorithm> <20 bytes HashValue> <4 bytes Public Key Exponent> <2 bytes Modulus Length> <Variable bytes Modulus>
Where:
<Hash Algorithm>: The only algorithm supported is SHA-1.The value is set to 0x01
<Encryption Algorithm>: The encryption algorithm in which this key is used. Currently support only one type: RSA. The value is set to 0x01.
<HashValue>: Which is calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent
<Public Key Exponent>: Actually, the real length of the exponent is either one byte or 3 bytes. It can have two values: 3 (Format is 0x00 00 00 03), or 65537 (Format is 0x00 01 00 01)
<Modulus Length>: **<LenL> <LenH>** Indicated the length of the next field.
<Modulus>: This is the modulus field of the public key. Its length is specified in the field above.
Note:
If CA Key RID does not exist, response 15 F2 05
If CA Key Index does not exist, response 15 F2 06

Remove CA Public Key

Command Body is 72 46 04 02 <5 bytes RID> <1 byte Index>.

Response Body is 06
Note:
If CA Key RID does not exist, response 15 F2 05
If CA Key Index does not exist, response 15 F2 06

Set CA Public Key

Command Body is 72 46 04 03 <5 bytes RID> <1 byte Index> <1 byte Hash Algorithm> <1 byte Encryption Algorithm> <20 bytes HashValue> <4 bytes Public Key Exponent> <2 bytes Modulus Length> <Variable bytes Modulus> <2Bytes MAC Length>

Where:
<Hash Algorithm>: The only algorithm supported is SHA-1.The value is set to 0x01
<Encryption Algorithm>: The encryption algorithm in which this key is used. Currently support only one type: RSA. The value is set to 0x01.
<HashValue>: Which is calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent
<Public Key Exponent>: Actually, the real length of the exponent is either one byte or 3 bytes. It can have two values: 3 (Format is 0x00 00 00 03), or 65537 (Format is 0x00 01 00 01)

<Modulus Length>: **<LenL> <LenH>** Indicated the length of the next field.
<Modulus>: This is the modulus field of the public key. Its length is specified in the field above.
<2Bytes MAC Length> is 2 bytes data – Fix is 0x00 0x00

Response Body is 06
Note:
Per <RID> has at least 6 CA Public Key.
Per <RID> has at most 16 CA Public Key.
If key has full, response 15 F2 07
If CA Key Hash Data is error, response 15 F2 08
If <Hash Algorithm> and <Encryption Algorithm> are not 0x01, response 15 F2 0E.

Remove All CA Public Key List

Command Body is 72 46 04 04.
Response Body is 06

Retrieve CA Public Key List

Command Body is 72 46 04 05.

Response Body is 06 <LenL> <LenH> <5Bytes RID1> <1 byte RID1 Index> <5Bytes RID2> <1 byte RID2 Index>….. <5Bytes RIDN> <1 byte RIDN Index>.

Note:
If any CA Key does not exist, response 15 F2 04

Start Transaction

Command Body is:
72 46 05 01- <FallBack> <TimeOut1> <TimeOut2> <App Data>.

Where:
<FallBack> (1byte). 0x01 indicates it supports FallBack to MSR, 0x00 indicate it not support FallBack.
<TimeOut1> (2 bytes, unit is Second). Timeout for Card is seated.
<TimeOut2> (2 bytes, unit is Second). Waiting time till “Authenticate Transaction” command.
<App Data> format is <TLV1> <TLV2> … <TLVn>. Refer to Transaction Data & Option Data List.

Response Body is:
First return: 06
Wait in seconds.

Second return:
06 <Response Code> <Attribution> <Output Data List> (All response except that MSR Fall Back). Or
06 <Response Code> <Output Data List> (MSR Fall Back response)
Where:
<Response Code> length is 2Bytes. Please refer to "Response Code" Section
<Attribution>: 1 Byte:
BIT0 – Card Type: 0 – Contact Card
BIT2,1 – Encryption Mode: 00 – TDES Mode, 01 – AES Mode
BIT7~3 - Reserve
<Output Data List> format is <TLV1> <TLV2> … <TLVn>.

Note in Transaction:
If have error in process, terminal will return error code and terminate transaction.
If command format error, terminal will response 15 F2 09
If no any application data in terminal, response 15 F2 00
If no terminal data in terminal, response 15 F2 01
After transaction start, terminal only can receive "Start Transaction", "Authenticate Transaction", "Cancel Transaction" and "Retrieve Transaction Data" command. If any other command was sent, it responds 15 F2 0A
If timeout occurred, response 15 81 00.
If a tag is not occurred in the ICC, the tag in output TLV list will not occur all the same. Amount, other amount, transaction type must be exist. If not, terminal will response error code 15 F2 0D.
For MSR fallback operation, the <Response Code> is 00 07, <Attribution> is None.
For MSR whose service code is not 2 or 6, the <Response Code> is 00 11.

Authenticate Transaction

Command Body is 72 46 05 02<ForeOnline> <TimeOut> <App Data>.

<ForeOnline> (1byte). 0x01 indicates it supports ForeOnline, 0x00 indicate not support.
<TimeOut> (2 byte, unit is Second).means terminal waiting time for host response when online.-
<App Data> format is <TLV> (V is output tag list.)

Response Body is:
First return: 06
Wait in seconds.
Second return: 06 <Response Code> <Attribution> <TLV1> <TLV2> … <TLVn>.
Where:
<Response Code> length is 2Bytes. Please refer to "Response Code" Section.
<Attribution>: 1 Byte:
BIT0 – Card Type: 0 – Contact Card
BIT2,1 – Encryption Mode: 00 – TDES Mode, 01 – AES Mode
BIT7~3 - Reserve

Note:
If command format error, terminal will response 15 F2 09
If an Option tag is not occurred in the Option Data List, the tag in output TLV list will not occur all the same.
If the tag '9f10' and '9f26' have value in the terminal, they will also occur in the command response TLV list.

Complete Transaction

Command Body is 72 46 05 03 <1Byte ComFlag> [<Authorization Response Code (TLV,Tag 8A)> <Issuer Authentication Data (TLV, Tag 91)> <Scripts (TLV, Tag 71/72)>] <App Data>

Where:
<1Byte ComFlag>:0x01 indicate online with host 0x00 indicate unable online.
Data in [ ] indicate these data is optional:
If ComFlag is 0x01, the Data exist.
If ComFlag is 0x00, the Data does not exist.
<App Data> format is <TLV> (V is output tag list.)

Response Body is:
First return: 06
Wait in seconds.
Second return: 06 <Response Code> <Attribution> <Output TLV Data>.
Where:
<Response Code>length is 2Bytes. Please refer to "Response Code" Section
<Attribution>: 1 Byte:
BIT0 – Card Type: 0 – Contact Card
BIT2,1 – Encryption Mode: 00 – TDES Mode, 01 – AES Mode
BIT7~3 - Reserve
Note:
If any parameter was error, response 15 F2 02.

Cancel Transaction

Command Body is 72 46 06.

Response Body is 06.

Retrieve Transaction Result

Command Body is 72 46 07 01<2 Byte Length> <Tags>.

<2 Byte Length>format is<LenL> <LenH> is the length of <Tags>.
<Tags> these tags will return TLV format. Supported tags refer to "Option Data List" Table in Section "Reference Data List".

Tags Example:
“9F02, 9F36, 95, 9F37” means total 4 tags (9F02, 9F36, 95, 9F37) reqested to in response.
Length is 7bytes.

Response Body is 06 <TLV1> <TLV2> … <TLVn>. For details please refer to P127 of EMV4.3 book3.
<1Byte Message type>,value 0x00 means transaction message, value 0x01 means advice message.
Note:
If an Option tag is not occurred in the Option Data List, the tag in output TLV list will not occur all the same.

Get EMV Level Two Version Number

Command Body is 72 46 08 01

Response Body is 06 <” EMV Common L2 VX.YY”>.
X.YY is version number. Fist version number is 1.10.

Get Internal EMV Level Two Version Number

Command Body is 72 46 08 02

Response Body is 06<” EMV Common L2 VX.YY.ZZZ”>.
X.YY.ZZZ is internal version number. Version number is start from 1.10.001

Retrieve Interface Device Serial Number

Command Body is 72 46 86 01

Response Body is 06 <Serial Number>

Note: Set Interface device's serial number. EMV serial Number can be set only once.

Set Interface Device Serial Number

Command Body is 72 46 86 03 <Serial Number>
Note: <Serial Number> 8 bytes ‘0’ ~ ‘9’, or ‘a’ ~ ‘z’, or ‘A’ ~ ‘Z’

Response Body is 06

Retrieve Terminal Identification

Command Body is 72 46 87 01

Response Body is 06 <Identification>

Set Terminal Identification

Command Body is 72 46 87 03 <Identification>
Note: <Identification> 8 bytes '0' ~ '9', or 'a' ~ 'z', or 'A' ~ 'Z'

Response Body is 06

Retrieve Certification Revocation List

Command Body is 72 46 85 01.

Response Body is 06<2Byte Length> <CRL1> <CRL2>…<CRLn>.
<2Byte Length>: <Low byte of length> <High byte of length>
<CRL>format is <5Bytes RID> <1Byte CA public key Index> <3Bytes Certificate Serial Number>

Remove Certification Revocation List

Command Body is 72 46 85 02 <2 Bytes Length> <CRL1> <CRL2>…<CRLn>
<2Byte Length>: <Low byte of length> <High byte of length>
<CRL>format is <5Bytes RID> <1Byte CA public key Index> <3Bytes Certificate Serial Number>Response Body is 06.

Set Certification Revocation List

Command Body is 72 46 85 03 <2 Bytes Length of CRL> <CRL1> <CRL2>…<CRLn> <2Bytes MAC Length>.
Where:
<2Byte Length of CRL>: <Low byte of length> <High byte of length>
<CRL> format is <5Bytes RID> <1Byte CA public key Index> <3Bytes Certificate Serial Number>
<2Bytes MAC Length> is 2 bytes data – Fix is 0x00 0x00

Response Body is 06.
Note:
Supported at least CRL number is 30 for a RID.

Remove All Certification Revocation List

Command Body is 72 46 85 04

Response Body is 06.

Get ICC L2 Kernel Check Value

Command Body is 72 46 09 01

Response Body is 06 <20 bytes L2 Kernel Check Value>

Get ICC L2 Configuration Check Value

Command Body is 72 46 09 02

Response Body is 06 <20 bytes L2 Configuration Check Value>
Note:
MiniSmart II Support 2C configuration.

Get ICC L2 AID List Check Value

Command Body is 72 46 09 13

Response Body is 06 <32 bytes L2 AID List Check Value>

Note:
1. Set AID List Order is different, the Check Value is different also.
2. Suggestion: User should load all application data once and not update it. Best practice is to remove all application data and load all application data again. Then order won't change.

Get ICC L2 CA Public Key Check Value

Command Body is 72 46 09 14

Response Body is 06 <32 bytes L2 CA Public Key (RID+Index) Check Value>

Note:
1. CA Public Key format is RID+Index
2. Set CA Public Key Order is different, the Check Value is different also.
3. Suggestion: User should load all CA Public Keys once and not update it. Best practice is to remove all CA Public Keys and load all CA Public Keys again. Then order won't change.

Get Internal EMV Level One Version Number

Command Body is 72 46 23 02

Response Body is 06<"IFMVx.yy.zzz">.

x.yy.zzz is internal version number. Version number is start from0.99.001

Remove Transaction Amount Log

Command Body is 72 46 0A 02

Response Body is 06

Smart Card Group (ICC EMV Level Two Task) – Output / Input

Directly Driver Output Body

Output Body Format

49 72 <DriverID> <DriverData>

Input Body Format

72 49 <DriverID> <ACK> or
72 49 <DriverID> <ACK> <ACK Data>or
72 49 <DriverID> <NAK> <Error Codes>

LCD display control

Output Body is 49 72 01 <Len_L of Control Data> <Len_H of Control Data> <m bytes Control Data>
Where:
<m bytes Control Data>

- Display mode – 1 byte
  1- Menu Display
  2- Normal Display get function key
  3- Display without key input (Do Not Receive Input Data)
  8 – Language Menu Display
  16-Clear Screen (Do Not Receive Input Data)

If Mode byte is “Clear Screen”, don’t need to send below field.

- Fix 0x00 0x00. – 2 bytes (Little-endian)

- Length of Display Message Language. - 2 bytes (Little-endian)
- Display Message Language, 2 byte
  EN - English (default)
  ES - Spanish
  ZH - Chinese
  FR – French
  …

• Length Display Message Control - 2 bytes (Little-endian)
• Display Message Control: repeatable combination of <Line> <Message> <0x1C>
<Line> - Display line number (1-First Line, n-nth Line), Maximum 16 lines.
  •The lower 7 bits is for line number.
  •The MSB is to indicate following message is a Message String or Message ID.
    •MSB – 0: Message String. (It is valid for "Menu Display" and "Language Menu Display")
    •MSB – 1: Message ID. (It is only valid for "Menu Display")
<Message> - Message String or Message ID.
Message String:
  •"Menu Display" : character in the range of 0x20 – 0x7f, Maximum 16 characters
  • "Language Menu Display" : 2 bytes Language ID
EN - English (default)
ES - Spanish
ZH - Chinese
FR – French
…
…
Message ID: 1 byte, check LCD Foreign Language Mapping Table
<0x1C> - separator

• Length Back Light On TimerValue - 2 bytes (Little-endian)
• Back Light On TimerValue in second, (Little-endian) (all 0-Back Light Off, all 0xff-Back Light always On)

Input Body is 72 49 01 <ACK> <Len_L of ACK Data> <Len_H of ACK Data> <n bytes ACK Data>
Where:
<n bytes ACK Data>

• Display mode – 1 byte
  0- Cancel (user presses cancel key on the key pad for mode 1)
  1 - Menu Display
  2- Normal Display get function key 8-
  Language Menu Display
If Mode byte is "Cancel" or "Display without key input", don't need to send below field.

• If Menu Display, Length of Menu value (If Normal Display, Length of Key (Get Function))
• If Menu Display, Menu value, sequence number of selected line, hex format (If Normal Display, ASCII format ('E' is Enter, 'C' is Cancel))

Get MSR Data control

Output Body is 49 72 03 <Len_L of Control Data> <Len_H of Control Data> <m bytes Control Data>
Where:
<m bytes Control Data>

• Length of Total timeout for Swipe MSR Card. – 2 bytes (Little-endian)
• Total timeout for Swipe MSR Card, in second, (Little-endian), default is 30 seconds.

• Length of Display Message Language. - 2 bytes (Little-endian)
• Display Message Language, 2 byte
EN - English (default)
ES - Spanish
ZH - Chinese
FR – French
…

•Length Display Message Control - 2 bytes (Little-endian)
• Display Message Control: <Line><Message><0x1C>
<Line> - lower 7 bits is for line number: Display line number (1-First Line, n-nth Line), Maximum 16 lines.
The MSB need be 1: is to indicate following message is Message ID.
<Message> - Message ID.
Message ID: 1 byte, check LCD Foreign Language Mapping Table. Normal Value is 19.
<0x1C> - separator

Input Body is 72 49 03 <ACK> <Len_L of ACK Data> <Len_H of ACK Data> <n bytes ACK Data>
Where:
<n bytes ACK Data>

n bytes MSR Data

Smart Card Group – Set/Get Commands Misc.

Set Key Type for Data encryption Key (ICC DUKPT Key)

Command Body is 72 53 01 01 01 <Option>

Key Type	Option
Data type Key	0x00
PIN type Key	0x01

Note: ICC Group (Task) only support A Fun of Setting/Review command except for Default All & Review All.
Response Body is 06

Get Key Type for Data encryption Key (ICC DUKPT Key)

Command Body is 72 52 01 01
Response Body is 06 72 01 01 01 <Option>
Note: The one byte followed Task ID (72) is Block Number Data.

Set Encryption Mode for Data encryption Key (ICC DUKPT Key)

Command Body is 72 53 01 02 01 <Option>

Encryption Mode	Option
TDES	0x00
AES	0x01

Note:
ICC Group (Task) only supports A Fun of Setting/Review command except for Default All & Review All.

Response Body is
06 (Data encryption Key (Data encryption Key (ICC DUKPT Key)) did not exist) or
15 6A 00 (NAK + Unsupported Command Error Code) (Data encryption Key (ICC DUKPT Key) existed)

Get Encryption Mode for Data encryption Key (ICC DUKPT Key)

Command Body is 72 52 01 02
Response Body is 06 72 01 02 01 <Option>

Encryption Mode	Option
No Data encryption Key (ICC DUKPT Key)	0xFF
TDES	0x00
AES	0x01

Note: The one byte followed Task ID (72) is Block Number Data.

Set Card Type Option

Command Body is 72 53 01 04 01 <Option>

Card Type	Option
EMV	0xFF
ISO	0x00

Response Body is 06

Get Card Type Option

Command Body is 72 52 01 04

Response Body is 06 72 01 04 01 <Option>

Set ICC L1 Transaction Timeout

Command Body is 72 53 01 05 01 <Timeout>
Where: <Timeout> is 1 byte. Value is **8 seconds** ~ 90 Seconds

Response Body is 06

Get ICC L1 Transaction Timeout

Command Body is 72 52 01 05

Response Body is 06 72 01 05 01 <Timeout>

Get EMV CT L2 Transaction Interval

This is the delay time between command responses during the various phases of the EMV transaction (start, auth, complete). For example, this is the delay time between sending Authenticate Transaction command and receiving ACK.

Command Body is 72 52 01 2F 01

Response Body is 06 72 01 2F 01 <CTL2Interval>

Where CTL2Interval is 0 (no delay) or a value in the range 30-100 (which means 300-1000 milliseconds).

Set EMV CT L2 Transaction Interval

This is the delay time between command responses during the various phases of the EMV transaction (start, auth, complete). You can delay the receipt of ACK using this command, for example. Adjustment of this value may be needed in certain instances if ACK is occurring too quickly to be retrieved reliably. (Set for 300 msec, then test. Adjust the delay upward until ACK is captured reliably.)

Command Body is 72 53 01 2F 01 <CTL2Interval>

Where:

CTL2Interval Value	Definition
0	**No Delay**
30 ~ 100	300ms ~ 1000ms

Example: 72 53 01 2F 01 **50** sets the delay to 800 msec (0x50 * 10 msec)

Set Pre/Post PAN Data Len

Command Body is 72 53 01 20 02 <PrePANCtlDataLen> <PostPANCtlDataLen>
Where:
<PrePANCtlDataLen> need be 0~6, Default is 4
<PostPANCtlDataLen> need be 0~4, Default is 4

Response Body is 06

Get Pre/Post Data Len

Command Body is 72 52 01 20

Response Body is 06 72 01 20 02 <PrePANCtlDataLen> <PostPANCtlDataLen>

Set ASCII Mask Data

Command Body is 72 53 01 21 01 <AsciiMaskData>
Where:
<AsciiMaskData> can be 0x20~0x7E, Default is 0x2A(*).

Response Body is 06

Get ASCII Mask Data

Command Body is 72 52 01 21

Response Body is 06 72 01 21 01 <AsciiMaskData>

Set BCD Mask Data

Command Body is 72 53 01 22 01 <BCDMaskData>
Where:
<BCDMaskData> can be can be 0x0A~0x0F, Default is 0x0C(*).
Note:
If 0x23 will be masked High BCD, result should be 0xC3
If 0x23 will be masked Low BCD, result should be 0x2C
If 0x23 will be masked all BCD, result should be 0xCC

Response Body is 06

Get BCD Mask Data

Command Body is 72 52 01 22

Response Body is 06 72 01 22 01 <BCDMaskData>

Default ICC Group All Setting

Command Body is 72 53 00

Response Body is 06

Below Setting should be reset to default value:

Function Name	Default Value
Key Type	Data Key
Card Type	EMV
ICC L1 Transaction Timeout	8
PrePANCtlDataLen	4
PostPANCtlDataLen	4
AsciiMaskData	0x2A
BCDMaskData	0x0C

Review ICC Group All Setting

Command Body is 72 52 00

Response Body is 06 72 07 01 01<Key Type Option> 02 01 <Encryption Mode Option> 04 01 <Card Type Option> 05 01 <L1 Transaction Timeout> 20 02 <PrePANCtlDataLen> <PostPANCtlDataLen> 21 01 <AsciiMaskData> 22 01 <BCDMaskData>

Note: The one byte followed Task ID (72) is Block Number Data.

Remote Key Injection

Initiate RKL

Command Body

55 52 4B 49 30

Where: 52 4B 49 is 'RKI'

Response Body

<ACK> <20 bytes ASCII code KSN of RKI-KEK> <10 bytes Serial Number> <16 bytes ASCII code Authentication Code 1> Or

<NAK> <Error Code>
Where:
<20 bytes ASCII code KSN of RKI-KEK> is 10 bytes KSN – It is KSN.
<10 bytes Serial Number> is set by itself command listed in Command Set File.
<16 bytes ASCII code Authentication Code 1> is 8 bytes Authentication Code 1
8 bytes Authentication Code 1 is 8 bytes Random generated by POS Device.
Note: If Serial Number is 9 bytes, it will be followed a 0x00 bytes to be 10 bytes in this response.

RKL Get Status

Command Body

55 52 4B 49 31 <32 bytes ASCII code Encrypted Data (EK1)>
Where:
52 4B 49 is 'RKI'
<32 bytes ASCII code Encrypted Data (EK1)> is 16 bytes Encrypted Data (EK1):
The Plaintext of 16 bytes Encrypted Data (EK1) is
<Authentication Code 2><Authentication Code 1>
Authentication Code 2 is 8 bytes Random generated by RKS
Authentication Code 1 is 8 bytes data (it should be validated)
16 bytes Encrypted Data (EK1) is encrypted by RKI-KEK according to KSN (CBC TDES algorithm, the IV are 0)

Response Body

<ACK><20 bytes ASCII code KSN of RKI-KEK><32 bytes ASCII code Encrypted Data (EK2)> Or
<NAK><Error Code>
Where:
<20 bytes ASCII code KSN of RKI-KEK> is 10 bytes KSN – It is advanced to next KSN. Refered as next-KSN hereafter)
<32 bytes ASCII code Encrypted Data (EK2)> is 16 bytes Encrypted Data (EK2):
The Plaintext of 16 bytes Encrypted Data (EK2) is
<Authentication Code 1><Authentication Code 2>
Authentication Code 1 is 8 bytes data
Authentication Code 2 is 8 bytes data
16 bytes Encrypted Data (EK2) is encrypted by RKI-KEK according to next-KSN (CBC TDES algorithm, the IV are 0)

New KSN/Key Pair

Command Body

55 52 4B 49 32 <M bytes TR-31 format Data (EK3)> Where:
52 4B 49 is 'RKI'
<M bytes TR-31 format Data (EK3)> is ASCII code "**TR-31 Edition B**" format data. The N byte Key data is encrypted by **KBEK** (generated from **KBPK**, **KBPK** is generated from

Data Key of RKI-KEK), and 4 bytes MAC is generated from data (detailed in below) encrypted with **KBMK** (generated from **KBPK**, **KBPK** is generated from Data Key of RKI-KEK). (**KSN** of RKI-KEK is advanced to next **KSN**. Refered as next-**KSN** hereafter. Please see the below detailed algorithm.)

TR-31 Block data format is: KBH + KBH_OB (Optional Block) **+ ASCII code Result + ASCII code MAC data.**

2N bytes ASCII code Encrypted Key – N bytes Encrypted Key

16 bytes ASCII code MAC data – 8 bytes MAC data

Response Body

<ACK><6 bytes ASCII code MAC Result>

<NAK><Error Code>

Where:

<6 bytes ASCII code MAC Result> is 3 bytes MAC Result (MAC Result of Key)

Using 8/16 bytes Key as Key, ECB DES/TDES Encrypt 8/16 bytes 0s.

Firmware Updating

Structure Note

1. Firmware Bootloader File is named FM file. The file name format is *.fm.
2. There are 'Block 0 - FM File Format Definition', 'Block 1', 'Block 2', 'Block 3' and 'Block 4' data.

Application/Bootloader FM File Structure

Block Number	Data Size	Data Note
Block 0	64 Bytes	Special Data – FM file format definition
Block 1	256 Bytes	<TLVn> format data: <Tag 0x01> data: 01 10 <App Version Data> (Such as 01 10 + "Version 0.99.003") <Tag 0x02> data: 02 < x> <x bytes Device Name Data> (Such as 01 0E + "Smart PIN L100", or 01 0C + "MiniSmart II") <Taq 0x03> data: 03 04 <Bootloader Version Number data> <Tag 0x04> data: 04 01 <Encryption Mode> <Tag 0xF0> data: F0 <Len> <0xFF … data>
Block 2	256 Bytes	256 Bytes Firmware Check Value
Block 3	Package 1	m bytes Data (at least 1K bytes)
…	Package 2	m bytes Data (at least 1K bytes)
…		…
…		…
…		…
	Package N	m bytes Data (at least 1K bytes)
Block 4	256 Bytes	Random Number

Note:

1. m value is confirmed by '2 byte ' data in 'FM File Format Definition' and Get Version From Bootloader Firmware.
2. N value is confirmed by '2 byte -Package Number' data in 'FM File Format Definition' and Calculation from FM file data.
3. 8 Bytes String "Version" + 8 Bytes Application Version need Ascii Code Data (Version 0.99.003 → Hex: 56 65 72 73 69 6F 6E 20 30 2E 39 39 2E 30 30 33)
4. x Bytes Device Name n need Ascii Code Data (Intuit III → Hex: 49 6E 74 75 69 74 20 49 49 49)
5. <Bootloader Version Number data> is from 0x00 00 00 00, 0x00 00 00 01, …, to 0xFF FF FF FF
6. <Encryption Mode>:
   - 0x00 – Plaintext.
   - 0x03 - Only Firmware RSA Key Encryption & RSAPSS RSA Data Format
   - 0x04 - Firmware Fix Key Encryption & RSSPSS RSA Raw Data Format.

Definitions

Block 0 Data

1. This Block data are 64 bytes data - FM File Format Definition

2. The Format is:

<6 byte PCI___, NONPCI> <1 byte File Type> <1 byte Mode> <2 byte file descriptor length> <2 byte block head size (byte)> <2 byte block tail size (byte)> <2 byte Package size (in byte)> <2 byte Package Numbers> <4 bytes Block 3 data First Package Start Address> <Surplus bytes Reserved>

- The Start 6 bytes is a magic text to identify if this file is for PCI or Non-PCI:
  - For PCI file, it is "PCI " (Hex is 50 43 49 20 20 20)
  - For Non-PCI file, it is "NONPCI" (Hex is 4E 4F 4E 50 43 49)
- <1 byte File Type>:'A'- Application, 'B'-Bootloader
- <1 byte Mode>: 'P'-Plaintext, 'A'-AES, 'T'-TDES
- 2 byte file descriptor length (LenL LenH) – The length of rest "File Descriptor", it always be 56.
- 2 byte block head size (byte) (SizeL SizeH):
  - It is -512 bytes (Block 1 Data: Version and Device Name…, and Block 2 Data: FM Check Value).
- 2 byte block tail size (byte) (SizeL SizeH):
  - It is 256 bytes (Random Number).
- 2 byte package size (in byte) (SizeL SizeH) – The byte size of package (m)
- 2 byte Package Numbers (NumberL NumberH) (N)
- 4 bytes Block 3 data First Package Start Address(Little Endian Mode)

Surplus bytes - Reserved

Block 1 Data

1. This Block data are 256 bytes data.

2. The 256 bytes data Format:

- 01 10 8 Bytes String "Version" + 8 Bytes Application Version need Ascii Code Data (Version 0.99.003 → Hex: 56 65 72 73 69 6F 6E 20 30 2E 39 39 2E 30 30 33)
  - Device need save the Version Number 1
- 02 x + x Bytes Device Name n need Ascii Code Data (UniPay II → Hex: 55 6E 69 50 61 79 20 49 49)
  - Device need verifythe Device Name. If it is not attributed to the device, response Error.
- 03 04 + 4 bytes Bootloader Version Number data (from 0x00 00 00 00, 0x00 00 00 01, …, to 0xFF FF FF FF)
- 04 01 + 1 byte Encryption Mode

Block 2 Data

1. This Block data are 256 bytes data - Firmware Check Value

2. The definition:
This Check Value is calculated from Full Packets Data of BIN File. If Full Data of BIN File (For Generating FM File) are not enough for Full Packages Data, FM File Generator Tool will Pad 0xFF to calculate the Check Value.

- RSSPSS RSA Raw Data Format, Device decrypts (using RSA-2048 algorithm & Firmware RSA Key) the 256 bytes data to be 256 bytes data (BASE Data):

After this Block data are received, Device decrypts (using RSA-2048 algorithm) the 256 bytes data to be 256 bytes data (BASE Data). The 256 bytes (BASE Data) should be saved in Special area. It is used for Bootloader Firmware.

Block 3 Data

1. This Block data are (N * m) bytes data.
It is Plaintext or Encrypted Firmware Data.
Per Package is m bytes data (At least 1K).
If Package Data of BIN File (For Generating FM File) are not enough for a total Package, FM File Generator Tool will Pad 0xFF.

2. The Block data could be Plaintext or Encryption.

3. The Encryption mode will be:
- Plaintext
- Only Firmware RSA Encryption Mode:
  - Per 256 bytes data of Plaintext BIN file will be Xor'ed 256 bytes (BASE Data) to be Encrypted Block 3 data. So Device need:
    - Get Base Data from Special area
    - Per 256 byte received data need be Xor'ed 256 bytes (BASE Data) to be Plaintext Block 3 data (Firmware Code).
- Firmware Fix Key Encryption Mode:
  - Need Load a fix key - Firmware Fix Key
  - All data of Plaintext BIN file will be encrypted by the key using CBC algorithm (IV is 0) to be Encrypted Block 3 data. So Device need decrypted received data by the key using CBC (IV is 0) to be Plaintext Block 3 data (Firmware Code).

Block 4 Data

It is End Flag.
After this Block data are received, Device Read per 256 bytes from Full Packets Data to Calculate per 32 bytes Signature Data, All these Signature Data are Xored to be a 32 bytes Signature Data (Sha-2 Signature Data 2).

- RSSPSS RSA Raw Data Format:
  - Device encrypts (using RSAPSS algorithm) the Below y bytes TLVn data to be 256 bytes data (BASE1 Data) , TLVn data are:

- <Tag 0x00> data: 00 <Len-32> <Sha-2 Signature Data 2>
- <Tag 0x02> data: 02 <Len x> <x bytes Device Name Data> (Such as 02 0E + "Smart PIN L100", or 01 0C + "MiniSmart II")

- BASE1 Data needs to be equal to BASE Data

Device Examples

Example 1
If Application Whole Size is 480K, BIN file is 396.6K, m is 1K.

The Check Value of Block 3 need be calculated from 397K data, FM Generator Tools need padding (397K – 396.6K) bytes 0xFF to calculate it.

Example 2
If Application Whole Size is 480K, BIN file is 400K, m is 2K.

The Check Value of Block 3 need be calculated from 400K data

Example 3
If ApplicationWhole Size is 480K, BIN file is 480K, m is 1K.

The Check Value of Block 3 need be calculated from 480K data.

Entering Bootloader Mode

1. Device enters into Bootloader Status method.
   While Device runs in Application IDLE Status, Host could send Command Commandy Body:
   78 46 7A 49 52 46 57 00 00 00 00 00 00 00 00 – Master Chip Bootloader
   78 46 7A 49 52 46 57 00 00 00 00 00 00 00 01 – Slave Chip Bootloader
   78 46 7A 49 52 46 57 FF FF FF FF 00 00 00 00 – Master Chip Bootloader need be changed
   78 46 7A 49 52 46 57 FF FF FF FF 00 00 00 01 – Slave Chip Bootloader need be changed

   Response Body:
   06 (Enter into Bootloader Status), or
   15 (Do Not Support Bootloader, or Do Not Enter into Bootloader Status).

2. Bootloader Work Status Display:

- Device with LED: LED display to indictor:
  - Dual-Color LED (Optional):
    - Without Battery Identifier: Flash Red & Green
    - With Battery Identifier: Flash Red (Battery is low), or Flash Green (Battery is enough)
  - Multi LEDs (Optional): Flash them
- Device with LCD, LCD display 'Bootloader …'

- Device without any Display Part, No Indictor.

Bootloader Status - Process
After entering into Bootloader mode, get status to determine device encryption, firmware packet size, and packet/block to start/resume the transmission on.

Command Body: 78 46 11

Response Body:
(Fix 28 bytes data): 06 <'**Bootloader FW** '> <PCI Indicator Flag> <' **Vx.yy**'> <Package Size_L> <Package Size_H> <4 bytes Request Indicator> Or

- <'**Bootloader FW** '>: 14 bytes Ascii Code Data, Hex Value is 42 6F 6F 74 6C 6F 61 64 65 72 20 46 57 20
- <PCI Indicator Flag>: 1 byte Ascii Code:
  - 0x31: PCI Device - Plaintext
  - 0x30: Non-PCI Device – Plaintext
  - 0x32: PCI Device – Encryption by TDES
  - 0x33: Non-PCI Device –Encryption by TDES
  - 0x34: PCI Device – Encryption by AES
  - 0x35: Non-PCI Device –Encryption by AES
  - 0x36: PCI Device – Encryption by Xored
  - 0x37: Non-PCI Device – Encryption by Xored
- <' **Vx.yy**' >: 7 bytes Ascii Code Data, Hex Value of " 1.00 " is 20 56 31 2E 30 30 20
- <Package Size_L> <Package Size_H>: 2 bytes data (m):
  - 1K size: 00 04
  - 2K size: 00 08
- <Request Package Num_L> <Request Package Num_H>:
  - For PCI device: Hex Value always 00 00.
  - For Non-PCI device: Indictor next Package Data requirement.
- <4 bytes Request Indicator>（Little Endian Mode）:
  - For PCI device Application & Bootloader Updating: Hex Value always 00 00 00 00
  - For Non-PCI device Application Updating: Indictor next Package Data requirement.
  - For Non-PCI device Bootloader Updating: Hex Value always 00 00 00 00

FM File Data - Process
Command Body is <42> <4 bytes Indicator> <Data (Xbytes)>
Where:
Type0 – 256 bytes Block 1 Data
42 00 00 00 00 <256 bytes Block 1 Data>

Type1 – 256 bytes Block 2 Data
42 01 00 00 00 <256 bytes Block 2 Data>

Type2 – m * 1K bytes Data of Block 3 Data

42 <4 bytes Flash Start Address of this Package Data> <m bytes Data>

- <4 bytes Flash Start Address of this Package Data>（Little Endian Mode）
- m will be 1024 or 2048……

Type3 – The latest 256 bytes Block 4 Data
42 FF FF FF FF <256bytes Data>

Response Body is
06 or
15 <Error Codes>

NOTE: The maximum interval between the last command and the new command is 30 seconds.
In Work State 1: If Timeout, Device response Error Code '0x5030', quit Bootloader Status and run in Application Status.
In Work State 2: If Timeout for PCI Device: Device Erase All Application. Device response Error Code '0x5031', does not quit Bootloader Status until finish implementing Bootloader function.

The interval between the command and the response is 20ms ~ 3 seconds

Full Commands with Packages Example – m=1K, N=397:

- 02 05 04 42 00 00 00 00 <Block 1 - 256 bytes data> <Lrc> <Sum> 03
- 02 05 04 42 01 00 00 00 <Block 2 - 256 bytes data "**Firmware Check Value**"> <Lrc> <Sum> 03
- 02 05 04 42 <4 bytes Address 1> <Block 3 - Package 1 - 1024 bytes of firmware> <Lrc> <Sum> 03
- 02 05 04 42 <4 bytes Address 2> <Block 3 - Package 2 - 1024 bytes of firmware> <Lrc> <Sum> 03
- …
- ……
- ……
- ……
- ……
- ……
- 02 05 04 42 <4 bytes Address 397> < Block 3 - Package 397 - 1024 bytes of firmware> <Lrc> <Sum> 03
- 02 05 01 42 FF FF FF FF <Block 4 - 256 bytes Data> <Lrc> <Sum> 03

Bootloader Error Codes

Error Codes	Definition	Software Process
0x5030	Timeout in Work State 1. Device quit Bootloader Status and run in Application Status.	Pause Bootloader function
0x5031	For PCI device: Timeout in Work State 2. Device need receive data from Block 0 Data. Device does not quit Bootloader Status until finish implementing Bootloader function.	Bootloader need send data from Block 1 Data.
0x5032	Data Error	Re-Send Current Package Data
0x5034	Application Version Error	Bootloader need send data from Block 1 Data.
0x5035	Erase flash or write flash Failed	Bootloader need send data from Block 1 Data.
0x5036	Firmware check value Error	Bootloader need send data from Block 1 Data.
0x5037	Device Name Error	Bootloader need send data from Block 1 Data.
0x5038	Encryption Mode Error	No firmware key，Bootloader only support plain text upgrade
0x5039	Firmware Address Error	Bootloader need send data from Block 1 Data.
0x6900	Invalid Command – Protocol is right, but task ID is invalid	
0x6A00	Unsupported Command – Protocol and task ID are right, but command is invalid	
0x6A01	Unsupported Command – Protocol and task ID are right, but command is invalid – In this State	
0x6B00	Unknown parameter in command – Protocol task ID and command are right, but parameter is invalid	
0x6C00	Unknown parameter in command – Protocol task ID and command are right, but length is out of the requirement.	

General Error Codes

Error Code	Definition
0x0400	Related Key was not loaded
0x0410	Non-SRED Device needs Load Manufacture Key, Firmware Key, and Check Values
0x0500	Key Same / Duplicate key detected
0x0702	PAN is Error
0x0D00	This Key was loaded
0x0F00	Encryption Or Decryption Failed
0x5500	No RKI-KEK
0x5501	RKI-KEK STOP
0x5504	Validate Authentication Code Error
0x5505	Encrypt Or Decrypt data failed
0x5506	Not Support the New Key Type
0x5507	New Key Index is Error
0x5508	Step Error
0x5509	Remote Key Injection Timeout (Latest Command is Timeout)
0x550A	MAC Error
0x550B	Key Usage Error
0x550C	Mode Of Use Error
0x550F	Other Error
0x6000	Save or Config Failed / Or Read Config Error, Flash Error
0x6200	No Serial Number
0x6900	Invalid Command – Protocol is right, but task ID is invalid
0x6A00	Unsupported Command – Protocol and task ID are right, but command is invalid
0x6A01	Unsupported Command – Protocol and task ID are right, but command is invalid – In this State
0x6B00	Unknown parameter in command – Protocol task ID and command are right, but parameter is invalid
0x6C00	Unknown parameter in command – Protocol task ID and command are right, but length is out of the requirement.
0x7300	DUKPT is STOP (21 bit 1)
0x7500	Self-Test Failed
0x8100	Timeout

0x8200	Wrong operate step
0x2C02	No Microprocessor ICC seated
0x2C06	No card seated to request ATR
0x8B10	ICC error on power-up
0xE313	IO line low -- Card error after session start
0x9042	Invalid LCL-KEK
0x9046	Invalid Data encryption Key
0x9047	Do not support this key
0x9052	Invalid RKI-KEK
0x9054	TR31 checks failed
0x9057	LCL-KEK exists
0xF002	ICC communication timeout
0xF003	ICC communication Error
0xF005	ICC Encrypted C-APDU Data Structure Length Error Or Format Error.
0xF200	AID List / Application Data does not exist
0xF201	Terminal Data does not exist
0xF202	TLV format is error
0xF203	AID List is full
0xF204	Any CA Key does not exist
0xF205	CA Key RID does not exist
0xF206	CA Key Index it not exist
0xF207	CA Key is full
0xF208	CA Key Hash Value is Error
0xF209	Transaction format error
0xF20A	The command will not be processing
0xF20B	CRL does not exist
0xF20C	CRL number exceed max number
0xF20D	Amount,Other Amount,Trasaction Type are missing
0xF20E	The Identification of algorithm is mistake
0xF20F	No Financial Card
0xF210	In Encrypt Result state, TLV total Length is greater than Max Length
0xF211	ICC L2 is not in idle state
0xF212	Transaction Type Error
0xF213	Major Config Error for Set Terminal Data

LCD Foreign Language Mapping Table

Some EMV responses include message IDs for messages that should be displayed on the POS, tablet, or LCD-equipped device. Consult the following table for those messages. Note that the ID (leftmost column) is decimal (not hex) notation.

ID	Message ID	English	French	Spanish	Chinese
0	MSG_NULL				
1	MSG_AMOUNT	AMOUNT	MONTANT	CANTIDAD	金额
2	MSG_AMOUNT_OK	AMOUNT OK?	MONTANT OK	MONTO CORRECTO?	确定金额
3	MSG_APPROVED	APPROVED	APPROUVE	APROVADO	通过
4	MSG_CALL_YOUR_BANK	CALL YOUR BANK	APPE VOTRE BANQE	LLAME A SU BANCO	请联系您的银行
5	MSG_CANCEL_OR_ENTER	CANCEL OR ENTER	ANNULE OU ENTRER	CANCEL O ENTRAR	取消或确定
6	MSG_CARD_ERROR	CARD ERROR	ERREUR CARTE	ERROR DE TARJETA	读卡错误
7	MSG_DECLINED	DECLINED	REFUSE	DECLINADO	卡被拒
8	MSG_ENTER_AMOUNT	ENTER AMOUNT	ENTRER MONTANT	INGRESE MONTO	输入金额
9	MSG_ENTER_PIN	ENTER PIN:	ENTRER PIN:	ENTRAR NPI:	请输入密码
10	MSG_INCORRECT_PIN	INCORRECT PIN	NIP INCORRECT	NPI INCORRECTO	密码错误
11	MSG_ICC_MSR1	SWIPE OR INSERT	PASSER OU INSERT	MOVER O INSERT	请刷卡或插卡
12	MSG_ICC_MSR2	CARD	CARTE	TARJETA	卡
13	MSG_INSERT_CARD	INSERT CARD	INSERT CARTE	INSERTAR TARJETA	请插卡
14	MSG_USE_CHIP_READER	USE CHIP READER	UTI LECTEUR CHIP	USO CHIP LECTOR	使用芯片卡
15	MSG_NOT_ACCEPTED	NOT ACCEPTED	PAS ACCEPTE	DENEGADO	无法接受
16	MSG_PIN_OK	GET PIN OK			密码正确
17	MSG_PLEASE_WAIT	PLEASE WAIT...	ATTENDRE...	POR FAVOR ESPERE	等候中
18	MSG_PROCESSING_ERROR	PROCESSING ERROR	ERREUR DE TRAITE	ERROR PROCESANDO	处理错误
19	MSG_USE_MAGSTRIPE	USE MAGSTRIPE	USAGE MAGSTRIPE	USO DE MAGSTRIPE	使用磁条卡
20	MSG_TRY_AGAIN	TRY AGAIN	REESSAYER	VUELV INTENTARLO	请重试
21	MSG_ONLINE	GO ONLINE	GO LIGNE	GO LINEA	在线
22	MSG_TRANSACTION_ERROR	TRANSACTION ERR	ERREUR DE TRANS	ERROR DE TRANSAC	交易错误
23	MSG_TERMINATE	TERMINATE	RESILIER	TERMINAR	终止
24	MSG_ADVICE	ADVICE	CONSEILS	CONSEJOS	建议
25	MSG_TIMEOUT	TIME OUT	TIMEOUT	TIEMPO DE ESPERA	超时
26	MSG_PROCESSING	PROCESSING...	PROCESSUS...	PROCESANDO...	处理中。。。
27	MSG_PIN_TRY_EX	PIN TRY LIMIT EX	PIN TRY DEPASSE	TRY PIN SUPERADA	密码尝试次数过多
28	MSG_ISSUER_AUTH_FAIL	ISSUER AUTH FAIL	EMETTEUR FAIL	EMISOR FALLA	与发卡机构认证
29	MSG_CONTINUE_PROCESS	CONTINUE PROCESS	CONTINUER LA	CONTINUAR PROCES	继续处理
30	MSG_GET_PIN_ERROR	GET PIN ERROR	GET PIN ERROR	OBTENER PIN ERR	密码错误
31	MSG_GET_PIN_FAIL	GET PIN FAIL	GET PIN FAIL	OBTENER PIN FALL	获取密码错误
32	MSG_NOKEY_GET_PIN	NO KEY GET PIN	NO KEY GET PIN	NO CLAVE GET PIN	无法输入密码
33	MSG_CANCELLED	CANCELLED	ANNULE	CANCELADO	取消
34	MSG_LAST_PIN_TRY	LAST PIN TRY			最后一次输入密码

Response Codes

Response Code(2Bytes)	Display on the LCD
0x00,0x00	APPROVED (offline)
0x00,0x01	DECLINED(offline)
0x00,0x02	APPROVED
0x00,0x03	DECLINED
0x00,0x04	GO ONLINE
0x00,0x05	CALL YOUR BANK
0x00,0x06	NOT ACCEPTED
0x00,0x07	USE MAGSTRIPE
0x00,0x08	TIME OUT
0x00,0x10	(start transaction success)
0x00,0x11	MSR Success
0x10, 0x01 (FILE_ARG_INVALID)	TERMINATE
0x10,0x02 (FILE_OPEN_FAILED)	TERMINATE
0x10,0x03 (FILE_OPERATION_FAILED)	TERMINATE
0x20, 0x01 (MEMORY_NOT_ENOUGH)	TERMINATE
0x30, 0x01 (SMARTCARD_OK)	TERMINATE
0x30, 0x02 (SMARTCARD_FAIL)	TERMINATE
0x30, 0x03 (SMARTCARD_INIT_FAILED)	TERMINATE
0x30,0x04 (FALLBACK_SITUATION)	TERMINATE
0x30, 0x05 (SMARTCARD_ABSENT)	TERMINATE
0x30,0x06	TERMINATE

(SMARTCARD_TIMEOUT)	
0x30,0x07 (MSR_CARD_ERROR)	TERMINATE
0x50, 0x01 (PARSING_TAGS_FAILED)	TERMINATE
0x50, 0x02 (CARD_DATA_ELEMENT_DUPLICATE)	TERMINATE
0x50, 0x03 (DATA_FORMAT_INCORRECT)	TERMINATE
0x50,0x04 (APP_NO_TERM)	NOT_ACCEPTED
0x50, 0x05 (APP_NO_MATCHING)	NOT_ACCEPTED
0x50,0x06 (MANDATORY_OBJECT_MISSING)	TERMINATE
0x50, 0x07 (APP_SELECTION_RETRY)	TERMINATE
0x50, 0x08 (AMOUNT_ERROR_GET)	TERMINATE
0x50, 0x09 (CARD_REJECTED)	TERMINATE
0x50, 0x10 (AIP_NOT_RECEIVED)	TERMINATE
0x50, 0x11 (AFL_NOT_RECEIVED)	TERMINATE
0x50, 0x12 (AFL_LEN_OUT_OF_RANGE)	TERMINATE
0x50, 0x13 (SFI_OUT_OF_RANGE)	TERMINATE
0x50, 0x14 (AFL_INCORRECT)	TERMINATE
0x50, 0x15 (EXP_DATE_INCORRECT)	TERMINATE
0x50, 0x16 (EFF_DATE_INCORRECT)	TERMINATE
0x50, 0x17 (ISS_COD_TBL_OUT_OF_RANGE)	TERMINATE
0x50, 0x18 (CRYPTOGRAM_TYPE_INCORRECT)	TERMINATE
0x50, 0x19 (PSE_BY_CARD_NOT_SUPPORTED)	TERMINATE
0x50, 0x20	TERMINATE

(USER_LANGUAGE_SELECTED)	
0x50, 0x21 (SERVICE_NOT_ALLOWED)	NOT_ACCEPTED
0x50, 0x22 (NO_TAG_FOUND)	TERMINATE
0x50, 0x23 (CARD_BLOCKED)	TERMINATE
0x50, 0x24 (LEN_INCORRECT)	TERMINATE
0x50, 0x25 (CARD_COM_ERROR)	TERMINATE
0x50, 0x26 (TSC_NOT_INCREASED)	TERMINATE
0x50, 0x27 (HASH_INCORRECT)	TERMINATE
0x50, 0x28 (ARC_NOT_PRESENCED)	TERMINATE
0x50, 0x29 (ARC_INVALID)	TERMINATE
0x50, 0x30 (COMM_NO_ONLINE)	TERMINATE
0x50, 0x31 (TRAN_TYPE_INCORRECT)	TERMINATE
0x50, 0x32 (APP_NO_SUPPORT)	TERMINATE
0x50, 0x33 (APP_NOT_SELECT)	TERMINATE
0x50, 0x34 (LANG_NOT_SELECT)	TERMINATE
0x50, 0x35 (TERM_DATA_NOT_PRESENCED)	TERMINATE
0x60, 0x01 (CVM_TYPE_UNKNOWN)	TERMINATE
0x60, 0x02 (CVM_AIP_NOT_SUPPORTED)	TERMINATE
0x60, 0x03 (CVM_TAG_8E_MISSING)	TERMINATE
0x60, 0x04 (CVM_TAG_8E_FORMAT_ERROR)	TERMINATE
0x60, 0x05 (CVM_CODE_IS_NOT_SUPPORTED)	TERMINATE
0x60, 0x06	TERMINATE

(CVM_COND_CODE_IS_NOT_SUPPORTED)	
0x60, 0x07 (CVM_NO_MORE)	TERMINATE
0x60, 0x08 (PIN_BYPASSED_BEFORE)	TERMINATE
Error Result Code	TERMINATE

Note:
First response byte format:
Bit 0 --- if transaction have advice, this bit is 1.
Bit 1 --- if transaction have reversal, this bit is 1.

Timings: General Purpose Commands

Description	Specific description	Send	Response Data	Response Time (ms)
General Purpose Commands				
Get Firmware Release Version		78 46 01	02 1A 00 06 49 44 54 45 43 48 20 4D 69 6E 69 53 6D 61 72 74 20 49 49 20 56 31 2E 30 30 02 52 03	6
Enter into Bootloader		78 46 7A 49 52 46 57 00	02 03 00 15 6A 00 7F 7F 0	
Get Serial Number	06 + 10 bytes ASCII code Serial Number Or 15 62 00 – No Serial Number	78 46 02	02 0b 00 06 20 32 33 34 3	6
Get Model Number	USB	78 46 20	02 09 00 06 4D 49 4E 49 3	6
Reset		78 46 49	02 01 00 06 06 06 03	6
Set Remote Key Injection Timeout (78 53 01 01 02 Timeout_H Timeout_L(Timeout_ H Timeout_L need be 120 seconds ~ 3600 seconds))	120 sec	78 53 01 01 02 00 78	02 01 00 06 06 06 03	6
	3600 sec	78 53 01 01 02 0e 10	02 01 00 06 06 06 03	6
	error command	78 53 01 01 02 00 77	02 03 00 15 6B 00 7E 80 0	6
Get Remote Key Injection Timeout	06 78 01 01 02 Timeout_	78 52 01 01	02 07 00 06 78 01 01 02 0	6
Default General Group	Remote Key Injection T	78 53 00	02 01 00 06 06 06 03	6
Set Date & Time	Command Body is 78 53 01 50 06 <Date Time> Where: <Data Time> is 6 bytes data – Year, Month, Date, Hour, Minute, Second	78 53 01 50 08 06 99 12	02 01 00 06 06 06 03	6
		78 53 01 50 08 06 15 01	02 01 00 06 06 06 03	6
		78 53 01 50 06 00 01 01	02 03 00 15 6a 00 7f 7f 03	6
Get Date & Time	Response Body is 06 78 01 50 06 <Date Time>	78 52 01 50	02 0B 00 06 78 01 50 06 0	6
Review General Group	06 78 02 01 02 <Timeout	78 52 00	02 0F 00 06 78 02 01 02 E	6

Smart Card Group (ICC EMV Level				
Get ICC Reader Status	06 + <Reader status> (1 byte) no seated no power on	72 46 24	02 02 00 06 00 06 06 03	6
	seated but no power on	72 46 24	02 02 00 06 02 04 08 03	6
	seated and power on	72 46 24	02 02 00 06 03 05 09 03	6
Power On (Get ATR)	06 + <ATR String>	72 46 6E	02 14 00 06 3B 6F 00 00 80 25 A0 00 00 00 68 54 08 00 0D 40 82 90 00 3C 18 03	243
Power Off		72 46 4D	02 01 00 06 06 06 03	6
Get EMV Level One Version Numb	06 <"IFM Vx.yy" >.	72 46 23 01	02 0A 00 06 49 4D 46 20 5	6
Smart Card Group – Set/Get Com				
Set Pre/Post PAN Data Len	Command Body is 72 53 01 20 02 <PrePANCtlDataLen> <PostPANCtlDataLen> PrePANCtlDataLen> need be 0~6, Default is 4 <PostPANCtlDataLen> need be 0~4, Default is 4	72 53 01 20 02 00 00	02 01 00 06 06 06 03	6
		72 53 01 20 02 06 04	02 01 00 06 06 06 03	6
		72 53 01 20 02 07 04	02 03 00 15 6B 00 7E 80 03	6
Get Pre/Post Data Len		72 52 01 20	06 72 01 20 02 06 04	6
Set ASCII Mask Data	Command Body is 72 53 01 21 01 <AsciiMaskData> Where: <AsciiMaskData> can be 0x20~0x7E, Default is 0x2A(*).	72 53 01 21 01 20	02 01 00 06 06 06 03	6
		72 53 01 21 01 7e	02 01 00 06 06 06 03	6
		72 53 01 21 01 7f	02 03 00 15 6B 00 7E 80 03	6
Get ASCII Mask Data		72 52 01 21	06 72 01 21 01 7e	6
Set ASCII Mask Data	Command Body is 72 53 01 21 01 <AsciiMaskData> Where: <AsciiMaskData> can be 0x20~0x7E, Default is 0x2A(*).	72 53 01 21 01 20	02 01 00 06 06 06 03	6
		72 53 01 21 01 7e	02 01 00 06 06 06 03	6
		72 53 01 21 01 19	02 03 00 15 6B 00 7E 80 03	6
Get ASCII Mask Data		72 52 01 21	06 72 01 21 01 7e	6

Set BCD Mask Data	Command Body is 72 53 01 22 01 <BCDMaskData> Where: <BCDMaskData> can be can be 0x0A~0x0F, Default is 0x0C(*). Note: l If 0x23 will be masked High BCD, result should be 0xC3 l If 0x23 will be masked Low BCD, result should be 0x2C l If 0x23 will be masked all BCD, result should be 0xCC	72 53 01 22 01 0A	02 01 00 06 06 06 03	6
		72 53 01 22 01 0F	02 01 00 06 06 06 03	6
		72 53 01 22 01 0C	02 01 00 06 06 06 03	6
		72 53 01 22 01 10	02 03 00 15 6B 00 7E 80 0	6
		72 53 01 22 01 09	02 03 00 15 6B 00 7E 80 0	6
Get BCD Mask Data		72 52 01 22	06 72 01 22 01 0C	6
Set Key Type for Data encryption Key (ICC DUKPT Key)	Data type Key	72 53 01 01 01 00	02 01 00 06 06 06 03	6
	PIN type Key	72 53 01 01 01 01	02 01 00 06 06 06 03	6
Get Key Type for Data encryption Key (ICC DUKPT Key)	06 72 01 01 01 <Option>	72 52 01 01	02 06 00 06 72 01 01 01 0	6
Set Encryption Mode for Data encryption Key (ICC DUKPT Key)	06 (ICC DUKPT Key existed) or 15 6A 00 (NAK + Unsupported Command Error Code) (ICC DUKPT Key did not exist) TDES	72 53 01 02 01 00	02 01 00 06 06 06 03	6
	AES	72 53 01 02 01 01	02 01 00 06 06 06 03	6
Get Encryption Mode for Data encryption Key (ICC DUKPT Key)	NONE (06 72 01 02 01 FF) (should send 7f 53 00 erase key)	72 52 01 02	02 06 00 06 72 01 02 01 F	6
	TDES(should loadkye af	72 52 01 02	02 06 00 06 72 01 02 01 0	6
	AES(should loadkye aft	72 52 01 02	02 06 00 06 72 01 02 01 0	6
Set Card Type Option	EMV	72 53 01 04 01 FF	02 01 00 06 06 06 03	6
	IOS	72 53 01 04 01 00	02 01 00 06 06 06 03	6

Get Card Type Option	06 72 01 04 01 <Option>	72 52 01 04	02 06 00 06 72 01 04 01 0	6
Set ICC L1 Transaction Timeout		72 53 01 05 01 <Timeou	02 06 00 06 72 01 05 01 0	6
Get ICC L1 Transaction Timeout		72 52 01 05	02 01 00 06 06 06 03	6
Default ICC Group All Setting	Key Type:Data Key, Encryption Mode:DES/TDES (ICC DUKPT Key existed) No change (ICC DUKPT Key did not exist) Card Type:EMV	72 53 00	02 01 00 06 06 06 03	6
Review ICC Group All Setting	Response Body is 06 72 0306 01 01<Key Type Option> 02 01 <Encryption Mode Option> 04 01 <Card Type Option> 20 02 <PrePANCtlDataLen> <PostPANCtlDataLen> 21 01 <AsciiMaskData> 22 01 <BCDMaskData>	72 52 00	02 16 00 06 72 06 01 01 00 02 01 FF 04 01 FF 20 02 04 04 21 01 2A 22 01 0C 73 2B 03	6
Key Loading Protocol				
Get Key Status	Account/ICC DUKPT Key: 0: None. 1: Exist 0xFF: STOP	78 46 30	02 07 00 06 00 00 00 00 0	6
		78 46 25		6
	Admin DUKPT Key: 0: None. 1: Exist 0xFF: STOP (when set Admin DUKPT key will be erase)	78 46 30	02 07 00 06 00 00 00 00 0	6
Get KSN	06 + <10 bytes KSN>	72 46 62	02 0B 00 06 62 99 49 01 6	6
	No Microprocessor ICC	72 46 62	02 03 00 15 2C 02 3B 43 0	6
	Related Key was not lo	72 46 62	02 03 00 15 04 00 11 19 0	6

	IO line low -- Card error	72 46 62	02 03 00 15 E3 13 E5 0B 0	6
Exchange APDU Plaintext	72 46 41 <C-APDU>	72 46 41 <C-APDU>	06 + 00 + <R-APDU>	depend on APDU length and card
Exchange APDU Encryption for sp	72 46 61	72 46 61<C-APDU>		depend on APDU length and card
Exchange APDU Encryption	force encryption APDU	72 46 63 <C-APDU>		depend on APDU

Timings: EMV Commands

Command	Details	Response Time (ms)
Retrieve Application Data	Command Body is 72 46 01 01 <LenL><LenH> <5~16 bytes AID> <- Where: <LenL><LenH> is the length of AID Response Body is 06 <TagCounterL> <TagCounterH> <TLV1> <TLV2>…<TLVn> or 15<ErrorCode>. Where: <TagCounterL> <TagCounterH> is the number of <TLV> Note: If AID List / Application Data does not exist, response 15 F2 00.	5.9
Remove Application Data	Command Body is 72 46 01 02 <LenL><LenH> <5~16 bytes AID> Where: <LenL><LenH> is the length of AID Response Body is 06 or 15<ErrorCode> Note: If AID List / Application Data does not exist, response 15 F2 00.	9.9
Set Application Data	Command Body is 72 46 01 03 <LenL><LenH> <5~16 bytes AID> <TagCounterL> <TagCounterH> <TLV1> <TLV2>…<TLVn>. Where: <TagCounterL> <TagCounterH>is the Number of <TLV>. <LenL><LenH> is the length of AID Response Body is 06 or 15 <ErrorCode> Note:	17

	If a <TLV> format was error, response 15 - F2 02 -If AID List has full (MAX is 16), response 15 F2 03	
Remove All Application Data	Command Body is 72 46 01 04 Response Body is 06	application data existed: 58 no application data:6
Retrieve Terminal Data	Command Body is 72 46 02 01 Response Body is 06 <TagCounterL> <TagCounterH> <TLV1> <TLV2>…<TLVn> or 15<ErrorCode> Where: <TagCounterL> <TagCounterH> is the number of <TLV> If Terminal Data does not exist, response 15 F2 01	10
Remove Terminal Data	Command Body is 72 46 02 02 Response Body is 06	14
Set Terminal Data	Command Body is 72 46 02 03 <TagCounterL> <TagCounterH> <TLV1> <TLV2>…<TLVn>. Where: <TagCounterL> <TagCounterH>: the Number of <TLV>. Response Body is 06 or 15 <ErrorCode> Note: If a <TLV> format was error, response 15 F2 02	30
Retrieve AID List	Command Body is 72 46 03 01 Response Body is 06 <NumberL><NumberH> <AID Block 1> <AID Block 2> … <AID Block N>. Where: <NumberL><NumberH> is Number of AID Blocks. <AID Block> format is <LenL> <LenH> <Several bytes AID>. Where: <LenL> <LenH> is Length of AID Note: If AID List does not exist, response 15 F2 00	6
Retrieve CA Public Key	Command Body is 72 46 04 01<5 bytes RID> <1 byte Index>. Response Body is 06 <5 bytes RID> <1 byte Index> <1 byte Hash Algorithm> <1 byte Encryption Algorithm> <20 bytes HashValue> <4 bytes Public Key Exponent> <2 bytes Modulus Length> <Variable bytes Modulus> Where: l <Hash Algorithm>: The only algorithm supported is SHA-1.The value is set to 0x01 l <Encryption Algorithm>: The encryption algorithm in which this key is used. Currently support only one type: RSA. The value is set to 0x01. l <HashValue>: Which is calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent l <Public Key Exponent>: Actually, the real length of the exponent is either one byte or 3 bytes. It can have two values: 3 (Format is 0x00 00 00 03), or 65537 (Format is 0x00 01 00 01)	5.9

	l <Modulus Length>: <LenL> <LenH> Indicated the length of the next field. l <Modulus>: This is the modulus field of the public key. Its length is specified in the field above. Note: If CA Key RID does not exist, response 15 F2 05 If CA Key Index does not exist, response 15 F2 06	
Remove CA Public Key	Command Body is 72 46 04 02 <5 bytes RID> <1 byte Index>. Response Body is 06 Note: If CA Key RID does not exist, response 15 F2 05 If CA Key Index does not exist, response 15 F2 06	CA Key existed:10 CA Key RID does not exist: 6
Set CA Public Key	Command Body is 72 46 04 03 <5 bytes RID> <1 byte Index> <1 byte Hash Algorithm> <1 byte Encryption Algorithm> <20 bytes HashValue> <4 bytes Public Key Exponent> <2 bytes Modulus Length> <Variable bytes Modulus> Where: l <Hash Algorithm>: The only algorithm supported is SHA-1.The value is set to 0x01 l <Encryption Algorithm>: The encryption algorithm in which this key is used. Currently support only one type: RSA. The value is set to 0x01. l <HashValue>: Which is calculated using SHA-1 over the following fields: RID & Index & Modulus & Exponent l <Public Key Exponent>: Actually, the real length of the exponent is either one byte or 3 bytes. It can have two values: 3 (Format is 0x00 00 00 03), or 65537 (Format is 0x00 01 00 01) l <Modulus Length>: <LenL> < LenH> Indicated the length of the next field. l <Modulus>: This is the modulus field of the public key. Its length is specified in the field above. Response Body is 06 Note: Per <RID> has 6 CA Public Key. Supported CA public key Max number is 16. If key has full, response 15 F2 07 If CA Key Hash Data is error, response 15 F2 08 If <Hash Algorithm> and <Encryption Algorithm> are not is 0x01, response 15 F2 0E.	18
Remove All CA Public Key List	Command Body is 72 46 04 04. Response Body is 06	no CA public Key: 6 CA public Key exist: 14
Retrieve CA Public Key List	Command Body is 72 46 04 05. Response Body is 06 <LenL> <LenH> <5Bytes RID1> <1 byte RID1 Index><5Bytes RID2> <1 byte RID2 Index>….. <5Bytes RIDN> <1 byte RIDN Index>. Note: If any CA Key does not exist, response 15 F2 04	CA Key existed:25 CA Key does not exist:5.9
Start Transaction	Command Body is: 72 46 05 01- <FallBack><TimeOut1> <TimeOut2> <App Data>. Where: <FallBack> (1byte). 0x01 indicates it supports FallBack to	First return:6 Expired Visa T = 1: 3.5sc Amex Retailer Terminal Test

	MSR, 0x00 indicate it not support FallBack. <TimeOut1> (2 bytes, unit is Second). Timeout for Card is seated. <TimeOut2> (2 bytes, unit is Second). Waiting time till "Authenticate Transaction" command. <App Data> format is <TLV1> <TLV2> … <TLVn>. Refer to Tanscation Data & Option Data List. Response Body is: First return: 06 Wait in seconds. Second return: 06 <Response Code> <Attribution> <Output Data List>. Where: <Response Code> length is 2Bytes. Please refer to "SSL2 Module Design Spec" <Attribution>: 1 Byte: l BIT0 – Card Type: 0 – Contact Card l BIT2,1 – Encryption Mode: 00 – TDES Mode, 01 – AES Mode l BIT7~3 - Reserve <Output Data List> format is <TLV1> <TLV2> … <TLVn>, refer to 7.7.6.20:Output Data List. Note in Transaction: 1. If have error in process, terminal will return error code and terminate transaction. 2. If command format error, terminal will response 15 F2 09 3. If no any application data in terminal, response 15 F2 00 4. If no terminal data in terminal, response 15 F2 01 After transaction start, terminal only can receive "Start Transaction", "Authenticate Transaction", "Cancel Transaction" and "Retrieve Transaction Data" command. If any other command was sent, it response 15 F2 0A 5. If timeout occurred, response 15 81 00. 6. If a tag is not occurred in the ICC, the tag in output TLV list will not occur all the same. 7. Amount , other amount, trsanction type must be exist. If not, terminal will response error code 15 F2 0D. 8. For MSR operation, the <Output Data List> is MSR data. TLVs are not included. 9. For MSR fallback operation, the <Response Code> is 00 07. 10. For MSR whose service code is not 2 or 6, the <Response Code> is 00 11.	Card :2.9sc RCTP International Maestro T = 1: 5.4sc Visa Credit Signature Test Card: 4.9sc Visa Credit Card Test:4.9sc RCTP Visa Electron T = 1 Card:4.9sc Diners Retailer Terminal Card:4.3sc JCB Retailer Terminal Test Card:3.8sc Retailer Terminal Test Card MasterCard Credit Card:5.1sc Retailer Terminal Test Card MasterCard Debit Test Card:5.0sc
Authenticate Transaction	Command Body is 72 46 05 02<ForeOnline><TimeOut> <App Data>-. <ForeOnline>(1byte). 0x01 indicates it supports ForeOnline,0x00 indicate not support. <TimeOut> (2 byte, unit is Second).means terminal waiting time for host response when online.- <App Data> format is <TLV> (V is output tag list.) Response Body is: First return: 06 Wait in seconds. Second return: 06 <Response Code> <Attribution> <TLV1> <TLV2> … <TLVn>. Where: <Response Code> length is 2Bytes. Please refer to "SSL2 Module Design Spec".	First return:96 Expired Visa T = 1: 586 Amex Retailer Terminal Test Card :592 RCTP International Maestro T = 1: 1.2sc Visa Credit Signature Test Card: 642 Visa Credit

	<Attribution>: 1 Byte: l BIT0 – Card Type: 0 – Contact Card l BIT2,1 – Encryption Mode: 00 – TDES Mode, 01 – AES Mode l BIT7~3 - Reserve Note: 1. If command format error, terminal will response 15 F2 09 2. If a Option tag is not occurred in the Option Data List, the tag in output TLV list will not occur all the same. 3. If the tag '9f10' and '9f26' have value in the terminal, they will also occur in the command response TLV list.	Card Test:639 RCTP Visa Electron T = 1 Card:644 Diners Retailer Terminal Card:745 JCB Retailer Terminal Test Card:944 Retailer Terminal Test Card MasterCard Credit Card:752 Retailer Terminal Test Card MasterCard Debit Test Card:752
Complete Transaction	Command Body is 72 46 05 03 <1Byte ComFlag> [<Authorization Response Code (TLV,Tag 8A)>< Issuer Authentication Data (TLV, Tag 91)>< <Scripts (TLV, Tag 71/72)>] <App Data> Where: <1Byte ComFlag>:0x01 indicate online with host,0x00 indicate unable online. Data in [ ] indicate these data is optional: l If ComFlag is 0x01, the Data exist. l If ComFlag is 0x00, the Data does not exist. <App Data> format is <TLV> (V is output tag list.) Response Body is: First return: 06 Wait in seconds. Second return: 06 <Response Code> <Attribution> <Output TLV Data>. Where: <Response Code>length is 2Bytes. Please refer to "SSL2 Module Design Spec" <Attribution>: 1 Byte: l BIT0 – Card Type: 0 – Contact Card l BIT2,1 – Encryption Mode: 00 – TDES Mode, 01 – AES Mode l BIT7~3 - Reserve Note: If any parameter was error, response 15F2 02.	First return:91 Expired Visa T = 1: 754 Amex Retailer Terminal Test Card :704 RCTP International Maestro T = 1: 920 Visa Credit Signature Test Card: 774 Visa Credit Card Test:770 RCTP Visa Electron T = 1 Card:776 Diners Retailer Terminal Card:544 JCB Retailer Terminal Test Card:704 Retailer Terminal Test Card MasterCard Credit Card:545 Retailer Terminal Test Card

		MasterCard Debit Test Card:562
Cancel Transaction	Command Body is 72 46 06. Response Body is 06. Retrieve Transaction Result Command Body is 72 46 07 01<2 Byte Length><Tags>. <2 Byte Length>format is<LenL><LenH>,is the length of <Tags>. <Tags> these tags will return TLV format. Supported tags refer to “Option Data List” Table in Section “Reference Data List”. Tags Example: “9F02,9F36,95,9F37,” means total 4 tags (9F02, 9F36, 95, 9F37) reqested to in response. Length is 7bytes. Response Body is 06<1Byte Message type><TLV1> <TLV2> … <TLVn>. For details please refer to P127 of EMV4.3 book3. <1Byte Message type>,value 0x00 means transaction message, value 0x01 means advice message. Note: If a Option tag is not occurred in the Option Data List, the tag in output TLV list will not occur all the same.	88
Get EMV Level Two Version Number	Command Body is 72 46 08 01 Response Body is 06<” EMVL2 X.YY” >. X.YY is version number. Fist version number is 1.00.	6
Retrieve Interface Device Serial Number	Command Body is 72 46 86 01 Response Body is 06 <Serial Number> Note: Set Interface device's serial number. EMV serial Number can be set only once.	6
Set Interface Device Serial Number	Command Body is 72 46 86 03 <Serial Number> Note: <Serial Number> 8 bytes ‘0’ ~ ‘9’, or ‘a’ ~ ‘z’, or ‘A’ ~ ‘Z’ Response Body is 06	
Retrieve Terminal Identification	Command Body is 72 46 87 01 Response Body is 06 <Identification>	6
Set Terminal Identification	Command Body is 72 46 87 03 <Identification> Note: <Identification> 8 bytes ‘0’ ~ ‘9’, or ‘a’ ~ ‘z’, or ‘A’ ~ ‘Z’ Response Body is 06	6
Retrieve Certification Revocation List	Command Body is 72 46 85 01. Response Body is 06<2Byte Length> <CRL1> <CRL2>…<CRLn>. <2Byte Length>: <Low byte of length> <High byte of length> <CRL>format is <5Bytes RID> <1Byte CA public key Index> <3Bytes Certificate Serial Number> <3Bytes Date>. <3Bytes Date> is the date that certificate was added to the revocation list. Format is YYMMDD. Note: If have no CRL exist, response 15 F2 0B.	5.9

Remove Certification Revocation List	Command Body is 72 46 85 02 <2 Bytes Length> <CRL1> <CRL2>…<CRLn> <2Byte Length>: <Low byte of length> <High byte of length> <CRL>format is <5Bytes RID> <1Byte CA public key Index> <3Bytes Certificate Serial Number>Response Body is 06.	6
Set Certification Revocation List	Command Body is 72 46 85 03 <2 Bytes Length of CRL> <CRL1> <CRL2>…<CRLn> <2Bytes MAC Length>. Where: <2Byte Length of CRL>: <Low byte of length> <High byte of length> <CRL> format is <5Bytes RID> <1Byte CA public key Index> <3Bytes Certificate Serial Number> <2Bytes MAC Length> is 2 bytes data – Fix is 0x00 0x00 Response Body is 06. Note: Supported at least CRL number is 30 for a RID.	1 CRL :5.9 CRL number is 30 : 33
Remove All Certification Revocation List	Command Body is 72 46 85 04 Response Body is 06.	6
Get ICC L2 Kernel Check Value	Command Body is 72 46 09 01 Response Body is 06 <20 bytes L2 Kernel Check Value>	6
Get ICC L2 Configuration Check Value	Command Body is 72 46 09 02 Response Body is 06 <20 bytes L2 Configuration Check Value> Note: Panasonic MiniSmart II Support 4C configuration.	6
Remove Transaction Amount Log	Command Body is 72 46 0A 02 Response Body is 06	6

EMV Tags

Most of the following tags are industry-standard EMVCo tags. Three-byte tags (e.g. DFEE15) are ID TECH proprietary.

<table>
<tr><th>Tag</th><th>Description</th><th>Format</th><th>Length</th></tr>
<tr><td>5F2A</td><td>Transaction Currency Code (Default: 08 40)</td><td>n3</td><td>2</td></tr>
<tr><td>5F36</td><td>Transaction Currency Exponent</td><td>n1</td><td>1</td></tr>
<tr><td>5F57</td><td>Account Type Selection</td><td>n2</td><td>1</td></tr>
<tr><td>8A</td><td>Authorization Response Code</td><td>b</td><td>2</td></tr>
<tr><td>98</td><td>Transaction Certificate (TC) Hash Value</td><td>b</td><td>20</td></tr>
<tr><td>99</td><td>Transaction Personal Identification Number (PIN) Data</td><td>b</td><td>1~64</td></tr>
<tr><td>9A</td><td>Transaction Date (YYMMDD)</td><td>n6</td><td>3</td></tr>
<tr><td>9B</td><td>Transaction Status Information</td><td>b</td><td>2</td></tr>
<tr><td>9C</td><td>Transaction Type</td><td>n2</td><td>1</td></tr>
<tr><td>9F01</td><td>Acquirer Identifier</td><td>n6-11</td><td>6</td></tr>
<tr><td>9F02</td><td>Amount, Authorized (Numeric)</td><td>n12</td><td>6</td></tr>
<tr><td>9F03</td><td>Amount, Other (Numeric)</td><td>n12</td><td>6</td></tr>
<tr><td>9F04</td><td>Amount, Other (Binary)</td><td>b</td><td>4</td></tr>
<tr><td>9F06</td><td>Application Identifier (AID) – terminal</td><td>b</td><td>5-16</td></tr>
<tr><td>9F09</td><td>Application Version Number (Default: 00 02)</td><td>b</td><td>2</td></tr>
<tr><td>9F15</td><td>Merchant Category Code</td><td>n4</td><td>2</td></tr>
<tr><td>9F16</td><td>Merchant Identifier</td><td>ans 15</td><td>15</td></tr>
<tr><td>9F1A</td><td>Terminal Country Code (Default: 08 40)</td><td>n3</td><td>2</td></tr>
<tr><td>9F1B</td><td>Terminal Floor Limit</td><td>b</td><td>4</td></tr>
<tr><td>9F1C</td><td>Terminal Identification</td><td>an 8</td><td>8</td></tr>
<tr><td>9F1D</td><td>Terminal Risk Management Data</td><td>b</td><td>1-8</td></tr>
<tr><td>9F1E</td><td>Interface Device (IFD) Serial Number (Default: 31 32 33 34 35 36 37 38)</td><td>an 8</td><td>8</td></tr>
<tr><td>9F21</td><td>Transaction Time (HHMMSS)</td><td>n6</td><td>3</td></tr>
<tr><td>9F22</td><td>Certification Authority Public Key Index</td><td>b</td><td>1</td></tr>
<tr><td>9F33</td><td>Terminal Capabilities (Default: E0 F8 C8)
Byte 1
<table>
<tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning</th></tr>
<tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Manual key entry</td></tr>
<tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Magnetic stripe</td></tr>
<tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>IC with contacts</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>RFU</td></tr>
</table>
Byte 2
<table>
<tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning</th></tr>
<tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Plaintext PIN for IC verification</td></tr>
<tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>X</td><td>x</td><td>Enciphered PIN for online verification</td></tr>
<tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>X</td><td>x</td><td>Signature(paper)</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>X</td><td>x</td><td>Enciphered PIN for offline verification</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>X</td><td>x</td><td>No CVM Required</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>RFU</td></tr>
</table></td><td>b</td><td>3</td></tr>
</table>

	x x x x x x X 0 RFU Byte 3 b8 b7 b6 b5 b4 b3 b2 b1 Meaning 1 x x x x x x x SDA X 1 x x x x x x DDA X x 1 x x x x x Card capture x x x 0 x x x x RFU x x x x 1 x x X CDA x x x x x 0 x x RFU x x x x x x 0 x RFU x x x x X X x 0 RFU		
9F34	Cardholder Verification Method (CVM) Results	b	3
9F35	Terminal Type (Default: 22) Environment / Financial Institution / Merchant / Cardholder **Attended** Online only: 11 / 21 / -- Offline with online capability: 12 / 22 / -- Offline only: 13 / 23 / -- **Unattended** Online only: 14 / 24 / 34 Offline with online capability: 15 / 25 / 35 Offline only: 16 / 26 / 36	b	1
9F37	Unpredictable Number	b	4
9F39	POS Entry Mode (Default: 07)	b	1
9F3A	Amount, Reference Currency	b	4
9F3C	Transaction Reference Currency Code	n3	2
9F3D	Transaction Reference Currency Exponent	n1	1
9F40	Additional Terminal Capabilities (Default: F0 00 F0 A0 01) Byte 1 b1 b2 b3 b4 b5 b6 b7 b8 Meaning 1 x x x x x x x Cash x 1 x x x x x x Goods x x 1 x x x x x Services x x x 1 x x x x Cashback x x x x 1 x x x Inquiry x x x x x 1 x x Transfer x x x x x x 1 x Payment x x x x x x x 1 Administrative Byte 2 b8 b7 b6 b5 b4 b3 b2 b1 Meaning 1 x x x x x x x Cash Deposit x 0 x x x x x x RFU x x 0 x x x x x RFU x x x 0 x x x x RFU x x x x 0 x x x RFU x x x x x 0 x x RFU x x x x x x 0 x RFU x x x x x x x 0 RFU Byte 3 b8 b7 b6 b5 b4 b3 b2 b1 Meaning	b	5

9F40	Additional Terminal Capabilities	b	5
9F41	Transaction Sequence Counter	n4-8	2-4
9F4E	Merchant Name and Location	ans	1~64
DF10	Multi Language (Default: “enfr”) Ex: “en”	an	2~128
DF11	Transaction Log Support (Default: Enable) 0 → Disable 1 → Enable	b	1
DF13	Terminal Action Code - Default	b	5
DF14	Terminal Action Code - Denial	b	5
DF15	Terminal Action Code - Online	b	5

b8	b7	b6	b5	b4	b3	b2	b1	
1	x	x	x	x	x	x	x	Numeric keys
x	1	x	x	x	x	x	x	Alphabetic and special characters keys
x	x	1	x	x	x	x	x	Command keys
x	x	x	1	x	x	x	x	Function Keys
x	x	x	x	0	x	x	x	RFU
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 4

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Print, attendant
x	1	x	x	x	x	x	x	Print, cardholder
x	x	1	x	x	x	x	x	Display, attendant
x	x	x	1	x	x	x	x	Display, cardholder
x	x	x	x	0	x	x	x	RFU
x	x	x	x	x	0	x	x	RFU
x	x	x	x	x	x	1	x	Code table 10
x	x	x	x	x	x	x	1	Code table 9

Byte 5

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Code table 8
x	1	x	x	x	x	x	x	Code table 7
x	x	1	x	x	x	x	x	Code table 6
x	x	x	1	x	x	x	x	Code table 5
x	x	x	x	1	x	x	x	Code table 4
x	x	x	x	x	1	x	x	Code table 3
x	x	x	x	x	x	1	x	Code table 2
x x x	x x x x	1 Code table 1						

<table>
<tr><td>DF17</td><td>Threshold Value for Biased Random Selection</td><td>b</td><td>4</td></tr>
<tr><td>DF18</td><td>Target Percentage to be Used for Random Selection</td><td>b</td><td>1</td></tr>
<tr><td>DF19</td><td>Maximum Target Percentage to be used for Biased Random Selection</td><td>b</td><td>1</td></tr>
<tr><td>DF21</td><td>Issuer Script Results</td><td>b</td><td>1~128</td></tr>
<tr><td>DF22</td><td>Force Online
0 → Disable
1 → Enable</td><td>b</td><td>1</td></tr>
<tr><td>DF25</td><td>Default DDOL</td><td>b</td><td>1~32</td></tr>
<tr><td>DF26</td><td>Revocation List Support (Default: Enable)
0 → Disable
1 → Enable</td><td>b</td><td>1</td></tr>
<tr><td>DF27</td><td>Exception File Support (Default: Disable)
0 → Disable
1 → Enable</td><td>b</td><td>1</td></tr>
<tr><td>DF28</td><td>Default TDOL</td><td>b</td><td>1~32</td></tr>
<tr><td>DFEE15</td><td>Application Selection Indicator</td><td>b</td><td>1</td></tr>
<tr><td>DFEE16</td><td>DUKPT or MK/SK Select for online PIN encrypted</td><td>b</td><td>1</td></tr>
<tr><td>DFEE17</td><td>ICC POS Entry Mode</td><td>b</td><td>1</td></tr>
<tr><td>DFEE17</td><td>ICC Terminal Entry Mode</td><td>b</td><td>1</td></tr>
<tr><td>DFEE18</td><td>MSR POS Entry Mode</td><td>b</td><td>1</td></tr>
<tr><td>DFEE18</td><td>MSR Terminal Entry Mode</td><td>b</td><td>1</td></tr>
<tr><td>DFEE19</td><td>Online DOL</td><td>b</td><td>1-256</td></tr>
<tr><td>DFEE1A</td><td>Output data element</td><td></td><td></td></tr>
<tr><td>DFEE1B</td><td>Authorization Response Code</td><td>b</td><td>8</td></tr>
<tr><td>DFEE1E</td><td>Contact Terminal Configuration (Default: F0 DC 3C F0 C2 9E 94 00)
Byte 1
<table>
<tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning</th></tr>
<tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Key Pad support</td></tr>
<tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>LCD support</td></tr>
<tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>PIN Pad support</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Print Support</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>RFU</td></tr>
<tr><td>x x x x x X 0</td><td>RFU</td><td></td><td></td><td></td><td></td><td></td><td></td><td></td></tr>
</table>
Byte 2
<table>
<tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning</th></tr>
<tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>PSE support</td></tr>
<tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Cardholder confirmation</td></tr>
<tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Preferred display order</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Multi language</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>EMV language selection method</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>Default DDOL</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>x</td><td>RFU</td></tr>
<tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>RFU</td></tr>
</table>
Byte 3
<table>
<tr><th>b8</th><th>b7</th><th>b6</th><th>b5</th><th>b4</th><th>b3</th><th>b2</th><th>b1</th><th>Meaning</th></tr>
<tr><td>0</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>RFU
(Revocation of Issuer Public Key Certificate (DF26))</td></tr>
</table></td><td>b</td><td>8</td></tr>
</table>

x	1	x	x	x	x	x	x	Manual action when CA PK loading fails
x	x	1	x	x	x	x	x	CA PK verified with check sum
x	x	x	1	x	x	x	x	Bypass PIN Entry
x	x	x	x	1	x	x	x	Subsequent bypass PIN Entry
x	x	x	x	x	1	x	x	Get data for pin try counter
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 4

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Amount before CVM processing
x	1	x	x	x	x	x	x	Floor limit checking
x	x	1	x	x	x	x	x	Random transaction selection
x	x	x	1	x	x	x	x	Velocity checking
x	x	x	x	0	x	x	x	RFU (Transaction Log (DF11))
x	x	x	x	x	0	x	x	RFU (Exception File (DF27))
x	x	x	x	x	x	0	x	RFU
x	x	x	x	x	x	x	0	RFU

Byte 5

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	X	Terminal action code support
x	1	x	x	x	x	x	x	Terminal action code can be change
x	x	1	x	x	x	x	x	Terminal action code can be deleted or disable
x	x	x	1	x	x	x	x	Default Action code processing before 1st GAC
x	x	x	x	1	x	x	x	Default Action code processing after 1st GAC
x	x	x	x	x	1	x	x	TAC/IAC default process when unable to go online (Skipped)
x	x	x	x	x	x	1	x	TAC/IAC default process when unable to go online (Normal)
x	x	x	x	x	x	x	0	RFU

Byte 6

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	x	x	x	x	x	x	x	Forced Online support
x	1	x	x	x	x	x	x	Forced acceptance support

	<table><tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Advices support</td></tr><tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Issuer referrals support</td></tr><tr><td>X</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>Batch data capture</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>Online data capture</td></tr><tr><td>X</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>Default TDOL</td></tr><tr><td>X</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>RFU</td></tr></table> Byte 7 <table><tr><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>Meaning</td></tr><tr><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>amount and pin entered on the same keypad</td></tr><tr><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>ICC/Magstripe reader combined</td></tr><tr><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Magstripe read first</td></tr><tr><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>x</td><td>Support account type selection</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>x</td><td>On fly script processing</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>x</td><td>Internal date management</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>1</td><td>x</td><td>Reversal Mode (1) Unable go online (2) ARC Error 0: (3) Online Approved but reader not approved. 1: (3) Online Approved but card response AAC.</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>0</td><td>RFU</td></tr></table> Byte 8 <table><tr><td>b8</td><td>b7</td><td>b6</td><td>b5</td><td>b4</td><td>b3</td><td>b2</td><td>b1</td><td>Meaning</td></tr><tr><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>x</td><td>RFU</td></tr></table>		
DFEE1F	Issuer script device limit Range: 0~255 (Default: 128)	b	1
DFEE20	ICC Power on detect waiting time. (Unit: Sec) (Default: 60S)	b	1
DFEE21	ICC L1 waiting time. (Unit: Sec)(Default: 10 S)	b	1
DFEE22	Driver (Menu, Get PIN, Get MSR) Timeout. (Unit: Sec) Byte1: Timeout for Menu. (Default: 30 S) Byte2: Timeout for Get PIN. (Default: 60 S) Byte3: Timeout for Get MSR. (Default: 60 S)	b	3
DFEE23	MSR Track Data	b	1~768
DFEE24	Force Acceptance (Default: 00)	b	1

Date	Change
3/2/2016	Original document (KT)
5/27/2016	Added command table (consolidated list, alphabetical). Added commands 72 46 09 13 and 72 46 09 14 (Get ICC L2 AID List Check Value and Get ICC L2 CA Public Key Check Value). Added command 72 46 50 (Implement Self-Test function for Non-PCI Device). New error codes added. Miscellaneous typos fixed. (KT)
5/25/2017	Correct miscellaneous grammar and usage problems. Update copyright. Refer to Universal SDK and download URL.
3/15/2018	Add 72 53 01 2F 01 command (and the Get version of it).
8/2/2019	Added Set Idle Waiting Time: 78 53 01 05 01