Supplement S3: Formal model, proof sketches and model checking

to Securing the Semantic Supply Chain: Meaning as a Foundation of Trust Infrastructure

Supplement to the research article

Dr. Carsten Stöcker
Spherity GmbH
carsten.stoecker@spherity.com
ORCID 0009-0003-7417-8872

Version 1.0, 16 November 2026 doi:10.5281/zenodo.24293613



Except where otherwise noted, this work is licensed under the Creative Commons Attribution 4.0 International License (CC BY 4.0). © 2026 Dr. Carsten Stöcker.

Contents

S3.1 The model (SSC-FM v5.2) **3**

S3.1.1 Basic objects . . . 3
S3.1.2 Meaning layers . . . 3
S3.1.3 Typed digests . . . 4
S3.1.4 Canonicalisation . . . 4
S3.1.5 Decision context . . . 5
S3.1.6 Actors . . . 5
S3.1.7 Adversary classes . . . 5

S3.2 The objectives as predicates **6**

S3.3 Theorems and proofs **7**

T1 — Signed is not meant (F1) . . . 7
T2 — `ecdsa-lfsc-2026`: agreement with the issuer's computation, or with its signed commitment (M17) . . . 7
T3 — Authenticate-first cost (M17, M18) . . . 8
T4 — Freshness and availability cannot both hold under withholding, and pins give neither . . . 9
T5 — Freshness bound anchored on the status assertion (M8) . . . 9

T6 — Rollback and permanence of withdrawal (M8, M14) . . . 10
T7 — Fork exclusion and conflict detection (M9) . . . 10
T8 — Deterministic verdicts (M19) . . . 11
T9 — Blank nodes, admission and lists (M18, F5) . . . 11
T10 — Codec binding (M15) . . . 12
T11 — Execution-bound attestation of decisions (M24) . . . 13
T11′ — Replay-validated receipts with a threshold key (alternative to isolation) . . . 15
T12 — Withdrawal as an authority-epoch event (M25) . . . 18
T13 — Extension downgrade safety (M21) . . . 18
T14 — Replay requires reconstructible decision-relevant state (M22) . . . 19
T15 — Certification time, not event time (M27) . . . 19
T16 — Parser agreement (L1) . . . 19
T17 — Scoped publisher authority (OBJ-1) . . . 19
T18 — Non-interference of resolution traffic (OBJ-9) . . . 20
T19 — Disclosure, sufficiency and the cost of assurance (M31) . . . 20
S3.3.20 Plan for mechanised verification . . . 25

S3.4 Model checking **25**

S3.5 Experiments against harness v0.3.0 (X1–X14) **27**

S3.6 Traceability **27**

This supplement belongs to the article *Securing the Semantic Supply Chain: Meaning as a Foundation of Trust Infrastructure* (C. Stöcker, 2026).

This supplement gives the formal model (SSC-FM v5.2) behind the article, §5. Proofs are sketches at the level of a security argument, with explicit premises; the cryptographic steps are standard reductions. The model checker (`model/mc.mjs`, 66 cases, strict `--check` with non-vacuity checks and the declared safe-but-unavailable `MC6.offline.deferEffect` exception) and the experiments are released with harness v0.4.0. Bounded model checking covers the instances checked, not every size; the general statements are the theorems.

Identifiers of the form R2-nn, R3-nn and R4-nn name review findings; S5 records them.

S3.1 The model (SSC-FM v5.2)

S3.1.1 Basic objects

Symbol	Meaning
$\mathbb{B}$	byte strings
$\mathbb{J}$	parsed JSON values under the profile's **I-JSON/JCS value domain**: UTF-8 text; no lone surrogates or noncharacters; unique member names; numbers parsed as finite IEEE 754 binary64 with round-to-nearest; -0 normalised to 0. RFC 7493 only says numbers SHOULD NOT exceed binary64 range and precision, with $\pm(2\hat{}53 - 1)$ the interoperable integer range. Requiring integers outside that range, and exact decimal quantities, to be carried as strings or typed values is a **profile rule**, stricter than RFC 7493
parse : $\mathbb{B} \to \mathbb{J} \cup \{\bot\}$	parsing under that profile; rejects input outside it
JCS : $\mathbb{J} \to \mathbb{B}$	RFC 8785 serialisation; injective on the normalised domain $\mathbb{J}$
$\mathbb{U}$	URLs and IRIs
Ctx : $\mathbb{U} \rightharpoonup \mathbb{B}$	a context environment: the bytes a party holds per URL
Λ	loader metadata that can affect processing: document URL and base, content type, profile, redirects
H, H_ℓ	SHA-256 and its ℓ-bit truncation; collision resistant. Quantitative second-preimage and birthday estimates additionally assume an idealised hash
Σ	a signature scheme; EUF-CMA secure

S3.1.2 Meaning layers

Layer	Object	Determined by
L0 bytes	b $\in \mathbb{B}$	the transmission
L1 JSON value	j = parse(b)	the parser profile
L2 RDF dataset	D = Exp_P(j, Ctx, Λ)	JSON-LD 1.1 expansion and toRdf under processor semantics P (version, `processingMode`, base IRI, `rdfDirection` and other options), using exactly the resources in **dep(j, Ctx, Λ, P)**
L3 interpretation	I = Int_E(D, V)	the vocabulary closure V (ontologies, shapes, rulebooks, code lists, crosswalks, selected by time, M27) **and the interpretation engine E with its configuration and built-in rules**

Layer	Object	Determined by
L4 decision	δ = Pol(I, π, Σ_ctx)	the policy π and the decision context Σ_ctx (S3.1.5)

For formats without an RDF layer (SD-JWT VC, mdoc), L2 is replaced by the format's own authenticated claim structure. The RDF-specific results (T2, T9, T10 I-RDFC) apply only to the RDF module.

Processing-aware dependency closure. dep(j, Ctx, Λ, P) is the set of all resources the processor dereferences while expanding j, together with the loader metadata and options that affect the result. It includes:

- contexts named at top level;
- contexts **embedded in node objects or values anywhere in the document**, inline or by URL;
- type-scoped and property-scoped contexts;
- `@import`, recursively.

It is defined operationally by recording the loader's requests under the pinned processor P, and is authenticated by a manifest. A verifier rejects any resolution outside the manifest (VER-16). Exp_P depends on Ctx only through dep.

The v4 definition started only from top-level contexts and context-to-context references. A context embedded in the document data escaped it; X12 changes the RDF this way.

S3.1.3 Typed digests

Earlier versions used "closure digest" for three different things. v5 separates them, with domain separation: H("ssc::v1" $\|$ JCS(object)).

Digest	Over	Used for
content digest	the bytes of one artefact	pinning, archives, the LFSC context list
context-manifest digest	the sorted list {url, content digest} of dep(j, Ctx, Λ, P), plus Λ and P	LFSC commitments, codec manifests, monitoring of loaded contexts (M23, M24 bundle events)
semantic-closure digest	the context-manifest digest, the content digests of the vocabulary closure V, and the identity and configuration of the interpretation engine E	receipts, admission, withdrawal tracking (M22, M25)
decision-context digest	the digests of every member of Σ_ctx	replay (T14)

A withdrawal of a rulebook in V must advance the epoch of every admission whose **semantic-closure** digest covers it. A context-manifest digest alone does not see V.

S3.1.4 Canonicalisation

Can maps datasets to byte strings with Can(D) = Can(D$'$) $\Leftrightarrow$ D $\cong$ D$'$. A bounded run returns Can(D) or $\perp$_B. **The meaning of B depends on the implementation.** `rdf-canonize` 3.x

bounds deep iterations per blank node; 5.x enforces a global budget derived from the number of non-unique blank nodes and `maxWorkFactor`. κ must record which.

S3.1.5 Decision context

Σ_ctx = (input bytes; dep with bytes; V; P and E with versions, options and executable semantics; policy π; status and revocation evidence; issuer-key resolution snapshot; trust-registry snapshot; authority epoch ε; decision time and its source; certification time of the evidence and its source; bound B and its semantics; any verifier state the decision reads).

A member that is irrelevant to every admitted verifier, or derivable from the others, may be omitted (T14).

S3.1.6 Actors

- **Issuer.** Holds key sk_I, authorised for assertion by the controller named in the credential's `issuer`. An honest issuer runs a specified Issue algorithm.
- **Publisher of a namespace.** TUF-style roles: threshold root, targets with delegations, snapshot, online timestamp.
- **Registry, transparency log, witnesses and monitors.**
- **Archive custodian.**
- **Verifier V.** State: metadata and target high-water marks, permanent withdrawal set, caches, clock.
- **Relying party.**
- **Edge device.** Measured boot chain, an isolated decision engine, attestation key AK, registers.
- **Agent kernel** with epoch ε.
- **Auditor.**
- **Network observer O** (for OBJ-9).

S3.1.7 Adversary classes

Class	Capabilities	Cannot
A-net	read, delay, drop, replay any published object; serve any bytes for a URL	forge signatures; control a trusted clock
A-clock	A-net plus set V's clock	forge signatures
A-input	send arbitrary input	sign as an allowed issuer
A-iss	an allowed issuer that signs arbitrary configurations and commitments	sign as another issuer
A-ins	an **authorised insider** publisher or threshold that releases harmful but authorised meaning	(nothing cryptographic: only review, testing and transparency limit it)
A-pub	one compromised publisher role key, for example the online timestamp key	meet a threshold it does not hold
A-log	an equivocating log or registry operator	forge honest witnesses' cosignatures
A-drift	none: the same URL serves updated bytes	—
D0	honest device (may reboot)	—

Class	Capabilities	Cannot
D1	root on the device outside the measured boot chain: runs modified code, chooses bundles, reboots, resets resettable registers, extends registers, and **invokes any TPM command the key policy permits, including using the engine key while the register shows the approved engine**	alter the measured boot chain; break isolation of an isolated engine runtime
O	network observer of resolution and update traffic	see inside the verifier

The following are outside the model and stated as limits: a compromised measured boot chain; compromised libraries or compilers (supply-chain compromise of the verifier itself); hardware side channels.

S3.2 The objectives as predicates

Objective	Predicate	Theorem
OBJ-1 authority	Every release V accepts, as a tuple (root epoch, namespace, type, operation, digest, version, time), has a valid scoped authorisation derivation in V's accepted trust state	**T17**
OBJ-2 integrity	(a) **publisher bytes:** the bytes V uses equal those the authorised publisher released. (b) **issuer graph (L2):** if V accepts c, V's dataset is isomorphic to the dataset the issuer's honest Issue computed; against A-iss, it equals the issuer's signed commitment. (c) **interpretation (L3):** V interprets IRIs only through the pinned V and E	T1, T2, T10; VER-13, VER-16
OBJ-3 freshness	If V uses v at t, an honest authorised status assertion declared v current at some $t' > t - \Delta_status - \varepsilon$	T4, T5
OBJ-4 rollback	Metadata and target versions never decrease except by authorised downgrade; an accepted withdrawal is never undone for current authorisation	T6
OBJ-5 consistency	(a) **fork exclusion:** accepted checkpoints lie on one history. (b) **conflict detection:** conflicting releases are eventually detected by a monitor	T7
OBJ-6 completeness	δ is a function of the input and the reconstructible Σ_ctx	T14
OBJ-7 bounded processing	Pre-authentication work is within the declared operation and memory budget, O(n log n) under the stated cost model; total work $\leq$ B	T3, T8, T9
OBJ-8 safe interpretation	V never accepts while ignoring a meaning-changing feature (T13 covers extension downgrade); loss in transformation, disclosure completeness and use binding are covered by VER-6, VER-8, VER-11, VER-13	T13; PoS P2
OBJ-9 privacy	Observable resolution and update traces are independent of the credentials processed, up to a declared leakage L	**T18**
OBJ-10 accountability	Replay over the reconstructible Σ_ctx reproduces every definitive verdict; an attested decision is bound to the engine computation and closure that produced it	T8, T11, T14, T15

Objective	Predicate	Theorem
OBJ-11 conditional availability	Under a declared envelope (admitted input language, workload, resources, dependency reachability, retention horizon), every admitted input obtains a definitive result within the declared bound. Nothing is promised under indefinite network suppression	(declared, measured; T4 states the trade-off)

Every safety objective above is satisfied by a verifier that rejects everything. OBJ-11 is what excludes that degenerate design, and the model checker requires an acceptance, use or commit for each "holds" result except the explicitly unavailable `MC6.offline.deferEffect` case, which demonstrates safe deferral rather than availability.

S3.3 Theorems and proofs

"Except with negligible probability" means: except with the adversary's advantage against EUF-CMA of Σ, collision resistance of H, or the attestation roots. Statements are scoped to their premises; each premise is traced to a requirement (S2; traceability in S3.6).

T1 — Signed is not meant (F1)

Statement. Let Ver(pk, j, σ) not depend on Ctx (JCS-based Data Integrity, JOSE/COSE envelopes, SD-JWT). Let j be accepted, and let its L2 output **actually depend** on mutable bytes at some u $\in$ dep(j, Ctx, Λ, P). Then there are Ctx, Ctx$'$ differing only at u under which the same (j, σ) is accepted with non-isomorphic meanings. Membership of an unused context in a conservative closure does not establish dependence.

Proof. Ver is the same under both; the meanings differ by the dependence assumption. ■ Witness: WE-1 B and D.

Corollary T1.1. Checking content digests for a set U guarantees L2 agreement against A-net **if** U $\supseteq$ dep(j, Ctx_I, Λ, P) and V uses the same P, Λ and options. If some dependency is unpinned, T1 applies to it. X2 (a type-scoped remote context) and X12 (a context embedded in the document) show that dependencies beyond the top-level list occur.

T2 — `ecdsa-lfsc-2026`: agreement with the issuer's computation, or with its signed commitment (M17)

Construction.

cfg = (suite, meta, ℓ = H(JCS(j)), s = H(Can(Exp_P(j, Ctx_I, Λ))), Γ = context-manifest of dep(j, Ctx_I, Λ, P), P)

σ = Sign(sk_I, H(JCS(cfg, @context))), with suite domain separation.

V accepts iff:

- **stage 1:** the input is in $\mathbb{J}$; `proofValue` has the suite's fixed length; H(JCS(j$'$)) = ℓ'; the verification method is authorised by the issuer named in j$'$ for assertion (VER-15); and Ver holds;

- **stage 2:** every resolution is in Γ' with matching bytes (VER-16), and H(Can_B(Exp_P(j′, Ctx_V, Λ))) = s′.

Statement.

- **(a) Honest issuance.** Let the named issuer be uncompromised and run the specified Issue algorithm. If V accepts, then except with negligible probability there is an Issue record with the same configuration, $j' = j$, and $Exp(j', Ctx_V, \Lambda) \cong Exp(j, Ctx_I, \Lambda)$.
- **(b) Malicious issuer (A-iss).** If V accepts, V's dataset is isomorphic to the dataset whose canonical hash the issuer **signed**. Nothing is implied about the truthfulness of that commitment or the correctness of the issuer's computation.

Proof sketch.

1. The verification method is authorised for the named issuer; by EUF-CMA and collision resistance, cfg' is a configuration that issuer signed.
2. Lexical-digest equality and injectivity of JCS on $\mathbb{J}$ give $j' = j$.
3. Equality of the canonical hashes and canonicality of Can give isomorphic datasets.
4. In (a), the honest Issue algorithm relates the commitment to its own expansion.

■

Scope.

- This is **agreement conditional on acceptance**. It does not say that every processor accepts.
- It binds L2, not L3 (VER-13), and says nothing about availability.
- The context digests Γ are not needed for the binding; they give early detection and the closure record.
- T2 is processor-independent in the sense that it compares outputs. Pinning (T1.1) compares inputs.
- **Use binding:** applications must consume the authenticated view (VER-11).

Corollary T2.1 (graph-native re-verification). Accepting stored quads Q with a stored configuration and a stored, authenticated issuer implies $Q \cong$ the dataset of (a) or (b) above, **if** the verification method is checked against that stored issuer. This re-verifies the **whole committed dataset**. It does not provide standalone proofs for subsets or single facts; that would need a separate membership scheme (R2-33).

Harness v0.3 gaps:

- X10: no issuer-key binding in `verifyLfsc`; no allow-list in `verifyFromQuads`.
- X2: top-level context list only.
- X12: no processing-aware closure.

T3 — Authenticate-first cost (M17, M18)

Cost model. n is the raw input size in bytes. Counted: parsing, the pre-scan, JCS, hashing, decoding of every unauthenticated field, issuer allow-list and key lookup in local data, memory. One signature verification costs a constant.

Statement. If every unauthenticated field decoded before the signature check has constant-bounded size, and key resolution is local, then the work of stages 0 and 1 is O(n log n) with comparison sorting of member names in JCS. It is O(n) with radix or trie ordering, because RFC 8785 orders names by UTF-16 code units. It requires constant memory beyond the input. This bounds stages 0 and 1, **not every rejection.** Replays of valid documents and A-iss can reach stage 2, which only B bounds (T8, T9).

Proof. From the stage structure; stage 2 is reached only after a valid signature over a matching configuration (T2, step 1). ■

Harness v0.3 gap (X11): `proofValue` is decoded in quadratic time before the signature check.

T4 — Freshness and availability cannot both hold under withholding, and pins give neither

Statement.

1. Couple two executions with identical observations. In W_0 the artefact stays current; in W_1 it is withdrawn at w, and A-net withholds all updates.
2. If V must still accept at some $t > w + \delta$ in W_0 (availability), it accepts with the same probability in W_1, violating bounded post-withdrawal cessation.
3. **No verifier achieves both** under indefinite withholding. A trusted clock lets V *stop* accepting when evidence expires; it does not deliver new evidence.
4. With digest pins only, or with an adversary-controlled clock, V cannot even stop.

Proof. Indistinguishability of the coupled executions. ■

Consequences.

- DST-6 (trusted time, bounded clock error) is Core for the expiry-based design.
- OBJ-11 makes availability an explicit, conditional objective with a declared reachability envelope.
- The earlier wording "trusted time is necessary for freshness" assumed non-interactive distribution. A nonce-bound status query also gives freshness without a clock, but VER-3 excludes such queries during verification.

T5 — Freshness bound anchored on the status assertion (M8)

Statement. Suppose:

- every use at t depends on an authenticated status assertion S, bound through checked digests and versions to the exact artefact and scope;
- an honest authorised signer issued S at t_0 declaring the artefact current;
- S expires at $e \leq t_0 + \Delta_status$;
- V requires $clock(t) < e$, with $|clock(t) - t| \leq \varepsilon$;
- V fails closed without such an S.

Then $t_0 > t - \Delta_status - \varepsilon$. If no honest assertion declares currentness after a withdrawal at w, no use occurs at or after $w + \Delta_status + \varepsilon$.

Proof. Combine the expiry inequalities. ■

What changed (R2-04). v4 bounded freshness by the **timestamp** lifetime. A compromised online timestamp key can sign a fresh timestamp that points to an older, unexpired snapshot or targets object. The verifier then uses a version that was not current within the timestamp lifetime. MC-5 confirms this (trace: withdrawal at t = 1, forged timestamp at t = 2 pointing to the status object from t = 0, use at t = 2). The bound is Δ_status, the lifetime of the honest status-bearing object, and DST-11 requires profiles to name that object and its lifetime.

T6 — Rollback and permanence of withdrawal (M8, M14)

Statement. Suppose that, for each root epoch, namespace and artefact, V keeps:

- durable metadata and target high-water marks;
- a permanent withdrawal set;

and accepts a lower target version only under a current, scoped downgrade authorisation. Then, by induction over guarded atomic updates:

- metadata versions are monotone;
- target versions are monotone except at authorised downgrades;
- an artefact is never used for **current authorisation** after V learned of its withdrawal.

Historical decoding of withdrawn meaning, for replay and archives, is a separate operation and remains permitted (T14, DST-8).

What the model checker shows.

- The earlier design (version counter only) lets a replayed older "current" object undo a withdrawal (MC1.v09).
- The complete design holds (MC1.m8).
- **Removing any single one of** the metadata counter, the version counter or the permanent-withdrawal guard **still holds in the small instance** (MC1.red-*): a single artefact with a monotone honest publisher cannot exhibit the attacks that require each guard independently (multiple artefacts, equal-version equivocation, crashes, role compromise).

Their necessity rests on T5, T6 and the role structure, not on MC-1.

T7 — Fork exclusion and conflict detection (M9)

Statement (fork exclusion). Assume:

- a common witness set of n, at most f Byzantine, quorum t with $2t - n > f$;
- unforgeable signatures;
- honest witnesses that are stateful and cosign only verified consistent extensions.

Then quorum-certified checkpoints lie on one append-only history.

Proof. Two conflicting branches would each need a quorum. The quorums intersect in an honest witness, which never signs inconsistent checkpoints. ■

Statement (detection). Conflicting releases are detected only if, additionally:

- a decidable conflict relation is defined over (publisher, namespace, identifier, version, epoch, status); legitimate succession is not a conflict;
- an honest monitor eventually retrieves every leaf under accepted checkpoints (DST-13).

Detection is after the fact. Witness-set reconfiguration and fresh-head policies need separate invariants.

T8 — Deterministic verdicts (M19)

Statement. Suppose:

- the verifier is a deterministic state machine with fixed executable semantics, recorded in κ;
- its metering has fixed semantics, and B counts operations under them;
- an interruption by wall-clock or resources yields *unknown*.

Then any two completed runs on the same input and decision context return the same definitive verdict. Operation counting alone does not create determinism.

Proof. Induction on identical deterministic transitions. ■ WE-5 shows that wall-clock limits make definitive verdicts machine-dependent.

T9 — Blank nodes, admission and lists (M18, F5)

Lemma T9.1 (cost). RDFC-1.0 calls N-degree hashing only for blank nodes whose first-degree hash is shared. Without such nodes the dataset is canonicalised by sorting and hashing, O(m log m) comparisons in a model where comparing two quads costs a bounded amount. ■

Lemma T9.2 (sufficient conditions). Expansion yields no blank nodes **if** none of the following sources occurs. The sources are evaluated on expanded IRIs and keyword semantics in every context of dep:

- S1: node objects without `@id` (also under `@reverse`, `@included`, `@nest`);
- S2: blank-node identifiers reached directly or through a term, compact-IRI prefix, `@vocab`, `@type` value or map key;
- S3: lists (explicit, or any value of a `@list` term);
- S4: graph containers and graph maps without `@id`;
- S5: `rdfDirection: compound-literal`;
- S6: `@id` redefined or aliased by a scoped context.

The conditions are sufficient, not necessary.

Admission design. A lexical scan cannot evaluate S1–S6 against arbitrary contexts. The design requires:

- a **restricted admission grammar**;
- a publisher **certificate** binding the complete closure, processor options, grammar version and the certification procedure (PRD-10);
- a verifier that enforces exactly that grammar (VER-4).

Until a separately specified certification checker is proved sound, the certifier remains a **trusted** party. Harness v0.4.0 implements the restricted admission grammar and has finite prototype

and regression evidence (E4 for those tested paths); this does not prove checker soundness for all inputs or remove the trust assumption.

Corollary T9.3 (F5, the tested family). For a simple RDF list (each list node has exactly the incident edges rdf:first, rdf:rest and one incoming rdf:rest or subject edge), interior list nodes are distinguished at the first-degree round only by their values. In a cell's incident quads the cell itself appears as `_:a` and every other blank node as `_:z`, so its neighbours' values are invisible at this round, and repeated interior values yield non-unique first-degree hashes. Whether a list completes within the default budget depends on how many cells fall in groups that share a hash, and on whether their neighbours are already unique. In `[0,1,0,1,0]` only the two interior 1s share a hash, and their neighbours are unique through value or end position. The default budget (`maxWorkFactor` 1) equals the number of such cells, and every entry into N-degree hashing, outer or recursive, consumes one unit (R3-24 and R3-25, checked in the library source).

Outcome on `rdf-canonize` 5.0.0 defaults	Lists
Fails	`[1,2,2,3]`, `[0,0,0,1]`, `[0,1,1,0]`, `[1,2,3,3,4,5]`, `[1,2,1,2,1,2]`, `[0,0,0,0]`
Completes	`[1,2,3,1]`, `[1,2,3,2,4]`, `[0,1,0,1,0]`, `[1,2,1,2]`, `[0,0,0]`

On the pinned v0.3 stack (per-node bound 1), lists of up to four equal values complete and five fail. An independent reviewer reproduced the whole matrix (X14).

These are measured outcomes for the tested family and two stacks. They do not establish a universal boundary or an asymptotic law. **Rejection can be an availability problem** wherever such lists are legitimate; whether it is exploitable depends on the use.

T10 — Codec binding (M15)

Statement.

- **Profile B (signed envelope).** If the manifest binds **every decoding input** and the envelope is signed by the issuer, identical inputs to the specified deterministic decoder yield identical output. Every decoding input means: registry and decoder or processing-model identifiers and versions, codec semantics, type and URL codecs, tables, the context-manifest, base and normalisation rules (DST-12).
- **Profile I-RDFC.** If the payload carries a graph-binding proof (RDFC or LFSC) and V uses its authenticated graph, V obtains L2 agreement, even if decoding provenance is incomplete.
- **Neither profile** binds L3, guarantees successful decoding, or authenticates the other's material.

Proof sketch. B reduces to equality of all deterministic inputs plus signature integrity; I-RDFC reduces to T2. ■

Quantitative bounds. In an idealised hash model, a second preimage against K target manifests succeeds with probability about q · K · 2^−ℓ, and a chosen collision needs about 2^(ℓ/2) work. Logging constrains which manifests are accepted; it does not add hash bits. **Default ℓ = 256.**

Case analysis (MC-4 v5: 6 drift kinds × rewrite × archive × 4 proofs × 3 manifest scopes):

Profile	Top-level manifest (v0.3)	Transitive contexts (first correction)	Full decoding manifest (current design)
A: signature over CBOR-LD bytes	fails	fails	fails
B: signature over the envelope	fails	**fails** (decoder-semantics drift)	**holds**
I-RDFC	holds (assumed)	holds (assumed)	holds (assumed)
I-JCS	fails	fails	fails

The I-RDFC verdict is **assumed** from T2. The case analysis does not test it (R2-07).

Experiments.

- X4: profile A with a rewritten envelope.
- X7: profile B with a transitive context.
- X8: inner JCS proof.

In harness v0.3.0, WE-2's I-RDFC label resisted the envelope rewrite and measured 337 bytes; X4's unsigned-label configuration changed from 151 to 168 bytes (+17). These are historical configurations. The v0.4.0 256-bit envelope measures 374 bytes for I-RDFC (314-byte payload, +60) and 356 bytes for profile B (129-byte payload, +227) in the reproduced candidate; post-quantum components are not included.

T11 — Execution-bound attestation of decisions (M24)

Statement. Suppose an accepted device identity binds a certified key K to an approved isolated runtime E (the decision engine), and:

- **(i)** attestation roots are intact;
- **(ii-a) exclusive key use (technical provenance).** The relying party has authenticated and pinned the exact receipt public key K through a technical key-authentication procedure whose integrity is assumed; organisational identity and mandate are separate premises in (ii-b). Accepted evidence binds K to the approved engine, platform boundary and key policy, and establishes, against the stated adversary, that the approved isolated engine E exclusively controls K's use. Possible evidence: a TEE attestation of key generation and use, or creation in a controlled ceremony followed by continuing platform protection. A TPM key certificate shows K's policy, not who holds its authorisation secret: a key the host creates with the approved policy passes it (harness v0.4.0, WE-3 scenario E3). Key certification, device provenance and organisational attribution alone do not establish exclusive engine use;
- **(ii-b) authorised enrolment (attribution), when claimed.** A current enrolment record, from an authority recognised for that function under the ecosystem's governance, binds K to:
  - the device identity;
  - the approved engine and key-policy versions;
  - the key epoch;
  - the responsible legal person L;
  - the authorised decision scope S;
  - validity and status.

The legal person's identity and mandate are established independently from designated authoritative sources. For attribution, the relying party authenticates this record under the designated enrolment authority, matches the exact receipt key and key epoch, and checks that the receipt's actual action, subject, purpose and audience fall within its authorised scope. The profile identifies the relevant time and the validity, freshness, compromise and revocation rules: current reliance uses the defined use or commit point; historical attribution requires authenticated evidence for its relevant time and the declared historical-verification policy;

- **(iii)** E's code, decision state and the immutable snapshot of input and closure are protected against host modification (**runtime isolation**);
- **(iv)** E signs exactly its computed outcome and the exact context digests;
- **(v)** receipts include a fresh relying-party challenge, recipient or purpose, and a decision identifier;
- **(vi)** the engine and key are authorised under the applicable technical policy when the receipt is produced, with rollback-resistant upgrades and revocation (`PolicyAuthorize` must not allow replay of revoked engine authorisations; NV-index lifecycle specified; MON-5). Current reliance additionally applies technical authorisation checks at the defined use or commit point. These do not replace the identity, mandate and scope checks of (ii-b).

Then an accepted receipt was produced by E over the named context, except with negligible probability.

Governance-attribution corollary. If (ii-b), including its relevant-time, key-epoch and actual-use scope checks, also holds, the receipt is bound to L's enrolled authorisation for that use within S under the stated governance and compromise/revocation policy. This is an evidentiary attribution claim; it does not prove substantive correctness, that L personally approved this particular computation, or legal liability.

Remark on (ii-a) and (ii-b).

- Attribution without exclusive use supplies evidence for accountability under the applicable governance and law; it establishes neither a correct computation nor who performed a disputed use after compromise.
- Exclusive use, with T11's other premises, establishes execution provenance; it does not by itself identify an accountable legal person or establish substantive correctness.
- Neither implies the other, and receipts should state which technical guarantee they claim. For execution-bound receipts the high-assurance profile requires T11's technical premises, including (ii-a); for replay-validated receipts it requires T11′'s authenticated group configuration, threshold-security and admissible-replay premises. In both cases MON-8 additionally requires organisational attribution. Replay validation does not require or establish the device's exclusive isolated-engine key use.

Proof sketch.

1. The receipt signature identifies K.
2. Exclusive use identifies E.
3. Isolation and the signing invariant identify the computation.
4. The challenge and context binding exclude replay and substitution.

Corollary argument. The authenticated enrolment record binds the same key and epoch to L and S; independent identity and mandate validation establishes the governance basis for that

binding. The acceptance checks place this receipt's actual use within S at the policy's relevant time, subject to its status and compromise/revocation rules. None of these organisational conclusions follows from TPM key certification or exclusive engine use alone.

■

Necessity (MC-3 v5):

Design	Honest device	Root outside the boot chain
V1 harness v0.3 (application receipts, resettable PCR 23, last event)	holds	**fails** (X6 on swtpm)
V2 first correction (application receipts, decision events, taint)	holds	**fails**
V3 attested engine, resettable identity register	holds	**fails**
V4 second correction (attested engine, non-resettable register, **PCR-only key policy**)	holds	**fails in two steps**: root invokes the engine key while the register shows the approved engine
V5 isolated engine with exclusive key use	holds	**holds**

Consequences.

- PCR equality discharges neither exclusive use nor runtime isolation. These must come from the platform: a TEE, or an OS-enforced isolated process whose key the host cannot invoke, with a policy that binds key use to that process.
- On swtpm, WE-3 v0.4.0 simulates the design with the engine as a separate OS process. All 20 scenario outcomes were reproduced. This is prototype evidence for receipt construction, key-policy checks and appraisal under an idealised protected-engine boundary, not a demonstration of isolation or hardware trust.
- Bundle-load measurement is monitoring (M23), not decision binding.

T11′ — Replay-validated receipts with a threshold key (alternative to isolation)

What it claims, and what it does not. T11 gives *execution provenance*: the receipt names the closure the isolated engine actually ran. T11′ gives something narrower: an accepted receipt's **outcome** is what deterministic evaluation of its committed, admissible context yields. Replay by a co-signer cannot observe what the device executed (R3-02).

Setting.

- **The signature scheme.** TS is a concrete t-of-n threshold signature scheme, $t \geq 2$. It is unforgeable under chosen-message attack in the threshold setting (threshold EUF-CMA), against a malicious coordinator and a fixed set of at most $f < t$ corrupt participants.
- **Corruption.** The device is a participant and counts as corrupt under D1. Corruption is static; an adaptive or mobile adversary needs the scheme's own adaptive theorem and proactive refresh.
- **Scheme conditions.** Key generation and registration, session scheduling, nonce generation and state recovery satisfy the chosen scheme's theorem. For FROST, that is the analysis RFC 9591 cites; threshold ECDSA by MPC needs its own named theorem.
- **Group key.** The mapping from key identifier and epoch to group key and profile is authenticated independently.

The receipt message. One injectively encoded, domain-separated message:

m = Encode("ssc:replay-receipt:v2", profile version, group-key identifier and epoch, authority epoch, status expiry, request or challenge, audience and purpose, decision identifier, action and scope, outcome, typed input digest, semantic-closure digest, decision-context digest, admissibility-policy identifier and version).

The authority epoch and the status expiry are signed, so that the relying party's effect gate can check them at its commit point (harness v0.4.0, R4-05).

The decision-context manifest binds every decision-relevant element of S3.1.5 by its role, not as an unordered set of hashes. That includes:

- policy and engine versions;
- time evidence and selection rules;
- status, key and trust snapshots;
- the authority epoch;
- the metering semantics.

Honest co-signer. Before it emits a usable signature share for m, an honest co-signer:

1. **Parses and authenticates.** It parses m, authenticates the request and its scope, reconstructs an immutable context Σ, and recomputes every committed digest.
2. **Checks admissibility.** It checks the profile's admissibility predicate A(request, Σ) against independently anchored roots, policy and state:
   - the closure is the one the request authorises, not merely an approved one;
   - status reflects the current authority epoch, or a declared freshness window that the receipt states;
   - time evidence is used with a declared uncertainty rule.

   Authentication alone is not admissibility. Device-only observations are typed as *claims*, whose truth this procedure does not establish.
3. **Replays.** It runs the deterministic decision under the recorded semantics and bound, and approves exactly the bytes m only if the definitive result equals the proposed outcome. It refuses *unknown* and indeterminate results.
4. **Signs.** It runs the signing protocol without message substitution or nonce reuse. Preprocessing is allowed where the scheme permits it, but no usable share is emitted before m is approved.

Relying party. It verifies:

- the authenticated group key and profile;
- the signature;
- the audience and request binding;
- the digests.

It also applies the current-authorisation checks required at its commit point (DEC-7), and consumes the challenge or decision identifier for single-use actions.

Statement. Under these premises, except with probability bounded by the threshold-EUF-CMA advantage plus collision, encoding and authentication failure probabilities, an accepted receipt certifies two things:

- its outcome equals deterministic evaluation of its committed context;
- that context satisfied the admissibility rule at signing.

Current authorisation at the time of the effect follows from the relying party's atomic commit check, not from the signature.

Proof sketch. Suppose an accepted receipt has an outcome inconsistent with its committed context, or its context fails A.

- If no honest participant approved m, a signature from at most $f < t$ corrupt participants is a forgery in the threshold-EUF-CMA game.
- If an honest participant approved m, its replay and digest checks establish the property.
- A signature copied from an earlier authorised session is not a forgery; the request, audience and single-use rules exclude its reuse. ■

Not established:

- the device's execution;
- the truth or completeness of device-only observations;
- that a signature is ever produced (liveness).

Corruption and progress.

- **Corruption bound.** A static bound $f < t$ is not "fewer than t at any instant": an adaptive extension bounds the distinct compromises per key epoch, and mobile security needs refresh and erasure.
- **Progress.** In 2-of-2 a malicious device can always withhold cooperation; security with abort is not liveness. Progress despite f withholding participants needs $n - f \geq t$, plus the protocol's robustness, authenticated communication and a working coordinator. RFC 9591 does not promise robustness against participant abort.
- **Common-mode failures.** Co-signers hosted in the same organisation or on the same TEE platform need a model of shared failures before they count as independent.
- **Enrolment of participants.** Each authorised participant is enrolled, together with:
  - its public verification share or commitment, and its participant identity;
  - its execution boundary;
  - the group public key, threshold and participant set;
  - the ciphersuite and key epoch.

  Secret shares are never published. Scheme-appropriate evidence of possession and consistency links the participants to one authenticated group configuration. Refresh, rotation or participant changes require an authenticated transition to the new configuration, updated verification shares or commitments and a configuration or share epoch even when the group public key is unchanged. The profile defines revocation of the old configuration, retention of historical bindings, and the chosen scheme's refresh, erasure and recovery conditions (MON-7).
- **What enrolment does not establish:** the honest-quorum assumption. A group signature normally does not identify the signing subset, so authenticated contribution records must be kept where individual attribution matters. Enrolment under different legal persons is evidence for separate failure domains, not proof.

Model checking.

- **MC-3** (R3-02). The honest co-signer sees only the claimed closure, the authenticated input and the proposed outcome:
  - V6 holds for the outcome property;
  - **V6x**, the same design checked against the execution-identity property of T11, fails as expected: the device ran A, claimed W, and the co-signer correctly signed W's outcome;
  - V6c, with the co-signer compromised, fails the outcome property.
- **MC-6** (R3-05, R3-06) checks each admissibility premise by removing it:
  - unauthenticated input;
  - counterfeit status;
  - an authentic but older status object, which fails the current-status property but satisfies an explicitly declared freshness-window property;
  - a device-chosen approved closure;
  - effects on provisional decisions while the co-signer is offline.

  Deferring effects is safe but leaves the action unavailable; MC-6 reports this as "no acceptance (by design)", not as success.

T12 — Withdrawal as an authority-epoch event (M25)

Statement. Suppose an action commits only at a defined **linearisation point**, where its action- and closure-bound admission, the current durable epoch, fresh status and the permanent withdrawal set are checked atomically. If the external effect lies outside the kernel, the effect itself must be fenced. Then no stale admission commits after the applicable withdrawal has been learned, and T5 bounds the delay when updates are withheld. The discrete bound $tw + D - 1$ in MC-2 is exact for its event ordering only.

Proof. The atomic guard excludes every stale or withdrawn state at the commit point. ■

Model checking (MC-2 v5).

- The complete design holds and is non-vacuous; both tightness mutants are caught.
- Removing the metadata counter **while keeping withdrawal memory** still holds (MC2.red-meta), as does replacing the epoch check by freshness and currentness checks (MC2.red-epoch). With one artefact and permanent memory these guards are redundant.
- Removing the counter **and** the memory fails (MC2.nometa). So do: no check, check-then-commit, push only, no fail-closed.

The v4 model's "without metadata counter" variant removed two safeguards at once. The v5 model reports isolated and compound removals separately.

T13 — Extension downgrade safety (M21)

Statement.

- If issuers and publishers authenticate a critical-extension declaration with scoped identifiers, and verifiers reject unsupported critical extensions and apply supported ones per the profile, no verifier accepts while ignoring a declared meaning-changing extension.
- Without marking, open-world verifiers cannot distinguish meaning-changing from ignorable extensions.

- Closed-world verifiers achieve this without marking, at the cost of rejecting harmless extensions.

This is **extension downgrade safety**, one part of OBJ-8. ■

T14 — Replay requires reconstructible decision-relevant state (M22)

Statement. If the verifier is deterministic (T8), and every member of Σ_ctx on which an admitted verifier may depend can be reconstructed from the receipt and the archive (DEC-1, DEC-6), then replay reproduces every recorded definitive verdict. **Necessity** applies only to information an admitted verifier may depend on and that cannot be derived otherwise. Replaying negative decisions requires retaining their inputs, subject to privacy and retention policy.

Proof. Induction over identical deterministic transitions from the reconstructed state. ■

T15 — Certification time, not event time (M27)

Statement. If the selection rule uses a validated TSA generation time, or a trusted bounded log-inclusion time bound to the evidence digest, the issuer cannot select an earlier **certification** interval without defeating that time service. This does not authenticate when the underlying event occurred, and the issuer still chooses when to submit. Selection must never use an issuer-asserted field such as `created` alone. M27 is corrected accordingly. ■

T16 — Parser agreement (L1)

Statement. Under a single raw-text and parsed-value profile (𝕁, S3.1.1), with the authenticated parsed value handed on immutably, the parser differences listed in v4 are removed:

- duplicate names;
- integers beyond the interoperable range and over-precise numbers;
- lone surrogates;
- non-UTF-8 input and byte-order marks;
- signed zero.

Any stricter numeric rule is a profile rule, not RFC 7493 (VER-14). ■

T17 — Scoped publisher authority (OBJ-1)

Statement. Assume:

- an authenticated root;
- valid root rotation under old and new thresholds;
- distinct-signer counting;
- delegations whose scopes (namespace, type, operation) intersect along the chain;
- validity intervals, revocation, and role and version links between metadata;
- durable rollback state;
- bounded update freshness (T5);
- no required threshold is compromised.

Then whenever V accepts a release tuple (root epoch, namespace, type, operation, digest, version, time), a valid authorisation derivation for exactly that tuple exists in V's accepted trust state. Revocation takes effect within the honest-status bound of T5; instantaneous delivery is not claimed.

Proof sketch. Induction over accepted root transitions and delegation depth. Each step either carries the required scoped threshold approval or constitutes forgery or threshold compromise; expiry and durable versions exclude stale authorisation beyond the window. ■

Scope. This establishes **authority**: not the truth of a vocabulary, its legal recognition, or benign behaviour by an authorised insider (A-ins). Premises: GOV-7, KEY-4, DST-3, DST-11.

T18 — Non-interference of resolution traffic (OBJ-9)

Statement. Fix the observer O and a permitted leakage L. Assume:

- the verification cell has no egress and cannot signal credential-dependent identifiers, cache misses or results into resolution scheduling;
- a separate fetcher uses only public state, on fixed or padded credential-independent schedules;
- cache contention, errors and retries introduce no undeclared dependence.

Then for any two credential streams with equal L, starting from low-equivalent public state with coupled public randomness, the distributions of observable resolution and update traces are equal: destinations, paths, sizes, timings and retries. They are computationally indistinguishable under the transport construction.

Proof sketch. Induction over fetcher transitions: equal public state and coupled randomness yield equal observations and preserve low-equivalence. ■ A fetch-on-miss channel falsifies the premise even when the verifier itself has no socket.

Scope. Presentation traffic and host side channels are excluded unless explicitly included. Premises: VER-3, DST-10, PRV-5.

T19 — Disclosure, sufficiency and the cost of assurance (M31)

An earlier version argued T19 wrongly in two places:

- it treated the number of admissible views as a lower bound on the cost of assurance (R3-01);
- it concluded that "linear assurance requires someone to see the whole graph".

Both are withdrawn. The revised result separates three questions:

- what the verifier can know from what it sees (information);
- what checking costs (computation);
- whether the evidence is complete (coverage).

Setting.

- Fix the semantic closure, the policy version and the reconstructible decision context C, and write $\pi_C(G) \in \{\text{grant, deny, unknown}\}$. The subscript matters: time, status, subject, action and authority can change a verdict with the same evidence.
- For a finite evidence graph G, a disclosure regime gives the admissible views R(G), with $V \subseteq G$ for every $V \in R(G)$.
- A certificate is not a view. Its observation $O = (h, C, \delta, \text{certificate, status, } \dots)$ is judged by an acceptance predicate.

Three properties, kept separate:

1. **Exact decision preservation** (decision-unique): $\pi_C(V) = \pi_C(G)$ for every admissible V.
2. **Grant soundness:** $\pi_C(V) = \text{grant}$ implies $\pi_C(G) = \text{grant}$.
3. **Deny soundness:** $\pi_C(V) = \text{deny}$ implies $\pi_C(G) = \text{deny}$.

Unknown is a third outcome, not a synonym for deny. For safe authorisation, grant soundness plus fail-closed handling of *unknown* can suffice. For reproducible three-way decisions it does not.

T19.1 — Optionality.

- **(a) Disagreement.** If $V, W \in R(G)$ and $\pi_C(V) \neq \pi_C(W)$, exact preservation fails. With independent statements, $R(G) = \{V : M \subseteq V \subseteq G\}$, this applies to $W = V \cup \{s\}$. With bundled disclosure, both views must be admissible.
- **(b) One-sided soundness.** If π_C is grant-monotone on the feasible pairs $V \subseteq G$ of the regime, the regime is grant-sound; deny-monotonicity likewise gives deny soundness. Global monotonicity is sufficient but not necessary.
- **(c) Quantifiers.** For a fixed G and M, an unsound grant exists exactly when some V with $M \subseteq V \subseteq G$ has $\pi_C(V) = \text{grant}$ while $\pi_C(G) \neq \text{grant}$. Global non-monotonicity of the policy does not by itself create one: a policy that grants unless an exception e is present has no unsound grant on $G = \{a, b\}$.
- **(d) Sufficiency of the mandatory set.** For $R(G) = \{V : M \subseteq V \subseteq G\}$, exact preservation holds iff π_C is constant on that interval, that is iff $\pi_C(V) = \pi_C(M)$ for all admissible V. This is a per-instance property; a uniform guarantee must quantify over all graphs and policy versions the profile admits.
- **(e) Counting.** Let the optional disclosure units be $B_1, \dots, B_b$ and F the permitted selections. The number of distinct admissible views is $N \leq |F| \leq 2\hat{}b$. Equality needs every subset to be selectable and distinct units to give distinct views. With singleton statements, $b = n - m$. One indivisible optional credential of 100 triples gives two views, not $2\hat{}100$.

Proofs. (a) Two different outputs cannot both equal $\pi_C(G)$. (b) Substitute $V \subseteq G$ into the monotonicity implication. (c) is the definition, with the admissibility conditions added; the counterexample is direct. (d) M and G are both admissible. (e) Count the image of the disclosure map. ■

T19.2 — Incomplete observation is a limit of information, not of computation. Let O be everything the verifier observes, and $\Gamma(O)$ the complete evidence graphs consistent with O and C.

- If $G_0, G_1 \in \Gamma(O)$ give different decisions, no procedure that uses only O decides correctly in both worlds.

- If π_C(G_0) = grant and π_C(G_1) $\neq$ grant, no procedure can grant in world 0 and be grant-sound in world 1. A safe procedure returns *unknown*.

Example. An authentic approval credential a, and in world 1 also an authentic disqualifying report e. The holder presents only a in both worlds, and nothing attests completeness. The transcripts are identical, so no amount of computation reveals e.

If a format authenticates an exhaustive manifest or other restrictive metadata, Γ(O) must respect it.

Proof. Identical observations give identical outputs. ■

T19.3 — Cost, with its premises. Let:

- A be the cost of obtaining and authenticating the evidence and checking its coverage;
- C_can the cost of canonicalisation;
- C_π the cost of one policy evaluation;
- N the number of distinct admissible views.

Task	Bound and premise
Full disclosure of a sufficient set K	A(K) + C_can + C_π(K). Correctness relative to the decision needs π_C(K) = π_C(U), where U is the policy-relevant evidence universe (T19.5)
Commit-and-certify, certification core	A(G) + C_can + hashing + C_π(G), plus signing and timestamping. Correctness of δ needs an honest certifier or a sound proof of the computation. Verified custodian deposit adds retrieval/opening, reconstruction, canonicalisation and another C_π(G) before successful issuance under DEC-13
Certificate check by the relying party	Signature or proof, scope, the selected certificate, status, closure and policy checks. Independent of n only when these are bounded independently of n
One presented view against a known G	C_π(V) + C_π(G). This checks that pair, not every possible presentation
Exact preservation over all views, with G known	At most N evaluations plus view generation. For a policy treated as a **black box**, N − 1 evaluations are necessary in the worst case: an unqueried view can differ
Exact decision from a holder-chosen view without coverage evidence	Impossible in general (T19.2); no cost assignment is meaningful
Dispute for an opened certificate	Retrieval, authentication and opening of the archive, context reconstruction, C_can + C_π(G). Shows computation consistency, not truth or completeness

Consequences:

- **Single-pass rules.** For single-pass rules over a canonical graph (C_π = O(n)), one certification costs O(n log n) plus acquisition and trust checks.
- **Richer policy languages.** For quadratic policy evaluation it is $O(n^2)$; for SHACL-SPARQL or OWL reasoning it is whatever the declared profile costs. The OWL 2 Profiles distinguish data from combined complexity, and a bound for one profile does not transfer to arbitrary reasoning.
- **Exhaustive checking** multiplies by N only as an upper bound. Structure can help: for "deny iff e is present", exact preservation is decided by one membership test, despite 2^q views.

- **Visibility.** "Someone must see the whole graph" is not a theorem. Independent sources can evaluate a policy jointly by secure multi-party computation without exchanging plaintext, given authenticated sources, complete coverage and a sound protocol.

The accurate headline is therefore: **one complete evaluation for the decision or certification core, plus representation, authentication and coverage overhead and any required deposit or dispute replays**, against a per-view analysis whose cost grows with optionality for policies that cannot be analysed structurally.

T19.4 — What opening and replay establish. Fix one selected certificate that authenticates (scope, h, C, δ, certification time). Assume:

- binding of h to the canonical representation;
- authentic retrieval of the certificate and its dependencies;
- deterministic replay (T14).

Then:

1. Except with the commitment's binding failure probability, two valid openings of h cannot give different representations, so parties relying on the same certificate refer to the same evidence.
2. $\delta = \pi_C(G)$ only if the certifier evaluated honestly or a sound computation proof is attached. Otherwise the signature authenticates an assertion.
3. With an available opening and a complete context, replay returns $\pi_C(G)$. A mismatch with δ shows an inconsistent claimed computation. A match does **not** show that the evidence is true or complete, that sources were selected correctly, or that the policy is legally correct.
4. Trusted certification time binds h to that time (T15); it does not date the underlying events.
5. A holder can obtain separate valid certificates over G_0 and G_1. That *one* snapshot is authoritative for a scope and epoch needs a separate selection and non-equivocation rule (DEC-15); new snapshots in later epochs are not attacks.

Proof sketch. Item 1 reduces two different openings to a binding break. Item 2 is the honest-evaluation premise. Item 3 applies T14 to the opened representation. Items 4 and 5 state what the premises do not give. ■

T19.5 — Completeness and sufficiency. Fix a decision scope s: subject identifiers, action, jurisdiction, time or snapshot, governing policy and closure. U_s is the policy-relevant authoritative evidence for s, not every fact in the world. Let the profile specify, independently of the holder:

- a closed inventory of sources and domains covering every dependency that can affect π_s, including dependencies reachable through the pinned semantics;
- source identity and authority, subject and key resolution, and complete query, pagination or issuer-push coverage;
- authenticated snapshot and freshness rules, with reconciliation of additions, deletions, supersession and revocation;
- where needed, scoped positive or absence attestations: subject, population or sample, category, method and time;
- a deterministic extraction procedure, and the rule that missing, stale or inconsistent coverage yields *unknown*.

If the sources are sound for their declared responsibilities, validated coverage establishes $\pi_s(G) = \pi_s(U_s)$, and the accepted output satisfies $\delta = \pi_s(G)$, then $\delta = \pi_s(U_s)$. The second premise can be discharged in four ways:

- full evaluation by the relying party;
- honest certification;
- a sound proof of the computation;
- a proved sufficient projection K.

Proof. Substitute the equalities. Without coverage, T19.2's example applies: every signature, commitment and evaluation can be correct while the grant is wrong. ■

This is a conditional lemma; it does not make sources truthful. The following hold:

- A source's own list does not prove a complete cross-issuer inventory.
- A status list answers for the entries it references; it does not enumerate everything a holder ever received.
- A laboratory's absence attestation covers its test scope, not arbitrary substances, batches or sampling errors.
- Sampling needs its own statistical contract: uniformly sampling q of N records misses all b bad ones with probability $C(N-b, q)/C(N, q)$, which is $1 - q/N$ for a single bad record.

Consequences for M31.

- **Coverage is the foundation.** No disclosure regime makes incomplete evidence complete: under full disclosure, commit-and-certify and selective disclosure alike, a holder can omit adverse evidence before anyone sees it. The coverage contract (DEC-9) is therefore the foundation of M31.
- **The regimes are choices, not consequences of a lower bound.** Full disclosure of a sufficient set and commit-and-certify are preferred because they evaluate one fixed sufficient evidence set and give no holder choice over policy-relevant evidence; commit-and-certify adds deposit replay and permits later dispute replay.
- **Selective disclosure stays possible.** It remains permissible for withholding policy-irrelevant fields, or for presenting proofs whose sufficiency for the exact policy, closure and required guarantee is established (DEC-8, VER-8).
- **The design rule becomes:** *fix the sufficient evidence for each purpose and decision in the profile, establish its coverage independently, evaluate that fixed set with the required verification replays, and leave the holder no choice over policy-relevant evidence.*

Privacy of the certificate transcript (R3-12, reproduced). A salted commitment h hides G, but a public decision-context digest that includes a deterministic digest of G reveals it to anyone who can enumerate candidates. Our two-candidate reproduction recovered G without the salt. Private inputs must therefore be bound in public digests only through h or another hiding binding (PRV-6). Stable certificate, status and locator identifiers remain linkable.

What it does not give.

- Without a sound computation proof, the relying party relies on trusted certification and custodian replay. Deposit replay precedes acceptance; dispute replay provides a later check of computation consistency.

- Verifiable computation and commit-and-prove proofs (for example Pinocchio, LegoSNARK) could remove trust in the evaluation, though not in the evidence. We have neither implemented nor benchmarked them for these policies, and make no claim about their practicality.

S3.3.20 Plan for mechanised verification

Statement	Tool and model	Obligation
T5, T6, T12	TLA+/Apalache: separate signed roles with independent expiry and versions, compromised keys, multiple closures, crashes, downgrades, external-effect linearisation	age at actual use; permanent withdrawal; action-specific epoch fencing; isolated guard mutations with reachability
T7	TLA+/Apalache or Alloy: checkpoint ancestry, persistent witness state, quorum identities, witness-set migration	fork exclusion across reconfiguration; monitor liveness as separate assumption
T2, T10, T17	Tamarin or ProVerif: signed message domains, issuer and publisher authorisation, compromise, manifest and payload substitution	every acceptance corresponds to the intended scoped issuance or publication
T11, T11′	Tamarin with execution and key-call events, plus a platform-specific refinement; for T11′, threshold signing with replaying co-signers and the exact receipt message	the host may invoke every TPM command its policy allows; accepted receipts imply `EngineComputed` (T11) or `ReplayedBy(t − f honest)` (T11′), not merely `KeySigned`
T19	Alloy or a SAT/SMT encoding of small policies with exceptions and bundled disclosure units	exact preservation, grant and deny soundness per instance; indistinguishable worlds (T19.2); coverage sufficiency (T19.5)
T18	ProVerif observational equivalence, or self-composed TLA+ fetch and cache schedules	equivalent low observations for different credential streams
T9.2, T8, T14	Coq or Lean over a restricted admission grammar and executable deterministic semantics	admission-checker soundness; dependency-complete replay

---

S3.4 Model checking

`model/mc.mjs`, Node 22, no dependencies, about 12 s. **66 cases**, each with a stable ID. `--check` fails unless:

- every expected case is present exactly once with its expected verdict;
- every "holds" case has a reachable acceptance, use or commit, except the explicitly declared `MC6.offline.deferEffect`, which is safe with no acceptance by design.

Self-test: deleting the MC-4 cases makes `--check` exit 1 with "missing case". v4's check had passed with cases missing (R2-31).

Model	Holds	Violated	Reading
MC-1 M8 (12 cases)	corrected M8; skew with window $\Delta + \varepsilon$; three **REDUNDANT** single-guard removals	pin only; no expiry; stateless; version counter only (withdrawal undone); adversarial clock; skew with window Δ; tightness mutant	The complete design holds; single guards are not individually necessary **in this instance**
MC-2 M25 (10 cases)	corrected M25; two **REDUNDANT** removals (counter with memory kept; epoch check replaced by freshness and currentness)	counter and memory removed; no check; check-then-commit; push only; no fail-closed; two tightness mutants	Compound removals fail; the discrete bound is exact for its ordering
MC-3 M24 (14 cases)	all designs with an honest device; **V5 isolated engine** against root (execution identity); **V6 threshold key with a replaying co-signer** against root (outcome property)	V1, V2, V3, **V4 (PCR-only key policy: root invokes the key)** against root; **V6x**: V6 checked for execution identity; V6c (co-signer compromised)	Isolation with exclusive key use binds execution; replay co-signing binds only the outcome
MC-6 T11′ premises (14 cases)	input authenticated to its source; status signature checked; current epoch known independently; a declared freshness window; request-bound closure; effects deferred (safe, **no acceptance by design**); co-signer back online; co-signer online before the withdrawal (the effect gate refuses after it; R4-27)	input as given; counterfeit status; older authentic status under a current-status property; device-chosen approved closure; effects on provisional decisions; bounded completion while offline	Each admissibility premise of T11′ is needed; deferral trades availability for safety
MC-4 M15 (12 reported cases, each a case analysis over 24 combinations)	B with full decoding manifest; I-RDFC (assumed)	A always; B with top-level or transitive manifest; I-JCS always	Decoder semantics must be bound for profile B
MC-5 M8/T5 (4 cases)	honest timestamp key with timestamp bound; compromised timestamp key with **status-object expiry checked and status bound**	compromised key with timestamp-only check; compromised key with long-lived status object and timestamp bound	Freshness is bounded by the status-bearing object's lifetime (T5)

Counts in this table refer to reported case identifiers; MC-4 is the case analysis summarised under T10.

Limits.

- Bounded instances; abstract cryptography; one artefact (MC-1, MC-2); time in ticks.
- MC-3 abstracts isolation and key policy as transition guards.
- MC-4 is a case analysis that assumes I-RDFC soundness.
- MC-5 has one compromised role.
- MC-3 V6 abstracts the threshold scheme as a co-signing guard over an idealised signature, and assumes deterministic replay. MC-6 is a case analysis over a single decision, with cryptography as an ideal predicate and a finite deadline, not unbounded liveness.
- OBJ-1 roles and thresholds, OBJ-9 traces and disclosure regimes are not model-checked (T17–T19 are proof sketches).

S3.5 Experiments against harness v0.3.0 (X1–X14)

#	Question	Result	Consequence
X1	Pre-scan soundness	`@graph` container, scoped list term, LFSC path without list terms: blank nodes accepted	M18 redesign (T9.2)
X2	Transitive context not listed by LFSC	caught at stage 2; commitment lists top-level only	T2; VER-16
X3	Parser differences (argument)	T16	VER-14
X4	Envelope rewrite with profile A	*restricted*, signature valid	T10
X5	Lists with repeated values	F5 matrix (T9.3); unbounded 1.1 s for 200 equal values	F5
X6	PCR 23 reset and replay	accepted	T11 (V1)
X7	Profile B with a substituted transitive context	signature valid, `y#Restricted`	T10
X8	Inner JCS proof with a rewritten envelope	inner proof valid, *restricted*	T10
X9	Five further pre-scan bypasses	accepted	T9.2
X10	Issuer-key binding; `verifyFromQuads`	another issuer's key accepted; arbitrary `did:key` accepted	T2; VER-15
X11	Oversized `proofValue` before signature check	528 ms at 60,000 characters	T3
X12	Context embedded in the document data (reproduced)	the loader requests A and B; changing B turns `<urn:public>` into `<urn:restricted>` with A and the document unchanged	dep (S3.1.2); VER-16
X13	Warm cache with substituted context bytes (reproduced)	honest/empty cache: valid; substituted/warm cache: **valid, cached**; substituted/empty cache: rejected (context digest mismatch)	PRF-5: cache the immutable authenticated view and scope reuse to its dependencies
X14	Independent replication (Node 22.23.5)	`npm test`: 20 pass, 1 skip (no swtpm); v4 model check: 32 as expected; F5 matrix reproduced on both stacks; isolated guard mutations as in S3.4 and the article, §5.3–5.4	F5 independently reproduced; MC changes in v5

Corroboration for F5. Fedify issue #824 (25 June 2026, now closed) reports "Maximum deep iterations exceeded (6)" in outbound Linked Data signing. Its comments do not identify RDF lists as the cause. It is a related symptom, **not** corroboration of the cause.

S3.6 Traceability

Objective	Theorems	Premises	Requirements (S2)
OBJ-1	T17	scoped delegation, thresholds, rotation, revocation, durable state	GOV-1, GOV-2, GOV-3, GOV-7, KEY-1, KEY-4, DST-3, DST-11
OBJ-2	T1, T1.1, T2, T10	honest Issue (or signed-commitment scope); issuer-key binding; $\mathbb{J}$; processing-aware closure; use binding; complete decoding manifest	VER-1, VER-2, VER-3, VER-6, VER-11, VER-13, VER-14, VER-15, VER-16, DST-1, DST-9, DST-12, PRD-3
OBJ-3	T4, T5	honest status object with Δ_status; trusted time with ε; fail-closed	DST-2, DST-5, DST-6, DST-11, DEC-5, VER-9
OBJ-4	T6	metadata and target high-water marks; permanent withdrawal; authorised downgrade; durability	DST-3, DST-7, M14
OBJ-5	T7	intersecting witness quorums; conflict relation; monitor	DST-4, DST-13
OBJ-6	T14	reconstructible decision-relevant state	DEC-1, DEC-6, VER-13
OBJ-7	T3, T8, T9	bounded pre-authentication decoding; cost model; deterministic metering semantics; certified admission grammar	PRF-1, PRF-5, VER-1, VER-4, VER-5, PRD-5, PRD-10
OBJ-8	T13, T19; PoS P2	scoped marking or closed world; transformation and disclosure rules; sufficient evidence with independent coverage for compliance	VER-6, VER-7, VER-8, VER-11, VER-13, DEC-8, DEC-9
OBJ-9	T18	no egress; credential-independent fetch schedules; declared leakage	VER-3, DST-10, PRV-5
OBJ-10	T8, T11, T11′, T14, T15, T19	determinism; archive; isolated engine with exclusive key, or threshold key with replaying co-signers; certification time; committed evidence graph	DEC-1, DEC-2, DEC-3, DEC-6, DEC-8 to DEC-15, DST-8, MON-2, MON-5, MON-6, MON-7
OBJ-11	T4 (trade-off)	declared envelope	PRF-6, DST-8

Open.

- T17, T18 and T19 are proof sketches with no model check.
- T11′ is model-checked only with the threshold scheme abstracted (MC-3 V6, V6x, V6c; MC-6); no threshold implementation exists yet.
- M31 has a synthetic worked example, WE-7, with a toy coverage profile and 23 scenarios. A legally and operationally adequate coverage profile for a real deployment, independent trust administration, and a complete archive service remain unestablished.
- The admission certificate (PRD-10) has no specified checker.
- M13 (private index audit) has no protocol.
- Presentation unlinkability (PRV-1) needs a disclosure scheme that supports presentation-specific identifiers.