



What we claim, and how sure we are

Aotearoa New Zealand · © John Stroh

Version 0.5 · revised 21 October 2026 · [this version as a PDF](#)

Everything this work asserts, with a grade on each: what follows from standard results, what is stated precisely enough to be tested but is not proved here, and what is an open question we have not answered. Every claim says what would show it to be wrong.

Claims register · Version 0.5 · October 2026

The claim

If the only copy of what your AI systems did is the one you hold, nobody outside your company can check it.

An organisation running AI agents can meet every record-keeping duty in force and still be unable to show anybody else what its agents did. The instruments that change that have been standard for twenty-five years, and we have not found them in use for this purpose. This work sets out what using them takes: built, measured once, and written up as a draft standard with its unsettled questions listed.

How to read the grades

GRADE	MEANS
Established	Follows from standard results. No novelty is claimed; the contribution is the composition.
Conjectured	Stated precisely enough to be tested. Not proved here.
Open	A question this work poses and does not answer.

A claim that depends on a conjecture cannot itself be established. E5 is absent from the established list below for that reason: it was reclassified as C4 on 12 September 2026.

What is established

	CLAIM	WHAT WOULD SHOW IT WRONG
E1	The substrate is a filtration: the past is annotatable, never editable, and a detector at any moment sees everything sealed before it.	A construction that alters a sealed record without breaking the chain.
E2	Commit-and-reveal inherits its unpredictability from the external clock alone. Without one, the committing party controls the order and the proof collapses.	A commitment scheme that is unpredictable to its own author without an outside time source.
E3	The batch interval Δ is a parameter of the evidence, not of cost. Every statistic carries an attestation grade.	Showing that events inside one batch can be independently ordered.
E4	The null-source trichotomy — learned, authored, randomised — is exhaustive for detectors whose inputs are the sealed history, the mandate and the seed.	A fourth kind of null model built from those inputs.
E6	A single fixed examination configuration can be driven towards zero detection by an adversary who can identify it. A mixture over a closed set has a floor.	A fixed configuration with a floor, or a mixture without one.
E7	Deriving a separate stream per component gives independence to an observer and joint reproducibility to an auditor.	Recovering one component's stream from another's observed behaviour.
E8	Among point processes of a given rate, homogeneous Poisson timing is the maximum-entropy choice: an observer learns least from it.	A timing distribution at the same rate from which an observer can infer more about the active configuration.

What is conjectured

	CLAIM	WHAT WOULD SETTLE IT
C1	A learned null's detection power degrades at least linearly in the fraction of history that is already compromised.	Proof or refutation for a specific estimator.
C2	The eight pattern types span the patterns definable over the substrate with first-order statistics on labels.	A pattern outside the basis. This is the load-bearing conjecture; C4 rests on it.
C3	At governance scale a near-uniform mixture is within a small constant of the optimal one.	Computing the optimum for a real rule set.
C4	The residual — what no detector can see — is exactly the set of acts the written authority permits, which is to say mandate error.	Downgraded from established, 12 September 2026. The identity holds only if C2 does. What is not in question is the weaker half — no detector flags an act the written authority permits, because there is nothing to flag against. Whether that set is <i>exactly</i> the residual depends on C2.

What is open

	QUESTION
O1	Compute the detection floor for a concrete rule set and adversary class. It is a number rather than a principle, and we have not produced one.
O2	Quantify what the content of released findings leaks about the active configuration.
O3	The practical cost of spreading a small examination budget across detectors.
O4	Whether C2 holds. If it does not, the basis needs extending and C4 goes with it.
O5	The recognition rate enters the detection probability multiplicatively and is unmodelled in the formal work.
OQ-15	What does any of this detect when an agent has no owner? Recorded 12 September 2026. The external clock and the unpredictable seed survive an ownerless agent. Sealer isolation survives only where there is an estate to run one. The fourth — a written authority somebody answers for — does not exist, and with no mandate the residual is everything.

The one measurement, what it shows and what it does not establish

Recognition rate q : the probability that a reviewer handed a record containing an improper act says so. It matters because sampling and recognition **multiply** — the chance of missing all n improper acts is $(1 - s \cdot q)^n$ for sampled share s , so a perfect sampling regime staffed by reviewers who do not recognise what they see catches almost nothing.

Measured once: **100, 92 and 58 per cent** across three levels of disguise, with no false alarms in thirty-six judgements on clean records. The records, both answer keys and the scoring code are published as a bundle.

It does not establish a threshold. The reviewers were machine reviewers; human recognition is unmeasured, and that is the largest open item in this work. Twenty-five units, with five planted breaches at the blatant level and four at each of the others after withdrawal, gives a magnitude — that recognition is neither near one nor near zero — and nothing more. The units are synthetic.

Two units planted as breaches were **withdrawn during scoring**. All three reviewers rejected them independently; on examination the units did not contain what the key said they contained. That moved two figures up, from 73 and 47 per cent. Both keys are in the bundle, so the correction can be checked.

The standard is a draft and says so

MIO-STD-01 is published at working-draft status. Its clause 9 lists what must be settled before it could be submitted anywhere:

1. q has been measured once, for model readers only — not enough to set a threshold.
2. The exhaustiveness conjecture (C2) is open.
3. The delta over PeerReview must be stated precisely.
4. No attestation authority is named, and the standing of a foreign one is unexamined.
5. The intellectual-property declaration differs between standards regimes.
6. Alignment with ISO/IEC 24970 is unexamined.

What is running, and what is not

An implementation of this draft standard runs in production. Its position against the standard is set out below in both directions. On 21 October 2026 every row was checked against the platform's code and the live service by AI agents working to the author's instruction, several of them auditors that had not written the rows. No person has audited the rows. AI agents made earlier checks on 12 and 15 September and on 12 and 19 October 2026; the 15 September check found this page had reported the receipt attestation as undeployed after it had been deployed. The four properties these rows bear on (sovereign, sealed, signed, timestamped) are set out with their limits on the record.

What is running.

The Governance API is live	Eighteen routes, under scope-based authentication, at <code>mysovereignty.digital</code> : sealed records, segments, sealing and receipt verification, mandates and the agent inventory, verdicts, attestation of any digest you supply, a committed-seed lookup, usage and policy. A second product of ours, an AI-visibility auditor, seals its reports through it.
Every record is hashed at emission and chained	Across forty-five record types, with a per-tenant Ed25519 signature over a canonical payload.
Receipts are signed fail-closed	A receipt is never issued unsigned, and never without an explicit, signed label naming which authority reached the verdict. That label is about the decision, not about the time.
Records are anchored to an outside authority, on a daily sweep	The sweep submits record digests to an RFC 3161 authority and stores the returned token whole, with its certificate chain, so a third party verifies it with <code>openssl ts -verify</code> and never has to contact us. The sweep stamps up to about fifty records per Village per night. For ordinary records the sweep is the only stamp, applied after the fact; for receipts it is the retry path behind the attestation taken at the moment of issue.
A receipt's time is attested by an outside authority at the moment it is issued	Every receipt carries a signed <code>time_attestation</code> block. Before the receipt is signed, its canonical digest is sent to an RFC 3161 authority; the attestation that comes back is included in what the tenant signature covers, so it cannot be added or altered afterwards — grade A (stamped before signing). If the authority does not answer inside the request window, the block records that instead, with the reason, and says so on its face — grade O (the authority did not answer in time). Note: a grade-O receipt stays grade O. The block is inside the signature, so the nightly sweep stamps it separately and the receipt in a holder's hands is unchanged. The field <code>decided_at</code> remains our own clock and is not the attested time. Committed 12 September 2026; running in production since 14 September. The authority behind every receipt by default is Certum (Poland). Its stamp is evidence of the time and carries no legal presumption (see below). Where a customer has the qualified second stamp switched on (Aruba PEC, Italy), the receipt carries that stamp as well — see the shortfall below.
Mandates and an agent inventory can be recorded, sealed and looked up	A tenant's key with the right scope can write a mandate for an agent (the actions it is permitted to perform; a later mandate supersedes an earlier one) and an agent-inventory entry (the agent's reference and the identity it is bound to). Each is a signed, hash-chained record with its own receipt, so anyone holding a saved copy can check it offline. A lookup returns the

	mandate in force for an agent at a given time, and the inventory lists its entries. Alongside these, scoped API keys record who minted them, what scopes they carry, when they expire, and when they were revoked or last used, and every proof-chain entry records the key, the signer and what evaluated the policy. This records which credential acted. An Email Decision Record receipt can name the mandate in force for the agent that proposed the message; a separate switch, which is not set on either production server, also refuses a send that mandate does not permit. See the shortfall below for what these records do not yet show.
The verifier, and now the platform, can check a signer named by a did:x509 identity against a trust anchor	In plain terms: it checks that a record was signed by an organisation whose certificate chains to an authority you have chosen to trust. Given a certificate chain in the bundle (<code>verification_certificates</code>), <code>--trust-anchor</code> (a PEM or DER file, repeatable) and <code>--at</code> (an ISO time with zone), the downloadable <code>verify-record.js</code> checks that the signer's identity chains to an anchor you chose, at a time you chose. With no anchor it prints <i>NOT PROVED: who issued it</i>. Since 20 October 2026 the platform does the same: a Village owner can set up to eight trust anchors, and a sealed record signed by a did:x509 identity is accepted only if its chain ends at one of them. Neither checks revocation. The time is the checker's, so neither shows when the signature was made. The platform does not yet sign with certificates itself. did:x509 is an individual IETF draft (draft-birkholz-did-x509-03, on the Independent Submission stream) that no working group has adopted. The code passes all 220 of Microsoft's published conformance vectors, 65 of which fall within the draft's scope; the rest exercise Microsoft's later additions. AI agents built it. Separate AI reviewers examined the library and verifier over ten rounds, and the platform part over further rounds. No person has audited it.
Post-quantum signing is built into the production code, behind a switch that is off	ML-DSA-65 (FIPS 204), a signature scheme designed to withstand future quantum computers, can add a second signature, bound to the Ed25519 one, on sealed custodian records, governance receipts, Email Decision Record receipts and proof-chain summaries. The code is in production behind a platform switch and a per-tenant switch. The platform switch is not set on either production server, so no record carries one today. The downloadable verifier already checks such a signature, and <code>--require-pq</code> makes it mandatory. AI agents built and reviewed it; no person has audited it.

A shipment's records can be checked offline

`consignment-verifier.js` checks a Provenance record (the custody trail) and an Email Decision Record together, each with its own unchanged checker pinned by hash, and reports whether a signed email names the consignment. It does not say that anything is lawful or complete. AI agents built and reviewed it over four rounds; no person has audited it.

What is not. Most are shortfalls against a clause of the standard; the last is research that is not part of the service.

Three timestamp authorities are in use; the standard's jurisdiction test is not yet met

The standard asks for at least two attestation authorities in distinct jurisdictions (5.3.1), neither compellable together under one instrument or mutual-assistance mechanism (5.3.2). In use: Certum (Poland) stamps every receipt and the daily record sweep; it is free and not eIDAS-qualified, so it supplies the evidence rather than a statutory presumption. Portugal's state authority (Cartão de Cidadão) stamps the daily estate root alongside Certum on both production servers; the code stores that stamp as not qualified. Aruba PEC (Italy, eIDAS-qualified) is under contract and its client is deployed behind a per-customer switch that we turn on at the customer's request (it is off on both production servers): where it is on, a receipt carries the Aruba stamp as well; where it is off, no receipt carries a qualified stamp. Separately, on 15 September 2026 Aruba attested the published record set directly — one token, granted under ETSI policy 0.4.0.2023.1.1, published with the records and checkable by anyone ([check the record yourself](#)). The Email Decision Record and Provenance checkers bundle the Certum, Portugal and Aruba PEC roots (list version 2026-10-11.1, in production since 11 October 2026); a holder can also run `openssl ts -verify` on any token, since each is stored whole with its chain. Either way nobody has to contact us. **Against the standard:** 5.3.1 is met by the estate root. 5.3.2 is not met by any pair in hand: Poland, Italy and Portugal are all EU member states under one mutual-assistance mechanism, and a receipt says so on its face. A New Zealand authority is what would answer 5.3.2; no qualified service operating under New Zealand law has been found. An approach to one New Zealand provider was sent on 1 October 2026 and no reply is recorded; the search for a New Zealand authority is described on our Provenance pages.

The batch interval was not set against the conduct it attests

The standard says the interval is a parameter of the evidence. The implementation did not set its interval against the conduct it must attest, and would have to. Money is no longer the reason: the stamping authority is free, and each receipt is stamped on its own. The nightly sweep is still capped to stay within that free service's rate tolerance.

Record signing has moved to a separate host and can be split between custodians; no tenant's key is split between parties yet

What exists, deployed. Since 20 October 2026 the record-signing key (`proof_signing`) of every active tenant is held by a separate signing service on its own host, reached over mutually authenticated TLS; one suspended demonstration tenant was left out. If the service is not configured or cannot be reached, signing is refused rather than falling back to a local key (since 4 October 2026 for an unreachable service,

and since 6 October for an unconfigured one). The old local keys are retired and kept; none has been destroyed. Split-key signing is built through the platform and deployed: FROST (RFC 9591), a scheme in which several parties each hold part of a signing key and none ever holds the whole of it, two of three, with the key generated by a ceremony in which no single machine ever holds the whole key (signed rounds, encrypted shares, a verifiable transcript, and each host seals its own share only after its operator confirms the group key and the transcript hash). Each custodian host runs its own service over mutual TLS, accepts requests only from an allowlisted coordinator, checks what it is asked to sign, and refuses a record that rolls back or forks a stream. A tenant set to threshold signing signs its Email Decision Record purposes with the group key and keeps its ordinary signer for everything else; its DID document lists the custodians (names, roles and verifying shares, no addresses) while the custodians are registered, and the result is an ordinary Ed25519 signature that the shipped verifier accepts. Registration at boot fails closed, and a build check fails if any code rebuilds the whole key. **Where it stops.** Only the record-signing key has moved. There is one signing host. A tenant's published DID document still lists the retired key as valid. No tenant is on split-key signing yet: the custodian hosts, share storage and a ceremony with real parties do not exist, so no tenant's key is held in shares by named parties today. A key generated this way is generated jointly by the named parties, each holding one share; how separate those parties are depends on who runs each custodian's host. We still hold the sealer, seed, inventory and observer keys, and the two routes specified for the seed, split custody or derivation from a source nobody controls, remain to be built.

Mandates and the inventory do not yet record lineage, expiry or revocation

A mandate names its agent and its permitted actions; it has no validity window and no revocation, only supersession by a later mandate. An inventory entry names an agent identity and what it is bound to, not which agent instantiated which. So the record does not yet show under whose written authority an agent was created, or when that authority lapses. The answer to "which mandate was in force" also rests on a record time the platform wrote: the ordering is not yet tamper-evident on the default signing path (MIO-STD-01 4.3), and the API returns that caveat with every answer that finds a mandate. The Email Decision Record also keeps a live register in which an agent can be suspended or offboarded; that register is configuration, not a sealed record.

Not every receipt gets its outside time stamp, and the default stamp has no legal presumption behind it	The attestation is attempted on every receipt, inside the request. It is not guaranteed: if the authority does not answer inside the window the receipt is issued grade O — labelled as such on its face, and retried by the nightly sweep, but at that moment its time rests on our word alone. We do not currently publish what proportion of receipts are grade A. That figure is the measure of this claim, and until it is published a reader should assume the weaker case. And a grade-A stamp by default comes from a free authority carrying no statutory presumption, so the strongest thing a receipt proves about its time is evidential, not legal, unless the customer has the qualified second stamp on.
The unpredictable examination regime is not running	It depends on the seed custody above. The code to take a seed from a public randomness beacon, and a route to look up a committed seed, exist; nothing yet writes a committed seed.
A zero-knowledge proof that a certificate chains to a root key has been generated and checked in research prototypes	A proof that a certificate chains to a given root key, is in date, and that the prover holds the certificate's key, without showing the certificate, has been generated and checked in research prototypes on two third-party proving systems, for a fixed certificate profile. They run on our own machine. On 19 October 2026 an AI agent placed a small record-keeping contract on a public test network belonging to one of those systems and wrote one entry of test data to it. The contract checks no proof and holds no customer data. Nothing runs on any production blockchain. The prototypes are unaudited and not part of the service, and AI agents were the only reviewers.

Where to go next

The argument for why this matters: [Cheaper not to look](#). What catches an agent: [What it takes](#). The evidence: [What we know and what we don't](#). How to build it: [How to build it](#). The formal statements: [Addendum M](#). The standard: [MIO-STD-01](#).

Drafted with AI assistance. The September text was checked and revised by the author. The October revisions were made by AI agents to the author's instruction, and no person has read them line by line.

►DISCLAIMER

How this was made. The version number counts drafts of the text. It does not measure the inquiry behind it, which has run over days and across several AI systems, with argument between those systems and within them, directed, refused and repeatedly redirected by the author. The source material was AI-generated, and then adversarially and iteratively refined across a range of tools — systems built by different companies in different jurisdictions, set against each other and against the author. No one of them produced this text, and no one of them reviewed it alone. **The plurality is deliberate rather than incidental.** A single model carries a single set of priors about which sources are authoritative, and this series argues that an evidence base narrowed in exactly that way is how a contested question comes to look settled. Using one model to investigate that claim would have been the claim refuting itself. To name a single model on it would credit that model with work that was neither its own nor done in a single pass. **The plurality was also necessary, and the record should say why.** In drafting, the assisting model repeatedly led with United States institutional sources — a national laboratory, an industry association, a market study nineteen years old — and presented conclusions drawn from them as the state of knowledge. On one occasion European measured data contradicting those conclusions was present in the same research return and was placed below them. Framings were proposed that would have argued against this series' own position using that evidence base, and offered as rigour. Each was refused by the author and the material rebuilt. That is the mechanism these documents describe, occurring in their own making, and it is recorded because a series arguing that evidence bases narrow without anyone deciding to narrow them cannot credibly claim its own production was exempt. The framing, the corrections and the judgements are the author's, and so are the errors. [How this site is written](#) sets out what is declared on every piece, who checks it, and where the per-piece record lives.

Cite this version

What we claim, and how sure we are, version 0.5 (October 2026).
`agenticgovernance.digital/downloads/what-we-claim-v0-5.pdf`

Licensed CC BY 4.0. Reuse is free, including commercially, provided the author is credited and the reuse does not suggest that the author endorses it. This link is frozen at version 0.5; [/what-we-claim](#) always shows the current one.

PREVIOUS

How to build it

Series contents

NEXT

What was visible at the time