Nick Clark · Published July 23, 2026 · PDF

Cross-Receipt Equivocation Detection: Catching an Agent That Tells Two Stories

Origin-equivalence normalization charges a semantic agent's refusal counter once per class, and that class is derived only from the agent's own append-only lineage field and the counterparty identity records it holds. An encounter receipt lets a third agent take in evidence for that derivation from an interaction it never observed, but the three-conjunct verification is stated over one presented receipt, so two attestations recording a single encounter and different determination classes can each be written. Cross-receipt equivocation detection, disclosed in U.S. Provisional Application No. 64/117,812 at Section 10.4, paragraph [0384], closes that opening inside the verifier's own holdings. Where two held attestations record an identical pair of identity primitives, an identical pair of epoch identifiers, and an identical scope-partition identifier yet record determination-class identifiers naming different members of the closed set, the two are irreconcilable, one encounter having produced one determination. The verifier appends a cross-receipt equivocation record, writes a withdrawn attribute into each attestation, and stops consuming either in the origin-equivalence class derivation, appending nothing adverse to any party.

1. The Gap

Origin-equivalence normalization charges the refusal counter (304) of a semantic agent (100) per origin-equivalence class (200) rather than per conduct evaluation artifact (116). Under Section 2.2 the agent evaluates each declared relation type retrieved from the signed policy object (112) in force between the present asserting party (118) and each party already assigned to a class, the present party joining that class where any declared type is evidenced and taking a new class where none is.

That evaluation has a hard boundary. Paragraph [0063] confines it to entries of the agent's own append-only lineage field (104) and to counterparty identity records (114) it holds, without reference to a centralized registry, without query to a directory, without participation in a consensus procedure, and without coordination with a further execution node. Locality is the point: unlike a Sybil-resistance scheme resolving identity against a shared registry, the class is computed from the agent's own records. The same boundary sets the limit, since an empty enumeration of declared relation types assigns every asserting party to a distinct class and restores the condition of Section 2.1, under which volume alone drives the authorization gate (300) to the withheld state (310).

Section 10.4 supplies material for that derivation which the verifier did not itself record. Under [0380], upon producing a determination of the closed set for an artifact received from an asserting party, the agent computes an encounter commitment over an ordered tuple of exactly four members and no other member: its own epoch identifier drawn from its lineage field (104), the counterparty epoch identifier attested in the artifact, a determination-class identifier naming exactly one of the four classes (122, 124, 126, 128), and a scope-partition identifier, with a retained opening value. The counterparty returns a counter-commitment over the same four members with the epoch roles exchanged, and the paired commitments are the encounter receipt, appended by both. The tuple admits no conduct descriptor, so an opening discloses no characterization of conduct.

Either party alone may later present that receipt to a third semantic agent (100) that was not a party to the recorded interaction. Paragraph [0381] has that agent run three conjuncts: both signatures verify against an identity primitive of each party and the binding verifies; each carried epoch identifier is a validated successor of a prior slope position the verifier holds in its counterparty identity record (114); and both commitments, opened from both opening values, recover one and the same determination class agreeing with the cleartext borne. Only on all three does the verifier write an encounter attestation into the counterparty identity record (114) of the presenting party.

Each conjunct is stated over a single presented receipt. Two attestations can therefore sit in one verifier's records, both properly written, while recording one encounter and different determination classes. The per-receipt test compares no attestation against another, and the construction reaches no registry, directory, or further execution node holding a competing record of that encounter.

2. Mechanism

Paragraph [0384] directs the third semantic agent (100) to compare the encounter attestations it holds against one another. The comparison is possible because of what an attestation records: a pair of identity primitives, a pair of epoch identifiers, a scope-partition identifier, and a determination-class identifier.

Three of those four are tested for identity and the fourth for difference. Where two attestations record an identical pair of identity primitives, an identical pair of epoch identifiers, and an identical scope-partition identifier, yet record determination-class identifiers naming different members of the closed set, the two are irreconcilable, one encounter having produced one determination. The matched elements say which two parties, at which recorded chain positions, within which scope partition. What is left

over is the determination itself, drawn from a closed set of four in which the admission evaluator (120) produces exactly one member per admitted artifact and no scalar confidence, probability, or graded weight in place of one.

Having found such a pair, the verifier does three things. It appends a cross-receipt equivocation record naming both attestations, both receipt lineage entries, the epoch pair, and the partition. It writes a withdrawn attribute into each attestation, neither attestation modified. And it ceases consuming either in the origin-equivalence class (200) derivation prospectively.

The third consequence is bounded by what the attestation was admitted for to begin with. Under [0381] it is evidence of origin-equivalence class derivation and nothing else: not a conduct evaluation artifact (116), not passed to the admission evaluator (120), modifying no scoped integrity vector (106). Ceasing to consume it removes an input to that derivation alone.

Both records are withdrawn, not one judged false. The withdrawn attribute goes into each of the two attestations. The filing prescribes no adjudication of which record is truthful and gives no ground for preferring one over the other, and the agent performing the comparison is, under [0381], one that was not a party to the recorded interaction.

Nothing adverse is appended. Paragraph [0384] states that the detection appends nothing adverse to any party beyond ceasing consumption. The filed response is exhausted by the equivocation record, the two attributes, and the prospective cessation.

Locality carries through to the detection itself, which [0384] states runs entirely over material the verifier already holds, without registry, directory, or coordination with any other execution node. That is the condition [0063] imposes on the derivation the attestation feeds, so the check adds no external dependency.

3. Operating Parameters

For the detection itself the filing declares no threshold, no tolerance, and no policy-declared quantity. The predicate is exact identity on three recorded elements together with difference on the fourth. Nothing is scored, and no count must accumulate before the finding.

Two structural facts bound the test: the committed tuple of [0380] has exactly four members and no other member, and the determination classes form the closed set of Section 1.8, being the accepted determination (122), the rejected determination (124), the not-determinable determination (126), and the not-applicable determination (128), with exactly one produced per admitted artifact. Withdrawal operates prospectively.

Policy-declared quantities do appear in the surrounding construction. Under [0385] the third agent decrements its own authorization budget (404) by a receipt-presentation decrement, taken from the signed policy object (112), responsive to writing an encounter attestation, metered per presenting origin-equivalence class (200) computed by the ordinary derivation over material the verifier already carries. A receipt-presentation register records, per class per window, whether that class has borne a decrement; a first write for an unrecorded class applies the decrement and updates the register, and each further write for a recorded class writes the attestation with no further decrement. Decrement units equal the units gating action dispatch and the issuance decrement, receipt of a presentation is free, and a party presenting from one origin cannot exhaust the budget (404) whatever the volume.

Paragraph [0386] coordinates that decrement with the first-encounter budget decrement: where one presenting party in one encounter occasions both and both meterings resolve to a common class within a common window, exactly one decrement is applied rather than two, and a coordination record names both registers, the class, the window, the decrement applied, and the register whose decrement was withheld.

The declared relation types the derivation consumes are enumerated in that same policy object (112), and for none of these quantities does the filing state a numeric value, a latency, or a bound.

4. Composition

This mechanism is the cross-artifact complement to a per-artifact test. Paragraph [0381] establishes that a presented receipt verifies; [0384] establishes that two held attestations are mutually consistent. Neither subsumes the other.

Its remedial primitive is shared with two neighboring paragraphs. Under [0382], a verifier holding no prior slope position for a party evaluates the second conjunct against predecessor epoch identifiers carried within the receipt, produces the not-determinable class (126) with respect to that conjunct rather than the rejected class (124), and writes the attestation bearing an unverified-slope attribute, consumed in the derivation on identical terms with no weight applied; on later acquiring a prior slope position it re-evaluates and either clears the attribute or writes a withdrawn attribute removing the attestation from consumption prospectively without modifying it. Under [0383], a party disowns a receipt bearing its own identity primitive by emitting a receipt repudiation observation on a ground of loss of control of that primitive or its supersession on substrate migration, and the verifier writes the same attribute into each held attestation recording that receipt. Three routes thus reach one attribute: later knowledge of a slope, the subject party's own disownment, and the verifier's cross-comparison. Each acts prospectively and leaves the attestation itself unmodified.

Timing matters downstream. Assignment of an asserting party to an origin-equivalence class (200) is, under Section 2.2, persistent and not window-scoped. Cessation of consumption is prospective, and the filing states no re-derivation of assignments already appended on the strength of a withdrawal.

A sibling check sits at Section 10.10. Under [0477], a receiving party observing two distinct valid successors of a single recorded epoch identifier appends a lineage-discontinuity record naming that epoch identifier, both presented successors, the presenting parties, and the recorded times of presentation. It declines to advance the locally retained slope position for either presenting party until exactly one successor is corroborated, denies neither on the ground of the discontinuity alone, and produces the not-determinable determination (126) where the discontinuity prevents resolution. The two key on different collisions, one encounter carrying two determinations against one epoch carrying two successors, each acted upon by the observing agent alone.

5. Prior-Art Distinction

Equivocation is a well-studied failure, and several established families address it. What separates them here is architectural precondition rather than result.

Transparency logs and split-view detection. Certificate transparency, key transparency, and comparable append-only log designs catch an operator serving inconsistent views by requiring a published log, inclusion and consistency proofs against it, and independent monitors that gossip what they saw. Detection is a property of surrounding infrastructure. Paragraph [0384] involves no log, no operator, and no gossip: a verifier compares attestations already in its own counterparty identity records (114), and the finding binds nothing beyond that verifier's own derivation.

Accountable Byzantine agreement and double-signing penalties. Accountable consensus protocols and proof-of-stake slashing catch a participant that signs two conflicting messages for one protocol position, then exclude it or take its stake. Each presupposes a common protocol view, a quorum, and an enforcement mechanism with authority over membership or stake. The filed detection presupposes none of the three and stops consuming two records rather than imposing a consequence.

Reputation and trust-graph systems. These resist inconsistent testimony by aggregating reports across raters and weighting by rater history, yielding a portable score. Under [0380] the committed tuple admits no conduct descriptor, the determination-class identifier names one of four classes and nothing further, and [0064] states that a class assignment is local to the deriving agent and confers no portable standing. There is accordingly nothing portable to aggregate.

Directory-based revocation. Revocation lists and status-response protocols withdraw trust in a credential by publishing a revocation for relying parties to fetch. A withdrawn attribute under [0384] is published nowhere, fetched by no one, and authoritative for no agent but the one that wrote it, on that agent's own comparison rather than a publisher's instruction.

None of the foregoing asserts anything about any product, service, or party, and no comparison here states that any implementation practices the disclosed subject matter.

6. Disclosure Scope

Cross-receipt equivocation detection is disclosed in U.S. Provisional Application No. 64/117,812, Section 10.4, paragraph [0384]. Supporting disclosure appears at [0380] for the content-blind encounter commitment and paired receipt, [0381] for the three-conjunct verification and the non-admitted attestation, [0382] for the unverified-slope branch, [0383] for receipt repudiation, [0385] for receipt-presentation metering, and [0386] for decrement coordination. The origin-equivalence class (200) derivation is disclosed at Section 2.2, the closed set of determination classes at Section 1.8, and counterparty-side fork detection at Section 10.10, paragraph [0477].

What is disclosed is a verifying semantic agent that compares encounter attestations it already holds, treats as irreconcilable any two recording an identical pair of identity primitives, an identical pair of epoch identifiers, and an identical scope-partition identifier while recording determination-class identifiers naming different members of

the closed set, appends a cross-receipt equivocation record naming both attestations, both receipt lineage entries, the epoch pair, and the partition, writes a withdrawn attribute into each attestation without modifying either, and ceases consuming either in the origin-equivalence class derivation prospectively, appending nothing adverse to any party beyond that cessation.

What is disclaimed is everything past that. No registry, directory, transparency log, notary, quorum, or coordination with a further execution node is claimed, none being required or consulted. No adjudication of which of two conflicting attestations is truthful is claimed, no determination concerning the conduct of any party, and no penalty attaching to any party by reason of the detection. No particular commitment scheme, hash construction, or signature algorithm is claimed, and no numeric threshold, latency, complexity bound, or benchmark is stated, the filed paragraph declaring none. Nothing here asserts infringement by any party, and this publication is a defensive technical disclosure of subject matter already on file.

Origin-Equivalence Normalization (/origin-equivalence)

All 49 steps → (/inventive-steps)

Metering that can't be gamed by splitting one accuser into many.

Provisional application

PRIMARY TECHNICAL DISCLOSURE

- Origin-Equivalence Normalization: Metering Refusal Per Source, Not Per Assertion (/articles/origin-equivalence/origin-equivalence-normalization)

SECONDARY TECHNICAL

- Content-Blind Encounter Commitment and the Paired Encounter Receipt (/articles/origin-equivalence/content-blind-encounter-commitment)

- **Cross-Receipt Equivocation Detection: Catching an Agent That Tells Two Stories (/articles/origin-equivalence/cross-receipt-equivocation-detection)**
- The Untested-Origin Saturation Bound: Pricing a Flood of Freshly Minted Identities (/articles/origin-equivalence/untested-origin-weighted-saturation-bound)
- Counterparty-Side Fork Detection: Recognizing a Split Identity From Outside (/articles/origin-equivalence/counterparty-side-fork-detection)
- The Attestation Origin-Equivalence Gate: An Anti-Collusion Bound (/articles/origin-equivalence/attestation-anti-collusion-bound)

TERMINOLOGY

- asserting party (/glossary#asserting-party)
- origin-equivalence class (/glossary#origin-equivalence-class)

Origin-Equivalence Normalization overview → (/origin-equivalence)