OSDP 2.3 Proposed PIV Enhancements

Security Industry Association

November 6, 2025

Contents

Changelog

2025-10-30 Clarified CRAUTH command/reply behaviour (including RSA, AES, and ECC support), overhauled PIV Mode configuration with TLV-based application and interface controls (including PIV Auto), added card status request/reply messages, expanded PIV data retrieval semantics, introduced PIV Put Data, TWIC privacy-key load, VCI pairing transmit, and VCI trust-anchor load/status commands, and documented the enhanced error-code set (including VCI/PIV Auto conditions).

2025-09-17 Major PIV Mode and SPE revisions.

2025-06-25 Added multi-AID usage and VCI support notes.

2025-05-27 New function code definitions for 2.3 biometric handling.

1 Introduction

This document captures proposed enhancements to the SIA Open Supervised Device Protocol (OSDP) needed to carry FICAM-compliant Personal Identity Verification (PIV) workflows. The material consolidates draft content, reviewer comments, and interoperability feedback collected during the 2025 working sessions and re-expresses it as normative guidance for command authors and device implementers.

The chapters that follow highlight changes required for existing OSDP 2.2 commands, introduce new transactions where no equivalent existed, and provide supplemental examples that reflect current federal credentialing practices. Wherever the proposal diverges from the baseline specification, the text records both the behavioral requirements and the error handling expectations so that access control units (ACUs) and peripheral devices (PDs) can adopt the profile without relying on vendor extensions.

2 Changes to OSDP 2.2

This section documents the normative deltas that apply when an OSDP 2.2 implementation adopts the Enhanced PIV profile. Each subsection references the original command definition and records the precise behavioral updates, encoding clarifications, and new error handling rules introduced by this proposal.

2.1 Updated Command Definitions

2.1.1 Authentication Challenge (osdp_CRAUTH)

OSDP 2.2 transports the ISO/IEC 7816-4 `GENERAL AUTHENTICATE` command through `osdp_CRAUTH`. The OSDP 2.3 profile maintains the same framing fields (`TOTAL`, `OFFSET`, `DATA_LEN`) but extends the semantics of the command payload so PDs can carry both the classic RSA signing flow and the enhanced elliptic-curve key-agreement flow.

- The Algorithm field continues to be sent in the first fragment only. Codes `0x05`, `0x06`, and `0x07` select RSA signing operations and preserve backwards compatibility with OSDP 2.2. Codes `0x08`, `0x0A`, and `0x0C` select AES encryption using 128-, 192-, and 256-bit keys respectively. Codes `0x11` and `0x14` select elliptic-curve key agreement using the P-256 and P-384 curves.
- The Key field retains its role as the card's reference identifier. PDs **SHALL NOT** reinterpret or remap the identifier.
- For RSA algorithms (`0x05`, `0x06`, `0x07`), the Challenge field carries the datum to be signed. The PD forwards the value unmodified; padding (for example PKCS#1 v1.5 or PSS) is the responsibility of the ACU.
- For AES algorithms (`0x08`, `0x0A`, `0x0C`), the Challenge field contains the plaintext block(s) to be encrypted. PDs **SHALL** send the data to the credential without padding; ACUs **SHALL** provide data aligned to the block size dictated by the selected key length and card profile.
- For algorithms `0x11` and `0x14`, the Challenge field conveys the peer public key. The PD **SHALL** issue `GENERAL AUTHENTICATE` with `CLA = 0x00` (or `0x0C` when a secure channel is active), `INS = 0x87`, `P1 = Algorithm`, and `P2 = Reference Identifier`. The command data **SHALL** be encoded as `7C Len 82 00 85 Len (04 || X || Y)`, where `X` and `Y` are the uncompressed coordinates for the selected curve and are left-padded with `0x00` as required. The credential performs elliptic-curve Diffie–Hellman in accordance with SP 800-73; the derived secret is returned in the reply fragments described below.

Table 1: Authentication Challenge (osdp_CRAUTH)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	0xA5
2	TOTAL	Length of the complete message, least-significant byte first.	0x0000–0xFFFF
2	OFFSET	Offset of this fragment within the complete message, least-significant byte first.	0x0000–0xFFFF
2	DATA_LEN	Length of the fragment payload, least-significant byte first.	0x0000–0xFFFF
1	Algorithm	Cryptographic mechanism. Supported values include 0x05/0x06/0x07 (RSA signature), 0x08/0x0A/0x0C (AES encryption with 128/192/256-bit keys), and 0x11/0x14 (ECC-P256/P384 ECDH).	0x00–0xFF
1	Reference Identifier	Key reference passed to GENERAL AUTHENTICATE.	0x00–0xFF
0–n	Challenge	For RSA algorithms, the datum to be signed (no padding applied by the PD). For AES algorithms, the plaintext block(s) to be encrypted, pre-padded by the ACU as required. For 0x11/0x14, the peer public key encoded as 7C Len 82 00 85 Len (04 || X || Y) with coordinates padded to curve length.	0x00–0xFF

2.1.2 Response to Challenge (osdp_CRAUTHR)

The reply continues to use the OSDP 2.2 multi-part format. Successful execution yields a response containing the credential's entire GENERAL AUTHENTICATE data (minus the ISO/IEC 7816 status word); PDs **SHALL NOT** strip outer BER-TLV tags or lengths. If the credential reports an error, the PD **SHALL** return osdp_NAK using the appropriate code from the standard set (for example 0x01 security condition not

met, 0x02 card removed, 0x03 PIN blocked, 0x04 wrong interface, 0x05 incorrect parameter, 0x06 PIN verification failed, 0x07 PIN verification timeout, 0x09 unsupported algorithm/reference, 0x0A general authentication failure).

Table 2: Response to Challenge (osdp_CRAUTHR)

Size (bytes)	Name	Meaning	Value
1	CMND	Reply identifier.	0x82
2	TOTAL	Length of the complete response, least-significant byte first.	0x0000–0xFFFF
2	OFFSET	Offset of this fragment within the complete response, least-significant byte first.	0x0000–0xFFFF
2	DATA_LEN	Length of the fragment payload, least-significant byte first.	0x0000–0xFFFF
0–n	Response Data	Signature, encrypted block, or shared-secret material returned by the credential. The credential's BER-TLV structure is preserved exactly as received (status word excluded).	0x00–0xFF

2.1.3 Get PIV Data (osdp_PIVDATA)

OSDP 2.2 defined `osdp_PIVDATA` with a five-byte payload that selected a PIV object and optional element. The original text did not explicitly specify how to pad identifiers shorter than three bytes, how nested tags should be returned, or how to interpret the data offset. OSDP 2.3 clarifies these behaviors so that ACUs and PDs apply a single normative approach.

- The ACU **SHALL** transmit the PIV Object Identifier as three bytes encoded most-significant byte first. When the identifier contains fewer than three significant bytes, the ACU **SHALL** left-pad the value with 0x00 or 0x0000, as required, so that the transmitted field is exactly three bytes. The PD **SHALL NOT** forward those pad bytes to the credential; it **SHALL** issue the ISO 7816-4 `GET DATA` APDU using only the significant identifier bytes. If VCI secure messaging is established, the PD **SHALL** use the appropriate secure channel automatically; no explicit selector is present in the command.
- The field previously described as "PIV element ID" is renamed PIV Tag. A tag value of 0x00 directs the PD to return the complete BER-TLV structure, including the outer 0x53 or 0x73 tag and length. Any non-zero value selects a single root-level tag within the object. The PD **MUST** remove the outer

wrapper only when it is 0x53 or 0x73; if the outer wrapper has any other value, the PD **MUST** reject the command unless the tag value is 0x00. Tags **SHALL** be a single byte in the range 0x01–0xFF; multi-byte DER tags are not supported and **MUST** be retrieved by issuing a tag value of 0x00 and parsing the returned object at the ACU.

- The data offset now supports one- or two-byte encodings. PDs **SHALL** accept either form and determine which was used based on the command length. The offset applies to the selected payload (entire object when the tag is zero, or the chosen tag otherwise). PDs **SHALL** return osdp_NAK with error code 0x0D when security prerequisites are not satisfied and 0x0E when the requested object or tag is absent; see Subsection 2.2.

Table 3: Get PIV Data (osdp_PIVDATA)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	0xA3
3	PIV Object Identifier	Three-byte identifier of the PIV data object per SP 800-73-4, encoded MSB first. Leading zero bytes SHALL be supplied by the ACU when the identifier is shorter than three bytes; the PD SHALL drop those pad bytes before issuing the APDU.	0x00–0xFF[1]
1	PIV Tag	Single-byte ASN.1 DER tag of a root-level element. Tag value 0x00 selects the entire object; non-zero tags select a root-level element and require the PD to strip the outer 0x53/0x73 wrapper, verify the length, and return only the tagged value. Multi-byte tags are not directly addressable and SHALL be retrieved by requesting tag 0x00.	0x00; or 0x01–0xFF
1–2	Data Offset	Byte offset within the requested content. Single-byte and two-byte encodings SHALL be accepted; the PD determines the format from the packet length.	0x00–0xFFFF

[1] Each position MAY carry any value from 0x00 to 0xFF; the ACU pads on the left with 0x00 when the identifier has fewer than three significant bytes.

2.1.4 PIV Data Reply (osdp_PIVDATAR)

The reply structure is unchanged from a field-count perspective but 2.3 aligns the semantics with the command refinements above:

- `TOTAL`, `OFFSET`, and `DATA_LEN` continue to be encoded LSB first; they now explicitly report the size, fragment base, and fragment length for the content selected by the `osdp_PIVDATA` request (entire object or chosen tag).
- `CARD_DATA` **SHALL** convey the exact bytes obtained from the credential after applying any tag stripping specified by the command. No additional framing or padding is permitted.

Table 4: PIV Data Reply (osdp_PIVDATAR)

Size (bytes)	Name	Meaning	Value
1	CMND	Reply identifier.	0x80
2	Total Length	Length in bytes of the complete data object or tagged element, least-significant byte first.	0x0000–0xFFFF
2	Fragment Offset	Offset in bytes of this fragment within the complete payload, least-significant byte first.	0x0000–0xFFFF
2	Fragment Length	Length in bytes of the data carried in this fragment, least-significant byte first.	0x0000–0xFFFF
0–n	Card Data	Requested data bytes exactly as returned by the credential. When a tagged element was selected, this sequence contains only the element's BER-TLV contents, excluding the selected tag and length.	0x00–0xFF

2.2 New Error Codes for Enhanced PIV

The following `osdp_NAK` error codes are introduced by the Enhanced PIV profile. They extend the set defined in OSDP 2.2 without reassigning existing values and are available to any command that requires them.

Table 5: Enhanced PIV Error Codes

Code	Meaning
0x0A	Function not supported.
0x0B	Insufficient PD memory to complete the requested operation.
0x0C	Insufficient credential memory or storage space.
0x0D	Security status not satisfied.
0x0E	Data object or tag not found.
0x0F	VCI not established.
0x10	PIV Auto not configured.

2.3 Updated Function Codes

2.3.1 Function Code 13 – Reader Interfaces

OSDP 2.2 defined Function Code 13 as a reader count indicator with a compliance byte fixed at `0x00`. Enhanced PIV-capable PDs **SHALL** continue to populate the reader count field, but they MAY set bits within the compliance byte to advertise the downstream credential technologies they expose. A value of `0x00` remains valid and retains the legacy meaning of "interface presence unspecified."

When a PD reports specific interfaces, it **SHALL** encode the compliance byte using the flags in Table 6. The reader count byte continues to report the number of attached credential interfaces.

Table 6: Function Code 13 Compliance Flags

Mask	Meaning
0x01	PD presents a contact card interface (ISO 7816 or equivalent).
0x02	PD presents a high-frequency contactless credential interface (13.56 MHz NFC).
0x04	PD supports the OSDP 2.3 Enhanced PIV command set (including `osdp_CRAUTH`, `osdp_PIVDATA`, `osdp_CARDSTATUS`, and `osdp_PIVMODE`).
0x08	PD presents a low-frequency credential interface (approximately 125 kHz prox).
0x10	PD presents a Bluetooth credential interface.
0x20	PD presents a barcode credential interface.
0x40	PD presents an ultra-high-frequency credential interface (UHF/RAIN RFID).
0x80	PD supports Virtual Contact Interface (VCI) secure messaging for applicable modes.

All other bits are reserved for future use and **SHALL** be transmitted as zero. PDs **SHALL** set Bit `0x04` when they implement the OSDP 2.3 PIV Mode TLV format and the associated automated flows described in Section 3.6.

2.3.2 Function Code 14 – Biometric Support

Function Code 14 remains backward compatible with OSDP 2.2. A compliance value of 0x00, 0x01, 0x02, or 0x03 retains its original meaning and indicates that the PD does not support the OSDP 2.3 biometric command set. Devices that support the enhanced commands **SHALL** set Bit 7 (0x80). When Bit 7 is asserted, the remaining bits of the compliance byte are interpreted as a capability bitfield. Bits not explicitly defined in Table 7 are reserved and **SHALL** be transmitted as zero.

Table 7: Function Code 14 Compliance Flags

Mask	Meaning
0x80	PD supports the OSDP 2.3 biometric command set.
0x01	PD performs off-card fingerprint comparison using FIPS 201 BIO ANSI INCITS 378 templates.
0x02	PD performs on-card fingerprint comparison using FIPS 201 OCC processes.
0x04	PD performs off-card iris comparison aligned with FIPS 201 BIO guidance.
0x08	PD performs off-card facial comparison using ANSI INCITS 385-2004 images stored on the credential.
0x10	PD supports OSDP 2.3 autonomous PIV operation (PIV Auto).

3 New or Enhanced PIV Commands

This section introduces commands defined exclusively for the OSDP 2.3 Enhanced PIV profile.

3.1 PIV Put Data (osdp_PIVPUTDATA)

The `osdp_PIVPUTDATA` command instructs the PD to deliver an ISO 7816-4 `PUT DATA` operation to the credential. The ACU **SHALL** construct the payload as a complete BER-TLV object, including the desired outer tag (`0x7E` for the Discovery Object, `0x7F61` for a BIT Group Template, `0x5C`/`0x53` sequences for other data objects, and so on). The PD **SHALL** forward the payload to the credential using `CLA = 0x00`, `INS = 0xDB`, `P1 = 0x3F`, and `P2 = 0xFF`. If VCI secure messaging is established, the PD **SHALL** select the secure channel automatically and adjust the `CLA` accordingly. Chained APDUs **SHALL** be generated by the PD when the payload exceeds a single-card APDU.

Table 8: PIV Put Data (osdp_PIVPUTDATA)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	TBA
2	MpSizeTotal	Total size of the complete PUT DATA payload, least-significant byte first.	0x0000–0xFFFF
2	MpOffset	Offset of this fragment within the complete payload, least-significant byte first.	0x0000–0xFFFF
2	MpFragmentSize	Number of payload bytes carried in this fragment, least-significant byte first.	0x0000–0xFFFF
0–n	Data	BER-TLV payload to be written to the credential. This field already includes the outer object tag selected by the ACU.	0x00–0xFF

Responses A successful write is acknowledged with `osdp_ACK`. Failures **SHALL** use `osdp_NAK` with one of the Enhanced PIV error codes defined in Subsection 2.2: `0x0A` when the function is not supported, `0x0B` when the PD lacks buffer space to stage the payload, `0x0C` when the credential has insufficient storage, or `0x0D` when security conditions on the credential are not met.

3.2 Card Status (osdp_CARDSTATUS)

The `osdp_CARDSTATUS` command requests the current credential context from the PD. The command code is to be assigned; PDs **SHALL** advertise support via Function Code 13 bit 2 before controllers attempt

to use this command. The payload selects the status record that should be returned (currently only the standard profile).

Table 9: Card Status Request (osdp_CARDSTATUS)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	TBA
1	Status Type	Requested status record. 0x01 selects the standard PIV card status profile. Values 0x02–0x7F are reserved and shall elicit osdp_NAK. Values 0x80–0xFF are reserved for private use.	0x00–0xFF

PDs **SHALL** return osdp_NAK with an appropriate error code (for example 0x03 unknown command, 0x05 incorrect parameter) if the requested status type is not supported.

3.3 Card Status Reply (osdp_CARDSTATUSR)

In response to osdp_CARDSTATUS, or when the credential state changes after a PIV mode has been selected, the PD reports the detected credential type, VCI state, and optional application data. When the card state changes asynchronously, the PD **MAY** send osdp_CARDSTATUSR as the next osdp_POLL reply following OSDP's standard unsolicited-report rules. ACUs may therefore receive the reply either immediately after issuing osdp_CARDSTATUS or as an unsolicited poll response when a credential is presented or removed.

Table 10: Card Status Reply (osdp_CARDSTATUSR)

Size (bytes)	Name	Meaning	Value
1	CMND	Reply identifier.	TBA
1	Detected Credential Type	Current card interface: 0x00 none, 0x01 ISO 7816 contact, 0x02 ISO 14443 contactless, 0x03 other ISO 7816 credential (for example DESFire, Seos), 0x04 low-frequency credential, 0x05 unknown type, 0x06–0x7F reserved, 0x80–0xFF private use.	0x00–0xFF
1	VCI Status	0x00 VCI not established, 0x01 VCI established. Values 0x02–0x7F reserved, 0x80–0xFF private use.	0x00–0xFF
1	VCI Trust Anchor ID	Identifier of the trust anchor used to establish VCI. 0x00 indicates no anchor in use.	0x00–0xFF
2	PIN Usage Policy	PIN Usage Policy bytes (tag 0x5F2F) returned verbatim when available. A value of 0xFFFF indicates the policy is not present or has not yet been read.	0x0000–0xFFFF
1	Selected AID Length	Length (0–16) of the selected AID. 0x00 indicates no application is currently selected.	0x00–0x10
0–16	Selected AID	Application Identifier read from the credential (for example from the FCI template). Present only when the length is non-zero.	0x00–0xFF

A PD that cannot provide the requested status **SHALL** return `osdp_NAK` and omit the reply. Successful replies include the entire selected AID (when available) and any PIN Usage Policy bytes read from the credential's Discovery Object (tag 0x5F2F). When no PIV PIN Usage Policy is present, the PD **SHALL** report a length of zero and omit the bytes. The VCI fields describe whether a secure channel is currently established, which trust anchor enabled the session, and whether the presented credential requires a pairing code for VCI operation.

3.4 TWIC Privacy Key Load (osdp_PIV_LOADTPK)

The osdp_PIV_LOADTPK command instructs the PD to cache a TWIC Privacy Key (TPK) for use with subsequent biometric operations against a TWIC credential. The command code is to be assigned. PDs **SHALL** advertise support for TWIC biometric processing via Function Code 14 before controllers attempt to use this command.

Table 11: TWIC Privacy Key Load (osdp_PIV_LOADTPK)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	TBA
1	Cache Timeout	Number of seconds to retain the cached TPK. 0x00 clears any cached key immediately; 0x01–0xFF permit caching for up to 255 seconds.	0x00–0xFF
1	Control Flags	Bit 0 set requests automatic clearing when the credential is removed; all other bits **SHALL** be zero.	0x00–0xFF
32	TPK	Trusted Processing Key bytes to cache for the next TWIC biometric operation.	0x00–0xFF

Controllers set Cache Timeout to define how long (in seconds) the PD may retain the key. A value of 0x00 clears any cached TPK immediately. Values 0x01–0xFF allow caching for up to 255 seconds. Bit 0 of Control Flags directs the PD to clear the cached TPK when the credential is removed; all other bits are reserved and **SHALL** be transmitted as zero. If the PD does not support TWIC TPK caching it **SHALL** return osdp_NAK 0x03. Invalid payloads (for example reserved bits set) **SHALL** elicit osdp_NAK 0x05.

3.5 PIV VCI Pairing Code Transmit (osdp_PIV_XMITPAIRING)

The osdp_PIV_XMITPAIRING command delivers an eight-digit pairing code to the PD for use during Virtual Contact Interface (VCI) establishment. The command code is to be assigned. Controllers **SHALL NOT** issue this command unless a PIV application is currently selected and VCI mode is active.

Table 12: PIV VCI Pairing Code Transmit (osdp_PIV_XMITPAIRING)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	TBA
8	Pairing Code	Eight ASCII digits (`'0'–'9'`) representing the VCI pairing code associated with the currently selected PIV application.	0x30–0x39

The pairing code is encoded as eight ASCII digits (`'0'–'9'`). A PD that has not yet established VCI **SHALL** return `osdp_NAK 0x0F` (VCI not established). If no PIV application is selected, the PD **SHALL** return `osdp_NAK 0x05`. PDs **SHALL** clear the stored pairing code when VCI is torn down or when the active PIV mode changes.

3.6 Set PIV Mode (osdp_PIVMODE)

The `osdp_PIVMODE` command defines the operating context for PIV, TWIC, and related credential transactions. OSDP 2.3 replaces the legacy "Application + AID" fields with a TLV list of configuration descriptors so controllers can specify multiple application profiles, preferred interfaces, and optional Application Identifiers. The command code remains to be assigned. PDs **SHALL** advertise support via Function Code 13 bit `0x04`.

Table 13: Set PIV Mode (osdp_PIVMODE)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	TBA
1	PIV Mode Flags	Global behaviour flags (Section 3.6).	0x00–0xFF
2	MpSizeTotal	Total size of the complete configuration payload, least-significant byte first.	0x0000–0xFFFF
2	MpOffset	Offset of this fragment within the configuration payload, least-significant byte first.	0x0000–0xFFFF
2	MpFragmentSize	Length of the fragment payload, least-significant byte first.	0x0000–0xFFFF
1	Config Count	Number of configuration entries (1–32).	0x01–0x20
0–n	TLV Data	Sequence of configuration descriptors: optional Global Interface Mask TLV (Tag `0x10`) followed by one or more Configuration Entry TLVs (Tag `0x01`).	–

The PIV Mode Flags byte enables global behaviours such as autonomous operation and VCI usage (Table 14). Bits not listed remain reserved and **SHALL** be transmitted as zero.

Table 14: PIV Mode Flags

Mask	Meaning
0x01	Enable autonomous PIV (PIV Auto). The PD interrogates credentials, establishes VCI when possible, and advances through the authentication flow without explicit ACU prompts.
0x02	Permit VCI usage when available.
0x04	Enforce FICAM strict mode (PIV applications only).

After the flags, the payload uses standard multi-part framing fields and a configuration count (1–32). Each configuration appears as a TLV:

- **Global Interface Mask (Tag `0x10`, optional)** — One-byte mask that limits the interfaces the PD may use for all configurations. Bit assignments follow Function Code 13 (bit 0 contact, bit 1 contactless, bit 2 low-frequency, bit 3 Bluetooth, bit 4 barcode, bit 5 UHF/RAIN). A value of `0x00` indicates no global restriction.

- **Configuration Entry (Tag 0x01)** — Contains the data-model identifier, a per-entry interface mask, the AID length, and the optional AID:
  - Data Model (0x01 PIV, 0x02 TWIC, values 0x04–0x7F reserved, 0x80–0xFF private use).
  - Interface Mask — Uses the same bit positions as the global mask; 0x00 means "any interface permitted by the global mask."
  - AID Length (0–16) and AID bytes when provided.

PDs process configuration entries in order, applying the global mask (when present) before honouring each entry's mask. Invalid masks, unknown data-model identifiers, or malformed TLVs **SHALL** result in osdp_NAK 0x05. Controllers **SHALL** send the command over a secure channel where required; the PD **SHALL** return osdp_NAK 0x06 if the necessary security conditions are not satisfied. Multi-part framing errors result in osdp_NAK 0x07. If the PD cannot cache the requested configuration set, it **SHALL** respond with osdp_NAK 0x0B.

When PIV Auto is enabled, PDs interrogate the credential on presentation, establish VCI automatically when possible, and advance through authentication without explicit ACU prompts. If a pairing code is required, the PD signals the state change via osdp_CARDSTATUSR; the ACU then provides the code using osdp_PIV_XMITPAIRING. Commands that rely on PIV Auto **SHALL** return osdp_NAK 0x10 when the feature has not been configured.

3.7 VCI Trust Anchor Load (osdp_PIV_VCILOADTA)

The osdp_PIV_VCILOADTA command loads, replaces, or deletes a Virtual Contact Interface (VCI) trust anchor. Trust anchors must be cached before a PD can verify card certificates during VCI establishment. The command code is to be assigned.

Table 15: VCI Trust Anchor Load (osdp_PIV_VCILOADTA)

Size (bytes)	Name	Meaning	Value
1	CMND	Command identifier.	TBA
1	Operation	0x00 load/replace anchor; 0x01 delete anchor; 0x02 query anchor inventory.	0x00–0x02
1	Trust Anchor ID	Controller-assigned identifier for the anchor (1–128).	0x00–0xFF
8	Issuer Identifier Number	Leftmost eight bytes of the subjectKeyIdentifier from the issuing content-signing certificate.	0x00–0xFF
2	Anchor Length	Total size of the anchor payload, least-significant byte first. Set to 0x0000 when deleting or querying inventory.	0x0000–0xFFFF
2	MpSizeTotal	Total size of the complete anchor data, least-significant byte first. Set to 0x0000 when deleting or querying inventory.	0x0000–0xFFFF
2	MpOffset	Offset of this fragment within the anchor data, least-significant byte first. Set to 0x0000 when deleting or querying inventory.	0x0000–0xFFFF
2	MpFragmentSize	Length of the data block in this fragment, least-significant byte first. Set to 0x0000 when deleting or querying inventory.	0x0000–0xFFFF
0–n	Anchor Data	Anchor bytes (DER or equivalent) present when loading. Omitted for delete and query operations. Subsequent fragments (load only) contain only this field.	0x00–0xFF

The metadata block (operation, identifier, IIN, anchor length) appears only in the first fragment where `MpOffset = 0`; subsequent fragments carry additional anchor bytes. Controllers set Operation to 0x00 to load or replace the anchor identified by Trust Anchor ID. Set Operation to 0x01 with Anchor Length 0x0000 to delete the anchor. The Issuer Identifier Number (IIN) is the leftmost eight bytes of the subjectKeyIdentifier in the content signing certificate used to protect the credential. The Anchor Length declares the total size of the anchor payload; controllers shall send the exact number of bytes in one or more fragments.

PDs **SHALL** reconstruct the anchor from the fragments before storing it. A successful operation returns `osdp_ACK`. Failures return `osdp_NAK` with:

- `0x03` when the PD does not support trust-anchor caching.
- `0x05` for malformed metadata (unknown operation, reserved bits, invalid lengths).
- `0x07` when multi-part sequencing fails.
- `0x0B` when the PD lacks storage space for the anchor.

PDs are not required to maintain a real-time clock; they need only verify that credential certificates presented later fall within the trust anchor's validity period. Implementers may perform stricter date checks if desired. PDs may retain either the full anchor blob or a compact record containing the public key, issuer identifier, validity period, and anchor number so long as the supplied IIN remains available for matching.

Trust anchors may be stored in volatile or non-volatile memory at the PD's discretion. Controllers should not assume retention beyond the device's documented behaviour.

3.8 VCI Trust Anchor Status (osdp_PIV_VCITASTATUS)

The `osdp_PIV_VCITASTATUS` command enumerates cached trust anchors and reports remaining storage capacity. The command code is to be assigned. The request carries no payload. The reply uses the standard multi-part fields (`TOTAL`, `OFFSET`, `DATA_LEN`) so PDs can stream the inventory across one or more fragments.

Table 16: VCI Trust Anchor Status Reply (osdp_PIV_VCITASTATUSR)

Size (bytes)	Name	Meaning	Value
1	CMND	Reply identifier.	TBA
2	TOTAL	Length of the complete status payload, least-significant byte first.	0x0000–0xFFFF
2	OFFSET	Offset of this fragment within the status payload, least-significant byte first.	0x0000–0xFFFF
2	DATA_LEN	Length of the data block in this fragment, least-significant byte first.	0x0000–0xFFFF
1	Total Anchor Count	Number of cached trust anchors.	0x00–0xFF
2	Available Bytes	Remaining storage capacity for trust anchors, least-significant byte first.	0x0000–0xFFFF
0–n	Anchor Records	Zero or more records. Each record contains: `Trust Anchor ID` (1 byte), `Issuer Identifier Number` (8 bytes), `Anchor Length` (2 bytes LSB first), `Attributes` (1 byte, reserved). Records that span fragments continue in subsequent replies.	–

The PD first reports the total number of anchors and the remaining free bytes. Each record thereafter lists a trust anchor by ID, IIN, anchor length, and reserved attributes (transmitted as zero). Controllers can use this information to decide whether to load additional anchors or delete existing ones. NAK responses follow the standard pattern: `0x03` for unsupported command, `0x07` for multi-part errors.

Table 17: Trust Anchor Record Structure

Field	Size	Meaning	Value Range
Trust Anchor ID	1 byte	Identifier assigned by the ACU; reused to replace or delete an anchor.	0x00–0xFF
Issuer Identifier Number (IIN)	8 bytes	Leftmost eight bytes of the subjectKeyIdentifier from the issuing content-signing certificate.	0x00–0xFF
Anchor Length	2 bytes (LSB first)	Total size of the stored anchor or compact representation.	0x0000–0xFFFF
Attributes	1 byte	Reserved for future use; transmitted as zero.	0x00

A References

- Security Industry Association, Open Supervised Device Protocol (OSDP) v2.2, 2020.
- National Institute of Standards and Technology, NIST Special Publication 800-73-4, Interfaces for Personal Identity Verification (PIV) — Part 2: PIV Card Application Card Command Interface, February 2015.
- National Institute of Standards and Technology, Federal Information Processing Standards Publication 201-3, Personal Identity Verification (PIV) of Federal Employees and Contractors, January 2022.
- U.S. General Services Administration, Federal Identity, Credential, and Access Management (FICAM) Architecture, Revision 3, 2023.
- Open Physical, High Speed Autonomous PIV: Concept Paper, 2024.

B Sample osdp_PIVDATA Requests (Informative)

This appendix illustrates representative requests that conforming ACUs may issue using the clarified osdp_PIVDATA command. Object identifiers align with NIST SP 800-73-4 Part 2, Table 6. Each example shows the bytes that appear in the OSDP payload after the CMND field; the PD strips leading zeroes before constructing the ISO 7816-4 APDU.

Cardholder Unique Identifier (CHUID) Object identifier 0x5FC102, tag 0x00, offset 0x0000. The PD transmits GET DATA for tag 0x5FC102 and returns the complete CHUID BER-TLV so the ACU can validate the FASC-N, GUID, and expiration fields.

Card Capability Container (CCC) Object identifier 0x5FC107, tag 0x00, offset 0x0000. Use this request during device discovery to read feature flags (contact/contactless support, optional biometrics) prior to issuing PIV Mode updates.

Discovery Object Object identifier 0x00007E, tag 0x00, offset 0x0000. The PD forwards tag 0x7E with its nested data so the ACU can parse the application identifier (tag 0x4F) and PIN Usage Policy (tag 0x5F2F). To retrieve only the AID, repeat the command with tag 0x4F.

Biometric Information Templates (BIT) Group Template Object identifier 0x007F61, tag 0x00, offset 0x0000. Large responses for fingerprint or iris templates require multiple osdp_PIVDATAR fragments; the TOTAL/OFFSET/DATA_LEN fields support reconstruction.

Selective Tag Retrieval Object identifier 0x00007E, tag 0x4F, offset 0x0000. The PD returns only the PIV Application Identifier nested inside the Discovery Object. Tags that occupy more than one byte (for example 0x7F61) are outside the scope of this selector and must be obtained by requesting tag 0x00.