

Technical Reference

Technical description of XPOS ESDC provided as FRCS accreditation evidence. Sections (S1-S17) are cited by the compliance response.

- [Architecture & Secure Element \(S1-S2\)](#)
- [Authentication & TaxCore \(S3-S4\)](#)
- [Identity, Tax & Invoice Processing \(S5-S7\)](#)
- [Digital Signatures & Verification \(S8\)](#)
- [Audit - Format, Remote, Local, Cadence \(S9-S12\)](#)
- [Clock, Logging & Persistence \(S13-S15\)](#)
- [Error Codes & Security \(S16-S17\)](#)

Architecture & Secure Element (S1-S2)

§1 — System architecture & components

XPOS ESDC is an application-based E-SDC for Windows 10/11 (x64). It is a single self-contained package that runs a background fiscal service and an operator UI in one process, so the UI and the POS-facing service share one Secure Element access lock, PIN cache and data store.

Component	Role
Operator UI (tray application)	Status, built-in POS, reports, audit controls, About (Manufacturer / Serial / Software version / Model / SE applet version).
POS-to-SDC service (<code>localhost:8888</code>)	Receives invoice requests from the built-in or a third-party POS (v3, legacy V2 supported).
Secure Element access layer	PC/SC (APDU) to the SE applet; PKCS#11 to the PKI applet. Serialised behind a single card lock.
Audit & command services	Background services for remote audit, proof-of-audit cadence, online-status and command processing.
Local store	Embedded database for invoice records and encrypted audit packages; plain-text log folder.
Token helper	Small native helper performing the mutual-TLS token request over the card key (see §3).

Program files install under Program Files (read-only at runtime); working data (records, logs, settings) is kept in the per-user application-data folder. All configuration originates from the Secure Element at runtime.

§2 — Secure Element interface

All fiscal signing and all fiscal counters are performed and held on the Secure Element (SE); XPOS ESDC never computes or alters fiscal signatures. The SE is reached through an external USB PC/SC reader using ISO 7816 APDUs.

- **PIN verification** — a Verify APDU unlocks signing. The PIN is supplied by the operator and held only in memory (§17).
- **Sign** — the invoice request (date/time, amount, counters, tax data) is sent to the SE, which returns the signature, encrypted internal data and updated counters.
- **Amount / limit** — the SE accumulates a signed amount and reports when an audit is required; XPOS ESDC reads this for status and audit triggering.
- **Start / End Audit** — Start Audit produces the audit request (ARP); End Audit applies a proof-of-audit and resets the held amount.

Signing does not require connectivity; an invoice can be fully fiscalised offline.

Authentication & TaxCore (S3-S4)

§3 — Authentication & token acquisition

Access to TaxCore.API requires a bearer token obtained by authenticating with the on-card **PKI certificate**. XPOS ESDC performs a mutual-TLS request to `/api/v3/sdc/token` in which the client certificate and private key are the SE's PKI key, exposed through **OpenSC PKCS#11**. The private key never leaves the card — the TLS handshake's client-certificate signature is computed on the SE.

TLS renegotiation

The FRCS token endpoint requests the client certificate via **TLS renegotiation** (the initial handshake asks for no certificate; the server then issues a HelloRequest to demand the card certificate). XPOS ESDC performs this exchange over TLS 1.2 with client-initiated renegotiation supported, using the card key via PKCS#11 for the certificate-verify signature (PKCS#1 v1.5, SHA-256). The server certificate is pinned on first use.

The returned token is cached and refreshed before expiry, and presented as the `TaxCoreAuthenticationToken` header on all subsequent data calls (§4).

§4 — TaxCore.API & command processing

All communication with TaxCore uses the token from §3. XPOS ESDC receives and executes the full command set, in the order received, and reports the result of each command back to TaxCore.

Command	Action
SetTaxRates	Install/replace tax-rate groups, including future-dated groups (§6).
SetTimeServerUrl	Set the NTP server used for time synchronisation (§13).
SetVerificationUrl	Set the base verification URL used on receipts (§8).
SetTaxCoreConfiguration	Apply TaxCore configuration parameters.

Command	Action
ForwardProofOfAudit	Apply a proof-of-audit to the SE via End Audit (§9, §12).
ForwardSecureElementDirective	Relay a directive to the SE.

Commands are processed consecutively (first to last); each result is reported so TaxCore has an accurate record of execution.

Identity, Tax & Invoice Processing (S5-S7)

§5 — Identity from the Secure Element

The taxpayer identity is read directly from the SE certificate — nothing is entered by the operator.

Field	Source in the certificate
TIN	Certificate OID (tag-6)
UID	SERIALNUMBER (subject)
Company name	Subject 0
Store / business unit	Subject OU
Address	Subject STREET
District	Subject S
TaxCore endpoint	Certificate extension (tag-5) — the environment URL

§6 — Tax calculation & rate management

Tax is calculated per item label against the active tax-rate group. Each tax category is applied by its type:

- **Tax on net** — added to the net amount.
- **Tax on total** — included within the price.
- **Per-quantity** — a fixed amount per unit.

Amounts are computed to **4 decimal places with half-round-up** and presented to 2 decimals on the response and journal. Invoices carrying a label outside the active group are rejected at validation (§17).

Effective-dated groups

Tax-rate groups are installed by `SetTaxRates` (online) and can also be applied from the local-audit file. Future-dated groups take effect from their effective date; where dates coincide, the higher revision applies. The group in force at the invoice's referent date is used (current date for a normal sale; the original date for a copy/refund).

§7 — Invoice processing pipeline

1. **Receive** — an invoice request arrives on the POS-to-SDC service as JSON.
2. **Validate** — structure, labels and amounts are checked; invalid requests are rejected with the proper error code (§16).
3. **Calculate** — taxes are computed on the active rates for the referent date (§6).
4. **Certificate check** — a proactive validity check ensures the SE certificate is valid for the current date before signing.
5. **Sign** — the request, SDC date/time and cached PIN are sent to the SE, which returns the signature, encrypted internal data and counters (§2).
6. **Journal** — a 40-column textual receipt is produced with the verification URL and QR code (§8).
7. **Store** — the encrypted audit package is committed to non-volatile storage *before* the response is returned (§9).
8. **Respond** — the fiscal result is returned to the POS.

Auditing runs on a background service that does not hold the card lock, so a POS sign is never queued behind a network operation.

Digital Signatures & Verification (S8)

§8 — Digital signature & verification URL

Every invoice type — Normal, Advance, Copy, Training, Proforma — is signed on the SE and produces a unique verification URL encoded into the receipt QR code. The URL payload is assembled to the FRCS specification and comprises:

- Format version;
- Secure Element UID and requesting-signer identifiers;
- Invoice and total counters;
- Total amount and timestamp;
- Encrypted internal data;
- The SE signature; and
- An integrity digest (MD5) over the payload.

The signature allows any party to verify the invoice's integrity and authenticity via the verification URL / FRCS portal. The exact byte layout and QR encoding conform to the FRCS E-SDC specification and are validated by the SDC Analyzer.

Signing mechanism

The invoice is not signed by XPOS ESDC itself — it is signed by the Secure Element. XPOS ESDC builds the signing payload (including the trusted SDC clock's timestamp) and hands it to the SE over PKCS#11; the SE's private key never leaves the card, and the signature it returns is what gets embedded in the verification URL. This is what makes the signature genuine cryptographic authenticity rather than a simple checksum: only the specific Secure Element holding that private key could have produced a signature that validates.

Because the FRCS token endpoint requires the client certificate via TLS renegotiation, which managed .NET TLS stacks cannot perform, this handshake and the signing call are carried out by a small bundled native helper (using the card key over PKCS#11/OpenSC) rather than XPOS ESDC's own managed code directly.

Verification

Anyone with the verification URL — scanned from the receipt QR code or read from the invoice — can independently confirm the invoice was genuinely signed by that Secure Element and has not been altered since, without needing access to XPOS ESDC, the Secure Element, or Defy Technologies' own systems.

Audit - Format, Remote, Local, Cadence (S9-S12)

§9 — Audit package format & storage

Each signed invoice yields an audit package containing the invoice `{ request, result }`. The package is encrypted as follows:

- **Payload** — encrypted with **AES-256 (CBC, PKCS7 padding)** using a one-time key and IV.
- **Key transport** — the one-time AES key and IV are **RSA-encrypted with the TaxCore public key**.
- **Package** — `{ key, iv, payload }`, identical in form for both remote and local audit.

Packages are written to non-volatile storage before the POS response returns, and are **never overwritten or erased without a proof-of-audit**: a package is retained until it is verified (accepted) or cleared by a valid POA. Applying a POA to the SE (End Audit) resets the held amount, after which the corresponding packages may be cleared. Packages are keyed by SE UID, so switching cards does not interrupt an in-flight audit.

§10 — Remote audit protocol

When connectivity is available, a background auditor submits audit packages to TaxCore continuously and flushes any previously unsent packages. The protocol uses the specified endpoints:

Operation	Purpose
Submit audit	Upload encrypted audit packages.
Audit proof	Receive the proof-of-audit for application to the SE.
Online status	Periodic heartbeat while a valid token is held.
Commands	Retrieve and acknowledge TaxCore commands (§4).

`auditRequired` is surfaced in Get Status when the SE nears or reaches its limit. Submission continues automatically whenever data and internet are available; this has been demonstrated live clearing a full SE (≈12M) to zero.

§11 — Local audit (removable media)

For prolonged offline operation, audit data is carried on USB/SD media using the same package format as remote audit.

Export

- A sub-folder named by the SE UID is created on the media if absent: `{root}\{UID}\`.
- The audit request produced by Start Audit is written as `{UID}.arp`.
- Each audit package is written as an individual JSON file in the specified `{UID}-{UID}-{n}.json` convention.
- Started / in-progress / completed status is shown in the Local Audit panel.

Import (proof-of-audit)

- The commands file `{UID}.commands` is read from the media and executed.
- The proof-of-audit is applied to the SE via End Audit.
- Results are written back as `{UID}.results`.

§12 — Proof-of-Audit cadence

A shared cadence guard governs both the manual and background audit paths so that only one Audit Start is outstanding at a time. This prevents a second Start from invalidating an earlier, still-pending proof.

- **Minimum between Starts:** 10 minutes (well above the 5-minute minimum).
- **Normal cadence:** 30 minutes; tightened as the SE approaches its limit.
- A proof-of-audit is applied to the SE as soon as it is received; memory is cleared only after the POA resets the held amount.

Clock, Logging & Persistence (S13-S15)

§13 — Real-time clock & time synchronisation

As a software E-SDC there is no dedicated hardware RTC. Time accuracy is maintained by **NTP synchronisation** against the server provided by FRCS (set via `SetTimeServerUrl`), performed **hourly** — well within the 48-hour requirement. Between syncs the host system clock is the time source. This is the documented alternative to a hardware RTC for an application-based E-SDC.

§14 — Logging

XPOS ESDC records the required error events: invoice, APDU, TaxCore, internal, initialization, fiscalization, audit and time-synchronisation errors. Log characteristics:

- **Chronological**, timestamped in local time to the second.
- **Human-readable** plain-text daily files, exportable from the UI ("Export logs").
- **Rolling** — daily files retained for the last 30 days; older files removed automatically.
- **Isolated** — logs live in a separate folder and do not affect invoice/package storage.

§15 — Persistence & offline capacity

Invoice records and encrypted audit packages are stored in a local embedded database that survives power loss and restart (non-volatile). No power is required to preserve fiscal data.

Offline capacity: an audit package is approximately 3–5 KB, giving at least **~100,000 unsent invoices per GB** of free storage. The device therefore sustains extended offline operation, reporting automatically once connectivity returns.

Error Codes & Security (S16-S17)

§16 — Error codes

XPOS ESDC returns only the prescribed FRCS error codes; any manufacturer-specific message is listed in the User Manual. Representative mappings:

Code	Meaning
1300	No card / Secure Element not present
1500	PIN required
2100 / 2110	PIN incorrect / attempts remaining
2210	Secure Element limit reached — audit required
6308	Certificate not valid for the current date

§17 — Security & prohibited functions

- **PIN handling** — the PIN is held only in working memory and is cleared on restart; it must be re-entered after any restart.
- **Fixed protocol** — the POS-to-SDC contract and its parameters are not user-editable.
- **Label enforcement** — invoices with labels outside the active tax group are rejected.
- **Prescribed responses only** — no non-standard error responses are emitted.
- **Fiscal integrity** — signing and counters are performed solely on the SE; XPOS ESDC cannot alter fiscal data or reset counters without a valid proof-of-audit.