ENGINEERING SPECIFICATION

version 10.0

INTERNET IDENTITY CARD ™

Provenance, implementation & verification method

Issuer	Https Card — Internet Identity Card Ltd
Company Registration	UK Companies House No. 09168431
Trademark	UK IPO UK00003166480 (2016) · INPI FR 4071419
US Copyright	TX 8-508-373 (2014/2017)
Office	124 City Road, London EC1V 2NX, United Kingdom
Disclosures	TDCommons #10079 · #10167 · #10394 · #10795 · #10796 · #11121 (CC BY 4.0)

This document supersedes the v9.0 edition. It records how the v10.0 corrections were implemented and, unusually for this series, how they were verified and where that verification failed. The second point is included because it changed what the release considers adequate evidence.

IIC v10.0 is a corrective release. It introduces no new capability: it repairs six defects found in the v9.0 implementation, four of which were reachable in normal use. Section references to v9.0 behaviour are retained where a reader of the previous edition would otherwise be misled.

1. Provenance & priority

The Internet Identity Card has been developed since 2013 by Https Card — Internet Identity Card Ltd (UK Companies House 09168431). Priority is evidenced by US Copyright TX 8-508-373 (2014/2017), UK trademark UK00003166480 (2016), INPI FR 4071419, and six defensive publications lodged with the Technical Disclosure Commons under CC BY 4.0: #10079, #10167, #10394, #10795, #10796 and #11121. Release digests are anchored on Bitcoin and Ethereum, giving authority-free proof of existence at a date.

2. Scope of the v10.0 release

v10.0 is corrective. No capability is added. Six defects in the v9.0 implementation are repaired:

Ref	Correction
D1	The vault encryption key is no longer stored. It is wrapped in two Argon2id envelopes and exists in memory for the session only.
D2	The salt-less fallback in memorable-word verification upgrades the stored hash on success, so the compatibility path removes itself instead of persisting.
D3	The signature sequence number comes from a persistent counter rather than the array length.
D4	Truncating the chain records an anchor, so the retained entries still verify to a known point.
D5	Signing-key generation on upgrade is guarded; an existing key is never replaced.
D6	A configured hybrid signature that cannot be produced causes refusal, not a classical receipt.

3. Boot sequence and the unlock gate

The vault load path is the largest structural change. In v9.0 the state was decrypted with a key read from storage, so loading was effectively synchronous. In v10.0 the key must be derived from a secret the holder supplies, so the boot sequence awaits user input before the interface is armed.

```
_a1()
  installBoot(AK)
  if no stored state            -> return          // fresh install
  if envelope record usable     -> prompt, unwrap, decrypt
  else                          -> load by the v9 path, then force migration
  normalise state fields

_a2()
  key = session key
  if absent and registered      -> provision the vault (prompt, phrase, confirm)
  if still absent               -> refuse to write
```

Two consequences were verified rather than assumed. The loader is dismissed before any dialog is shown, or it would cover it. And the error path in the previous loader, which swallowed every exception into an empty state, was removed: a failed unlock must not present as an absent vault, which would read to the holder as data loss.

4. Migration state machine

Migration is forced at first open of a v9.0 vault, with no defer option: a key stored beside its ciphertext protects nothing, and an indefinite postponement would leave it that way. The ordering of writes carries the safety property.

Step	Action	Recoverable if interrupted
1	Write the envelope record with confirmed = false. Leave the unencrypted key in place.	Yes — both paths still open the vault.
2	Display the phrase, require transcription, then prove that each envelope returns the original key.	Yes — nothing has been removed.
3	Set confirmed = true, persist, then delete the unencrypted key.	Complete.

A record found unconfirmed at next open is discarded and the migration restarts, because an unconfirmed record is one whose envelopes were never proven.

5. Signature chain implementation

- `sigSeq` is persisted with the vault and never derived from the array.
- Truncation writes `sigAnchor = { seq, hash, droppedTotal }` from the last dropped entry.
- Rotations are archived in `sigRotations` and are also truncated from the linear chain. An earlier design exempted them, which was wrong: a retained entry whose sequence number precedes the anchor breaks the linearity the anchor exists to preserve. Separating the two structures resolves it.
- The attestation message accepts EC P-256 keys only and encodes field lengths. An earlier version interpolated JWK fields for any key type, so every RSA key produced an identical message and bound to nothing. A canonical form must be injective over every input it accepts, or refuse the input.

6. Hybrid signing failure semantics

The v9.0 specification stated that card creation never fails if the post-quantum module is absent. The implementation delivered that by suppressing every error in the hybrid path, which made an absent module and a broken one indistinguishable in the output. v10.0 keeps the principle and separates the cases: absence remains a mode and yields a classical receipt; a configured suite that fails yields no receipt at all, with the cause reported. There is deliberately no configuration option that restores the previous behaviour.

7. Standards compliance

- FIPS 204 — ML-DSA-65, post-quantum signature component.
- FIPS 186-5 — ECDSA P-256, classical signature component.
- RFC 9106 — Argon2id, for card, backup and vault key derivation (96 MiB, t=4, p=4).
- NIST SP 800-38D — AES-256-GCM authenticated encryption.
- ANSSI transition guidance — hybrid classical plus post-quantum construction during the transition period.

Conformance to a standard in code is distinct from formal evaluation. Nothing in this release constitutes a FIPS 140 CMVP validation or an ANSSI qualification.

8. Verification method, and what it missed

This section is unusual for a specification and is included because it bears on how the remaining work should be checked.

The six defects above were found by reading the source. Three were independently confirmed by two separate automated reviewers working blind to each other, and those reviewers also found defects the author had not: a frozen sequence number, an orphaned truncation link, and a classifier that failed open. The method has demonstrable value.

It also has a bounded reach. Four further defects appeared only when the build was executed, all of them in the paths that write a vault: an unreachable branch guarded on a field that is never undefined; a deletion that removed two storage keys of three; an enrolment path that never created a vault record; and a save that was not awaited, so the dialog that followed closed the prompt before it was seen. Each is a property of a sequence of calls, not of any function read in isolation. Reviewers reading functions did not and structurally could not find them.

The practical conclusion, applied to this release: verification is organised around journeys — enrol, reload, unlock, sign, delete, re-enrol, restore — rather than around code regions, and a correction is not considered complete until its journey has been executed end to end from a `file://` URL with the network disabled.

9. Change history

Version	Engineering change
v8.4.1	Multi-zone canonical sealing; dual-passphrase architecture; TOTP-derived material.
v9.0	Hybrid ECDSA P-256 + ML-DSA-65 signing; crypto-agile suite registry; embedded offline verification engine; deterministic modular build.

v10.0	Vault key protection by dual envelope; Argon2id for vault derivation with per-envelope labelling; corrected signature chain counter and truncation anchor; guarded key generation on upgrade; refusal on hybrid signing failure; self-eliminating legacy hash path.

10. Issuer information

Https Card — Internet Identity Card Ltd, 124 City Road, London EC1V 2NX, United Kingdom. UK Companies House 09168431. ICO registration ZA457585. CNIL 1726245. D&B DUNS 220301971.

11. Disclaimer

This document is provided for informational purposes only. It describes intended behaviour and properties under the assumptions stated in it; where those assumptions do not hold, the described protections may not apply. No software system can be guaranteed to be free of defects or completely secure. Conformance to a cryptographic standard in code is distinct from formal evaluation: nothing here constitutes a FIPS 140 CMVP validation, an ANSSI qualification, or a certification of any kind, each of which requires assessment by an accredited third party. Internet Identity Card — Prove and Protect Your Online Identity ™ is a trademark of Https Card — Internet Identity Card Ltd, registered with the UK Intellectual Property Office (UK00003166480).