Provable Authority

A Protocol for Verifiable Authority Resolution and Control Across Human, Organizational, and Agentic Transactions

CHAINIT®

Executive White Paper

ChainIT Authority Protocol

Architecture and Source Basis

September 1, 2026

Authors

Jeremy Blackburn, Chief Executive Officer
Tim Hines, Chief Product Officer
Matthew Koepp, Chief Technology Officer
Russell Lessard, Chief Compliance Officer
Jason Patrick, Chief Commercial Officer

This paper describes the ChainIT Authority Protocol architecture. It does not constitute legal, regulatory, banking, accounting, security-certification, or patentability advice. Deployment configurations, partner rail support, and service levels vary by implementation.

© 2026, Scientia Potentia Est. II, LLC. All rights reserved.

This white paper and the information contained herein are the exclusive property of Scientia Potentia Est. II, LLC and are protected by United States and international copyright laws.

Permission is granted to recipients to access, view, download, print, and retain copies of this white paper solely for their personal, informational, or internal business purposes. Recipients may share an unmodified copy of this white paper in its entirety, provided all copyright and proprietary notices are retained, and no fee is charged for such distribution.

Except as expressly permitted herein, no part of this publication may be reproduced, distributed, transmitted, displayed, published, modified, translated, reverse engineered, incorporated into other works, or used for commercial purposes in any form or by any means, whether electronic, mechanical, photocopying, recording, or otherwise, without the prior written consent of Scientia Potentia Est. II, LLC.

No license, whether express, implied, by estoppel, or otherwise, is granted to any intellectual property rights, including patents, patent applications, copyrights, trademarks, trade secrets, or other proprietary rights of Scientia Potentia Est. II, LLC, except for the limited right to access and use this white paper as expressly provided above.

Unauthorized copying, modification, disclosure, distribution, publication, or commercial exploitation of this document, or the information contained herein is strictly prohibited.

Contents

Contents 1
Definitions 2
Executive Summary 6
1. The Problem: Authentication Is Not Portable Authority 8
2. The ChainIT Architecture: Identity, Authority, Evidence, and Execution 9
2.1 ChainIT ID and the Individual VDT 10
2.2 ChainIT Org ID and the Organization VDT 10
2.3 Authority Resolution Pactvera, Delegated Authority, and the Agent Authority VDT 10
2.4 Evidence Assertion VDTs and Context-Specific Decision VDTs 12
2.5 Business Rules Engine, Workflow Engine, and Execution Eligibility 13
2.6 Stable Agent Identity and Key Domains 13
2.7 Session and Execution Credentials 14
2.8 Agent Authentication and Session Bootstrap 14
2.9 Pactvera, ChainIT Pay, Lifecycle VDTs, and Valitorum 15
2.10 Relationship to Provable Compliance 15
3. Verification at Runtime: Status, Risk Tier, and Disposition 15
3.1 Risk Tier Selects Verification Rigor 16
3.2 Policy Disposition Selects the Outcome 16
4. Applying the Protocol to Payment Pre-Authorization 17
4.1 Subject Profiles: Human, Organizational, and Agent Initiation 18
4.2 Fiat Rail Profiles 19
4.3 Stablecoin and On-Chain Profiles 19
4.4 Outcome Events, Idempotency, and Unknown State 19
5. Exact-Digest Approval and Append-Only Lifecycle Evidence 20
5.1 Human Approval and Assurance 20
5.2 Canonical Transaction Digest: What You See Is What You Authorize 20
5.3 Operational EAC Versus Permanent Evidence 20
5.4 Append-Only Lifecycle VDTs and Valitorum 21
6. The Agent Subject Profile: Agentic Commerce and the Sovereignty Boundary 21
7. Design Principles, Credential Lifetimes, and Residual Risk 22
7.1 Zero-Trust Design Principles 22
7.2 Credential and Record Lifetimes 23
7.3 Residual Risk and Non-Goals 24
8. Illustrative Scenario 25

8.1 Human-initiated wire under the same ARP 25
8.2 Agent-initiated payment within the delegated scope 25
8.3 Exception handling under either subject profile 26
9. Conclusion 26
References and Source Basis 27

Definitions

Terms below are used as defined here throughout this paper. The Authority Protocol is the general authority and pre-execution control protocol and applies to every subject of authority and to fiat, card, instant-payment, stablecoin, and on-chain transactions alike. The Agent Subject Profile is the narrower profile covering the identity and containment controls that exist only because the acting subject is an autonomous workload. Definitions are provided for reader clarity and do not themselves constitute legal, regulatory, or product specifications.

Agent Authority VDT. The append-only record of a grant of delegated authority from an authorized human or organizational principal to a registered agent, and the agent-specialized form of the Delegated Authority VDT. Each entry immutably records the granting principal, the governing ARP and its version, the registered agent, the permitted purpose, counterparties, transaction classes, rails, per-transaction and aggregate value limits, effective period, step-up thresholds, redelegation rule, and revocation conditions. An agent holds no authority of its own; the Agent Authority VDT is the provable evidence that authority was granted, by whom, and within what bounds. The record is immutable, but present validity is resolved at the reliance event against current ARP status and the applicable freshness bound. It is not a bearer credential.

Agent Registration VDT. A versioned record identifying an autonomous workload's organization, responsible party or role, tenant, deployment, software or workload identity, permitted tools and APIs, risk class, attestation method, registered public key, status reference, and version history. It establishes what the workload is, not what it may do.

Agent Subject Profile. The identity and containment controls that exist because the acting subject is an autonomous workload rather than a person: the Agent Registration VDT, the attested workload key and ephemeral session key hierarchy, and the sovereignty boundary. The profile adds no authority concept of its own (an agent exercises delegated authority under the ARP in the same way a human delegate does, evidenced by an Agent Authority VDT).

Authority Policy Profile. The versioned rule set selected for a given reliance event, which may vary by organization, product, role, subject class, counterparty, transaction type, value, payment rail, jurisdiction, risk tier, and intended purpose. It is the authority-side counterpart to the Compliance Policy Profile described in ChainIT Provable Compliance, and it is versioned so that a later reviewer can determine exactly which rule set produced a decision.

Authority Protocol. The general ChainIT protocol for resolving whether an identified subject may perform a specific proposed action at the moment of reliance. It is not limited to agentic commerce and is not rail-specific: the same identity, authority, evidence, policy, digest, reservation, execution, and outcome-evidence controls govern a wire a finance director releases, a dual-approved ACH file, a card or instant-payment instruction, a stablecoin transfer, and an agent-initiated proposal.

Authority Resolution Pactvera (ARP). The canonical, versioned organizational authority record: who may act, in what role, for what purpose, within what limits, during what effective period, under what quorum or separation-of-duties requirements, and subject to what amendment or revocation.

Business Rules Engine (BRE). The deterministic predicate-evaluation engine. It evaluates configured conditions against subject identity and assurance, ARP authority, registration and attestation where the subject is a workload, evidence, status freshness, jurisdiction, scope, session risk, and for a transaction, the exact transaction, current capacity, required approvals, and rail-specific constraints, and returns explicit,

reason-coded true/false predicate results. The BRE does not itself perform risk scoring or assign a disposition; it runs within the ChainIT Workflow Engine, where the organization configures how predicate results combine into a disposition.

Canonical transaction digest. A deterministic hash of a canonical transaction object (payer, source, payee, destination, amount, currency or asset, rail, fees, nonce, validity window, and rail-specific parameters, among other fields) computed under a fixed encoding rule so that any material change to the transaction produces a different digest.

ChainIT ID. The verified identity anchor for an individual, established through authoritative-source verification, biometric and liveness controls, and privacy-preserving, portable identity storage.

ChainIT Org ID. The legal-entity identity anchor for an organization, which may reference entity registration, ownership, officers, governance, and authorized representatives.

ChainIT Pay. The ChainIT payment-execution platform intended to serve as the EAC enforcement boundary where the applicable Authority Protocol profile is deployed; a compatible partner rail adapter may serve this function in another implementation.

Compliance Decision VDT. The signed, time-bounded, purpose-specific record produced by an organization's configuration of the ChainIT Workflow Engine, recording whether present evidence and policy permit a proposed exercise of authority. Defined in ChainIT Provable Compliance and consumed here as the Compliance Decision Permits predicate. It gates present exercise; it does not amend, suspend, or erase the governing ARP.

Delegated Authority VDT. A portable artifact that references the governing ARP and exposes only the scope of authority needed by a given delegate, agent, or relying party, without replacing or superseding the ARP itself. Its agent-specialized form is the Agent Authority VDT.

Disposition. The explicit outcome produced by an organization's configuration of the ChainIT Workflow Engine from BRE predicate results (e.g., Allow, Allow with Controls, Step-Up, Review, Hold, Reject, Prohibit, or Stale/Unknown). The disposition set is configured by the organization to match its own policies and regulatory obligations; it is not a risk judgment ChainIT renders on the organization's behalf. Disposition and risk tier are evaluated independently (a high-risk transaction may still resolve to Allow, and a low-risk transaction may still be rejected).

Evidence Assertion VDT. A record of a single sourced fact, such as identity-document status, sanctions-screening result, or entity status, including its subject, source, source version, retrieval and effective time, expiration, jurisdiction, match or confidence state, and privacy classification. Evidence Assertion VDTs are never aggregated into a single "compliant" status. This is the same artifact class as the Evidence VDT defined in ChainIT Provable Compliance, which classifies each item as authoritative-source, qualified third-party, or institution-generated evidence and records what the source does and does not prove.

Execution Authorization Credential (EAC). A single-use, short-lived credential authorizing one exact transaction for one audience, bound to the acting session or approved subject, transaction digest, reservation, status epoch, and expiration, and consumed atomically at commit.

Execution Authorization Lifecycle VDT. A permanent, append-only evidence record of EAC issuance and terminal handling (consumed, expired, canceled, or rejected). Earlier lifecycle events are never modified or overwritten.

Pactvera. The ChainIT transaction-logic and append-only evidence framework linking Validated Data Tokens across a transaction's lifecycle.

Reliance event. The moment at which a relying party depends on identity, authority, evidence, or policy state — session issuance or renewal, a transaction proposal or payment initiation, a human step-up approval, a scheduled review, or a source-change or material risk event. Freshness and policy are evaluated at the reliance event rather than inherited from onboarding.

Reservation. The atomic hold of transaction capacity against a budget or limit, performed before Execution Authorization Credential issuance so that concurrent requests cannot commit against the same capacity twice.

Risk tier. A classification (Tier 1–Low, Tier 2–Standard, Tier 3–High) assigned to a transaction based on factors such as payee history, value, reversibility, and behavioral or device anomalies, which determines the minimum acceptable verification rigor (e.g., cached status snapshot versus live status introspection). The tier is assigned by the organization's configuration of the ChainIT Workflow Engine, not by the BRE; the assignment records the policy version applied, the input factors considered, the assignment time, applicable reason codes, and any manual override.

Session Authorization Credential (SAC). A short-lived, sender-constrained credential that permits proposal activity, evidence retrieval, and defined API activity, but which rail adapters must reject as payment authority. A session credential is never payment authority, whether the session belongs to a workload or to a signed-in person.

Session Eligibility Decision VDT. A session-context decision recording whether the authenticated subject may open or continue a session that enables consequential action, referencing identity assurance, the ARP, evidence, status, jurisdiction, scope, and policy version. Where the subject is a workload, it additionally references the Agent Registration VDT, Agent Authority VDT, attested deployment, and applicable tool scope. It is not transaction-specific and cannot authorize payment.

Sovereignty boundary. The architectural line beyond which an AI model or autonomous agent may reason, gather evidence, and propose an action, but cannot obtain unrestricted access to payment-account credentials, custody keys, or final execution authority, with that authority remaining with the deterministic control plane.

Status epoch / status freshness. The signed timestamp or snapshot boundary used to determine whether cached non-revocation status is still acceptable for a given risk tier, versus requiring live status introspection.

Subject of authority. The party whose authority is being resolved: a verified individual, a group acting under a quorum or separation-of-duties rule, an organization acting through an authorized representative, or a permitted software agent. The ARP, evidence, policy, and decision artifacts are the same across subject classes; what differs is how the subject is authenticated and registered, and what containment it requires. A scheduled workflow is not a fifth subject class; it is an organization-controlled initiation mechanism that acts on behalf of an identified organization and under the governing ARP, in the same manner as any other organizational or multi-party initiation.

Transaction Policy Decision VDT. A decision bound to one canonical transaction digest, recording risk tier, disposition, applicable policy version, evidence references, ARP and subject references (including agent references where the subject is a workload), required approvals or controls, expiration, and reason codes. It is newly derived per transaction and is not reusable as a standing status.

Validated Data Token (VDT). A cryptographically verifiable digital record representing validated data concerning a person, organization, asset, service, event, decision, or transaction. Within the Authority Protocol, evidence and decision VDTs are immutable or versioned; a change is represented by a new linked VDT rather than alteration of a prior immutable record (e.g., Evidence Assertion VDT, Agent Registration VDT, Transaction Policy Decision VDT).

Valitorum. The final sealed evidence record generated for a completed or otherwise terminal transaction lifecycle under a defined completeness rule. A sealed Valitorum is never altered; subsequent exceptions (returns, reversals, disputes) are captured in a linked Supplemental Valitorum that references the original.

Valitorum Service. The ChainIT service that creates, verifies, retrieves, or presents a Valitorum.

Workflow Engine. The ChainIT platform capability an organization configures to encode its own internal policies and regulatory requirements, orchestrating BRE predicate results into the disposition states and downstream actions the organization defines. The organization is responsible for the policy logic it configures and for its use of the resulting disposition, subject to applicable law and contractual allocation among ChainIT, the organization, payment providers, evidence providers, and other participating parties.

Workload key / session key. The key hierarchy underlying agent operations: a rotatable workload key, generated or protected in an approved security boundary (e.g., TPM, TEE, KMS, HSM), authorizes an ephemeral session key; Session Authorization Credentials and Execution Authorization Credentials are sender-constrained to the session key rather than the workload key.

Executive Summary

Payment systems authenticate users, devices, accounts, and cryptographic credentials. Financial institutions also enforce internal entitlements, limits, dual approval, sanctions screening, fraud controls, and account permissions. The persistent gap is portability: a counterparty, marketplace, payment rail, or autonomous workload generally cannot obtain compact, independently verifiable evidence that the party presenting one exact transaction possesses current organizational authority to do so under the applicable evidence, policy, budget, and approval state.

That gap is not created by autonomous agents and it is not limited to them. A finance director can hold a delegation that has since been amended. A dual-approval control can be satisfied on a screen that does not match the ACH file eventually transmitted. A wire can be released to a beneficiary account that changed after the invoice was approved. A per-transaction limit can be defeated by two concurrent requests reading the same balance. In each case the credential is authentic and the control is nominally present, but no relying party can independently verify that the exact instruction being committed is the one that current authority, evidence, and policy actually permit.

Agentic commerce makes that gap operationally urgent rather than novel. A software agent can produce a valid signature while acting outside its delegated purpose, after its human sponsor or organizational authority has changed, against stale or ambiguous evidence, beyond an aggregate budget, or on transaction details that differ from those a human approved. Cryptographic authenticity is necessary, but it does not answer the governance question: who stands behind the action, what authority was granted, whether that authority may be exercised now, and what precise instruction is being committed.

ChainIT Provable Authority is a pre-execution authority and evidence control plane, grounded in the ChainIT platform, patent, and application materials cited throughout. [1]-[6][16] It does not replace payment rails. ChainIT ID and ChainIT Org ID anchor verified subjects; a versioned Authority Resolution Pactvera is the governing organizational authority record; the Business Rules Engine returns deterministic predicate results against a versioned Authority Policy Profile and sourced evidence; a transaction-specific decision binds an exact canonical transaction digest; capacity is reserved atomically; a one-time Execution Authorization Credential is consumed at the payment boundary; and Pactvera, append-only lifecycle VDTs and Valitorum preserve what was decided, who acted, and what occurred. ChainIT Pay, or a compatible partner adapter, serves as the execution boundary where the applicable enforcement profile is deployed. These controls are subject-neutral and rail-neutral: they apply equally to a wire a finance director releases, an ACH file a shared-service team approves under a quorum rule, a card or instant-payment instruction, a stablecoin transfer, and an agent-initiated proposal.

Regulatory, identity, and risk information is not collapsed into a universal statement that a person is “compliant.” Each sourced fact is represented as an Evidence Assertion VDT with provenance, freshness, scope, and privacy controls, evaluated at the moment of reliance rather than inherited from onboarding. The BRE returns the predicate results; the organization’s configuration of the ChainIT Workflow Engine composes them into an explicit disposition (e.g., Allow, Allow with Controls, Step-Up, Review, Hold, Reject, Prohibit, or Stale/Unknown) and the organization is responsible for that policy logic and for its use of the resulting disposition, subject to applicable law and contractual allocation among ChainIT, the organization, payment providers, evidence providers, and other participating parties. This preserves the difference between organizational authority, sourced evidence, institutional policy, and present execution eligibility. The evidence, policy-profile and disposition mechanics are described in full in the companion paper, ChainIT Provable Compliance. [17]

An agent is a subject of authority, not a source of it. A verified principal who holds authority under an ARP may grant a registered agent a narrower authority to act on their behalf, bounded by purpose, counterparty,

transaction class, rail, per-transaction and aggregate value, and effective period. That grant is recorded in an append-only Agent Authority VDT: provable evidence of what authority was granted, by whom, and within what bounds. The record is immutable, while present validity resolves at the reliance event against current ARP status, so it is evidence of a grant rather than a bearer credential. Within the granted band the agent acts with no human in the loop; crossing the boundary (a new payee, a changed destination, a higher value, an irreversible rail) requires biometric exact-digest approval by a person with independent authority for that transaction. The Agent Subject Profile adds only the identity and containment controls a non-human subject requires (a versioned Agent Registration VDT, an attested workload key authorizing an ephemeral session key, and the sovereignty boundary that keeps final execution authority outside the model); it adds no authority concept of its own.

> **Core proposition**
>
> Before value moves or another consequential action executes, ChainIT is designed to establish: who or what is acting (a verified person, a group under quorum, an organization, or a permitted agent); whose authority supports the action; whether current evidence and policy permit that authority to be exercised now; what exact transaction is authorized; whether capacity is reserved; and what ultimately occurred.

1. The Problem: Authentication Is Not Portable Authority

Most payment infrastructure does more than merely validate a password or key. Banks and enterprises already maintain internal entitlements, approval limits, maker-checker controls, device authentication, sanctions screening, and fraud scoring. The missing capability is a portable, transaction-bound chain of authority that can be independently consumed across an organization, its bank, a processor, a marketplace, an agent runtime, and a counterparty.

The material question is therefore not only “Is this signature valid?” It is “Is this identified person or attested workload acting under a current organizational delegation, within the applicable evidence and policy state, against available capacity, on the same transaction that was evaluated and approved?” Without that answer, organizations either block agent-initiated activity or permit it through controls that are difficult to verify outside the system that created them.

The question is the same whether the initiating subject is a person, an organization initiating through a scheduled workflow, or an autonomous agent, and whether value moves over ACH, wire, card, an instant-payment rail, or a blockchain. Agent initiation raises the stakes because no human is present to notice an unfamiliar payee or a changed bank account. The control gap it exposes, however, is the ordinary fiat gap: authority, evidence, approval, and capacity are each verified inside systems the relying party cannot inspect, and none of them is cryptographically bound to the exact instruction that finally reaches the rail.

> - Authentication proves control of an accepted authenticator; it does not prove organizational power or transaction scope.
> - A valid organizational delegation may exist while its exercise is temporarily held for evidence refresh, enhanced review, or step-up.
> - A session may be valid while a particular payment is out of scope, over budget, directed to a new destination, or prohibited.
> - A human approval is not sufficient unless it is bound to the exact transaction ultimately presented to the rail.
> - A per-transaction limit is not enforceable under concurrency unless capacity is reserved atomically before commit authority is issued.
> - A signed credential is not proof of current non-revocation unless live status or a sufficiently fresh

> signed status snapshot supports the decision.

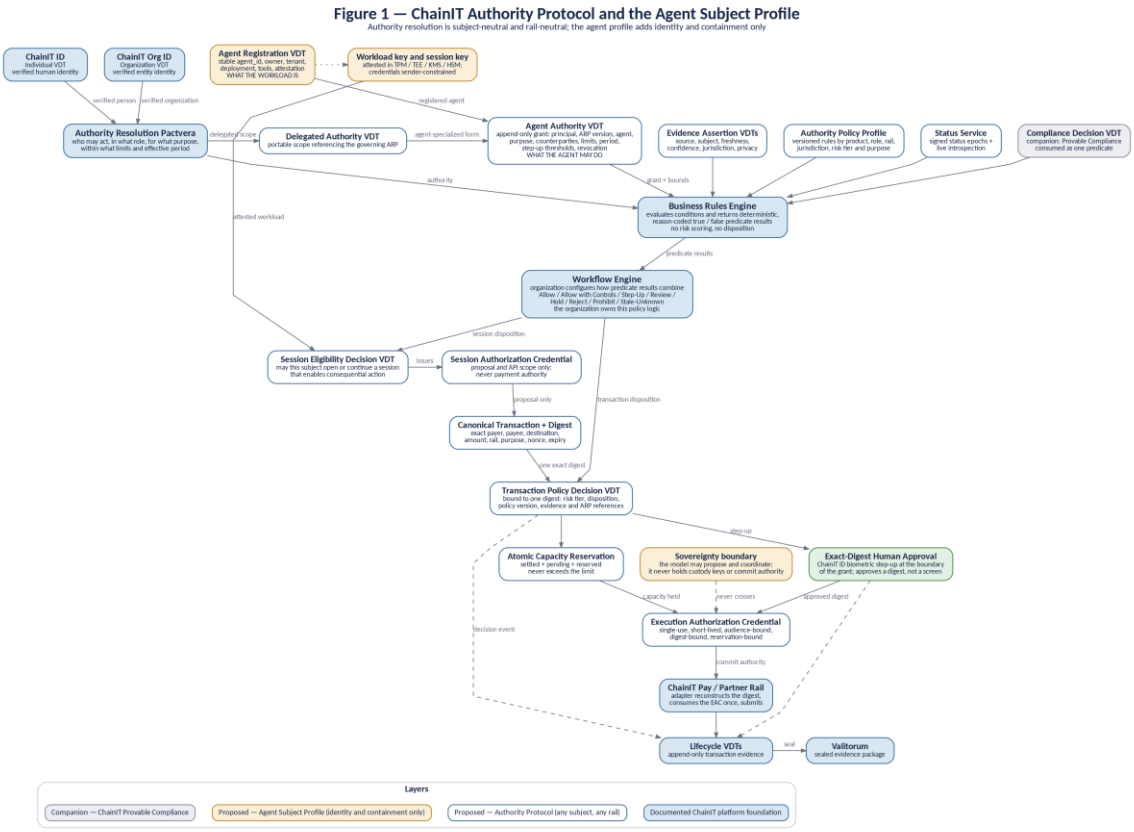
2. The ChainIT Architecture: Identity, Authority, Evidence, and Execution

The architecture separates durable identity, organizational authority, sourced evidence, session eligibility, transaction policy, operational credentials, payment execution, and outcome evidence. This separation prevents one opaque credential from becoming a substitute for every decision and allows each relying party to consume only the facts needed for its purpose.

The architecture is organized in two layers. The Authority Protocol is the general layer: it resolves whether an identified subject may perform a specific proposed action at the moment of reliance, and it is neither subject-specific nor rail-specific. The Agent Subject Profile is a narrower application layer that adds only the identity and containment controls a non-human subject requires: registration, workload attestation and key hierarchy, and the sovereignty boundary. It adds no authority concept of its own. An agent exercises delegated authority under the ARP exactly as a human delegate does, evidenced by an Agent Authority VDT, which is a protocol-layer artifact. Every subsection below belongs to the Authority Protocol unless it is expressly identified as part of the Agent Subject Profile.

The subjects of authority are a verified individual, a group acting under a quorum or separation-of-duties rule, an organization acting through an authorized representative, and a permitted software agent. The ARP, Evidence VDTs, Authority Policy Profile, BRE predicates, Workflow Engine disposition, canonical transaction digest, capacity reservation, Execution Authorization Credential, lifecycle VDTs, and Valitorum are the same across all four. What differs is how the subject is authenticated and registered, and what containment the subject requires.

Throughout this paper, the architecture is described in terms of the question each artifact answers, so that a relying party can consume only the facts its purpose requires.



Companion Figure 1. ChainIT Authority Protocol and the Agent Subject Profile.

2.1 ChainIT ID and the Individual VDT

ChainIT ID provides the verified identity anchor for an individual. The supplied identity application materials describe authoritative-source verification, biometric and liveness controls, privacy-preserving and selective disclosure, portable identity storage, and verification without querying the originating government system on every use. Transaction-time reauthentication should therefore be described as comparison against a previously source-validated ChainIT ID under the applicable refresh and assurance policy, not automatically as a new government query for each approval. [1]

The factor combination — biometric, device, cryptographic wallet, PIN, location, or other controls — and the BeingID Level and Token Grade taxonomy (ChainIT's internal classification of identity-proofing strength and credential quality) determine the assurance achieved. The required assurance is selected by risk and policy; a lower-assurance fallback cannot be described as equivalent to a higher-assurance approval merely because both occur through a ChainIT interface.

2.2 ChainIT Org ID and the Organization VDT

ChainIT Org ID establishes the legal-entity identity anchor and may reference entity registration, ownership, officers, governance, authorized representatives, and related evidence. The Organization VDT identifies the organization; it should not be used as a static container for every authorized person because authority changes independently. Authority belongs in a versioned ARP or a record derived from it. [1]-[3]

2.3 Authority Resolution Pactvera, Delegated Authority, and the Agent Authority VDT

Authority Resolution Pactvera (ARP) is the canonical, versioned organizational authority record: who may act, in what role, for what purpose, within what limits, during what effective period, under what quorum or

separation-of-duties requirements, and subject to what amendment or revocation. Under the Authority Protocol, a portable Delegated Authority VDT may reference the governing ARP and expose only the scope needed by a delegate, an agent, or a relying party. The portable artifact does not replace the ARP, and its continued validity depends on current ARP status. [2][3]

Authority is an organizational fact that can remain historically valid even when present exercise is suspended. An expired identity credential, an unresolved evidence alert, or a required enhanced review does not rewrite the historical delegation; it changes whether that authority may be exercised for the present context.

The division of labor with the companion paper follows from this. The ARP answers who may act, for what purpose, within what scope, and during what time. The Compliance Decision VDT answers whether present evidence and policy permit that authority to be exercised now. The two are evaluated together, and neither substitutes for the other. A corporate officer remains an officer while a transaction is held for enhanced due diligence, and an expired identity credential reduces assurance without proving the person lost every corporate role. [17]

Delegation follows the same rule. A principal who holds authority under the ARP may grant a narrower authority to someone or something acting on their behalf, and the grant is bounded by purpose, counterparty, transaction class, rail, per-transaction and aggregate value, and effective period. Delegation to a registered agent is an instance of this, not an exception to it. An agent holds no authority of its own; it exercises a grant that a verified principal made under an ARP that permitted the delegation.

The Agent Authority VDT is the append-only record of that grant, and the agent-specialized form of the Delegated Authority VDT. Each entry records the granting principal, the governing ARP and its version, the registered agent, the permitted purpose and counterparties, the transaction classes and rails, the per-transaction and aggregate limits, the effective period, the step-up thresholds, the redelegation rule, and the revocation conditions. Amendments, suspensions, and revocations create new linked entries rather than editing an earlier one, so a relying party, auditor, regulator, or reviewer can establish what authority the agent held at any past moment as well as what it holds now.

Immutable grant, resolved state. The distinction matters. The record is immutable; present validity is not carried by the record. Authority changes (a principal lowers a limit, the governing ARP is amended, the delegation is revoked) and an artifact that asserted current authority on its own would be a portable bearer credential of exactly the kind this architecture is designed to prevent. The Agent Authority VDT therefore proves what was granted and when; present validity resolves at the reliance event against current ARP status, revocation state, and the applicable freshness bound.

Necessary, not sufficient. The Agent Authority VDT establishes that authority was granted and within what bounds. It does not establish that the workload presenting it is the registered workload. That is attestation and sender-constrained proof. It does not establish that this transaction falls within scope right now. That is the policy evaluation at the reliance event. And it does not authorize the commit. That is the one-time Execution Authorization Credential bound to the exact canonical transaction digest. It is one required input to the execution predicate, not a credential that can be presented in place of the others.

The autonomous band and the step-up boundary. A grant defines a band within which the agent may act without a human in the loop, and a boundary beyond which it may not. Within the band, an approved counterparty, an established transaction class, a value below the configured threshold, the agent proposes and the deterministic control plane decides, with no human interaction. Crossing the boundary — a new payee, a changed destination, a higher value, an irreversible rail, or a materially changed evidence state — requires biometric exact-digest approval by a person holding independent authority for that transaction, as described in Section 5.1. Step-up is the boundary condition, not the ordinary case; a design that required

human confirmation of every agent transaction would make the delegation meaningless. Human approval at the boundary satisfies a separate transaction-specific authority path under the ARP. It does not expand the standing grant, and the Agent Authority VDT is not amended by it.

2.4 Evidence Assertion VDTs and Context-Specific Decision VDTs

Under the Authority Protocol (for any subject, human or agent) the architecture does not use a universal Compliance VDT asserting that a person or organization is broadly "in good standing." Instead, each sourced fact is represented separately as an Evidence Assertion VDT. An assertion records the subject, source, source version, retrieval time, effective time, expiration, jurisdiction, match or confidence state, privacy classification, and any material limitation on use or disclosure.

Evidence Assertion VDTs

Examples include identity-document status, entity status, ownership information, sanctions-screening result, PEP identification, license status, counterparty verification, beneficiary confirmation, device or location context, and unresolved monitoring alerts. Evidence is refreshed according to source-specific cadence, risk triggers, material changes, and policy. ChainIT need not re-query every external source for every transaction if a current assertion remains within its permitted freshness window. Each assertion is classified as authoritative-source, qualified third-party, or institution-generated evidence, and carries the freshness rule appropriate to that source class; the full taxonomy is set out in ChainIT Provable Compliance. [17]

Session Eligibility Decision VDT

A session-context decision answers whether a subject may open or continue a session that enables consequential action, and perform proposal or defined API activity within it. The artifact is protocol-general: a signed-in treasury session is subject to the same evaluation as a workload session. What differs is the input set — a person is admitted on identity assurance and role, a workload on registration, attestation, tool scope, and its Agent Authority VDT. It references the ARP, applicable evidence, status, jurisdiction, policy version, disposition, reason codes, and expiration. Where the subject is a workload, it additionally references the Agent Registration VDT, Agent Authority VDT, attested deployment, and applicable tool scope. It is not transaction-specific and cannot authorize payment.

Transaction Policy Decision VDT

A transaction-context decision is bound to one canonical transaction digest. It records the risk tier, disposition, applicable policy version, evidence references, ARP and subject references, required approvals or controls, expiration, and reason codes. It is created the same way whether the transaction was proposed by a person, an organization through a scheduled workflow, or an agent. It is newly derived for the proposed transaction and is not reusable as a permanent status attached to the person or organization.

Regulatory precision

- PEP identification is a risk factor evaluated through risk-based customer due diligence and monitoring; it is not itself an automatic disqualification. [12]
- OFAC exposure is not limited to a name on the SDN list; direct and indirect ownership analysis and false-positive resolution may be required. [13]
- AML is an institutional program of monitoring, investigation, and reporting. An alert is evidence requiring evaluation, not a portable fact that a person is "AML invalid."
- Where criminal-history information is used for employment or role qualification, relevance is contextual to the role, industry, jurisdiction, offense, and elapsed time; it is not a universal pass/fail condition. [15]
- A portable credential or shared ledger must not disclose a SAR or information that would reveal the existence or non-existence of a SAR. [14]

SAR confidentiality constraint

A relying party may receive a disposition such as “Hold: manual review required” without receiving the protected source or rationale that would disclose the existence or non-existence of a SAR. The evidence model must support restricted internal review and minimum counterparty disclosure.

2.5 Business Rules Engine, Workflow Engine, and Execution Eligibility

The BRE is the deterministic predicate-evaluation point. For a session, it evaluates identity assurance, the ARP, evidence, status, jurisdiction, and scope, together with registration, attestation, tool scope, and session risk where the subject is a workload. For a transaction, it evaluates those inputs together with the exact transaction, current capacity, required approvals, status freshness, and rail-specific constraints. Each result is explicit and reason-coded rather than a black-box “approved” flag.

The BRE does not assign the disposition. It runs within the ChainIT Workflow Engine, where the organization configures which conditions apply, how predicate results combine, and which disposition states and downstream actions follow. ChainIT supplies the evidence, the predicate-evaluation engine, and the workflow tooling; the organization is responsible for the policy logic it configures and for its use of the resulting disposition, subject to applicable law and contractual allocation among ChainIT, the organization, payment providers, evidence providers, and other participating parties. Collapsing predicate evaluation into a single "all checks pass?" boolean increases false positives, overblocking, and disclosure risk.

At the authority layer, the general execution predicate is the one stated in ChainIT Provable Compliance: MayExecute = IdentityAssuranceSatisfied AND AuthorityValidUnderARP AND ComplianceDecisionPermits AND TransactionWithinScope AND RequiredApprovalsSatisfied AND BudgetAndRailControlsSatisfied. The commit-time predicate below is that same test specialized to the payment boundary, where the exact instruction, the capacity reservation, and the one-time credential must also be verified. [17]

Commit-time execution predicate

MayCommit = SubjectAuthenticated AND AuthorityValidUnderARP AND EvidenceAcceptable AND ComplianceDecisionPermits AND WorkflowDispositionIsAllow AND StatusFresh AND CapacityReserved AND RequiredApprovalsSatisfied AND DigestMatches AND EACValidAndUnused

Every predicate is required. SubjectAuthenticated resolves to workload attestation where the subject is an agent and to ChainIT ID authentication at the required assurance level where the subject is a person. Allow with Controls or Step-Up must be completed and re-evaluated to Allow before EAC issuance or commit. Failure produces a controlled disposition, not silent fallback.

2.6 Stable Agent Identity and Key Domains

This subsection describes the Agent Subject Profile, which answers what the workload is rather than what it may do. Where the subject of authority is a verified person, the equivalent control is ChainIT ID authentication at the policy-required assurance level; no agent registration or workload key is involved, and no other part of the control flow changes. What the agent may do is a separate question, answered by the ARP and the Agent Authority VDT in Section 2.3.

Under the Agent Subject Profile, every autonomous workload is assigned a stable logical agent_id and a versioned Agent Registration VDT identifying its organization, responsible party or role, tenant, deployment, software or workload identity, permitted tools and APIs, risk class, attestation method, registered public key, status reference, and version history.

The key hierarchy is explicit. A rotatable workload key is generated or protected in an approved security boundary such as a TPM, TEE, KMS, HSM, or equivalent environment. That workload key authorizes an ephemeral session key. The Session Authorization Credential, per-request proofs, and Execution Authorization Credential are sender-constrained to the session key. ChainIT issuer keys sign and attest credentials; they do not act as the agent. The stable agent_id persists when workload or session keys rotate.

Sender-constrained proof reduces the value of a copied credential because an attacker must also control the bound key. It does not, by itself, authenticate the organization, establish authority, validate policy, or prevent malicious code already operating inside the legitimate workload from using that key. [7]

2.7 Session and Execution Credentials

Every artifact below is protocol-general. A session credential is never payment authority, and a session eligibility decision is never transaction authority, whether the session belongs to a workload or to a signed-in person; only a one-time credential bound to an exact canonical transaction digest may commit value. What varies by subject class is the input set each artifact evaluates, not the artifact itself. The agent-specific inputs (Agent Registration VDT and the attested key hierarchy) are described in Sections 2.6 and 2.8; the agent's authority to act at all is evidenced by the Agent Authority VDT in Section 2.3.

Artifact	Purpose	Critical limitation
Session Eligibility Decision VDT	Records that the authenticated subject may open or continue a session under the current context.	Not bound to an exact payment and cannot authorize value movement.
Session Authorization Credential (SAC)	Short-lived, sender-constrained credential permitting proposal, evidence retrieval, and defined API activity.	Rail adapters must reject it as payment authority.
Transaction Policy Decision VDT	Records the risk tier, disposition, policy version, evidence state, controls, and exact transaction digest.	Only a final Allow disposition—after required controls and step-up are completed and revalidated—may support EAC issuance.
Execution Authorization Credential (EAC)	Single-use, short-lived credential authorizing one exact transaction for one audience after reservation and approvals.	Bound to the session key, digest, reservation, status epoch, and expiration; consumed atomically.
Execution Authorization Lifecycle VDT(s)	Permanent evidence of EAC issuance and terminal handling (consumed, expired, canceled, or rejected), using one terminal record or separate linked issuance and disposition events.	Must not contain a reusable credential or private key; an earlier immutable event is never updated.

2.8 Agent Authentication and Session Bootstrap

This subsection also describes the Agent Subject Profile. The corresponding step for a human subject is ChainIT ID authentication at the assurance level the Authority Policy Profile requires for the session. Both paths then converge on the same session-eligibility evaluation.

Before issuing a session, ChainIT must authenticate the registered workload. The workload presents its Agent Registration VDT reference, attested workload key, approved deployment evidence, and proof of possession. The workload key then authorizes an ephemeral session key. The Session Eligibility Decision VDT is created for that context, and the SAC is sender-constrained to the session key.

A persistent bootstrap credential or a one-time enrollment credential may be used as an additional operational pattern, but neither is sufficient by itself. A persistent credential must be cryptographically bound to the registered workload. A one-time credential must be delivered through a channel authenticated to the correct workload, such as mutual Transport Layer Security (mTLS), private-key authentication, cloud workload identity, or another approved control.

2.9 Pactvera, ChainIT Pay, Lifecycle VDTs, and Valitorum

Pactvera and the Valitorum Service provide the transaction logic and final evidence. ChainIT Pay, or a compatible partner rail adapter, serves as the EAC enforcement boundary where the applicable Authority Protocol profile is deployed, and each material state transition is recorded as a separate append-only lifecycle VDT. This applies to every subject class; the enforcement point does not change because the proposal originated with an agent rather than a person. The Valitorum Service seals the completed or otherwise terminal evidence graph. [2]-[6][16]

2.10 Relationship to Provable Compliance

Provable Authority and Provable Compliance answer adjacent questions and are designed to be evaluated together. The ARP answers who may act, for what purpose, within what scope, and during what time. The Compliance Decision VDT answers whether present evidence and policy permit that authority to be exercised now, for this product, counterparty, rail, jurisdiction, and risk tier. Neither substitutes for the other, and neither rewrites the other. [17]

The practical consequence is a division of effect. A compliance disposition gates the present exercise of authority; it does not amend, suspend, or delete the ARP. A newly identified politically exposed person retains her corporate role while her transactions receive enhanced review. An expired driver's license reduces identity assurance without proving the person ceased to hold office. Only a confirmed mandatory prohibition, or another expressly defined policy condition, produces broader blocking. Conversely, a favorable compliance disposition confers no authority: a fully evidenced, unrestricted subject still cannot execute a transaction the ARP does not permit.

Both papers use the same platform mechanics. Evidence is captured as sourced, time-bounded VDTs; a versioned policy profile selects the applicable rules; the BRE returns deterministic predicate results; and the organization's configuration of the ChainIT Workflow Engine composes those results into a disposition the organization defines and owns. The Authority Protocol consumes the compliance result as a single predicate, Compliance Decision Permits, and contributes the authority resolution, transaction digest, capacity reservation, exact-digest approval, execution credential, and sealed outcome evidence that a compliance decision alone does not provide.

3. Verification at Runtime: Status, Risk Tier, and Disposition

A signature and expiration establish that a credential was issued, has not been altered, and is within its stated validity period. They do not establish that the credential, agent registration, ARP, person, organization, or supporting authority has not been revoked since issuance. Current non-revocation therefore requires live status or a signed status artifact within an approved freshness bound.

Freshness is evaluated at the reliance event, not on a universal schedule. The reliance events for this protocol are session issuance or renewal, a transaction proposal or payment initiation, a human step-up approval, a scheduled review set by source volatility or relationship risk, and a source-change or material risk event such as a sanctions-list update, a license change, an ownership change, a policy update, or an ARP amendment. Each required item carries its own maximum age, derived from the source, the evidence type, the transaction risk, the product, the jurisdiction, and the reliance event itself — the same source-specific

freshness rule stated in ChainIT Provable Compliance. Reliance events are identical in kind whether the subject is a person or an agent. [17]

Bounded offline freshness

StatusFresh = (CurrentTime - StatusSnapshot.IssuedAt) <= min(RiskTierStatusLimit, AuthorityPolicyStatusLimit)

EvidenceFresh = AND, for each Evidence Assertion VDT i: (CurrentTime - Evidence_i.EffectiveTime) <= SourceAndPolicyLimit_i

StatusFresh tests whether the cached non-revocation snapshot for a credential, ARP, or registration is still within its approved bound. EvidenceFresh separately tests each underlying Evidence Assertion VDT (sanctions, identity, licensing, beneficial-ownership, and similar sourced facts) against its own source- and policy-specific limit. A current status snapshot does not by itself establish that this underlying evidence is current. Commit requires both StatusFresh and EvidenceFresh; EvidenceFresh is one of the conditions within the broader EvidenceAcceptable test applied at commit (Section 2.5). If either test fails, or the risk tier requires live status, the verifier must introspect or re-retrieve rather than accepting the cached value.

The status service publishes signed status epochs or compact status snapshots that verifiers can cache. A compatible compact status structure may follow established credential-status patterns, while a live service may follow introspection semantics adapted to ChainIT authority and transaction context. [8][10]

3.1 Risk Tier Selects Verification Rigor

Risk tier	Illustrative condition	Minimum verification mode
Tier 1 - Low	Known payee, low value, reversible flow, normal device and behavior.	Signed status snapshot within the longest approved bound; routine authentication and monitoring.
Tier 2 - Standard	Normal vendor payment within established limits and ordinary risk.	Tighter status bound; live status when evidence changed or snapshot approaches its limit.
Tier 3 - High	New payee, changed destination, high amount, cross-border, irreversible rail, anomalous device, or critical evidence change.	Live status, stronger authentication, step-up or quorum as required; no stale offline acceptance.

3.2 Policy Disposition Selects the Outcome

The table below illustrates one disposition set an organization might configure in the ChainIT Workflow Engine using BRE predicate results as inputs. These are not decisions ChainIT renders on the organization's behalf. An organization may define a narrower or broader set to match its own policies and regulatory obligations and should align that set with the disposition states used in its Compliance Policy Profile.

Disposition	Meaning
Allow	All required predicates are satisfied; proceed to the next controlled stage. Commit still requires a valid reservation, EAC, and adapter checks.
Allow with Controls	Proceed only after the specified additional control is completed and the transaction is re-evaluated to Allow.
Step-Up	Additional human authentication, approval, quorum, or role validation is required.

Disposition	Meaning
Review	A potential match, elevated risk, conflicting evidence, or a contextual judgment requires an authorized reviewer before the request can be dispositioned. A case is opened or referenced; reviewer authority and confidentiality are preserved.
Hold	Evidence is incomplete, stale, ambiguous, or under review; no execution pending manual resolution.
Reject	The request lacks valid authority, uses a revoked or expired credential, is outside scope, is malformed, or otherwise fails policy; execution is denied but not necessarily legally prohibited.
Prohibit	A mandatory legal or sanctions restriction, or an authorized emergency kill-switch rule, prevents execution.
Stale / Unknown	The system cannot support a current decision within the required freshness and confidence bounds. Fail controlled — hold, review, or deny according to the risk profile — rather than assuming the prior result still holds.

Risk and disposition are independent. A Tier 3 transaction may ultimately be allowed after live status and dual approval. A Tier 1 transaction may still be rejected because the ARP was revoked. An unresolved sanctions name match ordinarily produces Hold or Review; a confirmed applicable prohibition produces Prohibit.

> **Recorded status evidence**
> Every decision should record the signed snapshot or introspection result, status epoch, evaluation time, maximum permitted staleness, and reason the selected verification mode was acceptable. “Verified offline” is therefore a bounded and auditable statement, not a claim of instant universal revocation.

4. Applying the Protocol to Payment Pre-Authorization

Before value moves, the control plane evaluates five distinct questions: who or what is acting; whose authority supports the action; whether the current evidence and policy permit that authority to be exercised; whether capacity is available and reserved; and whether the instruction presented to the rail is exactly the transaction evaluated and authorized.

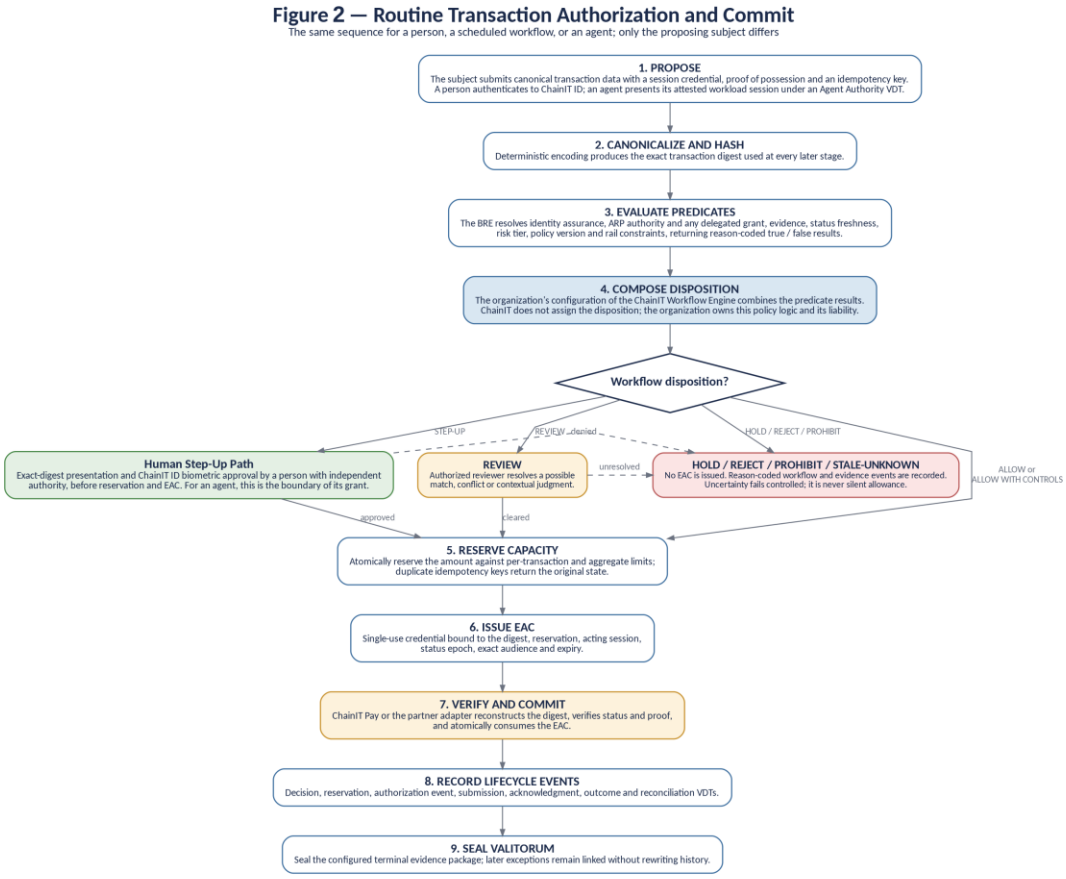
The routine state sequence is: Propose -> Canonicalize -> Evaluate -> Verify Status -> Reserve -> Step-Up if Required -> Re-evaluate -> Issue EAC -> Validate and Consume -> Submit -> Record Outcome -> Reconcile -> Seal.

The disposition set in Section 3.2 is illustrative of one execution configuration an organization might implement in the ChainIT Workflow Engine, not a set ChainIT dictates. For that illustrative set, the execution effect of each disposition is as follows.

- Allow: continue to the next controlled stage.
- Allow with Controls: proceed only after the specified additional control is completed, then re-evaluate to Allow.
- Step-Up: complete the required human or quorum action, then re-evaluate to Allow.
- Review: pause execution pending authorized review; re-enter evaluation after a recorded result.
- Hold: pause execution while evidence, status, or another prerequisite remains unresolved.
- Stale/Unknown: obtain fresh status or evidence, or apply the organization's configured fail-controlled outcome.

- Reject: terminate the current execution attempt.
- Prohibit: terminate execution and apply any required legal, sanctions, incident, or kill-switch process the organization has configured.

Only a final Allow disposition permits EAC issuance.



Companion Figure 2. Routine Transaction Authorization and Commit.

The sequence is the same for a person, an organization initiating through a scheduled workflow, or an agent. A session credential permits proposal activity; only the one-time EAC authorizes commit of the exact canonical transaction.

4.1 Subject Profiles: Human, Organizational, and Agent Initiation

The control sequence above is subject-neutral. What changes across subjects is the front of the sequence (how the proposal originates and how the initiating subject is authenticated) not the authority, evidence, policy, digest, reservation, execution, or outcome-evidence controls that follow it.

Human initiation. A finance director prepares or approves a payment directly. She authenticates to ChainIT ID at the assurance level the Authority Policy Profile requires, her authority is resolved against the current ARP version, the canonical transaction object is constructed from the instruction actually being submitted, and capacity is reserved before an EAC is issued. The controls that matter most in ordinary fiat operations, that the approval is bound to the exact ACH or wire instruction transmitted, that a changed beneficiary account produces a different digest and therefore a new decision, and that two concurrent releases cannot draw on the same limit, are protocol controls, not agent controls.

Organizational or multi-party initiation. A shared-service team, a scheduled disbursement run, or a treasury workflow initiates on behalf of the entity. Authority is resolved to the acting representatives and to the quorum or separation-of-duties rule the ARP imposes, and each required approver approves the same

digest. The Human Approval VDTs are separate, individually attributable events; a batch approval that is not bound to the individual canonical transactions does not satisfy the RequiredApprovalsSatisfied predicate.

Agent initiation. A registered workload prepares the proposal under an Agent Authority VDT granted by a principal with authority to delegate. The additional controls are those described in Sections 2.6, 2.8 and Section 6: registration, workload attestation, an ephemeral session key, and the sovereignty boundary. Authority resolution, session eligibility, evidence, policy, digest, reservation, approval and execution are unchanged. Within the granted band the sequence completes without human interaction; beyond it, the step-up path in Section 5.1 applies and a person with independent authority approves the exact digest. From the canonical transaction object onward, the sequence is identical to human initiation.

In every case the rail adapter enforces the same object: a one-time Execution Authorization Credential bound to a digest the adapter independently reconstructs. The rail profiles that follow describe how that enforcement is achieved on fiat rails and on-chain, and they apply to all three subject classes.

4.2 Fiat Rail Profiles

> **Authority Protocol adapter with partner rail integration.**

This profile is the ordinary fiat case and does not presuppose an agent: the same preflight authorization, orchestration gate, or authorization call governs a wire a person releases. Existing ACH, wire, card, instant-payment, and bank interfaces do not universally accept arbitrary ChainIT credentials. The protocol therefore uses a rail-specific enforcement profile rather than assuming that the EAC is inserted into every native payment message. Reasonable models include a preflight authorization API, a ChainIT Pay orchestration gate, a bank or processor authorization call, a sidecar evidence object, or a compact authorization reference in supported metadata or ISO 20022 extensions.

The authoritative adapter must reconstruct the canonical transaction digest from the rail instruction, validate the EAC audience, session-key proof, status epoch, reservation, policy decision, approvals, expiration, and one-time-use state, and then consume the EAC through a recoverable commit path before or atomically with submission.

4.3 Stablecoin and On-Chain Profiles

> **Authority Protocol adapter with custody, wallet, smart-contract, or network integration.**

For stablecoin or tokenized value, the control may be enforced by a policy-controlled custody wallet, smart-contract gate, account abstraction flow, relayer, paymaster, controlled signing service, or off-chain preflight tied to an approved transaction hash. The canonical object must bind, at minimum, the chain or network, token contract or asset, source wallet, destination wallet, amount, nonce, contract method or calldata hash, fee or gas policy, finality policy, and expiration.

4.4 Outcome Events, Idempotency, and Unknown State

Rail acknowledgments and outcomes must be authenticated and idempotent. A signed or mTLS-authenticated event should identify the ChainIT transaction and digest, rail transaction identifier, event type, sequence or ordering metadata, occurrence and receipt times, amount, currency or asset, reason code, prior-event reference, and idempotency key. Duplicate or out-of-order delivery must not create a duplicate payment or rewrite prior evidence.

A timeout after possible submission is an unknown state, not an instruction to retry blindly. The adapter must query or reconcile by idempotency key and rail reference before any new submission. This behavior is a core control, not a documentation detail.

5. Exact-Digest Approval and Append-Only Lifecycle Evidence

5.1 Human Approval and Assurance

> **Biometric identity, exact-digest presentation, and subject-transaction binding operate together: the person approves the instruction, not a screen.**

When policy requires human action, the approver must possess independent organizational authority for the proposed transaction. Approval cannot cure a lack of authority. The approval experience presents the material transaction fields, risk warnings, relevant evidence or exceptions, and the approving role. The person then completes the authentication assurance required by policy and approves or denies the exact digest. This requirement is the same whether the proposal originated with another person, an organization's scheduled workflow, or an agent: the approver approves a digest, not a screen, and approval of a different digest is not approval of this transaction. Where an agent proposal has crossed the step-up boundary set in its Agent Authority VDT, this is the control that resolves it. The approval satisfies a separate transaction-specific authority path under the ARP; it does not widen the agent's standing grant, and the next transaction is evaluated against the same unchanged bounds.

Assurance is a property of the authentication method achieved - factors, phishing resistance, key protection, replay resistance, and identity binding - not simply whether the interaction occurred in an app, browser, or messaging channel. A browser flow may be high assurance; an SMS-only flow ordinarily is not. Higher-risk or irreversible transactions should require the approved high-assurance profile. [11]

5.2 Canonical Transaction Digest: What You See Is What You Authorize

A canonical transaction object is encoded deterministically before hashing. The profile must define field names, schema version, decimal and currency representation, timestamp format, Unicode normalization, null handling, ordering, and amendment rules. RFC 8785 is one available model for deterministic JSON serialization. [9]

Illustrative minimum digest inputs

Digest = Hash(schema_version | payer | source | payee | destination | amount | currency_or_asset | rail | purpose_or_invoice_hash | fees | nonce | not_before | expiration | rail_specific_parameters)

Any material change requires a new object, digest, policy decision, reservation, approval, and EAC as applicable.

The same digest is bound to the Transaction Policy Decision VDT, Reservation VDT, Human Approval VDT, EAC, Execution Authorization Lifecycle VDTs, rail instruction, and lifecycle evidence. The rail adapter independently reconstructs and compares it before submission.

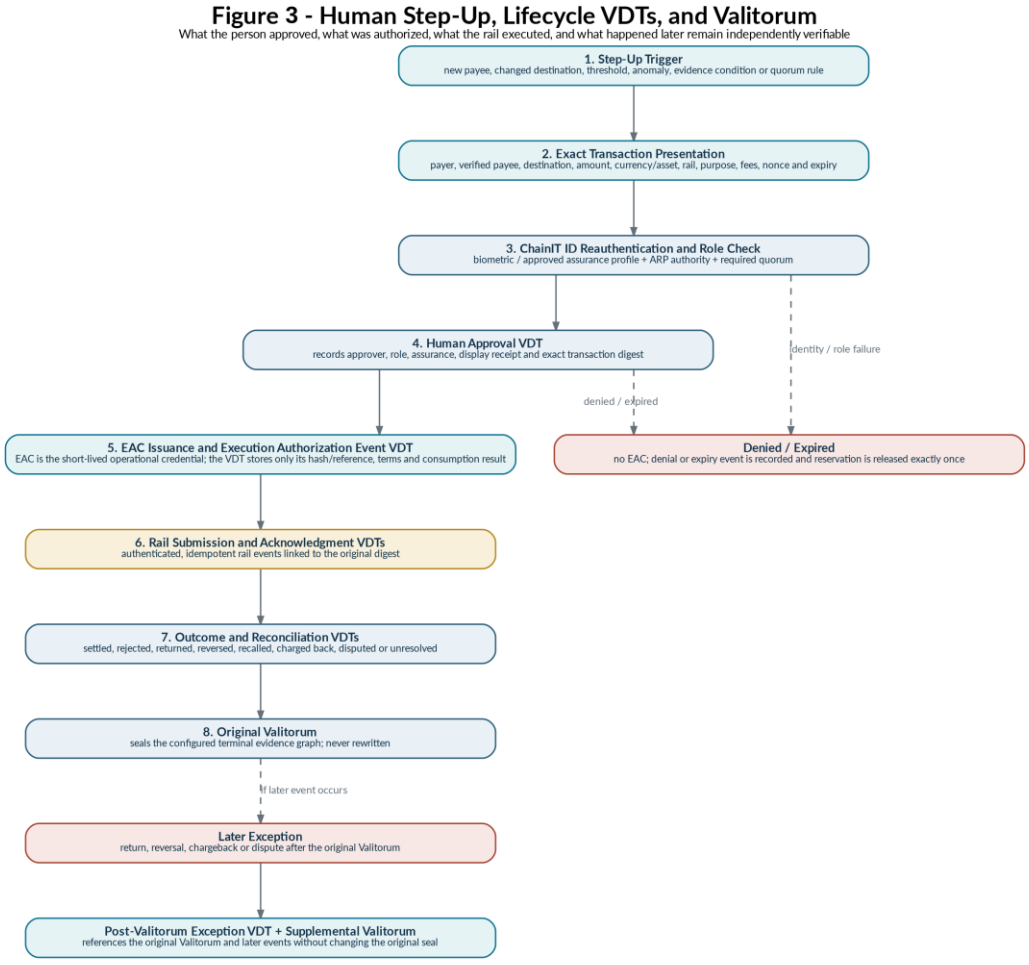
5.3 Operational EAC Versus Permanent Evidence

The EAC is a short-lived operational credential. It is not stored as a reusable ledger artifact. The permanent evidence layer uses Execution Authorization Lifecycle VDTs. At minimum, a terminal event is written when the EAC is consumed, expires, is canceled, or is rejected; it records the EAC identifier or cryptographic hash, issuer, audience, transaction digest, reservation, status epoch, approvals, issuance and expiration times, validation result, terminal time, and linked submission reference. Where contemporaneous evidence at

issuance is required, the system creates separate immutable EAC Issuance and EAC Consumption or Disposition VDTs. No earlier VDT is updated, and no private key or reusable credential is included.

5.4 Append-Only Lifecycle VDTs and Valitorum

A sealed record is never modified. The transaction is represented by linked, immutable events: Payment Intent, Session or Transaction Decision, Reservation, Human Approval or Denial, Execution Authorization Issuance and Consumption or Disposition events, Rail Submission, Rail Acknowledgment, Settlement or other outcome, Reconciliation, and any exception event. Each event references the relevant parent or predecessor and preserves the source, time, signature, digest, and privacy classification.



Companion Figure 3. Human Step-Up, Lifecycle VDTs, and Valitorum.

The EAC is operational authority. Execution Authorization Lifecycle VDTs are permanent evidence. The original Valitorum never changes.

The Valitorum Service seals the completed or otherwise terminal evidence graph under a defined completeness rule. If a later return, reversal, chargeback, recall, dispute, or post-settlement exception occurs, a new linked event is created. The original Valitorum remains immutable. The Valitorum Service may seal a Supplemental Valitorum over the later exception graph, referencing the original Valitorum so a relying party can verify both the original outcome and the subsequent change.

Pactvera and the Valitorum Service provide structured evidence of identity, authority, terms, approvals, events, and outcomes. They do not by themselves guarantee enforceability, evidentiary admissibility, or allocation of liability in every jurisdiction or rail relationship.

6. The Agent Subject Profile: Agentic Commerce and the Sovereignty Boundary

Everything to this point describes the Authority Protocol, which applies to any subject and any rail. This section describes the Agent Subject Profile: the identity and containment controls that exist only because the acting subject is autonomous. The agent's authority itself is not part of this profile. It is delegated under the ARP and evidenced by the Agent Authority VDT described in Section 2.3, in the same way authority is delegated to a human representative.

Agentic commerce requires two-sided verification. A relying party needs confidence that the workload presenting the request is the registered workload and that a verified, currently authorized human or organization stands behind the requested action. A legitimate agent identity can still act outside scope; a valid delegation can still be misused by an unregistered or compromised workload. Both sides must be proven together.

The architecture establishes a sovereignty boundary. An AI model or autonomous agent may reason, gather evidence, compare invoices, prepare terms, and propose an action. It does not receive unrestricted access to payment-account credentials, custody keys, or final execution authority. A deterministic control plane resolves authority, evidence, policy, status, capacity, and approval before issuing the one-time EAC that the rail adapter enforces. Section 8.2 works through this boundary end to end: an agent proposing routine invoices within its granted band, and the same agent's proposal for a new, higher-value vendor crossing into the biometric exact-digest approval path described in Section 5.1.

A verifier can trust a bounded set of ChainIT issuer keys and status services rather than maintaining every agent key independently. The cryptographic work per verification remains approximately constant, although total operational cost scales with transaction volume, evidence refresh, status distribution, support, and partner processing. Comparative economics therefore depend on deployment profile.

7. Design Principles, Credential Lifetimes, and Residual Risk

7.1 Zero-Trust Design Principles

No implicit trust from prior success

A prior identity proof, session, policy decision, or approval is not assumed to remain valid beyond its scope and freshness. The system reevaluates the predicates required for the present context.

Least privilege and intersection of constraints

Effective authority is the intersection of human, organization, ARP, agent, counterparty, jurisdiction, policy, budget, and time constraints - never their union.

Separate identity, authority, evidence, policy, and execution

Each artifact answers a defined question and can be refreshed, revoked, or disclosed independently.

Authority is separate from evidence and policy

The ARP records organizational power and delegation. Evidence and compliance state govern whether that power may be exercised now. A risk indicator does not erase a valid delegation, and a generous delegation does not override a mandatory prohibition.

Organization-owned policy logic

The BRE returns predicate results; the organization's configuration of the ChainIT Workflow Engine assigns the disposition. ChainIT supplies the evidence, the predicate engine, and the workflow tooling. The policy logic remains the organization's. The organization is responsible for its use of the resulting disposition, subject to applicable law and contractual allocation among ChainIT, the organization, payment providers, evidence providers, and other participating parties.

Sender-constrained operational credentials

SACs and EACs require proof from the bound session key. Copied credentials are insufficient without the key. [7]

Bounded status freshness

Offline verification is accepted only within the risk-tier limit; live introspection is required when stale or high risk. [8][10]

Atomic reservation and idempotency

Capacity is reserved before EAC issuance; retries return the authoritative prior result rather than committing a second payment.

Fail controlled

Invalid signature, unknown key, stale status, digest mismatch, unavailable required evidence, or ambiguous rail state produces Hold, Reject, Prohibit, or reconciliation - not silent acceptance.

Privacy by minimum disclosure

Counterparties receive the decision and proof needed for their purpose, not raw biometrics, government records, SAR information, or unrelated identity details.

7.2 Credential and Record Lifetimes

Authority Protocol artifacts apply to every subject class. Only the Agent Registration VDT, the workload key, and the ephemeral session key are Agent Subject Profile artifacts. The Delegated Authority VDT and its agent-specialized form the Agent Authority VDT, the Evidence Assertion VDT, the Session Eligibility Decision VDT, the SAC, the Transaction Policy Decision VDT, capacity reservation, EAC, Execution Authorization Lifecycle VDTs, and Supplemental Valitorum are protocol-general; their inputs vary by subject class.

Identity, Authority, and Session Artifacts

Credential or record	Layer	Typical lifetime or state	Control boundary
Individual VDT / ChainIT ID	Platform foundation	Durable, versioned identity anchor; subject to refresh, suspension, revocation, and recovery.	User-controlled credential architecture and ChainIT verification services, as implemented.
Organization VDT / ChainIT Org ID	Platform foundation	Durable, versioned entity anchor.	ChainIT Org ID and governed organization records.
ARP	Platform foundation	Versioned; effective until amended, superseded, expired, or revoked.	Organization authority domain.
Delegated Authority VDT	Authority Protocol	Bounded period; validity depends on governing ARP status.	Portable reference held or presented by authorized component.

Credential or record	Layer	Typical lifetime or state	Control boundary
Agent Authority VDT	Authority Protocol	Append-only grant record; each amendment, suspension, or revocation is a new linked entry. Present validity resolves at the reliance event against current ARP status.	Granting principal under the ARP; ChainIT evidence ledger.
Evidence Assertion VDT	Authority Protocol	Source-specific effective and expiration periods.	Evidence service; selective disclosure.
Session Eligibility Decision VDT	Authority Protocol	One session context; expires and is re-derived.	Authority gateway / BRE.
Agent Registration VDT	Agent Subject Profile	Versioned across deployments and key rotations.	Agent registry.
Workload key	Agent Subject Profile	Rotatable; protected in approved TPM/TEE/KMS/HSM or equivalent boundary.	Registered workload security boundary.
Ephemeral session key	Agent Subject Profile	One session or shorter.	Authorized by workload key; used for request proofs.
SAC	Authority Protocol	Minutes to hours; proposal and API scope only.	Session and authority gateway.

Transaction, Execution, and Evidence Artifacts

Credential or record	Layer	Typical lifetime or state	Control boundary
Transaction Policy Decision VDT	Authority Protocol	One exact transaction and limited evaluation window.	BRE / transaction control plane.
Capacity reservation	Authority Protocol	Until commit, release, cancellation, or expiration.	Reservation ledger.
EAC	Authority Protocol	Seconds to minutes; single-use.	Issued to session; consumed by rail adapter.
Execution Authorization Lifecycle VDT(s)	Authority Protocol	Permanent immutable evidence events for issuance and terminal handling.	ChainIT evidence ledger.
Lifecycle VDTs	Platform foundation	Permanent once written; new events represent later state.	Pactvera / evidence ledger.
Valitorum	Platform foundation	Permanent immutable seal over the configured terminal evidence graph.	Valitorum verification package.
Supplemental Valitorum	Authority Protocol	Permanent seal over a later exception graph referencing the original Valitorum.	Post-terminal exception verification package.

7.3 Residual Risk and Non-Goals

Provable Authority is designed to prevent, limit, or expose specified classes of unauthorized execution. It does not eliminate:

- A properly authorized person being deceived about the business purpose, beneficiary, or legitimacy of an invoice.
- Coercion, collusion, or malicious conduct by people who satisfy the formal authority and approval rules.

- Incorrect, delayed, or incomplete data from an authoritative source or evidence provider.
- Compromise of ChainIT issuer infrastructure, an HSM or KMS administrator, a payment partner, custody system, or payment rail.
- Compromise of a legitimate workload while it remains able to use its protected key.
- Residual exposure accepted by a verifier that uses a signed status snapshot up to the approved maximum staleness.
- Irreversible settlement after a valid but fraudulently induced approval.

> **Claims discipline**
>
> Provable Authority is designed to reduce unauthorized, stale, replayed, substituted, duplicate, and out-of-scope transaction risk. It is not fraud-proof, non-spoofable, unstealable, tamper-proof, or instantaneously revoked everywhere, and it is not described as such. The residual risks above are stated so that a deploying organization can control for them.

8. Illustrative Scenario

Acme Corp verifies its organization through ChainIT Org ID. Its governing body executes an ARP granting Jane, the finance director, personal authority to approve vendor payments up to $100,000 per transaction, subject to a second authorized approver above $25,000. The same ARP permits Jane to delegate a much narrower autonomous execution authority to a registered procurement agent: approved cloud-services vendors only, up to $5,000 per transaction and $25,000 per month, in U.S. dollars, with no redelegation. The ARP separately permits the agent to prepare and route higher-value or new-payee proposals for human decision, but not to execute them without the required authorized approvers.

8.1 Human-initiated wire under the same ARP

Jane initiates a $60,000 wire to a long-standing supplier. No agent is involved. She authenticates to ChainIT ID at the high-assurance profile the Authority Policy Profile requires for an irreversible rail. The BRE evaluates her identity assurance, her authority under the current ARP version, payee and destination evidence, sanctions and entity evidence within their permitted freshness windows, and the second-approver requirement that applies above $25,000. Acme's configuration of the ChainIT Workflow Engine composes those predicate results into a Step-Up disposition.

The canonical transaction object is built from the instruction that will actually be transmitted, including the beneficiary bank account. Capacity is provisionally reserved before the approval request is issued to Jane and the second approver, so that no concurrent request can consume the same capacity while approval is pending. After both authorize the same digest, policy and status are revalidated, the reservation is confirmed, and a one-time EAC is issued to the wire adapter, which independently reconstructs the digest before submission. Had the beneficiary account been changed after the first approval (the ordinary vendor-impersonation pattern) the digest would differ, the existing approvals would not apply, and a new decision, reservation, and approval cycle would be required. None of these controls depends on an agent being present.

8.2 Agent-initiated payment within the delegated scope

The agent is registered under Acme's Org ID. Jane's delegation is recorded as an Agent Authority VDT referencing the governing ARP version, naming her as the granting principal, and fixing the approved-vendor scope, the $5,000 and $25,000 limits, the currency, the no-redelegation rule, and the thresholds above which biometric step-up is required. Its workload key is attested and authorizes an ephemeral session key. ChainIT creates a Session Eligibility Decision VDT and issues a sender-constrained SAC permitting invoice

review and payment proposals. The agent does not receive unrestricted access to Acme's bank account, wallet, or rail credentials.

For a routine $3,800 invoice to an established vendor, the agent creates the canonical transaction object and per-request proof. The BRE evaluates the applicable predicates against current status, evidence, and the agent's authorized scope, and Acme's configuration of the ChainIT Workflow Engine creates a Transaction Policy Decision VDT and composes those predicate results into an Allow disposition. The reservation ledger atomically reserves $3,800, and the authority gateway issues a one-time EAC bound to the session key, digest, reservation, status epoch, rail adapter, and expiration. The adapter reconstructs the digest, consumes the EAC once, submits the instruction, and records the authorization, submission, acknowledgment, settlement, and reconciliation events before the Valitorum Service seals the completed evidence graph.

Later, two $4,000 proposals arrive simultaneously while only $5,000 of monthly autonomous capacity remains. The reservation ledger serializes the competing requests. One reservation succeeds; the other receives an insufficient-capacity disposition. Both requests cannot read the same balance and both proceed.

The agent then prepares a $50,000 proposal to a new vendor. The amount is outside the agent's autonomous execution scope but within the proposal-routing authority and Jane's personal approval authority. The Transaction Policy Decision VDT requires live status, verified payee and destination evidence, a provisional capacity reservation, Jane's exact-digest approval, and the second authorized approver required above $25,000. After both authorized people approve the same digest, the system revalidates policy and status, confirms the reservation, and may issue a one-time EAC. Human approval does not retroactively expand the agent's standing delegation; it satisfies a separate transaction-specific authority path under the ARP. The Agent Authority VDT is unchanged, and the next proposal is evaluated against the same $5,000 and $25,000 bounds.

8.3 Exception handling under either subject profile

If a routine vendor later produces a possible sanctions name match, the transaction is held for resolution rather than automatically treated as prohibited. If the match is resolved as false, a new decision may allow execution. If an applicable prohibition is confirmed, the disposition is Prohibit and no EAC is issued. If the signed status snapshot for the vendor's entity or sanctions evidence ages past its approved bound at the moment of reliance, the disposition resolves to Stale/Unknown rather than proceeding on the cached result; the verifier must obtain a live status introspection before the proposal can be re-evaluated to Allow. If a settled payment is later returned, the Settlement VDT remains unchanged; a linked Return VDT and, when configured, a Supplemental Valitorum records the later event.

9. Conclusion

The control question in this paper is not specific to agents. It is whether an identified subject (a person, a group acting under quorum, an organization, or a permitted workload) is acting under a current organizational delegation, within the applicable evidence and policy state, against available capacity, on the same transaction that was evaluated and approved. That question often cannot be independently answered outside the systems that enforce the underlying authority, approval, and capacity controls, none of which is ordinarily bound to the instruction that finally reaches the rail. Agentic commerce does not create the question. It removes the human who would otherwise have noticed that the answer was wrong.

ChainIT Provable Authority supplies a structured answer by connecting verified identity, organizational authority, sourced evidence, context-specific policy, transaction semantics, human approval, payment execution, and immutable outcome evidence. Applied to autonomous workloads, the Agent Subject Profile additionally keeps AI outside unrestricted private-key and final execution control while allowing agents to

propose and coordinate real economic activity. Where the companion Compliance Decision VDT supplies the evidence-and-policy predicate, the Authority Protocol supplies the authority resolution, transaction binding, capacity control, approval, execution, and sealed outcome evidence around it. [17] The result is not merely proof that a credential was presented. It is verifiable evidence of who or what acted, whose authority supported the action, what was authorized, which rules and evidence were applied, how capacity was controlled, and what ultimately occurred.

References and Source Basis

The architecture is grounded in the ChainIT patent and application materials and in the official technical and regulatory references listed below. External references are used for protocol design and regulatory framing, not as a substitute for deployment-specific legal or compliance review.

[1] Portable Self-Sovereign Digital Identity with Immutable Trust & Privacy-Preserving Verification, ChainIT application materials, including authoritative-source identity, biometric authentication, selective disclosure, portable verification, and credential update/revocation concepts.

[2] Pactvera Applications A and B, ChainIT application materials, including ChainIT ID, ChainIT Org ID, Authority Resolution Pactvera, BRE-controlled execution, VDTs, conditional value transfer, transaction validation, and Valitorum.

[3] System and Method for Managing Verified Digital Identities, Tokenized Assets, and Conditional Transactions in a Blockchain-Based Marketplace, ChainIT application materials, including Validated Data Token terminology, ARP, BRE, Pactvera, ZKP, and Valitorum.

[4] U.S. Patent No. 11,216,772 B2, Use of Blockchain-Based Distributed Ledger and Smart Contracts for a Construction Project, including condition-driven electronic payment and immutable ledger records.

[5] U.S. Patent No. 11,288,761 B2, Decentralized System for Verifying Participants to an Activity, including authenticated participants, devices, location, time, metadata, and immutable event records.

[6] U.S. Patent No. 12,321,940 B2, System for Verification and Management for Digitally Cash Transactions, including payment records, transaction participants, amount, location, time, and persistent storage.

[7] IETF RFC 9449, OAuth 2.0 Demonstrating Proof of Possession (DPoP), https://www.rfc-editor.org/rfc/rfc9449.html.

[8] IETF RFC 7662, OAuth 2.0 Token Introspection, https://www.rfc-editor.org/rfc/rfc7662.html.

[9] IETF RFC 8785, JSON Canonicalization Scheme (JCS), https://www.rfc-editor.org/rfc/rfc8785.html.

[10] W3C Recommendation, Bitstring Status List v1.0: Privacy-Preserving Status Information for Verifiable Credentials (May 15, 2025), https://www.w3.org/TR/vc-bitstring-status-list/.

[11] NIST Special Publication 800-63B-4, Digital Identity Guidelines: Authentication and Authenticator Management (July 2025), https://csrc.nist.gov/pubs/sp/800/63/b/4/final.

[12] FFIEC BSA/AML Examination Manual, Politically Exposed Persons, https://bsaaml.ffiec.gov/manual/RisksAssociatedWithMoneyLaunderingAndTerroristFinancing/20.

[13] U.S. Department of the Treasury, Office of Foreign Assets Control, FAQ 401 and the 50 Percent Rule, https://ofac.treasury.gov/faqs/401.

[14] 31 U.S.C. § 5318(g)(2); Financial Crimes Enforcement Network Advisory FIN-2010-A014, Maintaining the Confidentiality of Suspicious Activity Reports, https://www.fincen.gov/resources/advisories/fincen-advisory-fin-2010-a014; and Financial Crimes Enforcement Network Advisory FIN-2012-A002, addressing confidentiality of the existence or non-existence of a SAR, https://www.fincen.gov/resources/advisories/fincen-advisory-fin-2012-a002.

[15] U.S. Equal Employment Opportunity Commission, Arrest and Conviction Records: Resources for Job Seekers, Workers and Employers, https://www.eeoc.gov/arrestandconviction.

[16] ChainIT Pay official product materials, About ChainIT Pay and Money Movement, describing ChainIT Pay as the ChainIT ecosystem's money-movement layer across supported bank, card, instant-payment, wallet, and stablecoin rails; integration varies by deployment, https://chainitpay.com/about/ and https://chainitpay.com/money-movement/.

[17] ChainIT, Provable Compliance: A Protocol for Runtime Compliance Evidence and Transaction-Specific Decisioning, CIT-PC-WP-3.0, companion white paper, including modular Evidence VDTs and source classes, the Compliance Policy Profile, the Compliance Decision VDT, Workflow Engine disposition configuration, source-specific freshness, and the May Execute predicate.