

POI SDK Specification

Change History

Revision	Modification	Auditor	Date
0.1	Create document	Cedar	2019-08-27
1.0	Update section 2.2	Kevin	2020-10-31

Contents

目录

1. Acronyms & Terms.....	6
2. POI SDK API.....	7
2.1. Class POIGeneralAPI.....	7
2.1.1. setDateTime.....	7
2.1.2. getDateTime.....	7
2.1.3. getApVersion.....	8
2.1.4. getSpVersion.....	8
2.1.5. getSdkVersion.....	8
2.1.6. setLed.....	9
2.1.7. setLedFlash.....	9
2.1.8. setBeep.....	10
2.1.9. getVersion.....	10
2.1.10. getAntennaPerformance.....	11
2.1.11. setApplicationScene.....	12
2.1.12. getCurrentApplicationScene.....	13
2.2. Class POICardManager.....	13
2.2.1. PosIccCardReader.....	13
2.2.2. PosPiccCardReader.....	16
2.2.3. PosMagCardReader.....	19
2.2.4. PosMifareCardReader.....	23
2.2.5. PosPsamCardReader.....	27
2.2.6. PosSidCardReader.....	29
2.2.7. PosViccCardReader.....	31
2.3. Class POIHsmManage.....	35
2.3.1. PedCalDes.....	35
2.3.2. PedWriteKey.....	36
2.3.3. PedWriteTIK.....	39
2.3.4. PedGetPinBlock.....	40
2.3.5. PedGetMac.....	42
2.3.6. PedGetMacDukpt.....	42
2.3.7. PedVerifyPlainPin.....	43
2.3.8. PedVerifyCipherPin.....	44
2.3.9. PedGetKcv.....	45
2.3.10. PedErase.....	46
2.3.11. PedGetRsaKey.....	46

2. 3. 12.	PedWriteRsaKey.....	46
2. 3. 13.	PedReadRsaKey.....	47
2. 3. 14.	PedCalcRsa.....	48
2. 3. 15.	PedDukptDes.....	48
2. 3. 16.	PedGetDukptKsn.....	49
2. 3. 17.	PedDukptIncreaseKsn.....	49
2. 3. 18.	PedGetRandom.....	50
2. 3. 19.	PedKeyManage.....	50
2. 3. 20.	PedCancelPinBlock.....	51
2. 3. 21.	SysHwSelfCheck.....	51
2. 3. 22.	registerListener.....	52
2. 3. 23.	unregisterListener.....	54
2. 3. 24.	SysSetAppServiceState.....	54
2. 3. 25.	SysGetAppServiceState.....	55
2. 3. 26.	SysSetWriteKeyResult.....	56
2. 3. 27.	SysGetWriteKeyResult.....	56
2. 4.	Class POIPrinterManager.....	56
2. 4. 1.	open.....	56
2. 4. 2.	close.....	57
2. 4. 3.	setPrintGray.....	57
2. 4. 4.	setPrintFont.....	57
2. 4. 5.	setLineSpace.....	58
2. 4. 6.	cleanCache.....	58
2. 4. 7.	addPrintLine.....	58
2. 4. 8.	beginPrint.....	59
2. 4. 9.	getPrinterLength.....	59
2. 4. 10.	getBeforePrinterLength.....	59
2. 5.	Class POIEmvCoreManager.....	60
2. 5. 1.	startTransaction.....	60
2. 5. 2.	IPosEmvCoreListener.....	60
2. 5. 3.	stopTransaction.....	62
2. 5. 4.	EmvSetAid.....	62
2. 5. 5.	EmvDeleteAid.....	63
2. 5. 6.	EmvGetAid.....	63
2. 5. 7.	EmvSetCapk.....	63
2. 5. 8.	EmvDeleteCapk.....	64
2. 5. 9.	EmvGetCapk.....	64
2. 5. 10.	EmvSetExceptionFile.....	64
2. 5. 11.	EmvDeleteExceptionFile.....	65

2. 5. 12.	EmvGetExceptionFile.....	65
2. 5. 13.	EmvSetRevocationIPK.....	65
2. 5. 14.	EmvDeleteRevocationIPK.....	66
2. 5. 15.	EmvGetRevocationIPK.....	66
2. 5. 16.	EmvSetTerminal.....	66
2. 5. 17.	EmvGetTerminal.....	67
2. 5. 18.	EmvSetDRL.....	67
2. 5. 19.	EmvDeleteDRL.....	68
2. 5. 20.	EmvGetDRL.....	68
2. 5. 21.	EmvSetService.....	69
2. 5. 22.	EmvDeleteService.....	69
2. 5. 23.	EmvGetService.....	69
2. 5. 24.	onSetSelectResponse.....	70
2. 5. 25.	onSetCardInfoResponse.....	70
2. 5. 26.	onSetPinResponse.....	70
2. 5. 27.	onSetOnlineResponse.....	71
2. 5. 28.	PosEmvErrorCode.....	71
2. 5. 29.	PosEmvAid.....	72
2. 5. 30.	PosEmvCapk.....	74
2. 5. 31.	PosEmvRevocationIPK.....	74
2. 5. 32.	PosEmvExceptionFile.....	74
2. 5. 33.	EmvDrlConstraints.....	75
2. 5. 34.	EmvServiceConstraints.....	75
2. 5. 35.	EmvTermCfgConstraints.....	77
2. 5. 36.	EmvTransDataConstraints.....	85
2. 5. 37.	EmvPinConstraints.....	86
2. 5. 38.	EmvOnlineRequestConstraints.....	88
2. 5. 39.	EmvOnlineResultConstraints.....	88
2. 5. 40.	EmvProcessResultConstraints.....	89
3.	POI SDK Error Code.....	90
3. 1.	ICCR Error Code.....	90
3. 2.	PCD Error Code.....	91
3. 3.	MSR Error Code.....	93
3. 4.	Keypad Error Code.....	93
3. 5.	RTC Error Code.....	94
3. 6.	HSM Error Code.....	95

1. Acronyms & Terms

Abbreviation	Description
PCD	Proximity coupling device, also be known as PICC or contactless reader.
CTLS	Contactless reader, same with PCD.
ICCR	IC Card Read.
IFM	Interface module, same with ICCR.
PED	PIN entry device.
MSR	Magnetic Strip Reader.
SP	Security Processor.

2. POI SDK API

2.1. Class POIGeneralAPI

2.1.1. setTime

Function description

Set SP time

Parameter

Field	Type	Description
time	byte[]	The date and time value, for a total of 6 bytes of BCD code. For example: October 1,2013,13:05' 10, it is expressed as: 131001130510

Return

None

2.1.2. getTime

Function description

Get SP time

Parameter

None

Return

Field	Type	Description
time	byte[]	The date and time value, for a total of 6 bytes of BCD code. For example: October 1,2013,13:05' 10, it is expressed as: 131001130510

2.1.3. getApVersion

Function description

Get the Android system version.

Parameter

None

Return

Field	Type	Description
version	String	Version number

2.1.4. getSpVersion

Function description

Get SP version number

Parameter

None

Return

Field	Type	Description
version	String	Version number

2.1.5. getSdkVersion

Function description

Get SDK version number

Parameter

None

Return

Field	Type	Description
version	String	Version number

2.1.6. setLed

Function description

Turn On/Off LED indicator.

Parameter

Field	Type	Description
color	int	Version type : 0x10 - Led Blue 0x20 - Led Green 0x40 - Led Yellow 0x80 - Led Red Constant type : LED_RED LED_YELLOW LED_GREEN LED_BLUE
onOff	boolean	true - ON false - OFF

Return

None

2.1.7. setLedFlash

Function description

Set LED blink.

Parameter

Field	Type	Description
color	int	Version type : 0x10 - Led Blue

		0x20 - Led Green 0x40 - Led Yellow 0x80 - Led Red Constant type : LED_RED LED_YELLOW LED_GREEN LED_BLUE
onMs	int	The period of LED on, in milliseconds
offMs	int	The period of LED off, in milliseconds

Return

None

2. 1. 8. setBeep

Function description

Turn On/Off beeper.

Parameter

Field	Type	Description
onOff	boolean	true - On, false - Off
freq	int	Frequency in HZ
timeMs	int	Keep time in millisecond

Return

None

2. 1. 9. getVersion

Function description

Get specified type of version or serial number.

Parameter

Field	Type	Description
type	int	Version type : 0x01 - SP Firmware Version 0x02 - Terminal Modem Version 0x03 - Terminal Software Version 0x04 - Terminal hardware version 0x05 - POI SDK version 0x06 - Terminal Equipment PSN 0x07 - Terminal DSN 0x08 - Terminal DSN2 (KSN on POS) 0x09 - Terminal kernel version Constant type : VERSION_TYPE_SP ... VERSION_TYPE_CUSTOMER_SUBNAME

Return

Field	Type	Description
version	String	Version number

2. 1. 10. getAntennaPerformance

Function description

Get NFC Antenna Performance.

Parameter

Field	Type	Description
out	PosByteArray	Antenna Performance result (PosByteArray): Len - response data length Buffer - response data Tlv Format : 01040000006B 0204000000D7

		01 : amplitude 02 : phase
--	--	------------------------------

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 1. 11. setApplicationScene

Function description

Set the SDK scene.

Parameter

Field	Type	Description
type	int	Version type : 0x00 - scene unencrypted 0x01 - scene mag encrypted 0x02 - scene encrypted Constant type : PAYMENT_APPLICATION_SCENE_UNENCRYPTED PAYMENT_APPLICATION_SCENE_MSD_ENCRYPTED PAYMENT_APPLICATION_SCENE_ENCRYPTED Note : This flag can be update from lower level to higher level in only single direction and can not switch back from higher level to lower level.

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 1. 12. getCurrentApplicationScene

Function description

Get the SDK scene.

Parameter

None

Return

Field	Type	Description
type	int	Version type : 0x00 - scene unencrypted 0x01 - scene mag encrypted 0x02 - scene encrypted others are failures

2. 2. Class POICardManager

2. 2. 1. PoslccCardReader

Operate IC card equipment function.

2. 2. 1. 1. open

Function description

Open ICC Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 1. 2. detect

Function description

Detecting the IC card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 1. 3. reset

Function description

Power-on and reset card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 1. 4. getCardReaderInfo

Function description

Get card information list.

Parameter

None

Return

Return value	Description
PosCardReaderInfo	Card Information List : mCategory : Card Reader Category (ICC Card, PICC Card, Mag Card)

	mCardType : Card type mCardChannel : Card logical channel number mSerialNum : Serial number of the card, BCD encoding mAttribute : Card Attribute (ATR)
--	--

2.2.1.5. transmitApdu

Function description

APDU instruction interaction.

Parameter

Field	Type	Description
inAPDU	byte[]	APDU command
outAPDU	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data
outSW	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.1.6. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 2. PosPiccCardReader

Operate PICC card equipment function.

2. 2. 2. 1. open

Function description

Open PICC Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 2. 2. detect

Function description

Detecting the PICC card.

Parameter

Field	Type	Description
mode	String	<p>Picc card reader detect mode.</p> <p>Default : 0x00</p> <p>0x00 - MODE ISO14443</p> <p>0x01 - MODE EMV</p> <p>0x0A - MODE A</p> <p>0x0B - MODE B</p> <p>Constant type :</p> <p>CARDREADER_DETECT_MODE_ISO14443</p> <p>CARDREADER_DETECT_MODE_EMV</p>

Field	Type	Description
		CARDREADER_DETECT_MODE_A
		CARDREADER_DETECT_MODE_B

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 2. 3. getCardReaderInfo

Function description

Get card information list.

Parameter

None

Return

Return value	Description
PosCardReaderInfo	Card Information List : mCategory : Card Reader Category (ICC Card, PICC Card, Mag Card) mCardType : Card type mCardChannel : Card logical channel number mSerialNum : Serial number of the card, BCD encoding mAttribute : Card Attribute (ATR)

2. 2. 2. 4. transmitApdu

Function description

APDU instruction interaction.

Parameter

Field	Type	Description
inAPDU	byte[]	APDU command
outAPDU	PosByteArray	Card response result (PosByteArray):

		Len - response data length Buffer - response data
outSW	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 2. 5. removeCard

Function description

Remove card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 2. 6. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 3. PosMagCardReader

Operate Mag card equipment function.

2. 2. 3. 1. open

Function description

Open Mag Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 3. 2. open(Encryption)

Function description

Open Mag Card Encryption mode.

Parameter

Field	Type	Description
dataType	int	Version type : 0x01 - PLAIN 0x02 - ENCRYPT 0x03 - ENCRYPT ZIOSK 0x04 - ENCRYPT TRANSARMOR Constant type : CARDREADER_DATA_TYPE_PLAIN = 1; CARDREADER_DATA_TYPE_ENCRYPT = 2; CARDREADER_DATA_TYPE_ENCRYPT_ZIOSK = 3;

Field	Type	Description
		CARDREADER_DATA_TYPE_ENCRYPT_TRANSAR MOR = 4;
keyType	int	Version type : 0x01 - TDK 0x02 - DUKPT MAC 0x03 - DUKPT DATA REQUEST 0x04 - DUKPT DATA RESPONSE 0x05 - DUKPT PIN Constant type : CARDREADER_KEY_TYPE_TDK = 1; CARDREADER_KEY_TYPE_DUKPT_MAC = 2; CARDREADER_KEY_TYPE_DUKPT_DATA_REQUEST = 3; CARDREADER_KEY_TYPE_DUKPT_DATA_RESPONSE = 4; CARDREADER_KEY_TYPE_DUKPT_PIN = 5;
keyIndex	int	Key Index
mode	int	Version type : 0x01 - ECB 0x02 - CBC Constant type : CARDREADER_MODE_ECB = 1; CARDREADER_MODE_CBC = 2;
padding	byte	DES padding data
vector	byte[]	Initial vector, CBC mode

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.3.3. detect

Detecting the Mag card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.3.4. getTraceData

Function description

Get trace data from MAG Card trace.

Parameter

Field	Type	Description
index	int	Trace index(Such as 1, 2, 3) Constant type : CARDREADER_TRACE_INDEX_1 = 1; CARDREADER_TRACE_INDEX_2 = 2; CARDREADER_TRACE_INDEX_3 = 3;

Return

Field	Type	Description
result	byte[]	Default: Trace Data,Failed to return null. Encryption: Ciphertext track Data, If encryption fails, the card number shielding data will be returned.

2. 2. 3. 5. getTraceDataKsn

Function description

Obtain track data encryption SN from MAG card. Only used for DUKPT.

Parameter

Field	Type	Description
index	int	Trace index(Such as 1, 2, 3) Constant type : CARDREADER_TRACE_INDEX_1 = 1; CARDREADER_TRACE_INDEX_2 = 2; CARDREADER_TRACE_INDEX_3 = 3;

Return

Field	Type	Description
result	byte[]	Ksn means success,Failed to return null.

2. 2. 3. 6. getTraceMaskData

Function description

Obtain the orbit data Mask card number from the MAG card. Only used in encryption mode.

Parameter

Field	Type	Description
index	int	Trace index(Such as 1, 2, 3) Constant type : CARDREADER_TRACE_INDEX_1 = 1; CARDREADER_TRACE_INDEX_2 = 2; CARDREADER_TRACE_INDEX_3 = 3;

Return

Field	Type	Description
result	byte[]	Mask Card means success,Failed to return null.

2.2.3.7. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.4. PosMifareCardReader

Operate M1 card equipment function.

2.2.4.1. open

Function description

Open M1 Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.4.2. detect

Function description

Detecting the M1 card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 4. 3. getCardReaderInfo

Function description

Get card information list.

Parameter

None

Return

Return value	Description
PosCardReaderInfo	<p>Card Information List :</p> <p>mCategory: Card Reader Category (ICC Card, PICC Card, Mag Card)</p> <p>mCardType: Card type</p> <p>mCardChannel: Card logical channel number</p> <p>mSerialNum: Serial number of the card, BCD encoding</p> <p>mAttribute: Card Attribute (ATR)</p>

2. 2. 4. 4. auth

Function description

M1 card authentication

Parameter

Field	Type	Description
KeyType	int	<p>Key type:</p> <p>'A' or 'a': A password</p> <p>'B' or 'b': B password</p>
blkNo	int	Block number
keyBuf	byte[]	Key data

serialNum	byte[]	Card serial number
-----------	--------	--------------------

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 4. 5. read

Function description

Read data from the specified block number of the M1 card.

Parameter

Field	Type	Description
blkNo	Int	Block number
outApdu	PosByteArray	Read data (PosByteArray): Len -- response data length Buffer -- response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 4. 6. write

Function description

Write data to the specified block number of the M1 card.

Parameter

Field	Type	Description
blkNo	int	Block number
buffer	byte[]	Written data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 4. 7. transmitApu

Function description

APDU instruction interaction.

Parameter

Field	Type	Description
inApu	byte[]	Apu command
outApu	PosByteArray	Read data (PosByteArray): Len -- response data length Buffer -- response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 4. 8. removeCard

Function description

Remove card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 4. 9. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 5. PosPsamCardReader

Operate PSAM card equipment function.

2. 2. 5. 1. open

Function description

Open PSAM Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 5. 2. detect

Detecting the PASM card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 5. 3. reset

Power-on and reset card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 5. 4. getCardReaderInfo

Function description

Get card information list.

Parameter

None

Return

Return value	Description
PosCardReaderInfo	Card Information List : mCategory : Card Reader Category (ICC Card, PICC Card, Mag Card) mCardType : Card type mCardChannel : Card logical channel number mSerialNum : Serial number of the card, BCD encoding mAttribute : Card Attribute (ATR)

2. 2. 5. 5. transmitApdu

Function description

APDU instruction interaction.

Parameter

Field	Type	Description
inAPDU	byte[]	APDU command
outAPDU	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data
outSW	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 5. 6. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 6. PosSidCardReader

Operate Sid card equipment function.

2. 2. 6. 1. open

Function description

Open Sid Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 6. 2. detect

Function description

Detecting the Sid card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 6. 3. getCardReaderInfo

Function description

Get card information list.

Parameter

None

Return

Return value	Description
PosCardReaderInfo	Card Information List : mCategory : Card Reader Category (ICC Card, PICC Card, Mag Card) mCardType : Card type mCardChannel : Card logical channel number mSerialNum : Serial number of the card, BCD encoding mAttribute : Card Attribute (ATR)

2. 2. 6. 4. transmitCmd

Function description

CMD instruction interaction.

Parameter

Parameter	Type	Description
inApdu	byte[]	Apdu command
outApdu	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 6. 5. removeCard

Function description

Remove card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 6. 6. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 7. PosViccCardReader

Operate VICC card equipment function.

2.2.7.1. open

Function description

Open VICC Card.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.7.2. inventory

Inventory from VICC Card.

Parameter

Parameter	Type	Description
outBuf	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.7.3. select

Function description

Select.

Parameter

Field	Type	Description
UidByte	byte[]	

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 7. 4. reset

Function description

Reset ViccCard.

Parameter

Field	Type	Description
UidByte	Byte[]	

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 7. 5. readBlock

Function description

Read block data.

Parameter

Field	Type	Description
BlockNo	Int	block number
OutApdu	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 7. 6. writeBlock

Function description

Write data to block.

Parameter

Field	Type	Description
blockNo	int	block number
outApdu	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 2. 7. 7. getSystemInfo

Function description

Get VICC Card info.

Parameter

Field	Type	Description
outApdu	PosByteArray	Card response result (PosByteArray): Len - response data length Buffer - response data

Return

Field	Type	Description
result	int	0 means success , others are failures

2.2.7.8. close

Function description

Turn off card reader.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3. Class POIHsmManage

2.3.1. PedCalDes

Function description

Data encryption and decryption

Parameter

Field	Type	Description
tdkIdx	int	Index of TDK
mode	int	Version type : ECB decryption: 0x00 ECB encryption: 0x01 CBC decryption: 0x02 CBC encryption: 0x03 Constant type : PED_CALC_DES_MODE_ECB_DEC PED_CALC_DES_MODE_ECB_ENC PED_CALC_DES_MODE_CBC_DEC PED_CALC_DES_MODE_CBC_ENC

inBuf	byte[]	Data to be encrypted/decryption
rspBuf	PosByteArray	Processed data

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.2. PedWriteKey

Function description

Write key into POS

Parameter

Field	Type	Description
keyInfo	PedKeyInfo	Key written
kcvInfo	PedKcvInfo	Key check value

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.2.1. Class PedKeyInfo

Function description

Key information

Parameter

Field	Type	Description
keyType	int	Source key type Version type : NOT: 0x00 No original key. Write in clear text. TLK : 0x01 TMK : 0x02 Constant type :

		<p>PED_TLK</p> <p>PED_TMK</p>
keyIdx	int	<p>Source key index</p> <p>Plain text when the source key index is 0</p> <p>When the target key type is TLK, the source key Index must be 0</p> <p>TLK :</p> <p>Only 1 group is supported, and its index range is [1,1]</p> <p>TMK :</p> <p>Supports 64 groups, the index range is [1, 64]</p> <p>TPK :</p> <p>Supports 64 groups, the index range is [1, 64]</p> <p>TAK :</p> <p>Supports 64 groups, the index range is [1, 64]</p> <p>TDK :</p> <p>Supports 64 groups, the index range is [1, 64]</p> <p>TEK :</p> <p>Supports 64 groups, the index range is [1, 64]</p> <p>TTK :</p> <p>Supports 64 groups, the index range is [1, 64]</p>
keyType	int	<p>Target key type</p> <p>Version type.</p> <p>TLK : 0x01</p> <p>TMK : 0x02</p> <p>TPK : 0x03</p> <p>TAK : 0x04</p> <p>TDK : 0x05</p> <p>TEK : 0x06</p>

		TTK : 0x09 Constant type. PED_TLK ... PED_TTK
keyIdx	int	Target key index
keyAlgorithm	int	Target Key algorithm definition Version type. TDEK : 0x00 AES : 0x01
keyLen	int	Target Key length 8, 16, 24, 32
keyData	byte[]	Target key data Plaintext or ciphertext Plaintext if the source key index is 0

2.3.2.2. Class PedKcvInfo

Function description

Key check value

Parameter

Field	Type	Description
checkMode	int	Mode 0 the value of checkBuf is invalid, KCV is not need to verify, checkBuf can be invalid data. Mode 1 the checkBuf[0]=4 , and one block of 0x00 encrypted by destination key, the first 4 bytes of result are KCV. Mode 2

		<p>Performs odd parity on the destination key and then checks by Mode=1.</p> <p>Mode 3</p> <p>Performs parity check on the destination key, and then checks by Mode=1.</p>
checkBuff	byte[]	<p>KCV valid</p> <p>checkBuf[1] to checkBuf[N-1] pointing to KCV values.</p>

2.3.3. PedWriteTIK

Function description

Write DUKPT Initial Key

Parameter

Field	Type	Description
grpIdx	int	<p>DUKPT key group index number</p> <p>Supports 10 groups, the index range is [1, 10]</p>
tlkIdx	int	TLK index number
tlkLen	int	TIK length n, now DUKPT algorithm supports 8/16 bytes length key.
tlkData	byte[]	<p>Initial encryption key data.</p> <p>If TIKIdx is 0, this is clear-text.</p> <p>If TIKIdx is not 0, this is cipher-text encrypted by TLK.</p>
ksn	byte[]	Initialize KSN
kcvInfo	PedKcvInfo	Key check value

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.4. PedGetPinBlock

Function description

Get PIN-BLOCK cipher-text.

Parameter

Field	Type	Description
fetchMode	int	Key management type: 0x01: Get Pin Block using MK pin key. 0x02: Get Pin Block using DUKPT pin key.
idxOrGrpId	int	If Mk, index of PIN Key (TPK). If DUKPT, group id of DUKPT.
fmt	int	Format of PIN BLOCK. If MK: 0x00 - ISO9564 format 0 0x01 - ISO9564 format 1 0x02 - ISO9564 format 3 0x04 - ISO9564 format 4 If DUKPT: 0x00 - ISO9564 format 0 KSN auto-increment. 0x01 - ISO9564 format 1 KSN auto-increment. 0x02 - ISO9564 format 2 KSN auto-increment. 0x20 - ISO9564 format 0 KSN not auto-increment. 0x21 - ISO9564 format 1 KSN not auto-increment. 0x22 - ISO9564 format 2 KSN not auto-increment.
timeoutMs	int	Timeout (ms), 0: no timeout, PED do not do timeout control.

data	byte[]	<p>Consisting of two parts.</p> <p>The first 16 bytes:</p> <p>For ISO9564 format 0/1/3:</p> <p>The card number without check digit, if the account number is less than 16 characters fill '0' at the left of card number.</p> <p>Fir ISO9564 format 4:</p> <p>The first byte (byte[0]) is the number of PAN digits.</p> <p>The PAN digits is filled into the bytes array start from byte[1] in BCD form, every byte contain 2 BCD digits.</p> <p>The last 8 bytes are:</p> <p>Participate in Pin Block formatted 8-byte data (according to ISO9564 specification, the data can be random number, transaction serial number or timestamp, etc., but the upper 4 bits and lower 4 bits of each byte must be 0xA~0xF between.</p>
expPinLenInd	String	<p>Expect PIN length list.</p> <p>For example, supports length of pins is 4/6/8 and support pressing the enter key without entering a password.</p> <p>This string should be "0, 4, 6, 8", The '0' means allowed press the enter key directly.</p> <p>PIN length is up to 12 digits.</p>

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.5. PedGetMac

Function description

Calculate MAC of data input (using MK MAC_KEY).

Parameter

Field	Type	Description
takIdx	int	Index of MAK key in MK (TAK).
operationMode	int	Operation mode: 0x00: CBC-MAC Use MAC-Key to encrypt message in CBC mode, the last block of cipher-text is the MAC result. 0x01: XOR-ECB-MAC Do xor operation with each block of message, use MAC-Key encrypt the finally XOR result. 0x02: ANSI-X9.19 MAC 0x03: ANSI-X9.9 MAC
data	byte[]	Message input to calculate MAC.
rspBuf	PosByteArray	MAC output

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.6. PedGetMacDukpt

Function description

Calculate MAC of data input (using DUKPT MAC_KEY).

Parameter

Field	Type	Description
grpIdx	int	DUKPT group id
operationMode	int	<p>MAC operation control code:</p> <p>This code is coded by two part: X + Y. E.g 21</p> <p>X can be:</p> <p>00 - Use "both ways variant", KSN auto-increment.</p> <p>20 - Use "both ways variant", KSN not auto-increment.</p> <p>40 - Use "Response variant", KSN not auto-increment.</p> <p>Note: the 20 and 40 is decimal number.</p> <p>Y can be:</p> <p>0 - CBC-MAC</p> <p>1 - XOR-ECB-MAC</p> <p>2 - ANSI-X9.19 MAC</p>
data	byte[]	Message input to calculate MAC.
macBuff	PosByteArray	MAC output
ksnBuff	PosByteArray	KSN output

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.7. PedVerifyPlainPin

Function description

Offline PIN plain text verify

Parameter

Field	Type	Description
slot	int	IC Card slot ID. Default 0.
mode	int	Mode, fixed 0.
timeoutMs	int	Timeout in millisecond.
expPinLenInd	String	<p>Expect PIN length list.</p> <p>For example, supports length of pins is 4/6/8 and support pressing the enter key without entering a password.</p> <p>This string should be "0, 4, 6, 8", The '0' means allowed press the enter key directly.</p> <p>PIN length is up to 12 digits.</p>

Return

Return value	Description
int	<p>Operation result, 0 -- success, other values failed</p> <p>Offline Plain Text verification requires the user to enter the PIN, which must be executed asynchronously. All execution results are notified to the upper layer through the callback function: onPedVerifyPin, type 0x86</p>

2.3.8. PedVerifyCipherPin

Function description

Offline Cipher Text PIN verify.

Parameter

Field	Type	Description
slot	int	IC Card slot ID. Default 0.
mode	int	Mode, fixed 0.
timeoutMs	int	Timeout in millisecond.
expPinLenInd	String	Expect PIN length list.

		<p>For example, supports length of pins is 4/6/8 and support pressing the enter key without entering a password.</p> <p>This string should be "0, 4, 6, 8", The '0' means allowed press the enter key directly.</p> <p>PIN length is up to 12 digits.</p>
rsaPinKey	PedRsaPinKey	<pre> PedRsaPinKey { byte[] modData; //Public key modulus. byte[] expData; // Public key exponent. byte[] iccRandomData; // Random number from IC card } </pre>

Return

Return value	Description
int	<p>The result of the operation</p> <p>0 - success, other values - failed.</p> <p>Offline cipher text verification requires the user to enter the PIN, which must be executed asynchronously. All execution results are notified to the upper layer through the callback function: onPedVerifyPin, type 0x87</p>

2.3.9. PedGetKcv

Function description

Get the KCV data of KEY

Parameter

Field	Type	Description
keyType	int	Key type
keyIdx	int	Key index
kcvInfo	PedKcvInfo	Key check value
rspBuff	PosByteArray	Kcv data output

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.10. PedErase

Function description

Clear all MK/SK and DUKPT keys in POI device.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.11. PedGetRsaKey

Function description

Generate RSA key

Parameter

Field	Type	Description
publicKeyIndex	int	Public key index
privateKeyIndex	int	Private key index
size	int	Key size

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.12. PedWriteRsaKey

Function description

Write RSA key

Parameter

Field	Type	Description
keyIdx	int	Key index
modData	byte[]	Modulus
expData	byte[]	Exponent

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.13. PedReadRsaKey

Function description

Read RSA key

Parameter

Field	Type	Description
keyIndex	int	Key index
rspData	PosByteArray	RSA Info Format : Byte1 Key Type : 0x00 : does not exist 0x01 : Public key 0x02 : Private key Byte2-3 RSA Key Number E.g: 0800 bit== 2048 bit == 256 byte
rspModulus	PosByteArray	Public Modulus
rspExponent	PosByteArray	Public Exponent

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.14. PedCalcRsa

Function description

RSA encryption and decryption

Parameter

Field	Type	Description
keyIndex	int	Key index
mode	Int	Mode
data	byte[]	Input data
rspBuf	PosByteArray	Data output

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.15. PedDukptDes

Function description

Data encryption or decryption by DUKPT data-key.

Parameter

Field	Type	Description
grpIdx	int	DUKPT group ID: 1~10
keyType	int	Key type: 0x00 - Use Request or both ways MAC key 0x01 - Use Request or both ways Data key 0x02 - Use DATA Response key 0x03 - Use PIN Encryption key
initVector	byte[]	8-byte initial vector, required for CBC encryption and decryption.
mode	int	Operation mode:

		0x00 - ECB decryption 0x01 - ECB encryption 0x02 - CBC decryption 0x03 - CBC encryption
data	byte[]	Data input.
rspKsn	PosByteArray	KSN output (10 bytes)
rspBuff	PosByteArray	Data output

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 3. 16. PedGetDukptKsn

Function description

Get the KSN of next DUKPT operation.

Parameter

Field	Type	Description
grpIdx	int	DUKPT group ID: 1~10
rspBuff	PosByteArray	KSN output (10 bytes)

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 3. 17. PedDukptIncreaseKsn

Function description

KSN+1, the DUKPT key for each KSN, can only be used up to 256 times.

When a single key is used 256 times, it will return error, user need to call this interface, to make KSN increase.

Parameter

Field	Type	Description
grpIdx	int	DUKPT group ID: 1~10

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.18. PedGetRandom

Function description

Get random number compliance with NIST SP800-90A.

Parameter

Field	Type	Description
randomLen	int	Random number length
rspBuff	PosByteArray	Random number output

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.19. PedKeyManage

Function description

Password management

Parameter

Field	Type	Description
type	int	Management type: 0x80 - Verify unlock password. 0x81 - Verify Admin-A password. 0x82 - Verify Admin-B password. 0x83 - Hold sensitive state. 0x84 - Set new unlock password. 0x85 - Set new Admin-A password. 0x86 - Set new Admin-B password.

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.20. PedCancelPinBlock

Function description

Cancel PIN entering process.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2.3.21. SysHwSelfCheck

Function description

Hardware self-test

Parameter

Field	Type	Description
hwDeviceMask	int	<p>The hardware part bit map.</p> <p>0x01 - MSR</p> <p>0x02 - ICCR</p> <p>0x04 - CTLS</p> <p>0x08 - Memory</p> <p>0x10 - Keypad</p> <p>0x40 - Antenna</p> <p>0xFF - All hardware components.</p>

Return

Return value	Description
int	0 means success , others are failures

onHwSelfCheckRet function returns.

2. 3. 22. registerListener

Function description

Register Listener

Parameter

Field	Type	Description
listener	EventListener	Security broadcast

Return

None

2. 3. 22. 1. Interface EventListener

Function description

Security listener callback interface.

Interface

Interface	Description
onInfo	Called to indicate an info or a warning. What: what the type of info or warning. Extra: extra an extra code, specific to the info. Typically implementation dependent.
onError	Called to indicate an error. What: what the type of info or warning. Extra: extra an extra code, specific to the info. Typically implementation dependent.
onKeyboardShow	Called when the Pin Block module should show UI with TP

	<p>as Soft Keys.</p> <p>Keys:</p> <p>E.g BCD 38 35 37 36 30 32 39 31 33 1B 34 0D</p> <p>The keyboard sequence is displayed according to the Keys parameter.</p>
onKeyboardInput	<p>Button input value, input return value is defined as follows.</p> <p>NumKeys:</p> <p>0xN -- returns 0xN keys correctly</p> <p>0x90 -- touch screen has been bounced by point</p> <p>0x95 -- touch screen is not in range</p> <p>0x96 -- Confirm button pressed</p> <p>0x97 -- Cancel button pressed</p> <p>0x98 -- number key pressed</p> <p>0x99 -- The entered key has exceeded the set pin length</p>
onPedPinBlockRet	<p>PIN block returns, Support PED or PSAM</p> <p>Type:</p> <p>0x83 -- DEFAULT PED_PINBLOCK_RET_TYPE_DEFAULT</p> <p>0x92 -- PSAM PED_PINBLOCK_RET_TYPE_PSAM</p> <p>PinBuf:</p> <p>08 62BC7A46801A6C46 0A FFFF9080102495000001</p> <p>LEN + PINBLOCK + LEN + KSN</p>
onPedVerifyPin	<p>PIN verification, Support for Offline ciphertext PIN, offline plaintext PIN.</p> <p>Type:</p> <p>0x86 -- CipherText PIN PED_VERIFY_PIN_TYPE_PLAIN</p> <p>0x87 -- PlainText PIN PED_VERIFY_PIN_TYPE_CIPHER</p> <p>PinBuf:</p> <p>Card status code. For example 9000, 6A83</p>

onHwSelfCheckRet	<p>Self-test results</p> <p>Type:</p> <p>Check type.</p> <p>CheckResult:</p> <p>0x00 success</p>
onHwSensorTriggered	<p>Safe trigger</p> <p>Triggered:</p> <p>SP firmware status.</p> <p>0x00 -- SP_STATUS_OK</p> <p>0x01 -- SP_STATUS_TRIGGERED</p> <p>0x02 -- SP_STATUS_LOCK</p> <p>0x03 -- SP_STATUS_KEY_WIPED</p> <p>SensorValue:</p> <p>Trigger information</p> <p>TriggerTime:</p> <p>Trigger Time</p>

2. 3. 23. unregisterListener

Function description

Unregister listener.

Parameter

Field	Type	Description
listener	EventListener	See registerListener for details.

Return

None

2. 3. 24. SysSetAppServiceState

Function description

Set the App Run Status

Parameter

Parameter	Type	Description
InOrExit	boolean	<p>True - the system will mask the power long button to disable the screen, and disable the low power and other prompt boxes that will affect the Pin block processing.</p> <p>False - recovery</p> <p>Note: Applications use the SDK interface are recommended to be set it to true when the application is running, and must be restored when the application switch to the background. Otherwise, the system power key cannot be processed.</p>

Return

None

2. 3. 25. SysGetAppServiceState

Function description

Get App Run Status.

Parameter

None

Return

Return value	Description
boolean	<p>true -- set</p> <p>false -- not set</p>

2. 3. 26. SysSetWriteKeyResult

Function description

Set key filling result.

Parameter

Parameter	Type	Description
result	int	0 - success 1 - Unfilled 2 - Filling failed 3 - Filling is successful but verification failed

Return

None

2. 3. 27. SysGetWriteKeyResult

Function description

Get key filling result.

Parameter

None

Return

Field	Type	Description
result	int	0 means success , others are failures

2. 4. Class POIPrinterManager

2. 4. 1. open

Function description

Open Printer.

Parameter

None

Return

None

2.4.2. close

Function description

Close Printer.

Parameter

None

Return

None

2.4.3. setPrintGray

Function description

Set the print grayscale, if don't set default value is 2000. The value smaller the lighter, the larger the thicker.

Parameter

Field	Type	Description
gray	int	

Return

None

2.4.4. setPrintFont

Function description

Set the print font, font path: "/system/fonts/simsun.ttc".

Parameter

Field	Type	Description
fontName	String	"/system/fonts/simsun.ttc"

Return

None

2.4.5. setLineSpace

Function description

Set line space.

Parameter

Field	Type	Description
gray	int	

Return

None

2.4.6. cleanCache

Function description

clean cache. It is used with beginPrint interface.

Parameter

None

Return

None

2.4.7. addPrintLine

Function description

add print content.

Parameter

Field	Type	Description
line	PrintLine	print content model(TextPrintLine/BitmapPrintLine)

Return

None

2.4.8. beginPrint

Function description

start to print.

Parameter

Field	Type	Description
Listener	IPrinterListener	print listener

Return

None

2.4.9. getPrinterLength

Function description

Print the total length of the data.

Parameter

None

Return

Field	Type	Description
length	int	

2.4.10. getBeforePrinterLength

Function description

Print data current length.

Parameter

None

Return

Field	Type	Description
length	int	

2. 5. Class POIEmvCoreManager

2. 5. 1. startTransaction

Function description

Start EMV process request.

Parameter

Field	Type	Description
bundle	android.os.Bundle	Business data required for the EMV process. (see the EmvTransDataConstraints class definition for details)
listener	IPosEmvCoreListener	EMV process processing listener. (see the IPosEmvCoreListener class definition for details)

Return

see the PosEmvErrorCode class definition for details.

2. 5. 2. IPosEmvCoreListener

Function description

EMV process listener callback interface, contains the following callbacks. Please refer to the How to start EMV for description of each method.

Interface

Interface	Description
onEmvProcess	Check to the card. The transaction is ready to start. type : DEVICE_CONTACT DEVICE_CONTACTLESS DEVICE_MAGSTRIPE

	EMV_MULTI_CONTACTLESS
onSelectApplication	<p>When there are multiple candidate AIDs in an EMV transaction, the cardholder is required to make a selection. ICC only.</p> <p>list : Candidate list name</p> <p>isFirstSelect : FirstSelect</p>
onConfirmCardInfo	<p>Confirm card information. ICC only</p> <p>mode : Determine the type.</p> <p>CMD_TRY_OTHER_APPLICATION</p> <p>CMD_AMOUNT_CONFIG</p> <p>CMD_ISSUER_REFERRAL</p> <p>bundle : (see the EmvCardInfoConstraints class definition for details)</p>
onKernelType	<p>Returns the type of card issuer. For example (VISA, MasterCard, AMEX, RuPay, etc.). PICC only</p> <p>type :</p> <p>EMV_CARD_NOT</p> <p>EMV_CARD_VISA</p> <p>EMV_CARD_UNIONPAY</p> <p>EMV_CARD_MASTERCARD</p> <p>EMV_CARD_DISCOVER</p> <p>EMV_CARD_AMEX</p> <p>EMV_CARD_MIR</p> <p>EMV_CARD_RUPAY</p> <p>EMV_CARD_INTERAC</p>
onSecondTapCard	<p>Some issuers support script processing in PICC transactions. A second call is required. This callback is required by the application to display a prompt. Let the cardholder re-card. PICC only</p>

onRequestInputPin	Input PIN. Support offline plain text PIN. Offline cipher text PIN. Online PIN bundle : (see the EmvPinConstraints class definition for details)
onRequestOnlineProcess	Online connection is requested. bundle : Online connection data (see the EmvOnlineRequestConstraints class definition for details)
onTransactionResult	EMV process result result : (see the PosEmvErrorCode class definition for details) bundle : (see the EmvProcessResultConstraints class definition for details)

2.5.3. stopTransaction

Function description

Stop EMV Transaction.

Parameter

None

Return

None

2.5.4. EmvSetAid

Function description

Add AID parameter.

Parameter

Field	Type	Description
-------	------	-------------

param	PosEmvAid	see the PosEmvAid class definition for details
-------	-----------	---

Return

Return value	Description
int	result

2.5.5. EmvDeleteAid

Function description

Delete AID parameter.

Parameter

None

Return

None

2.5.6. EmvGetAid

Function description

Get AID parameter.

Parameter

None

Return

Field	Type	Description
param	List<PosEmvAid>	see the PosEmvAid class definition for details

2.5.7. EmvSetCapk

Function description

Add CAPK parameter.

Parameter

Field	Type	Description
param	PosEmvCapk	see the PosEmvCapk class definition for details

Return

Return value	Description
Int	result

2.5.8. EmvDeleteCapk

Function description

Delete CAPK parameter.

Parameter

None

Return

None

2.5.9. EmvGetCapk

Function description

Get CAPK parameter.

Parameter

None

Return

Field	Type	Description
param	List<PosEmvCapk>	see the PosEmvCapk class definition for details

2.5.10. EmvSetExceptionFile

Function description

Add Exception File parameter.

Parameter

Field	Type	Description
param	PosEmvExceptionFile	see the PosEmvExceptionFile class definition for details

Return

Return value	Description
int	result

2.5.11. EmvDeleteExceptionFile

Function description

Delete Exception File parameter.

Parameter list

None

Return

None

2.5.12. EmvGetExceptionFile

Function description

Get Exception File parameter.

Parameter

None

Return

Field	Type	Description
param	List<PosEmvException File>	see the PosEmvExceptionFile class definition for details

2.5.13. EmvSetRevocationIPK

Function description

Add CAPK Revocation parameter.

Parameter

Field	Type	Description
param	PosEmvRevocList	see the PosEmvRevocList class definition for

		details
--	--	---------

Return

Return value	Description
int	result

2.5.14. EmvDeleteRevocationIPK

Function description

Delete CAPK Revocation parameter.

Parameter

None

Return

None

2.5.15. EmvGetRevocationIPK

Function description

Get CAPK Revocation parameter.

Parameter

None

Return

Field	Type	Description
param	List<PosEmvRevocList>	see the PosEmvRevocList class definition for details

2.5.16. EmvSetTerminal

Function description

set terminal information in EMV kernel.

Parameter

Field	Type	Description
-------	------	-------------

type	int	TYPE_TERMINAL TYPE_INTERAC
data	android.os.Bundle	see the EmvTerminalConstraints class definition for details

Return

Return value	Description
int	result

2.5.17. EmvGetTerminal

Function description

get terminal information in EMV kernel.

Parameter

Field	Type	Description
type	int	TYPE_TERMINAL TYPE_INTERAC
data	android.os.Bundle	see the EmvTerminalConstraints class definition for details

Return

Return value	Description
int	result

2.5.18. EmvSetDRL

Function description

Add DRL configuration parameter.

Parameter

Field	Type	Description
type	int	TYPE_VISA TYPE_AMEX
bundle	android.os.Bundle	see the EmvDrlConstraints class definition for details

Return

Return value	Description
int	result

2.5.19. EmvDeleteDRL

Function description

Delete DRL configuration parameter.

Parameter

Field	Type	Description
type	int	TYPE_VISA TYPE_AMEX

Return

None

2.5.20. EmvGetDRL

Function description

Delete DRL configuration parameter.

Parameter

Field	Type	Description
type	int	TYPE_VISA TYPE_AMEX
bundle	android.os.Bundle	see the EmvDrlConstraints class definition for details

Return

Return value	Description
int	result

2. 5. 21. EmvSetService

Function description

Add RuPay service parameter.

Parameter

Field	Type	Description
bundle	android.os.Bundle	see the EmvServiceConstraints class definition for details

Return

Return value	Description
int	result

2. 5. 22. EmvDeleteService

Function description

Delete RuPay service parameter.

Parameter

None

Return

None

2. 5. 23. EmvGetService

Function description

Add RuPay service parameter.

Parameter

Field	Type	Description
bundle	android.os.Bundle	see the EmvServiceConstraints class definition for details

Return

Return value	Description
--------------	-------------

int	result
-----	--------

2. 5. 24. onSetSelectResponse

Function description

When EMV processes request to select Application (**onSelectApplication()**), EMV select the application.

Parameter

Field	Type	Description
index	int	Application selection is from "1", "0" means cancellation.

Return

None

2. 5. 25. onSetCardInfoResponse

Function description

When EMV processes request to confirm the card information(**onConfirmCardInfo()**), it is called after user confirmation.

Parameter

Field	Type	Description
confirm	boolean	

Return

None

2. 5. 26. onSetPinResponse

Function description

When EMV processes request to PIN (**onRequestInputPin()**).

Parameter

Field	Type	Description
-------	------	-------------

info	android.os.Bundle	see the EmvPinConstraints class definition for details
------	-------------------	---

Return

None

2. 5. 27. onSetOnlineResponse

Function description

When EMV processes request for on-line operation(**onRequestOnlineProcess()**) and the value is returned, the online data is passed into the process through this interface.

Parameter

Field	Type	Description
data	android.os.Bundle	see the EmvOnlineResultConstraints class definition for details

Return

None

2. 5. 28. PosEmvErrorCode

Error code	Error name	Comment
0xAA	EXCEPTION_ERR	Exception error
0x00	EMV_OK	Handle OK
0x01	EMV_APPROVED	Transaction offline approval
0x02	EMV_APPROVED_ONLINE	Transaction online approval
0x03	EMV_DECLINED	Transaction declined
0x04	EMV_FORCE_APPROVED	Transaction force approval
-1	EMV_PARAMETER_ERROR	EMV parameter error
-2	EMV_OTHER_INTERFACE	Please try another communication interface

-3	EMV_COMMAND_FAIL	Read the card failure
-4	EMV_CARD_BLOCK	Card locked
-5	EMV_APP_EMPTY	No application
-6	EMV_APP_BLOCK	Application locked
-7	EMV_FALLBACK	Fallback
-8	EMV_SEE_PHONE	See Phone
-9	EMV_NOT_ALLOWED	No allowed
-10	EMV_NOT_ACCEPTED	No accepted
-11	EMV_TERMINATED	Trans terminated
-12	EMV_CANCEL	Trans Canceled
-13	EMV_TIMEOUT	Read card Timeout
-14	EMV_MULTI_PICC	Read card multi Picc
-15	EMV_ICC_INTERFACE	Trade IC card. Not pure magnetic strip
-20	EMV_ENCRYPT_ERROR	Trade encrypt error
-21	EMV_UNENCRYPTED	Trade unencrypted
-999	EMV_OTHER_ERROR	Other error

2.5.29. PosEmvAid

Property	Describe	Value Type	Example
AID	Application id. Tag 9F06	byte[]	"A000000004"
Version	Application version. Tag 9F09	byte[]	"0001"
SelectIndicator	Application select flags: FULL_MATCH or PART_MATCH Partial match between 9F06 and ICC 4F supported	boolean	True: FULL_MATCH False: PART_MATCH

TypeIndicator	Application Type	boolean	True: Contact False: Contactless
AcquirerId	Acquirer Id. Tag 9F01	byte[]	"A00000000001"
dDOL	dDol. Tag 97	byte[]	"9F3704"
tDOL	tDol. Tag 9F49	byte[]	"9F3704"
TACDenial	TAC Denial	byte[]	"0000000000"
TACOnline	TAC Online	byte[]	"0000000000"
TACDefault	TAC Default	byte[]	"0000000000"
Threshold	Threshold of bias random selection	int	
TargetPercentage	Target percentage of random selection	int	
MaxTargetPercentage	The maximum target percentage of offset random selection	int	
FloorLimit	Floor Limit	int	
ContactlessTransLimit	Contactless Trans Limit	int	
ContactlessCVMLimit	Contactless CVM Limit	int	
ContactlessFloorLimit	Contactless Floor Limit	int	
DynamicTransLimit	Dynamic Trans Limit	int	
TerminalType	Terminal Type 9F35	byte[]	"22"
TerminalCapabilities	Terminal Capabilities 9F33	byte[]	"E0E888"
AdditionalTerminalCapabilities	Additional Terminal Capabilities 9F40	byte[]	"D8E00000"
TerminalRiskManagementData	Terminal Risk Management Data 9F1D	byte[]	"2000000000000000"
TerminalCountryCode	Terminal Country Code 9F1A	byte[]	"0356"
MerchantCategoryCode	Merchant Category Code 9F15	byte[]	"0356"

TransCurrencyCode	Trans Currency Code 5F2A	byte[]	"0356"
TransCurrencyExp	Trans Currency Exp 5F36	byte[]	"02"
TerminalType, TerminalCapabilities, AdditionalTerminalCapabilities, TerminalCountryCode, MerchantCategoryCode, TransCurrencyCode, TransCurrencyExp. If it is set, the parameters in the AID will be used. If not set, the parameters in the Terminal will be used by default.			

2.5.30. PosEmvCapk

Property	Describe	Value Type	Example
RID	Application id. 9F06	byte[]	"A000000004"
CapkIndex	key id	byte	0x01
HashInd	HASH algorithm flag	byte	AIRTH_IND_RSA AIRTH_IND_SM
AlgorithmInd	RSA algorithm flag	byte	HASH_IND_NO HASH_IND_SHA1
Modul	Module content.	byte[]	
Exponent	Exponent content	byte[]	
CheckSum	key checksum	byte[]	

2.5.31. PosEmvRevocationIPK

Property	Describe	Value Type	Example
RID	Application id. 9F06	byte[]	"A000000004"
CapkIndex	key id	byte	0x01
SerialNo	Certificate serial number	byte[]	

2.5.32. PosEmvExceptionFile

Property	Describe	Value Type	Example
----------	----------	------------	---------

ucPan	Pan	byte[]	
ucSerialNo	Serial number	byte[]	

2. 5. 33. EmvDrlConstraints

Constant names	Describe
DRLDATA	DRL Data
DRL_SET_DELIMITER_TAG	DF01 variable DRL delimiter DF01 length is equal to the length of the following TLV
DRL_SET_ID_TAG	9F5A 1 to 16 Mandatory. Application program Identifier (AMEX does not have this configuration.)
DRL_SET_FLOOR_LIMIT_TAG	DF23 6 Optional. Reader Contactless Floor Limit
DRL_SET_TRANSACTION_LIMIT_TAG	DF24 6 Optional. Reader Contactless Transaction Limit
DRL_SET_CVM_REQUIRED_LIMIT_TAG	DF26 6 Optional. Reader CVM Required Limit
DRL_SET_ENTRY_POINT_TAG	DF30 Entry Point Byte 1 B8: StatusCheckSupportFlag b7: ZeroAmountAllowedFlag b6: ContactLessTransactionLimitFlag b5: ContactLessFloorLimitFlag b4: CVMRequiredLimitFlag b3..b1 : RFU
DRL_SET_STATUS_ZERO_AMOUNT_TAG	DF32 Status Zero Amount Allowed Flag Byte 1 0x01: option1 = online cryptogram request 0x02: option2 = not allowed
TLV Example : DF01 26 9F5A023031 DF2406323030303030 DF23053130303030 DF26053130303030 DF3001F8 DF320101 DF01 2B 9F5A0730313133323333 DF2406323030303030 DF23053130303030 DF26053130303030 DF3001F8 DF320101	

2. 5. 34. EmvServiceConstraints

Constant names	Describe
----------------	----------

DF48 8 Mandatory PRMacq KEY

DF01 820107 DF160431303130 DF170A36303130313042353030

DF48204436393437303545444630444642423532303233433133344345423935344535 DF5406413332424241

DF01 820107 DF160431303131 DF170A36303130313142353030

DF48204436393437303545444630444642423532303233433133344345423935344535 DF5406413332424241

2. 5. 35. EmvTermCfgConstraints

Constant names	Describe	Bundle Value Type	Example
MERCHNAME	Merchant Location 9F4E	byte[]	"Railway Operator"
MERCHCATECODE	Merchant Category Code 9F15	byte[]	"0356"
MERCHID	Merchant Id 9F16	byte[]	"0000000000000000"
TERMID	Terminal Id 9F1C	byte[]	"Terminal"
TERMINALTYPE	Terminal Type 9F35	byte[]	"22"
TERMINALCAPABILITY	Terminal Capabilities 9F33	byte[]	"E0E888"
TERMINALEXCAPABILITY	Additional Terminal Capabilities 9F40	byte[]	"D8E00000"
REFERCURREXP	Trans Ref Currency Exp 9F3D	byte[]	"02"
REFERCURRCODE	Transaction Ref Currency Code 9F3C	byte[]	"0356"
TRANSCURREXP	Trans Currency Exp 5F36	byte[]	"02"
TRANSCURRCODE	Trans Currency Code 5F2A	byte[]	"0356"
TERMINALCOUNTRYCODE	Terminal Country Code 9F1A	byte[]	"0356"
IFDSERIALNUMBER	IFD Serial Number 9F1E	byte[]	"12345678"
TERMINALENTRYMODE	Terminal Entry Mode 9F39	byte[]	"05"
PSE	PSE	boolean	Default:true
CARDHOLDERCONFIRM	Card Holder Confirm	boolean	Default:true
PREFERREDORDER	Preferred Order	boolean	Default:false
LANGUAGESELECT	Language Select	boolean	Default:false
REVOCATIONOFISSUERPUBLICKEY	Revocation Issuer Public Key	boolean	Default:true

DEFAULTDDOL	Default DOL	boolean	Default:true
DEFAULTTDOL	Default DOL	boolean	Default:true
BYPASSPINENTRY	Bypass PIN Entry	boolean	Default:true
GETDATAFORPINCO UNTER	Get Data For PIN Counter	boolean	Default:true
SUBSEQUENTBYPASS PINENTRY	Subsequent Bypass PIN Entry	boolean	Default:true
RANDOMTRANSACTION SELECTION	Random Transaction Selection	boolean	Default:true
VELOCITYCHECKING	Velocity Checking	boolean	Default:true
TRANSACTIONLOG	Transaction Log	boolean	Default:true
EXCEPTIONFILE	Exception File	boolean	Default:true
ISSUERREFERRAL	Issuer Referral	boolean	Default:false
FLOORLIMITCHECKING	Floor limit checking	boolean	Default:true
UNABLETOGOONLINE	Unable to go online	boolean	Default:false
FORCEDONLINE	Forced Online	boolean	Default:false
FORCEDACCEPTANCE	Forced Acceptance	boolean	Default:false
<p>TerminalType, TerminalCapabilities, AdditionalTerminalCapabilities, TerminalCountryCode, MerchantCategoryCode, TransCurrencyCode, TransCurrencyExp.</p> <p>If it is set, the parameters in the AID will be used. If not set, the parameters in the Terminal will be used by default.</p>			

2. 5. 35. 1. MasterCard

Constant names	Describe
MASTER_SET_DEFAULT_UDOL_TAG	DF1A Default UDOL
MASTER_SET_KERNEL_ID_TAG	DF0C Kernel ID
MASTER_SET_MAGSTRIPE_AVN_TAG	9F6D Magstripe Application Version Number

MASTER_SET_MAGSTRIPE_CVM_CAPABILITIES_TAG	<p>DF1E Magstripe CVM Capabilities CVM Required</p> <p>Byte 1</p> <p>B8..b5:</p> <p>0000: No CVM</p> <p>0001: Obtain Signature</p> <p>0010: Online PIN</p> <p>1111: N/A</p> <p>Other: RFU</p> <p>b4..b1 : RFU</p>
MASTER_SET_MAGSTRIPE_NO_CVM_CAPABILITIES_TAG	<p>DF2C Magstripe CVM Capabilities No CVM Required</p> <p>Byte 1</p> <p>B8..b5:</p> <p>0000: No CVM</p> <p>0001: Obtain Signature</p> <p>0010: Online PIN</p> <p>1111: N/A</p> <p>Other: RFU</p> <p>b4..b1 : RFU</p>
MASTER_SET_MOBILE_SUPPORT_INDICATOR_TAG	<p>9F7E Mobile Support Indicator</p>
MASTER_SET_CVM_CAPABILITIES_TAG	<p>DF18 CVM Capabilities CVM Required</p> <p>Byte 1</p> <p>B8: Plaintext PIN for ICC verification</p> <p>b7: Enciphered PIN for online verification</p> <p>b6: Signature (paper)</p> <p>b5: Enciphered PIN for offline verification</p> <p>b4: No CVM required</p> <p>b3..b1: RFU</p>
MASTER_SET_NO_CVM_CAPABILITIES_TAG	<p>DF19 CVM Capabilities No CVM Required</p> <p>Byte 1</p> <p>B8: Plaintext PIN for ICC verification</p> <p>b7: Enciphered PIN for online verification</p> <p>b6: Signature (paper)</p> <p>b5: Enciphered PIN for offline verification</p> <p>b4: No CVM required</p> <p>b3..b1: RFU</p>
MASTER_SET_KERNEL_CONFIG_TAG	<p>DF1B Kernel Configuration</p> <p>Byte 1</p> <p>b8: only EMV supported</p> <p>b7: only Magstripe supported</p>

	b6: On device cardholder verification supported b5: Relay resistance protocol supported b4..b3 : RFU
MASTER_SET_MIN_GRACE_RRP_TAG	DF32 Minimum Relay Resistance Grace Period
MASTER_SET_MAX_GRACE_RRP_TAG	DF33 Maximum Relay Resistance Grace Period
MASTER_SET_CAPDU_EXPECTED_RRP_TAG	DF34 Terminal Expected Transmission Time For Relay Resistance C-APDU
MASTER_SET_RAPDU_EXPECTED_RRP_TAG	DF35 Terminal Expected Transmission Time For Relay Resistance R-APDU
MASTER_SET_ACCURACY_THRESHOLD_RRP_TAG	DF36 Relay Resistance Accuracy Threshold
MASTER_SET_MISMATCH_THRESHOLD_RRP_TAG	DF37 Relay Resistance Transmission Time Mismatch Threshold
TLV Example : DF1A039F6A04 DF1E0110 DF1E0100 DF180128 DF190160 DF1B0130	

2. 5. 35. 2. Visa

Constant names	Describe	Example
VISA_SET_QUALIFIERS_TAG	9F66 Terminal Transaction Qualifiers Byte 1 bit 8: 1 = MSD supported (Mandatory no support) bit 7: RFU (0) bit 6: 1 = qVSDC supported bit 5: 1 = EMV contact chip supported bit 4: 1 = Offline-only reader bit 3: 1 = Online PIN supported bit 2: 1 = Signature supported bit 1: 1 = Offline Data Authentication (ODA) for Online Authorizations supported. Byte 2 bit 8: 1 = Online cryptogram required bit 7: 1 = CVM required bits 6-1: RFU (00000) Byte 3 bit 8: 1 = Issuer Update Processing supported bit 7: 1 = Mobile functionality supported (Consumer Device CVM)	

	bits 6-1: RFU (000000) Byte 4 RFU ('00') TTQ byte 2 bits 8-7 are transient values, and reset to zero at the beginning of the transaction. All other TTQ bits are static values, and not modified based on transaction conditions. TTQ byte 3 bit 7 shall be set by the acquirer-merchant to 1b.
VISA_SET_KERNEL_CONFIG_TAG	DF1B Kernel Configuration Byte 1 b1: DrlSupportFlag
VISA_SET_ENTRY_POINT_TAG	DF30 Entry Point Byte 1 B8: StatusCheckSupportFlag b7: ZeroAmountAllowedFlag b6: ContactLessTransactionLimitFlag b5: ContactLessFloorLimitFlag b4: CVMRequiredLimitFlag b3..b1 : RFU
VISA_SET_STATUS_ZERO_AMOUNT_TAG	DF32 Status Zero Amount Allowed Flag Byte 1 0x01: option1 = online cryptogram request 0x02: option2 = not allowed
TLV Example : DF3001F8 DF320101 DF1B0100 9F660436004000	

2. 5. 35. 3. Discover

Constant names	Describe
DISCOVER_SET_QUALIFIERS_TAG	9F66 Terminal Transaction Qualifiers Byte 1 bit 8: 1 = MSD supported bit 7: RFU (0) bit 6: 1 = EMV supported bit 5: 1 = EMV contact chip supported bit 4: 1 = Offline-only reader bit 3: 1 = Online PIN supported bit 2: 1 = Signature supported bit 1: RFU Byte 2

	<p>bit 8: 1 = Online cryptogram required</p> <p>bit 7: 1 = CVM required</p> <p>bits 6-1: RFU (00000)</p> <p>Byte 3</p> <p>bit 8: 1 = Issuer Update Processing supported</p> <p>bit 7: 1 = Mobile functionality supported (Consumer Device CVM)</p> <p>bits 6-1: RFU (000000)</p> <p>Byte 4</p> <p>RFU ('00')</p> <p>TTQ byte 2 bits 8-7 are transient values, and reset to zero at the beginning of the transaction. All other TTQ bits are static values, and not modified based on transaction conditions.</p> <p>TTQ byte 3 bit 7 shall be set by the acquirer-merchant to 1b.</p>
DISCOVER_SET_ENTRY_POINT_TAG	<p>DF30 Entry Point</p> <p>Byte 1</p> <p>B8: StatusCheckSupportFlag</p> <p>b7: ZeroAmountAllowedFlag</p> <p>b6: ContactLessTransactionLimitFlag</p> <p>b5: ContactLessFloorLimitFlag</p> <p>b4: CVMRequiredLimitFlag</p> <p>b3..b1 : RFU</p>
DISCOVER_SET_STATUS_ZERO_AMOUNT_TAG	<p>DF32 Status Zero Amount Allowed Flag</p> <p>Byte 1</p> <p>0x01: option1 = online cryptogram request</p> <p>0x02: option2 = not allowed</p>
<p>TLV Example :</p> <p>DF3001F8 DF320101 9F6604B6004000</p>	

2. 5. 35. 4. Amex

Constant names	Describe
AMEX_SET_CAPABILITIES_TAG	<p>9F6E Enhanced ContactLess Reader Capabilities</p> <p>Byte 1</p> <p>bit 8: 1 = EMV contact chip supported</p> <p>bit 7: 1 = MSD supported</p> <p>bit 6: 0 = (Fixed parameter)</p> <p>bit 5: 1 = (Fixed parameter)</p> <p>bit 4: 1 = (Fixed parameter)</p> <p>bit 3: 1 = Try Another Interface after a decline</p>

	<p>bit 2: RFU</p> <p>bit 1: RFU</p> <p>Byte 2</p> <p>bit 8: 1 = Mobile CVM supported</p> <p>bit 7: 1 = Online PIN supported</p> <p>bit 6: 1 = Signature supported</p> <p>bits 5-1: RFU (00000)</p> <p>Byte 3</p> <p>bit 8: 1 = Terminal is offline only</p> <p>bit 7: 1 = CVM Required</p> <p>bits 6-1: RFU (000000)</p> <p>Byte 4</p> <p>RFU ('00')</p> <p>Byte 3 bit 7 are transient values, and reset to zero at the beginning of the transaction.</p>
AMEX_SET_KERNEL_CONFIG_TAG	<p>DF1B Kernel Configuration</p> <p>Byte 1</p> <p>b3: DelayedAuthorizationFlag</p> <p>b2: UseTACDefaultIfUnableToGoOnlineFlag</p> <p>b1: DrlSupportFlag</p>
AMEX_SET_ENTRY_POINT_TAG	<p>DF30 Entry Point</p> <p>Byte 1</p> <p>B8: StatusCheckSupportFlag</p> <p>b7: ZeroAmountAllowedFlag</p> <p>b6: ContactLessTransactionLimitFlag</p> <p>b5: ContactLessFloorLimitFlag</p> <p>b4: CVMRequiredLimitFlag</p> <p>b3..b1 : RFU</p>
AMEX_SET_STATUS_ZERO_AMOUNT_TAG	<p>DF32 Status Zero Amount Allowed Flag</p> <p>Byte 1</p> <p>0x01: option1 = online cryptogram request</p> <p>0x02: option2 = not allowed</p>
<p>TLV Example :</p> <p>DF3001F8 DF320101 DF1B0100 9F6E04C0E00000</p>	

2. 5. 35. 5. Quics

Constant names	Describe
----------------	----------

QUICS_SET_QUALIFIERS_TAG	<p>9F66 Terminal Transaction Qualifiers</p> <p>Byte 1</p> <p>bit 8: RFU</p> <p>bit 7: 1 = Full transaction flow in ContactLess interface Support (Mandatory no support)</p> <p>bit 6: 1 = EMV supported</p> <p>bit 5: 1 = EMV contact chip supported</p> <p>bit 4: 1 = Offline-only reader</p> <p>bit 3: 1 = Online PIN supported</p> <p>bit 2: 1 = Signature supported</p> <p>bit 1: RFU</p> <p>Byte 2</p> <p>bit 8: 1 = Online cryptogram required</p> <p>bit 7: 1 = CVM required</p> <p>bits 6-1: RFU (00000)</p> <p>Byte 3</p> <p>bit 8: RFU</p> <p>bit 7: 1 = Mobile functionality supported (Consumer Device CVM)</p> <p>bits 6-1: RFU (000000)</p> <p>Byte 4</p> <p>bit 8: 1 = fDDA v1.0 Supported (Mandatory support)</p> <p>bits 7-1: RFU (000000)</p> <p>TTQ byte 2 bits 8-7 are transient values, and reset to zero at the beginning of the transaction. All other TTQ bits are static values, and not modified based on transaction conditions.</p> <p>TTQ byte 3 bit 7 shall be set by the acquirer-merchant to 1b.</p>
QUICS_SET_ENTRY_POINT_TAG	<p>DF30 Entry Point</p> <p>Byte 1</p> <p>B8: StatusCheckSupportFlag</p> <p>b7: ZeroAmountAllowedFlag</p> <p>b6: ContactLessTransactionLimitFlag</p> <p>b5: ContactLessFloorLimitFlag</p> <p>b4: CVMRequiredLimitFlag</p> <p>b3..b1 : RFU</p>
QUICS_SET_STATUS_ZERO_AMOUNT_TAG	<p>DF32 Status Zero Amount Allowed Flag</p> <p>Byte 1</p> <p>0x01: option1 = online cryptogram request</p> <p>0x02: option2 = not allowed</p>
<p>TLV Example :</p> <p>DF3001F8 DF320101 9F660436004000</p>	

2. 5. 36. EmvTransDataConstraints

Constant names	Describe	Bundle Value Type	Example
TRANSTYPE	Trans Type(9C)	int	PosEmvCoreManager.EMV_CASH, PosEmvCoreManager.EMV_GOODS,
TRANSAMT	Trans Amt(9F02)	int	1 == 0.01, 100 == 1.
CASHBACKAMT	Trans Cashback Amt(9F03)	int	1 == 0.01, 100 == 1.
TRANSDATE	Trans Date(9A)	String	YYMMDD
TRANSTIME	Trans Time(9F21)	String	hhmmss
TRANSMODE	Trans Card Type	int	Support for the ICC, PICC, MSD PosEmvCoreManager. DEV_ICC DEV_PICC DEV_MAG; Support for the ICC, PICC PosEmvCoreManager. DEV_ICC DEV_PICC;
TRANSFALLBACK	Trans Fallback	boolean	Whether the IC card is degraded.
TRANSTIMEOUTMS	Card Search Time	int	60ms
OPENENCRYPT	Open Encrypt	int	Dynamic encryption configuration Support for the ICC, PICC, MSD : ENCRYPT_OPEN_ICC ENCRYPT_OPEN_PICC ENCRYPT_OPEN_MAG; Only support MSD : ENCRYPT_OPEN_MAG
ENCRYPTCONTACT	Encrypt Contact	Bundle	Encryption configuration.
ENCRYPTCONTACTLESS	Encrypt Contactless	Bundle	Encryption configuration.

ENCRYPTMAGSTRIPE	Encrypt MagStripe	Bundle	Encryption configuration.
ENCRYPTKEYINDEX	Encrypt Key Index	int	
ENCRYPTKEYTYPE	Encrypt Key Type	int	Version type : 0x01 - TDK 0x02 - DUKPT MAC 0x03 - DUKPT DATA REQUEST 0x04 - DUKPT DATA RESPONSE 0x05 - DUKPT PIN 0x06 - RSA TRANS ARMOR
ENCRYPTPADDING	Encrypt Padding	byte	"0"
ENCRYPTMODE	Encrypt Mode	int	Version type : 0x01 - ECB 0x02 - CBC Constant type : CARDREADER_MODE_ECB = 1; CARDREADER_MODE_CBC = 2;
ENCRYPTVECTOR	Encrypt Vector	byte []	Hex : 0000000000000000
RSATRANSARMORPOSID	RSA Trans Armor POS Id	byte []	Byte: 12345678
RSATRANSARMORKEYID	RSA Trans Armor Key Id	byte []	Hex: 166257982464
ENCRYPTBASE64	Encrypt Base64	boolean	

2. 5. 37. EmvPinConstraints

Constant names	Describe	Bundle Value Type	Example
PINTYPE	Pin Type	int	PIN_TYPE_OFF_PIN, PIN_TYPE_ONLINE_PIN,

			PIN_TYPE_OFF_CIPHER_PIN,
PINCARD	Card(5A)	String	<p>“444212131313”</p> <p>If Encrypt is turned off, it will return; if it is not turned off, it will not return.</p>
PINENCRYPTCARD	Encrypt Card	String	<p>If Encrypt is turned on, it will return. If it is not turned on, it will not return.</p>
PINALLOWBYPASS	Pin ByPass	boolean	
PINOFFTRYCNT	Pin TryCnt	int	
PINCARDRND	Off-line cipher text random number	byte[]	
PINPUBEXP	Off-line cipher text pub exponents	byte[]	
PINPUBMODEL	Off-line cipher text pub model	byte[]	
OUTPINBLOCK	Pin input results, PinBlock	byte[]	
OUTPINOFFTRYCNT	Pin input results, PinOffTryCnt	int	
OUTPINVERIFYRESULT	Pin input results, PinVerifyResult	int	<p>PosEnvCoreManager.</p> <p>EMV_VERIFY_SUC,</p> <p>// succeed</p> <p>EMV_VERIFY_NO_PINPAD,</p> <p>// No password keyboard or keyboard</p> <p>EMV_VERIFY_NO_PASSWORD,</p> <p>// No password or user ignores the password input</p> <p>EMV_VERIFY_PIN_ERROR,</p>

			// Password error 63cx EMV_VERIFY_PIN_BLOCK, // Pin locking
--	--	--	---

2. 5. 38. EmvOnlineRequestConstraints

Constant names	Describe	Bundle Value Type	Example
EMVDATA	EMV Data	byte[]	“9F4104000000019F1E08393030303 03030319F3303E0F9C89F350122950 500000000009F3704F18FDDAD9F020 60000000121019F030600000000000 0”
ENCRYPTDATA	Encrypt Data	byte[]	DF30 : Track 1 56 DF31 : Track 2 57 DF32: PAN 5A DF33 : PAN Mask
ENCRYPTRESULT	Encrypt Result	int	0 means encryption is successful and other failures

2. 5. 39. EmvOnlineResultConstraints

Constant names	Describe	Bundle Value Type	Example
REQUESTAC	Authorization Code Tag 8A	int	EMV_ONLINE_APPROVE 3030 EMV_ONLINE_FAIL !3030 EMV_ONLINE_DENIAL Connection failed, connection failed (Y3 / Z3)

REQUESTAUTHCODE	Issuer Data Tag 91	byte[]	"9108E64A2FE21FD88672"
REQUESTSCRIPT	Issuer Script Tag 71,72	byte[]	"72219F180411223344860D8424000 008AABCCDD112233448609842400 000455667788"

2. 5. 40. EmvProcessResultConstraints

Constant names	Describe	Bundle Value Type	Example
EMVDATA	EMV Data	byte[]	"9F4104000000019F1E08393030303 03030319F3303E0F9C89F350122950 500000000009F3704F18FDDAD9F020 60000000121019F030600000000000 0"
CVM	CVM	int	CVM_NO_CVM, CVM_SIGNATURE, CVM_CONFIRMATION_CODE_VERIFYIE D, CVM_SEE_PHONE
REQUSETISSUESCRIPT	Script Result	byte[]	
ENCRYPTDATA	Encrypt Data	byte[]	DF30 : Track 1 56 DF31 : Track 2 57 DF32: PAN 5A DF33 : PAN Mask
ENCRYPTRESULT	Encrypt Result	int	0 means encryption is successful and other failures

3. POI SDK Error Code

3.1. ICCR Error Code

Error code	Error name	Comment
F7CC(-2100)	SC_VCCERR	Voltage mode error
F7CB(-2101)	SC_SLOTERR	Cartoon error
F7CA(-2102)	SC_PARERR	Parity error
F7C9(-2103)	SC_PARAERR	Parameter error
F7C8(-2104)	SC_PROTOCOLERR	Protocol error
F7C7(-2105)	SC_DATALENERR	Wrong data length
F7C6(-2106)	SC_CARDOUT	Card out
F7C5(-2107)	SC_NORESET	Not initialized
F7C4(-2108)	SC_TIMEOUT	Card communication timeout
F7C3(-2109)	SC_PPSERR	PPS error
F7C2(-2110)	SC_ATRERR	ATR error
F7C1(-2111)	SC_APDUERR	Card communication failed
F7AD(-2131)	MEMIC_ERR_FAIL	Operation failed
F7AC(-2132)	MEMIC_ERR_NOCARD	No card
F7AB(-2133)	MEMIC_ERR_CARD_REMOVED	The card was removed during the operation
F7AA(-2134)	MEMIC_ERR_UNSUP_TYPE	Card type is not supported
F7A9(-2135)	MEMIC_ERR_MFC	Card manufacturer code is incorrect
F7A8(-2136)	MEMIC_ERR_PARAM	Parameter error

F7A7(-2137)	MEMIC_ERR_UNSUP_OPT	Operation is not supported yet
F7A6(-2138)	MEMIC_ERR_AUTH	Authentication failed
F7A5(-2139)	MEMIC_ERR_CHECK	Bad password, limited number of times
F7A4(-2140)	MEMIC_ERR_VERIFY	Wrong password, no limit
F7A3(-2141)	MEMIC_ERR_READ	Failed to read
F7A2(-2142)	MEMIC_ERR_WRITE	Write failure
F7A1(-2143)	MEMIC_ERR_ERASE	Erase failed
F7A0(-2144)	MEMIC_ERR_RESC	Failed to reset password verification
F79F(-2145)	MEMIC_ERR_NACK	Did not receive ACK
F79E(-2146)	MEMIC_ERR_WRITE_P	Write protection bit failed
F79D(-2147)	MEMIC_ERR_PROTECTED	Has been write protected, not operational
F79C(-2148)	MEMIC_ERR_KEY_UPDATE	Password update failed
F79B(-2149)	MEMIC_ERR_CARD_LOCKED	Card is locked

3. 2. PCD Error Code

Error code	Error name	Comment
F447(-3001)	RET_RF_ERR_PARAM	Parameter error
F446(-3002)	RET_RF_ERR_NO_OPEN	RF module is not turned on
F445(-3003)	RET_RF_ERR_NOT_ACT	Card not activated
F444(-3004)	RET_RF_ERR_MULTI_CARD	Multiple cards
F443(-3005)	RET_RF_ERR_TIMEOUT	Timeout is not responding
F442(-3006)	RET_RF_ERR_PROTOCOL	Protocol error
F441(-3007)	RET_RF_ERR_TRANSMIT	Communication transmission error

F440(-3008)	RET_RF_ERR_AUTH	M1 card authentication failed
F43F(-3009)	RET_RF_ERR_NO_AUTH	Sector not certified
F43E(-3010)	RET_RF_ERR_VAL	Numeric block data format is incorrect, or the file size is large in DES Fire card operation tiny error
F43D(-3011)	RET_RF_ERR_CARD_EXIST	The card is still in the sensing area inside
F43C(-3012)	RET_RF_ERR_STATUS	Card status error (such as a/b card call m1 card interface)
F43B(-3013)	RET_RF_ERR_OVERFLOW	
F43A(-3014)	RET_RF_ERR_FAILED	DES Fire card should answer data error
F439(-3015)	RET_RF_ERR_COLLERR	
F438(-3016)	RET_RF_ERR_FIFO	Application buffer empty in DES Fire card operation Lack of
F437(-3017)	RET_RF_ERR_CRC	
F436(-3018)	RET_RF_ERR_FRAMING	
F435(-3019)	RET_RF_ERR_PARITY	
F434(-3020)	RET_RF_ERR_DES_VAL	DES Fire card response Data DES operation result is inconsistent
F433(-3021)	RET_RF_ERR_NOT_ALLOWED	The operation is not allowed. For example, when the currently selected file is not a record file, the read record cannot be executed.
F3E4(-3100)	RET_RF_ERR_CHIP_ABNORMAL	Operating interface chip does not exist Or abnormal
F37F(-3201)	RET_RF_DET_ERR_INVALID_PARAM	
F37E(-3202)	RET_RF_DET_ERR_NO_POWER	
F37D(-3203)	RET_RF_DET_ERR_NO_CARD	
F37C(-3,204)	RET_RF_DET_ERR_MULTIPLE_CARDS	
F37B(-3,205)	RET_RF_DET_ERR_ACT	
F37A(-3,206)	RET_RF_DET_ERR_PROTOCOL	

F31B(-3,301)	RET_RF_CMD_ERR_INVALID_PARAM	
F31A(-3,302)	RET_RF_CMD_ERR_NO_POWER	
F319(-3,303)	RET_RF_CMD_ERR_NO_CARD	
F318(-3,304)	RET_RF_CMD_ERR_TX	
F317(-3,305)	RET_RF_CMD_ERR_PROTOCOL	
FF38(-200)	PHILIPS_MIFARE_ERR_NACK	
FF37(-201)	PHILIPS_MIFARE_ERR_COMM	
FF36(-202)	PHILIPS_MIFARE_ERR_AUTHEN	
FF35(-203)	PHILIPS_MIFARE_ERR_CRC	
FF34(-204)	PHILIPS_MIFARE_ERR_REJECT	

3. 3. MSR Error Code

Error code	Error name	Comment
0xFFFF(-1)		Open, close the magnetic stripe card failed
0xFD41(-703)	ERR_MSR_READERR	Read the card, but read the card wrong
0xFD42(-702)	ERR_MSR_OPENERR	
0xFD43(-701)	ERR_MSR_NOSWIPED	

3. 4. Keypad Error Code

Error code	Error name	Comment
FFFF(-1)		Time out
FFFD(-3)		User cancel input (ESC key)

FFFC(-4)		IC card is removed
FFFB(-5)	ERR_PIN_LEN	Password length mode setting is not required to be greater than 4 Less than 12 digits
FFFA(-6)	KEY_ERRPAD	Keyboard (TP) module error
FC0B(-1013)	KB_USER_ENTER_Exit	User cancel input (command mode)
FC09(-1015)	KB_USER_SLEEP	User actively goes to sleep
FC03(-1021)	TP_TAG_ERR	Did not receive 0x73 this TLV
FC02(-1022)	TP_TLV_ERR	The number of button coordinates received is incorrect.
FC01(-1023)	TP_EXG_SIZE_ERR	The coordinates received are incorrect, upper left and lower right Coordinates are not logical
FC00(-1024)	TP_CMDEXG_TO_ERR	Coordinate exchange command wait timeout

3. 5. RTC Error Code

Error code	Error name	Comment
FC7C(-900)	TIME_FORMAT_ERR	Time format error
FC7B(-901)	TIME_YEAR_ERR	Year error
FC7A(-902)	TIME_MONTH_ERR	Month error
FC79(-903)	TIME_DAY_ERR	Day error
FC78(-904)	TIME_HOUR_ERR	Hour error
FC77(-905)	TIME_MINUTE_ERR	Minute error
FC76(-906)	TIME_SECOND_ERR	Second error
FC75(-907)	TIME_WEEK_ERR	Weekly error

FC74(-908)	TIME_SET_ERR	Setup failed
FC73(-909)	TIME_GET_ERR	Acquisition failed
FC72(-910)	TIME_RAMADDR_OVER	Time content address is exceeded
FC71(-911)	TIME_RAMLEN_OVER	Length of time exceeded

3. 6. HSM Error Code

Error code	Error name	Comment
FED3(-301)	PED_RET_ERR_NO_KEY	The key does not exist
FED2(-302)	PED_RET_ERR_KEYIDX_ERR	Key index error, parameter index not in range
FED1(-303)	PED_RET_ERR_DERIVE_ERR	Source key class when the key is written The type error or level is lower than the destination key
FED0(-304)	PED_RET_ERR_CHECK_KEY_FAIL	Key verification failed
FECF(-305)	PED_RET_ERR_NO_PIN_INPUT	Did not enter the pin
FECE(-306)	PED_RET_ERR_INPUT_CANCEL	User cancels input pin
FECD(-307)	PED_RET_ERR_WAIT_INTERVAL	Function call is less than minimum interval time
FECC(-308)	PED_RET_ERR_CHECK_MODE_ERR	KCV mode is wrong, not supported
FECB(-309)	PED_RET_ERR_NO_RIGHT_USE	Not authorized to use this key, PED current key tag value and want to make The key tag values used are not equal
FECA(-310)	PED_RET_ERR_KEY_TYPE_ERR	Key type error
FEC9(-311)	PED_RET_ERR_EXPLEN_ERR	Expected pin length string wrong
FEC8(-312)	PED_RET_ERR_DSTKEY_IDX_ERR	The destination key index is wrong, not Within the scope
FEC7(-313)	PED_RET_ERR_SRCKEY_IDX_ERR	The source key index is incorrect, not in scope or when the key is written, the value of the source key type is greater than the destination Key type, will return the key

FEC6(-314)	PED_RET_ERR_KEY_LEN_ERR	Key length error
FEC5(-315)	PED_RET_ERR_INPUT_TIMEOUT	Input pin timeout
FEC4(-316)	PED_RET_ERR_NO_ICC	ICC card does not exist
FEC3(-317)	PED_RET_ERR_ICC_NO_INIT	ICC card is not initialized
FEC2(-318)	PED_RET_ERR_GROUP_IDX_ERR	DUKPT group index number error
FEC1(-319)	PED_RET_ERR_PARAM_PTR_NULL	The pointer parameter is illegally empty
FEC0(-320)	PED_RET_ERR_TAMPERED	PED has been attacked
FEBF(-321)	PED_RET_ERROR	PED generic error
FEBE(-322)	PED_RET_ERR_NOMORE_BUF	No free buffer
FEBD(-323)	PED_RET_ERR_NEED_ADMIN	Need to get advanced permissions
FEBC(-324)	PED_RET_ERR_DUKPT_OVERFLOW	DUKPT has overflowed
FEBB(-325)	PED_RET_ERR_KCV_CHECK_FAIL	KCV checksum failed
FEBA(-326)	PED_RET_ERR_SRCKEY_TYPE_ERR	Source key id when writing the key Key type and source key type do not match
FEB9(-327)	PED_RET_ERR_UNSPPT_CMD	Command not supported
FEB8(-328)	PED_RET_ERR_COMM_ERR	Communication error
FEB7(-329)	PED_RET_ERR_NO_UAPUK	No user authentication public key
FEB6(-330)	PED_RET_ERR_ADMIN_ERR	Failed to take system sensitive service
FEB5(-331)	PED_RET_ERR_DOWNLOAD_INACTIVE	PED is inactive download state
FEB4(-332)	PED_RET_ERR_KCV_ODD_CHECK_FAIL	KCV odd parity failed
FEB3(-333)	PED_RET_ERR_PED_DATA_RW_FAIL	Failed to read PED data
FEB2(-334)	PED_RET_ERR_ICC_CMD_ERR	ICC card operation error (offline Text, Cipher text password verification)
FEB1(-335)	PED_RET_ERR_KEY_VALUE_INVALID	The key written is all zero or equal, there are equal components, and the 16/24 byte key has two components equal.

FEB0(-336)	PED_RET_ERR_KEY_VALUE_EXIST	Already have the same key value
FEAF(-337)	PED_RET_ERR_UART_PARAM_INVALID	Serial port parameters are not supported
FEAE(-338)	PED_RET_ERR_KEY_INDEX_NOT_SELECTED_OR_NOT_MATCH	The key index is not selected or is not related to the selected key index. Wait
FEAD(-339)	PED_RET_ERR_INPUT_CLEAR	User presses clear key to exit Input pin
FEAC(-340)	PED_RET_ERR_LOAD_TRK_FAIL	
FEAB(-341)	PED_RET_ERR_TRK_VERIFY_FAIL	
FEAA(-342)	PED_RET_ERR_MSR_STATUS_INVALID	
FEA9(-343)	PED_RET_ERR_NO_FREE_FLASH	
FEA8(-344)	PED_RET_ERR_DUKPT_NEED_INC_KSN	DUKPT KSN needs to be added 1
FEA7(-345)	PED_RET_ERR_KCV_MODE_ERR	KCV mode error
FEA6(-346)	PED_RET_ERR_DUKPT_NO_KCV	NO KCV
FEA5(-347)	PED_RET_ERR_PIN_BYPASS_BYFUNKEY	Press the FN/atm4 key to cancel the pin
FEA4(-348)	PED_RET_ERR_MAC_ERR	
FEA3(-349)	PED_RET_ERR_CRC_ERR	
FEA2(-350)	PED_RET_ERR_ALG_ERR	
FEA1(-351)	PED_RET_ERR_STATE	
FEA0(-352)	PED_RET_ERR_PWD	
FE9F(-353)	PED_RET_ERR_NEWPWD	
FE9E(-354)	PED_RET_ERR_PWD_OVERRUN	
FE9D(-355)	PED_RET_ERR_REQ_SSA	
FE9C(-356)	PED_RET_ERR_UNLOCK_TIMEOUT	
FE9B(-357)	PED_RET_ERR_UNLOCK_PARAM	
FE9A(-358)	PED_RET_ERR_UNLOCK_NO	

FE99(-359)	PED_RET_ERR_UNLOCK_CHECK	
FE98(-360)	PED_RET_ERR_NEED_RESET	
FE97(-361)	PED_RET_ERR_RNG	
FE96(-362)	PED_RET_ERR_NEED_RMT_AUTH	
FE70(-400)	PED_RET_ERR_KEY_KCV_TAB_NULL	
FE6F(-401)	PED_RET_ERR_PED_CFG_RW_FAIL	
FE6E(-402)	PED_RET_ERR_SEK_VERIFY_FAIL	
FE6D(-403)	PED_RET_ERR_SEK_ALL_ZERO	
FE6C(-404)	PED_RET_ERR_PINBLOCK_FMT	
FE6B(-405)	PED_RET_ERR_PIN_LEN	
FE6A(-406)	PED_RET_ERR_PAN_LEN	
FE69(-407)	PED_RET_ERR_DATA_LEN	
FE68(-408)	PED_RET_ERR_KEY_LEN	
FE67(-409)	PED_RET_ERR_CMAC	
FE66(-410)	PED_RET_ERR_READ_CFG	
FE65(-411)	PED_RET_ERR_KEY_STRENGTH	
FE64(-412)	PED_RET_ERR_PARAM	
FE63(-413)	PED_RET_ERR_WORK_MODE	
FE62(-414)	PED_RET_ERR_KEY_INDEX	
FE61(-415)	PED_RET_ERR_KEYBLOCK_VER	
FE60(-416)	PED_RET_ERR_FORBIDDEN	
FE5F(-417)	PED_RET_ERR_INVALID_SIGN	
FE5E(-418)	PED_RET_ERR_INVALID_USE	