PKOC NFC Card Specification

Security Classification:	Protected
Version:	1.1
Revision:	Rev0
Control:	Uncontrolled when printed.
Date	12/1/2023

Document History

Version	Date	Author	Description
0.0-.99	Oct 8, 2022	Mohammad Soleimani	Initial Document
1.0RC1	June 2, 2023	Mohammad Soleimani, Ryan Littleton	Updates, Changed to PSIA Stationary
1.0	June 13, 2023	Mohammad Soleimani	Minor improvements for clarity
1.1	Dec 1, 2023	Mohammad Soleimani	Clarified the necessary number of bits for reader output. Minor edits.

Disclaimer

THIS DOCUMENT IS PROVIDED "AS IS" WITH NO WARRANTIES WHATSOEVER, INCLUDING ANY WARRANTY OF MERCHANTABILITY, NONINFRINGEMENT, FITNESS FOR ANY PARTICULAR PURPOSE, OR ANY WARRANTY OTHERWISE ARISING OUT OF ANY PROPOSAL, SPECIFICATION OR SAMPLE. Without limitation, PSIA disclaims all liability, including liability for infringement of any proprietary rights, relating to use of information in this specification and to the implementation of this specification, and PSIA disclaims all liability for cost of procurement of substitute goods or services, lost profits, loss of use, loss of data or any incidental, consequential, direct, indirect, or special damages, whether under contract, tort, warranty or otherwise, arising in any way out of use or reliance upon this specification or any information herein.

You are hereby granted a license to copy and distribute (but not to modify) the information set forth in this document; and to make and sell, including commercially, products using the specification. Any rights that PSIA has not expressly granted are hereby reserved.

PSIA encourages you to explore membership in the organization to obtain the full set of benefits to be received from use of its specifications.

Contents

Abbreviations:

Abbreviation	Complete Text
PKOC	Public Key Open Credential
AID	Application Identifier
PACS	Physical Access Control System

Variables:

Variable Names	Value
AID	A000000898000001
Protocol Version	0100
Reader Identifier	128 bits (16 bytes) of site key identifier + 128 bits (16 bytes) of reader location identifier

Introduction:

This is a specification for a PKOC (Public Key Open Credential) implementation used for NFC-enabled physical access cards.

PKOC- Credential Creation and Provisioning

1) The card will generate a public/private key pair using the NIST secp256r1 (P256) curve, stored in an internal secure element. The private key never leaves the card, and the public key is used as a user's Public Key Open Credential (PKOC).
2) The card's PKOC will be registered in the PACS cloud during card enrollment.
3) For sending the credential from the Reader to the PACS panel, the Reader **Shall** send:
   a) **For newer systems that use bi-directional communication to the panel**, the 256 bits of the X component of the key as the credential
   b) For older systems that cannot support credentials this large, the lower 75 bits of the X component of the key (recommended) or the lower 64 bits of the X component (minimum) as the credential.
4) The length of the credential should be settable during provisioning to match the capabilities of the panel, as applicable.

See below for examples of extracting the credential from the full 65-byte Public Key.

For 256 bits -
04**BEA02AA1320054CFF1DFD2F88FA583B5B059833BA87CEC415ABDAE0791F0EC66**A913C7104A725F6497B8C0
8FF91217B106FEF7B51ACD4ADF6645E765E4E88D84

For 64 bits -
04BEA02AA1320054CFF1DFD2F88FA583B5B059833BA87CEC41**5ABDAE0791F0EC66**A913C7104A725F6497B8C0
8FF91217B106FEF7B51ACD4ADF6645E765E4E88D84

TLV

Within the NFC 14443A communication protocols, TLV (type-length-value or tag-length-value) is used for conveying informational elements. A TLV-encoded data stream contains code related to the record type, the record value's length, and finally the value itself. There is no strict ordering of TLVs within a message, the reader and card must be able to support any ordering.

Type	Details	Use	Length (Bytes)
0x5C	Protocol Version	Card → Reader (SELECT Response) Reader → Card (Authentication Command)	2
0x4C	Transaction Identifier (Reader Nonce)	Reader → Card (Authentication Command)	16 to 65
0x4D	Reader Identifier	Reader → Card (Authentication Command)	32
0x9E	Digital Signature	Card → Reader (Authentication Response)	64
0x5A	Uncompressed Public Key ECC P-256	Card → Reader (Authentication Response)	65

Note: For maintaining forward compatibility, any TLV command not recognized should be ignored by the card and the reader.

APDU (Application Protocol Data Unit)

SELECT Command

Field	Value (hex)	Description
CLA	00	Default class
INS	A4	Select
P1	04	Select by DF Name (AID)
P2	00	First or only occurrence
Lc	08	Data Length
Data	A000000898000001	AID
Le	00	Maximum length of expected response data

SELECT Response

Field	Value (hex)	Description
Supported Protocol Versions TLV	0100	One or more protocol versions supported by the Card. If the card supports multiple protocols, it will include all in the list, sorted by largest (most recent) first.
SW1/SW2	9000	Success Status (or other code on error)

Authentication Command

This command allows the reader to initiate the authentication process. It consists below information (TLVs):

- Selected Protocol Version
- Transaction ID – 16 bytes
- Reader Identifier - 32 bytes

Field	Value (hex)	Description
CLA	80	Manufacturer specific class
INS	80	Authentication
P1	00	shall be set to 0
P2	01	shall be set to 1
Lc	38	Data Length
Data	Var	Transaction ID, Protocol Version, Reader Identifier
Le	00	Maximum length of expected response data

Authentication Response

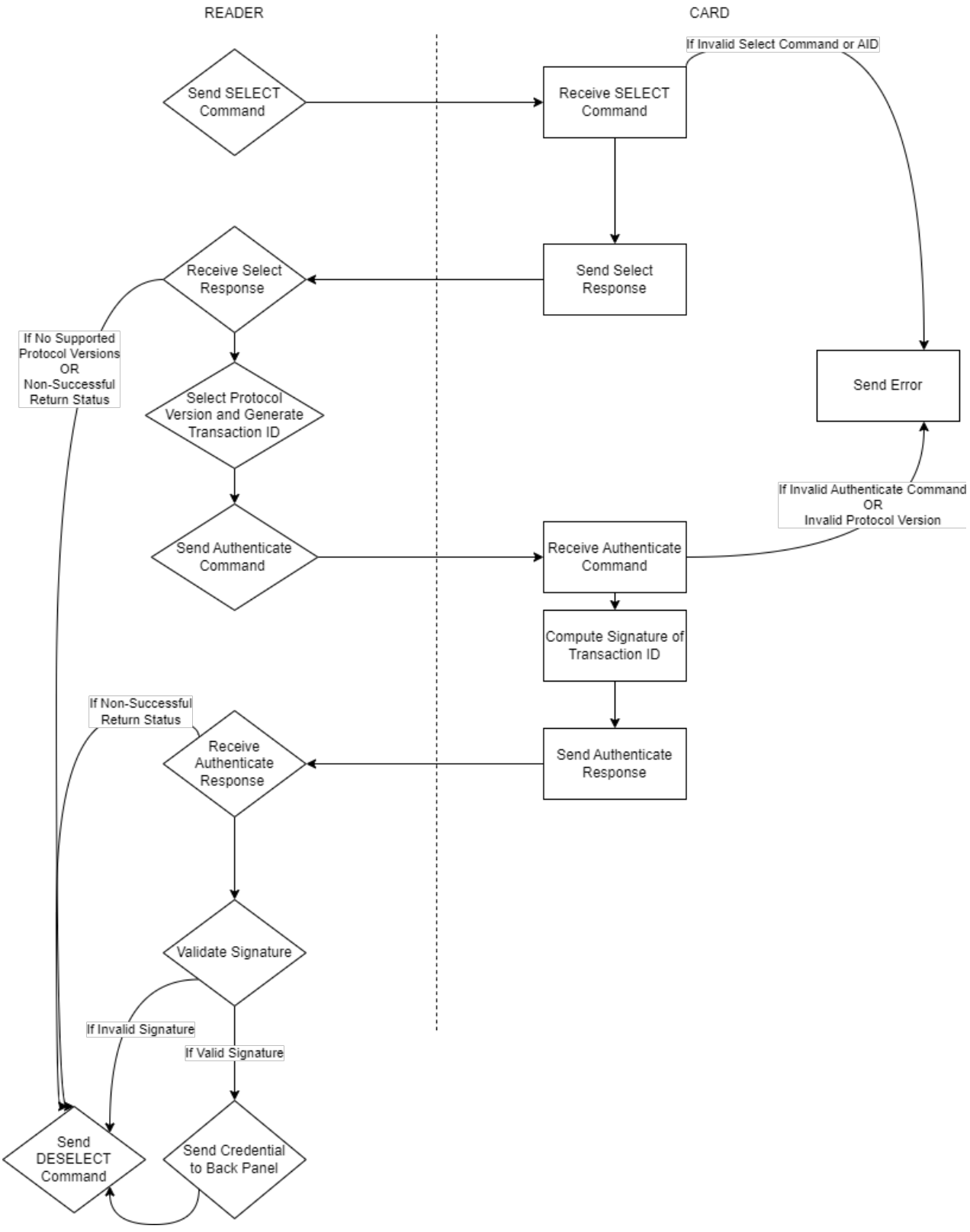
Field	Value	Description
Uncompressed Public Key TLV	Var	Public Key of card (65 bytes)
Digital Signature TLV	Var	Signature of Transaction ID (64 bytes)
SW1/SW2	0x9000	Success Status (or other code on error)

Status Codes

These status codes match ISO 7816-4 definitions and are only sent from the Card to the Reader. In the event of a Reader error the reader will issue a DESELECT command and stop communications with the card.

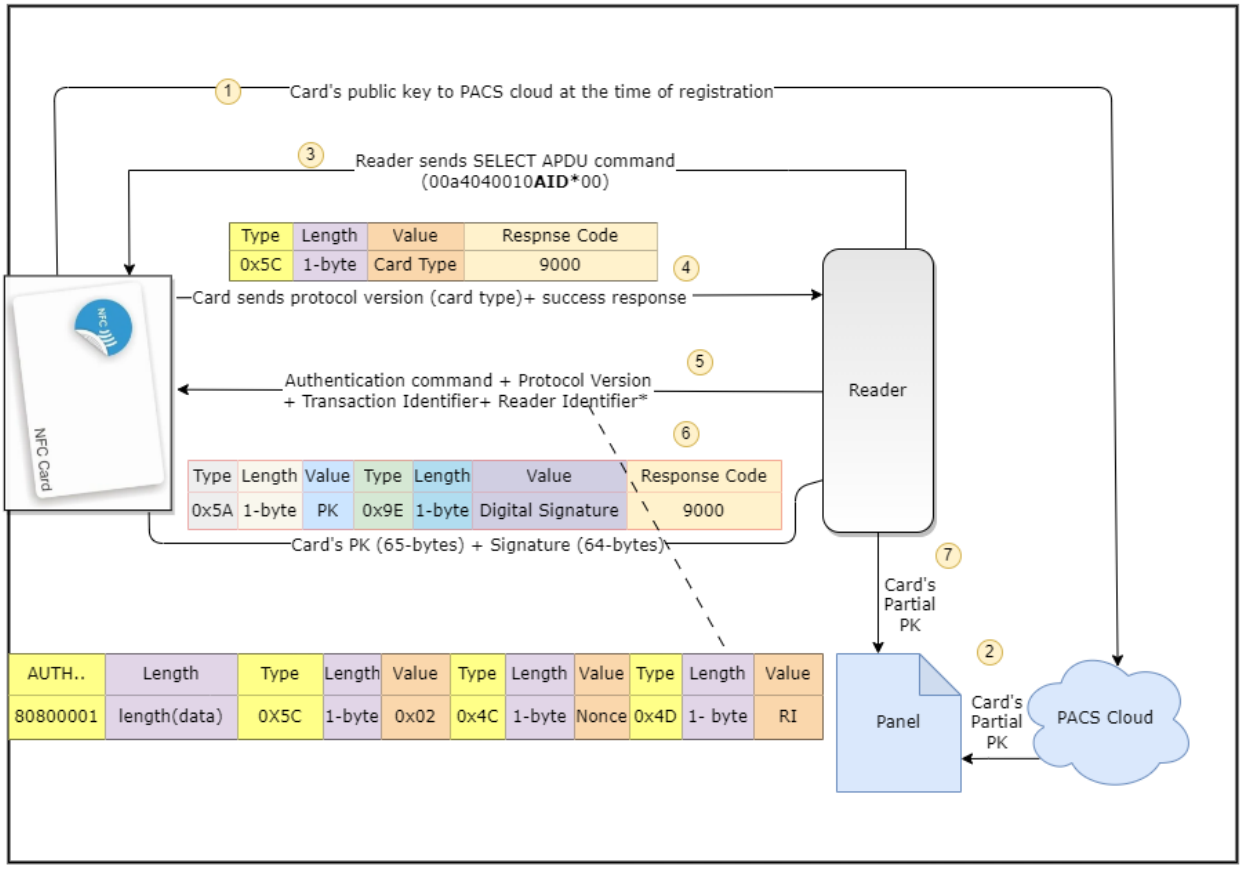
Code	Description
6700	Wrong length in LC
6985	Unsupported Protocol Version selected by Reader (Condition of use not satisfied)
6B00	Incorrect P1 or P2
6D00	Invalid INS
6E00	Invalid CLA
6F00	General Error
9000	Success

Flow Diagram



Flow Description

1. During card registration into the PACS, the public key is read off the card and registered in the PACS
2. PACS will send partial PKOC to the panel.
3. A reader sends the SELECT APDU Command to the card.
4. The card sends the Protocol Version in TLV along with Success Response to the Reader.
    1. Protocol Version (0x5C) + Length (Var) + Value (Protocol Version)
    2. 9000 – success response
5. The reader validates that it understands the requested Protocol Version and sends the Authentication Command (Protocol Version + Transaction ID + Reader Identifier) to the card.
6. The card generates a digital signature using ECDSA (NIST 186-5, Section 6), with the received Transaction ID as the input. The Digital Signature must only be the R & S components concatenated, not the ASN.1 encoding. The card then sends Digital Signature (64 bytes) and public key (65 Bytes) in TLV format to the Reader along with success response code 9000.
7. The reader validates the signature and, if valid, sends the partial public key as the credential to the panel.



Example:

Notes:

- Keys are in DER encoded format, the bolded sections are the 32 byte private and 65 byte public keys.
- There are multiple valid signatures for the same input data and key, this test data's signature may not exactly match another example.

1. Static Keys (Card)
    1. Private Key: 308187020100301306072a8648ce3d020106082a8648ce3d030107046d306b0201010420**c0c93d0ee2c83d077a91448478f438d633f0c9f863799f9574151fa1260d1349**a144034200040ec5d87dc39d14a2c5480686da860c82b16be0b6903b525f84848b79fd463e32bbda1f0252c33503c5287035e6eac55d138d0650dcfb5281d59a9cf4124d2831
    2. Public Key: 3059301306072a8648ce3d020106082a8648ce3d030107034200**040ec5d87dc39d14a2c5480686da860c82b16be0b6903b525f84848b79fd463e32bbda1f0252c33503c5287035e6eac55d138d0650dcfb5281d59a9cf4124d2831**
2. The reader sends SELECT APDU to CE: **00a40400**08A00000089800000100
3. The card sends Protocol Version + Success Response to RE: **5C**0201009000
4. A reader sends the Authentication Command + Protocol Version + Transaction ID + Reader Identifier to the Card: **80800001**38**5c**0201004**c**106fcf5012b224043b09350a4fc5e56a8f**4d**207a25432a462d4a404e635266556a586edfee8022966311eda1eb0242ac12000200
    1. Protocol Version - **5c**020100
    2. Transaction ID - **4c**106fcf5012b224043b09350a4fc5e56a8f
    3. Reader Identifier - **4d**207a25432a462d4a404e635266556a586edfee8022966311eda1eb0242ac120002
5. The card sends Public key (65 Bytes) and Digital Signature (64 bytes) in TLV format to the Reader along with success response code 9000. **5A**41040ec5d87dc39d14a2c5480686da860c82b16be0b6903b525f84848b79fd463e32bbda1f0252c33503c5287035e6eac55d138d0650dcfb5281d59a9cf4124d2831**9E**40b98613070c78010b04ed306d143f94ee6dc4eca2585b621405731fb3a53cd877a21685de18435da7cbcc38f1d926300a454efee3594cec5effe28c7feac03d7d9000
    1. Public Key - **5A**41040ec5d87dc39d14a2c5480686da860c82b16be0b6903b525f84848b79fd463e32bbda1f0252c33503c5287035e6eac55d138d0650dcfb5281d59a9cf4124d2831
    2. Signature - **9E**40b98613070c78010b04ed306d143f94ee6dc4eca2585b621405731fb3a53cd877a21685de18435da7cbcc38f1d926300a454efee3594cec5effe28c7feac03d7d
6. The reader verifies the signature. If verified, it converts the full Public Key into a partial key (credential) which is used to validate access.

