Whitepaper · Version 1.0 · April 2026 · github.com/KYE-Protocol · **Download PDF (2.3 MB)**

KYE Protocol™: the Entity Authority Protocol for AI Governance

> **KYE Protocol**™ proves *who or what is acting*, *on behalf of whom*, using *which capability*, under *what authority*, *in what state*, and with *what audit trail* — for every action a human, business, AI agent, service, model, tool or workflow takes. The technical and evidentiary foundation for the **KYE Chain of Authority**™ — the linked, attenuating delegation chain over which **Authority Finality**™ is asserted. For AI-agent systems: accountability, compliance, dispute resolution, legally defensible audit trails.

Contents

Abstract

The agentic stack — AI agents, models, tools, workflows acting autonomously on behalf of humans and businesses — is reaching production at a velocity that has outrun the identity, authorization and audit infrastructure beneath it. KYC verifies humans. KYB verifies businesses. KYA (just emerging in 2026 from Visa, Skyfire, Persona, Sumsub, Trulioo) verifies agents. Each layer is siloed; each verifies once, at registration; none answers *"is the answer still true 200 ms from now when the next call arrives?"*

KYE Protocol™ — *Know Your Entity*™ — is the **Entity Authority Protocol for AI governance**. It proves *who or what is acting*, *on behalf of whom*, using *which capability*, under *what authority*, *in what state*, and with *what audit trail*. Every entity (human, business, agent, service, model, tool, workflow) shares one URN format. Every action is bound to a signed delegation chain with attenuable scope. Every decision is recorded with a standardised reason code. Every revocation cascades through the trust graph in milliseconds. Every audit event hash-links to the previous one. Every decision can be exported as a signed proof bundle a regulator can verify with public keys alone. The result: a complete **KYE Chain of Authority**™ over which **Authority Finality**™ holds — replayable proof for accountability, compliance, dispute resolution, and legally defensible audit trails in agentic systems. KYE™ does not replace legal agreements, signatures, or regulatory obligations; it provides the technical and evidentiary foundation for them.

1 · Problem

Modern agentic workflows generate three classes of pain that legacy identity stacks cannot resolve.

1.1 Fragmented identity

One agent typically holds three or four identities at once: a SPIFFE SVID for workload attestation, an OAuth client_id for the API gateway, a vendor-specific KYA passport for payment rails, and a model card for inference governance. Each format is reconcilable only by hand. Auditors reconstructing an incident traverse four logging systems and stitch traces by timestamp.

1.2 Static authorization

OAuth scopes and KYC files describe state at issuance. Neither propagates a revocation. When a delegated agent is compromised, the human delegator may not learn about it until the next compliance review. Stop signals (entity stopped, credential revoked, attestation stale) need to ripple through dependent grants in real time, recursively, with cryptographic guarantees they were applied.

1.3 Unprovable history

Audit logs are usually JSON lines in a search engine. They are searchable but not provable: a malicious operator can edit the past. Regulators increasingly demand cryptographic non-repudiation — an append-only chain whose head hash is publicly committed.

2 · Prior art & gaps

OAuth 2.1 / GNAP

Human authorization, token introspection

No agent identity, no delegation chain, no cascade

SPIFFE / SPIRE

Workload identity (SVIDs)

No first-class delegation, no decision vocabulary

Anthropic MCP

Agent ↔ tool communication

No identity, no policy, no audit

Google A2A / ADK

Agent registration metadata

No delegation, no cascade, no proof

OpenID AuthZEN

Standard decision API shape

No URN, no chain, no signals

OpenSSF SCITT

Transparency receipts

No identity, no scope, no cascade

OpenID SSF / CAEP

Stop-event distribution

No delegation chain, no proof bundle

Visa Trusted Agent / Skyfire / Persona KYA

Agent passport (AID), spend caps (TXG)

Vendor-specific, agent-only, no unified URN, no cascade

Each of these solves a slice. None composes into a single contract that an auditor can read end-to-end. KYE Protocol™ is that contract layer; it does not replace these, it composes them.

3 · Design principles

KYE Protocol™ stands on **16 protocol-core principles** grouped in three tiers: six runtime-governance semantics that define what KYE™ *decides*, eight protocol-design disciplines that define how KYE™ *classifies and ships*, and two developer-adoption disciplines that define how KYE™ *reaches developers*.

3.1 Tier A — Runtime governance (what KYE™ decides)

1. **Authority-first.** KYE™'s centre is authority, not identity. Every action answers *who or what is acting, on behalf of whom, using which capability, under what authority, in what state, with what audit trail*. Core records: `KYEAuthorityGrant`, `KYEAuthorityScope`, `KYEDelegation`, `KYEActingOnBehalfOf`, `KYEAuthorityState`, `KYEAuthorityDecision`, `KYEAuthorityRevocation`, `KYEAuthorityEvidence`.
2. **State-first.** Authority is meaningless without state. Every authorize call composes `entity_state`, `authority_state`, `delegation_state`, `credential_state`, `capability_state`, `payload_state`, `recovery_state`, `risk_state`. KYE™ is state-aware authority infrastructure.
3. **Decision-first.** Runtime systems need an answer. `POST /v1/runtime/authorize` returns one of `allow` / `allow_with_constraints` / `require_approval` / `require_step_up` / `require_human_review` / `require_recovery` / `quarantine` / `deny` — with reason code, matched policies, obligations, evidence refs — in milliseconds.
4. **Evidence-first.** KYE™ turns authority into evidence. Every decision produces a `policy_decision_id` binding to validation results, audit events, transparency receipts, signals, and a signed `evidence_pack` consumable by GRC, auditors, regulators, dispute panels.
5. **Audit-trail-first.** No authority without audit. No audit without evidence. Every material change emits an append-only, hash-linked, timestamped, reason-coded, actor-bound, policy-bound, decision-bound audit event — exportable, replayable, externally verifiable.

3.2 Tier B — Protocol design (how KYE™ classifies and ships)

6. **Schema-first.** JSON Schema 2020-12 with absolute `$id` at `kye-protocol.github.io/schemas`. OpenAPI, SDKs, validators, docs, and conformance tests are derived from schemas, not hand-written.
7. **Dictionary-first.** Canonical vocabulary for entity types, all state dimensions, decisions, reason codes, capability kinds, side-effect levels, data classes, signal types, redaction fields, taxonomies, graph types.

8. **Taxonomy-first.** 16 V1 canonical taxonomies (entity_type, capability_type, action_type, resource_type, data_class, side_effect_level, risk_state, environment, decision, reason_code, evidence_type, compliance_framework, sector, jurisdiction + state taxonomies). Versioned, status-bound, mappable to framework controls.
9. **Metadata-first.** Every KYE™ object carries a normative metadata block — `labels`, `classifications`, `ownership`, `lineage`, `compliance`. Field values draw from registered taxonomies. Metadata *influences decisions*; the runtime exposes it to the policy layer as `request.{actor,capability,resource,authority}.metadata`.
10. **Graph-first.** Authority is relational. Every entity, delegation, capability, credential, policy, state, decision, and evidence object is a node; every authority relationship is a typed edge. Authority Graph™, Decision Map™, Evidence Graph™, Blast Radius Map™, Compliance Map™. Storage substrate is implementation choice.
11. **Profile-first.** Core stays small. 33+ profiles add domain semantics; profiles never modify Core.
12. **Registry-first.** Every object is resolvable. `/.well-known/kye` advertises versions, profiles, crypto suites, JWKS, dictionaries, taxonomies, metadata schemas, registries.
13. **Conformance-first.** 41-fixture black-box pack that any conformant Gateway must pass. Vendor self-attestation via `conformance-report.json`.

3.3 Tier C — Developer adoption (how KYE™ reaches developers)

14. **API-first.** 193 OpenAPI operations across 87+ runtime endpoints across runtime, registry, taxonomy, metadata, and graph endpoints. `POST /v1/runtime/authorize` is the headline; `GET /v1/decisions/{id}/map` returns a Decision Map™.
15. **SDK-first.** TypeScript, Python, Go SDKs ship with schema types, local validators, decision clients, signing/verification helpers, policy adapters, audit emitters, evidence-pack builders, taxonomy resolvers, metadata classifiers, graph traversal clients, decision-map renderers.

Protocol-design corollaries that fall out of the 16 principles:

- *Delegation is a graph, not a flag.*
- *Scope must attenuate, not extend.*
- *Revocation and quarantine signals fan out before the next request.*
- *Profiles, not forks.*
- *Mechanism designs sit behind the patent track* (decision evaluation, audit-record linking, signal propagation, scope attenuation, signing-suite construction, graph traversal).

4 · Conceptual model

KYE™ governs acting entities, principal entities, capability entities, resource entities, credential entities, and evidence artefacts — and keeps them strictly distinct. Payloads are *evidence artefacts*, never authority-bearing entities; the **KYE™ Payload Trust Profile™** binds signed payloads to entity authority, capability state, policy decisions, and replayable audit evidence.

KYE Protocol™ defines nine first-class records:

- `Entity` — any actor or resource. Has a status, a lifecycle state, an immutable block, classification, assurance, optional sectoral profile.
- `Delegation` — a record that *actor* may act for *subject*, granted by *delegator*, within *scope*, for *allowed_actions*.
- `Scope` — named bundle of constraints + obligations. Attenuable through `parent_scope_id`.
- `AccessRight` — fine-grained, resource-level grant.
- `Credential` — signed assertion about an entity. Issuer + holder + subject + claims + Ed25519 signature.
- `Attestation` — workload identity binding (SPIFFE / EAT / build provenance).
- `Signal` — reactive event on the bus. Stop / quarantine / revoke / cascade.
- `PolicyDecision` — record of an authorize call: decision, reasons, obligations, stop_conditions.
- `AuditEvent` — append-only, hash-linked entry whose construction algorithm is not disclosed publicly.

Two derived records support proof & observability:

- `ProofBundle` — signed export of decision + supporting audit events. Verifiable by external auditors with public keys alone.
- `TransparencyReceipt` — signed receipt that a statement is included in the public log at a given index.

4.1 Six-dimension state model

An entity is not a single state. KYE Protocol™ runtime evaluates authorization against six independent state dimensions:

- `entity_state` — lifecycle of the actor itself (provisional, active, suspended, quarantined, stopped, tombstoned).

- `authority_state` — whether the actor currently holds any active authority (none, scoped, elevated, break_glass, frozen).
- `delegation_state` — integrity of the chain of delegations behind the actor (active, parent_revoked, self_revoked, expired, scope_violated).
- `credential_state` — verifiable credentials held by the actor (none, valid, expired, revoked, signature_invalid).
- `recovery_state` — whether the actor is healthy, mid-recovery, post-compromise, or rotated.
- `risk_state` — telemetry-derived posture that informs sPDP without changing identity (nominal, elevated, watch, denylisted).

Each dimension transitions independently and emits its own signal class. “Can this actor act now?” is a single composed test against the six dimensions plus scope. The full vocabulary is open and pinned in the conformance pack.

4.2 Cascade revocation

When an entity is stopped, quarantined, revoked or marked compromised, downstream authorities derived from it become unusable before the response to the originating request returns. Dependent delegations, payment authorities, access rights, capability grants, and recovery decisions are all affected. The mechanism that effects this propagation — including its ordering, atomicity contract, audit emission, and on-the-wire summary — is part of the patent track and is **not published** in this repository.

4.3 · Implementation architecture

KYE Protocol™ stands on **16 protocol-core principles** in three tiers (see §3 above). Tier A — runtime governance — defines what KYE™ *decides*: *authority-first, state-first, decision-first, policy-bound, evidence-first, audit-trail-first*. Tier B — protocol design — defines how KYE™ *classifies and ships*: *schema-first, dictionary-first, taxonomy-first, metadata-first, graph-first, profile-first, registry-first, conformance-first*. Tier C — developer adoption — defines how KYE™ *reaches developers*: *API-first, SDK-first*. Canonical JSON Schemas drive OpenAPI, SDK code generation, validators, documentation, and conformance fixtures. Shared dictionaries (entity types, states, decisions, reason codes, capability kinds, side-effect levels, data classes, signal types, taxonomies, graph types) make implementations interoperable. Versioned taxonomies (16 V1 canonical taxonomies) and a normative metadata model (labels, classifications, ownership, lineage, compliance) drive policy decisions at runtime. Profiles keep Core small while letting domain extensions (Capability, Recovery, Payments, Payload Trust, Taxonomy & Metadata, Graph, Healthcare, EU AI Act, ISO 42001, sector overlays) compose without modifying it. Registries make every object resolvable. The runtime API is decisioning-led, with `POST /v1/runtime/authorize` as the headline endpoint and `GET /v1/decisions/{id}/map` returning a Decision Map™ for every adjudication. SDKs in TypeScript, Python, and Go ship with schema types, signing/verification helpers, evidence-pack builders, taxonomy resolvers, metadata classifiers, graph traversal clients. Every adjudicated action produces a signed `evidence_pack` a regulator can verify with public keys alone. A 41-fixture black-box conformance pack tests vendor and reference implementations byte-for-byte the same.

5 · Contract surface

v1.0 publishes **193 OpenAPI operations** across the Core + Payments specs (87+ runtime endpoints in the KYE Reference Gateway™), spanning resource families: Entities, Delegations, Scopes, Access Rights, Credentials, Attestations, Capabilities, Capability Grants, Runtime (authorize / invoke), KYE Signal Bus™ (publish / subscribe / replay / webhook-endpoints / deliveries), Audit, Proof Bundles, Transparency, Federation, Recoveries, Break-Glass Grants, Compromise Reports, Keys, Conformance, plus the Payments families (Wallets, Payment Authorities, Payment Intents, Beneficiaries). **82 JSON Schemas** with absolute `$id` URIs at `https://kye-protocol.github.io/schemas/`; **87 example payloads** pinned to schemas and validated in CI. **41 conformance fixtures** assert deterministic behaviour under happy paths and edge cases.

Every state-changing endpoint accepts an `Idempotency-Key` header and returns the original response on replay; conflicting bodies under the same key return HTTP 409. Every state-changing endpoint emits an audit event whose `correlation_id` matches the originating request.

6 · Reference runtime

The reference deployment ships:

- A **KYE Reference Gateway**™ in Node.js (no Express, no external runtime dependency) that implements every endpoint and demonstrates conformant cascade behaviour.
- An **embedded PDP library** (`@kye/epdp`) for low-latency local decisions backed by signed bundles.
- An **Express PEP middleware** (`@kye/pep-express`) that fails closed for high-risk actions when the central PDP is unreachable.
- **Three SDKs** — TypeScript, Python, Go — each with idempotency-key support and webhook helpers.
- A **conformance runner** that executes the fixture pack against any candidate Gateway.
- A **Docker image** + **docker-compose** + **Helm chart skeleton** for production deployment patterns.

The reference is illustrative. It does not implement the patent-track decision algorithm; conformant production Gateways are expected to substitute that mechanism while preserving the open contract surface.

7 · Profiles — the full v1.0 inventory

Profiles add vocabulary, schemas, endpoints, and conformance fixtures to Core. **18 base profiles ship as Normative draft v1.0:**

Base profiles:

- **KYE-Core** — entity registration, delegations, scopes, credentials, attestations, runtime authorize / evaluate, audit chain, signal bus.
- **KYE-Gateway** — API surface, headers, idempotency, content types, signing, error envelopes; the contract any conformant implementation honours.
- **KYE-Federation** — cross-trust-domain entity import with attenuated scope and origin metadata; signed assertions; transferred entities preserve provenance.
- **KYE-Credentials** — issue / verify / present / revoke with Ed25519 detached signatures over canonical payloads.
- **KYE-Attestation** — SPIFFE / EAT / build-provenance bindings; explicit revocation; stale detection.
- **KYE-Signals** — pub/sub bus with subscribe / ack / replay; signed webhook delivery with replay-window enforcement and key rotation. The specific signing-suite vectors are part of the normative specification and are not published in this repository.
- **KYE-Transparency** — append-only statement log + signed inclusion receipts; verifiable with the gateway's public keys alone.
- **KYE-Conformance** — 41-fixture black-box pack + reporter; tests vendor stacks, internal stacks, and the KYE Reference Gateway™ with the same pack.
- **KYE-Treasury** — treasury authority chain (sweeps, rebalances, intercompany transfers, FX); reconciliation hooks bind state-changing intents back to authority + scope.
- **KYE-Custody** — asset custody chain of authority; workload attestation binds runtime, custody authority binds wallet, audit chain binds timeline.
- **KYE-Healthcare** — HIPAA-aligned overlay; binds the agent to consent credentials, attaches redaction obligations and external-send blocks.
- **KYE-Telemetry** — structured authorization-decision telemetry (policy, inputs, reason code, decision time, entity chain).
- **KYE-Capability** — skills / tools / MCP tools / functions / connectors / prompts / workflows / playbooks / runbooks / model_profiles; register, grant, invoke (allow / deny / require-approval / quarantine), supersede, revoke.

- **KYE-Recovery** — recovery requests, decisions, signed proofs, time-boxed break-glass grants, compromise reports; replaces ad-hoc admin reset.
- **KYE-Payments** — payment authorities, beneficiaries, intents; sPDP with currency / amount / rail / approval gating; PSD3-aligned obligations.
- **KYE-Payload-Trust** — payload artefacts as first-class evidence (not entities). Signed, hashed, replay-resistant request bytes carry state but never authority. 13 lifecycle + denial states; 10 deny reason codes.
- **KYE-Taxonomy-&-Metadata** — 16 V1 canonical taxonomies plus a normative metadata model (labels, classifications, ownership, lineage, compliance). Metadata is not decorative; it influences runtime decisions and binds to compliance-framework controls.
- **KYE-Graph** — the Authority Graph™ contract. Canonical node + edge schemas; Decision Map™, Evidence Graph™, Blast Radius Map™, Compliance Map™ as first-class projections. Storage substrate (Postgres, Neo4j, Neptune, Memgraph, TigerGraph, ArangoDB, RDF) is implementation choice.

Payment overlays (jurisdictional):

- **kye-payments-eu-1.0** — PSD2/PSD3 + DORA alignment.
- **kye-payments-card-1.0** — PCI DSS 4.0 alignment.
- **kye-payments-high-assurance-1.0** — ISO 20022 alignment for high-value flows.

Sector overlays beyond these (healthcare-clinical / payer / research, 42 CFR Part 2) are intentional placeholders for v1.1.

7.1 · Sector coverage

The 18 base profiles compose into nine sector-ready bundles:

- **Retail & commercial banking** — Core + Payments + Treasury + Federation + Capability + Recovery.
- **Payments & cards** — Core + Payments + Credentials + Signals + Telemetry.
- **Healthcare & life sciences** — Core + Healthcare + Credentials + Capability + Telemetry.
- **Capital markets & treasury** — Core + Treasury + Custody + Attestation + Transparency + Recovery.
- **Custody & digital-asset operators** — Core + Custody + Attestation + Credentials + Recovery + Capability.
- **Insurance & underwriting** — Core + Credentials + Federation + Capability + Telemetry.
- **AI labs & agent platforms** — Core + Capability + Attestation + Signals + Recovery + Telemetry.
- **Public sector & defence** — Core + Federation + Attestation + Transparency + Recovery.
- **Marketplaces & platforms** — Core + Federation + Capability + Signals + Telemetry.

7.2 · Compliance & control mapping

The compliance addendum ships **266 control mappings** across **13 horizontal frameworks** (published in the public conformance mirror; the source-of-truth normative spec ships under commercial licence). Each row points to the specific KYE™ artefact — endpoint, schema, profile section, or conformance fixture — that satisfies it. Horizontal frameworks covered:

- SOC 2 Trust Services Criteria 2017 (rev. 2022)
- ISO/IEC 27001:2022 Annex A
- PCI DSS 4.0
- PSD2 + PSD3 / PSR
- DORA (Digital Operational Resilience Act)
- NIS2
- EU AI Act (Title III high-risk system obligations)
- ISO/IEC 42001 (AI Management System)
- NIST AI Risk Management Framework
- NIST SP 800-207 (Zero-Trust Architecture)
- NIST Cybersecurity Framework 2.0
- GDPR / UK GDPR
- FedRAMP

Sector overlays bind on top of the horizontal mappings: HIPAA Privacy + Security Rule (US healthcare), MiCA (EU crypto-asset markets), FFIEC (US bank exam guidance), IEC 62443 (industrial cybersecurity), 42 CFR Part 2 (US substance-use confidentiality).

The mapping is the audit-evidence skeleton an enterprise GRC team uses to certify a deployment.

The **KYE Compliance Mapping Rail™** (`schemas/compliance-mapping.json`) binds each framework control to the specific KYE™ runtime events that produce evidence for it; the **KYE™ EU AI Act Profile™** (`kye-euaiact-v1.md`) is the first such mapping, covering the EU AI Act with 10 controls (KYE-EUAIACT-001 through KYE-EUAIACT-010): entity accountability, AI system registry, capability manifest + risk classification, human-oversight decision gates, runtime authority decision logs, technical documentation evidence pack, corrective action and revocation trail, provider/deployer/operator role mapping, high-risk workflow profile, and post-market monitoring evidence hooks. KYE™ EU AI Act does not replace conformity assessments, notified bodies, or fundamental-rights impact assessments — it provides the signed evidence those processes consume.

Sector profiles in v1.0 cover financial services, healthcare, custody, treasury, AI labs, public sector, marketplaces, defence, critical infrastructure, energy, manufacturing, oil & gas, mining, automotive, maritime & shipping, logistics, and aviation. Each composes with KYE™ EU AI Act when AI systems or AI agents participate.

7.3 · Conformance & certification

v1.0 ships a **41-fixture black-box conformance pack** covering registration, delegation, scope attenuation, lifecycle and stop-cascade, capability invoke (allow / deny-quarantined / require-approval), capability-grant cascade, recovery flow, state transitions (allowed / rejected), break-glass grants, compromise cascade, key rotation, point-in-time replay, payments allow / deny / approval, audit integrity, idempotency, federation transfer, transparency log append + receipt, signal cascade, webhook delivery, proof bundle export, workload attest. Fixtures speak only HTTP — any conformant implementation can be tested with the same pack regardless of language, runtime, or topology. The conformance reporter emits machine-readable evidence for the auditor (`conformance-report.json`, normative as of v1.0).

7.4 · Normative addenda (gap-closure register)

An independent v1.0 normative review identified 15 gaps across blockers, important and polish classes. Each is closed at the spec contract level; the underlying mechanisms remain in the patent track.

Blockers (5 / 5 closed)

- **State-composition transition matrix** — the matrix publishes the 10 illegal compositions, 5 break-glass entry conditions and 3 terminal states across the 6-tuple state. Normative version ships in the procurement pack; the public conformance mirror exposes the test-fixture form.
- **Wire-protocol version negotiation** — `kye-gateway-v1.md` §11: `Accept-Version` header, optional URN version segment, `/.well-known/kye::kye_supported_versions`, 18-month backward-compatibility floor.
- **Cascade atomicity contract** — `kye-gateway-v1.md` §12: caller-visible end-state semantics. The contract shape, the on-the-wire summary, and the algorithm are part of the patent track and remain unpublished.
- **RFC 7807 problem+json error envelope** — `kye-gateway-v1.md` §13 + `schemas/problem-detail.json`, with status-code map and reason-code pinning.
- **MCP elicitation / sampling / versioning** — `kye-capability-v1.md` §11: gating per MCP operation, sampling budgets, tool versioning rules.

Important (5 / 5 closed)

- **Recovery m-of-n approval** — `kye-recovery-v1.md` §11 (`approval_quorum`, ordered/unordered, reject-wins, window-expiry escalation).
- **Conformance-report schema** — promoted from v1.1 placeholder to v1.0 in `schemas/conformance-report.json`.
- **Payments post-execution lifecycle** — `kye-payments-v1.md` §12: 10 states (hold, release, reversed, disputed, charged_back, dispute_resolved, settled, …), 9 signal types, ISO 20022 message-class alignment.
- **Telemetry redaction / sampling / export MUST** — `kye-telemetry-v1.md` §9-10: redaction field list, per-decision-class sampling floors, OTLP and CloudEvents 1.0 exports.

- **Quantitative SLA conformance tiers** — `kye-gateway-v1.md` §15: Tier-1 Bank / Tier-2 Mid-market / Tier-3 Reference targets for p50, p99, throughput, cascade latency.

Polish (5 / 5 closed)

- **Selective disclosure + GDPR right-to-erasure** — `kye-credentials-v1.md` §9-10 (SD-JWT, BBS+, Article 17 erasure flow).
- **Cryptographic agility** — `kye-gateway-v1.md` §14: `Accept-Crypto-Suite` negotiation; opaque suite names; algorithms remain in patent track.
- **Capability dependency + supply chain** — `kye-capability-v1.md` §12-13: DAG resolution, supply-chain attestation MUST, `state=blocked_by_dependency` cascade.
- **Multi-region geo-replication + conflict resolution** — `kye-federation-v1.md` §10: replication topology metadata, 6 conflict-resolution rules, 18-month key archival floor.
- **Vocabulary completeness** — `public/vocabulary/payments-decision-codes.md`, `healthcare-break-glass-categories.md`, `signal-types.md`, `redaction-fields.md`.

7.5 · Continuity & Discoverability

Two new normative profiles ship in v1.0 alongside the sectoral inventory above. Both extend Core: **Continuity** preserves *alignment* across the chain, **Discoverability** makes the chain *operational*.

7.5.1 KYE Continuity Profile™

Where Core records *who acted, on whose authority, in what state, with what evidence*, KYE Continuity Profile™ records whether the action **remained continuous** from intent to execution. It introduces three new dimensions on top of the seven Core records: **intent** (declared goal + constraints + declared_by), **interpretation** (interpreted goal + confidence + material_drift_detected), and **incentive / pressure context** (optimisation goal, commercial / affiliate / commission flags, urgency / coercion / social-engineering signals). Six normative objects: `KYEContinuityProfile`, `KYEContinuityContext`, `KYEIntentTrace`, `KYEContinuityDecision`, `KYEAgencyDriftEvent`, `KYEContinuityEvidencePack`. Decision values: `continuity_preserved` · `continuity_degraded` · `continuity_broken` · `require_human_review` · `require_reconfirmation` · `deny` · `quarantine`. Ten drift types (`intent_drift`, `authority_drift`, `scope_drift`, `state_drift`, `capability_drift`, `execution_drift`, `incentive_drift`, `oversight_drift`, `evidence_drift`, `delegation_drift`) each emit a signed `KYEAgencyDriftEvent` on KYE Signal Bus™ and hash-chain into the audit ledger. Spec: `kye-continuity-v1.md`.

7.5.2 KYE Discoverability Profile™

Identity, authority, scope, state, decision, audit, evidence are what KYE™ *records*. KYE Discoverability Profile™ turns the recorded graph into a **policy-filtered, audited, masked** discovery surface. Six discovery types (entity, authority, capability, evidence, risk, ecosystem); six discovery modes (`private_tenant`, `workspace`, `cross_workspace`, `federated`, `public_registry`, `certification_registry`); seven normative objects (`KYEDiscoverabilityProfile`, `KYEDirectoryEntry`, `KYEDiscoveryQuery`, `KYEDiscoveryResult`, `KYEDiscoveryPolicy`, `KYEAuthorityPathDiscovery`, `KYEDiscoveryAuditEvent`). Every query is purpose-bound (`security_review`, `audit`, `incident_response`, `compliance`, `procurement_review`, `operations`, `investigation`), audit-required by default, and runs against an explicit `KYEDiscoveryPolicy` that declares allowed / masked / denied fields. The profile is explicit about what it never returns: raw credentials, private keys, secret_refs, personal data outside the requesting tenant, and patent-track mechanism content. Spec: `kye-discoverability-v1.md`.

7.5.3 Positioning

The refined v1.0 thesis: **KYE™ makes delegated agency observable, governable, revocable, replayable — and now continuous and discoverable.**

8 · Security & threat model

The reference implementation defends against:

- **Replay attacks** — webhook signatures include a Unix timestamp and are rejected outside a 5-minute window. Idempotency keys cache responses for 24 hours.
- **Tampered audit events** — each event's canonical encoding includes its predecessor's hash; the verify endpoint detects breaks end-to-end.
- **Stale revocations** — downstream-derived authorities become unusable before the originating request returns. Mechanism not disclosed in this repository.
- **Forged credentials** — Ed25519 signatures verified against the gateway's published JWKS at `/.well-known/jwks.json`.
- **Approval timeout abuse** — pending approvals carry a `required_by` deadline; the runtime expires them and emits a deny signal.
- **Compromised actor** — compromise reports cause downstream-derived authorities to become unusable; re-activation is gated by a signed, time-boxed recovery flow. Cascade and recovery algorithms are part of the patent track.
- **Lost or rotated keys** — key rotation requires a valid `X-Break-Glass-Grant-Id`; appends a `key.rotated` entry to the audit chain; previous keys remain verifiable for a configurable retention window. The signing-suite construction is part of the normative specification and is not disclosed publicly.
- **Forensic back-dating** — point-in-time audit replay reconstructs entity / authority / state at any sequence or timestamp; investigators verify behaviour rather than reading current state.

Out of scope for the reference: HSM-backed key custody, multi-tenant gateway hardening, transport-level mTLS configuration. These are deployment concerns; production Gateways must address them.

9 · Governance

Vocabulary, schemas, OpenAPI specs and KYE Reference Gateway™ behaviours are published openly under Apache License 2.0 in github.com/KYE-Protocol. Specific mechanism designs (decision algorithms, hash-chain construction, cascade ordering, attenuation propagation) are intentional placeholders pre-filing in `private/mechanisms/` and are not disclosed publicly to preserve patent novelty. Conformant implementations may license the mechanism designs royalty-free for any conformant use; full terms are forthcoming with the Linux Foundation / OpenWallet Foundation track.

Trademark policy: KYE™, KYE Protocol™, and Know Your Entity™ refer to the protocol as published. Forks, modifications and unrelated projects must not use the marks to identify themselves.

10 · Roadmap

- **v1.1** — Sector overlays: `kye-healthcare-clinical-1.0`, `kye-healthcare-payer-1.0`, `kye-healthcare-research-1.0`, 42 CFR Part 2. Extended signal-bus durability options. `conformance-report.json` + `conformance-fixture.json` schemas.
- **v1.2** — Conformance certification programme; independent test-vector runners; vendor self-attestation portal.
- **v2.0** — Federation v2 with multi-hop attenuation and cross-jurisdiction proofs. Patent-track algorithms moved to royalty-free open standard.

References

1. Visa. Trusted Agent Protocol.
2. Persona. Know Your Agent (KYA).
3. Sumsub. Agent Verification.
4. Trulioo / PayOS. Digital Agent Passport.
5. Anthropic. Model Context Protocol.
6. OpenID Foundation. AuthZEN, SSF, CAEP.
7. SPIFFE Project. SPIFFE Identity Specification.
8. OpenSSF. SCITT Architecture.
9. NIST. SP 800-207 Zero Trust Architecture.

Appendix — verify a sample evidence pack live

The whitepaper's claims about offline-verifiable evidence packs are not abstract. Below is the same KYE Evidence Pack™ Viewer that ships at `widgets.html#evidence`: it verifies a signature against the publisher's JWKS, replays the bound decision, walks the audit chain, and projects to SOC 2 / ISO 27001 / EU AI Act / PSD3 / DORA. No signup, no install — the same flow your auditor will run on production packs.

> Cite as: KYE Protocol™ Project. *KYE Protocol™: an open trust layer for the agentic economy.* Whitepaper v1.0, April 2026. `https://kye-protocol.github.io/whitepaper.html`