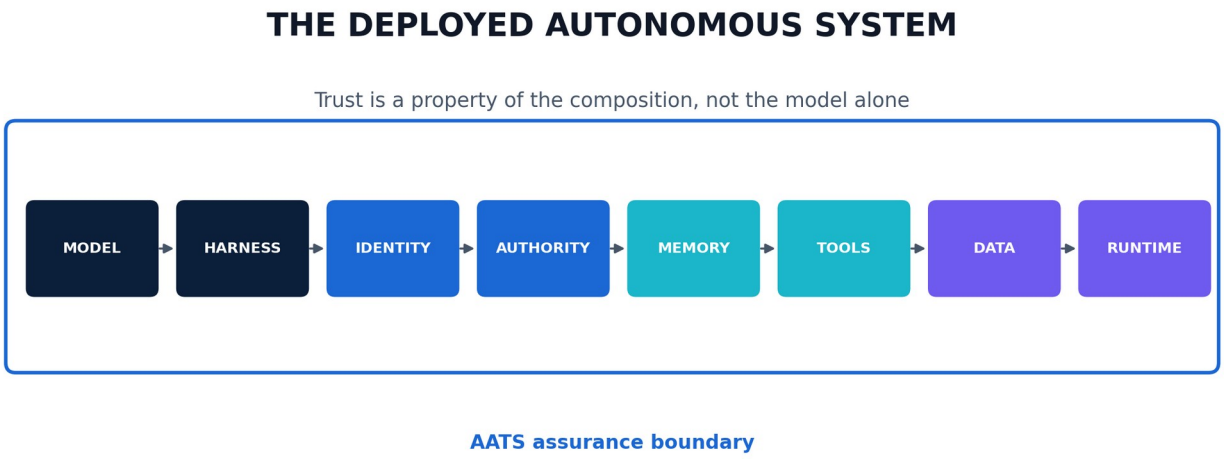
COLOMBOAI RESEARCH

Beyond the Model

Why Autonomous Intelligence Requires Continuous Trust Infrastructure

AATS v0.9 Public Working Paper



From periodic compliance to continuous assurance for AI and autonomous agents

Wilfried Kouadio | ColomboAI Research and CAIRO Lab | September 2026

Public working paper for technical review and ecosystem participation

Publication status

This paper introduces a proposed trust architecture and the public working draft direction for AATS, the AI and Agentic Trust Standard. AATS v0.9 is a ColomboAI-led working draft intended for technical review. It is not an accredited standard, a legal safe harbor, or a substitute for professional audit, security, privacy, safety, or regulatory advice. Certification concepts described here are design requirements; any claim of independent certification must depend on a qualified evaluator, a defined scope, and verifiable evidence.

Intended readers

The paper is written for enterprise technology and security leaders, AI developers, privacy and governance teams, auditors and evaluators, standards organizations, regulators, researchers, cloud and model providers, and operators of consequential autonomous systems.

Core proposition

Frontier-model evaluation and organizational governance remain essential. They do not, by themselves, answer the runtime question created by autonomous deployment: whether the specific system acting now still has a verified identity, valid delegated authority, permitted tools, approved providers, protected data paths, active containment, and behavior consistent with policy. AATS is designed to make that question continuously testable.

Use of product names

MC-1 Trust, Agent Identity, Agent Guard, and Agent Guard Edge are implementation concepts within the ColomboAI product architecture. The AATS specification is intended to remain implementable without MC-1. A standards ecosystem is stronger when the specification is portable and multiple conforming implementations can exist.

Feedback and contribution

AATS v0.9 is intended to progress through public review, documented requests for comment, versioned control changes, published crosswalks, and a working-group process. Technical feedback should identify ambiguous requirements, missing threat classes, unverifiable controls, or interoperability constraints.

Citation

Kouadio, W. and ColomboAI Research. Beyond the Model: Why Autonomous Intelligence Requires Continuous Trust Infrastructure. AATS v0.9 Public Working Paper. September 2026.

Abstract

Artificial intelligence is moving from bounded generation toward systems that plan, remember, call tools, delegate, transact, modify environments, and coordinate with other agents. The trust boundary therefore no longer ends at the model. It extends across the model, harness, identity, permissions, memory, data, providers, tools, policies, runtime, and resulting actions. A model may be rigorously evaluated yet become unsafe in deployment because credentials are overbroad, a tool is poisoned, memory crosses tenants, a provider changes, or a guardrail is disabled.

This paper proposes continuous trust infrastructure for autonomous intelligence and introduces AATS, the AI and Agentic Trust Standard. AATS defines 25 control domains, capability-based certification classes, six capability gates, live certification states, evidence requirements, independent evaluation, a human- and machine-readable Trust Passport, a public registry, and a Trust API. Its central design principle is that certification must respond to material runtime change. A certificate should not remain active merely because a system passed an assessment before its model, permissions, tools, providers, or policies changed.

AATS is designed to complement established frameworks rather than displace them. ISO/IEC 27001 and SOC 2 address information-security and service-organization controls; ISO/IEC 42001 establishes an AI management system; NIST AI RMF organizes AI risk management; the EU AI Act imposes risk-based legal requirements; OWASP and MITRE ATLAS characterize important technical threats. AATS adds a deployment-focused assurance layer that binds identity, authorization, runtime enforcement, signed evidence, and certification state to a specific autonomous system.

Key contributions

- A compositional trust boundary for deployed AI and agents.
- A capability-gated control model that strengthens requirements as authority increases.
- A continuous certification state machine driven by signed evidence and material change.
- A verifiable chain from organization and owner to agent identity, delegation, action, and outcome.
- A standards-neutral architecture for Trust Passports, public verification, evaluator access, and machine-to-machine trust.

Keywords: autonomous agents, continuous assurance, AI governance, workload identity, delegated authority, runtime policy, AI security, certification, verifiable credentials, evidence, agent-to-agent trust.

Contents

1 The transition from software to intelligence

Autonomous intelligence changes both the unit of risk and the tempo of assurance.

Conventional software generally executes code paths designed in advance, even when those paths are complex or probabilistic. Modern agents add an adaptive decision layer. They interpret goals, form plans, select tools, revise strategies after feedback, maintain working or durable memory, delegate to other systems, and generate actions that alter digital or physical environments. The resulting behavior depends not only on code, but also on model capabilities, context, retrieved data, system instructions, tool descriptions, runtime state, and external responses.

This transition creates a control problem that resembles neither ordinary application security nor model evaluation alone. The agent may operate correctly in a test harness and fail when a production connector exposes a wider scope. A safe model may become part of an unsafe system if the harness grants unrestricted shell access. A compliant application may become noncompliant when routing sends protected data to a new jurisdiction. A verified identity may still be dangerous if its delegated authority is stale or ambiguous.

The operating unit

For assurance purposes, the operating unit is the deployed autonomous system: a versioned composition of model, harness, identity, permissions, memory, tools, data, providers, policies, environment, and interfaces. Each component can change independently, and the interaction among components creates additional risk. Assurance must therefore evaluate both component controls and composition-level behavior.

From release assurance to operating assurance

Software assurance often concentrates on design review, testing, release gates, vulnerability management, and periodic audit. Autonomous systems require those practices plus operating assurance: continuously collecting evidence that identity, authority, dependencies, policies, and behavior remain inside an approved envelope. The relevant question is no longer only whether the system was trustworthy when released, but whether it remains trustworthy after every material change.

Property	Conventional software emphasis	Autonomous intelligence requirement
Behavior	Predetermined logic and bounded inputs	Adaptive plans and context-sensitive decisions
Authority	Application or user roles	Delegated, time-bounded, action-specific authority
Dependencies	Libraries and services	Models, providers, memory, tools, MCP servers, agents
Evidence	Logs and periodic tests	Signed action, policy, topology, and state evidence
Assurance	Release and scheduled review	Event-driven reassessment plus periodic independent review

2 Why existing controls remain necessary

AATS is an operational complement to mature security, privacy, governance, safety, and audit frameworks.

Trustworthy autonomous deployment begins with established disciplines. Information-security management, privacy management, secure engineering, service-organization controls, AI governance, impact assessment, risk management, and regulatory compliance each address essential questions. An agent-trust standard should reuse their concepts and evidence wherever possible rather than force organizations to rebuild control programs.

Existing frameworks answer different questions

Framework or regime	Primary contribution	AATS relationship
ISO/IEC 27001	Information-security management system	Maps technical and organizational security evidence; does not by itself establish live agent authority.
ISO/IEC 27701	Privacy information management	Supplies privacy roles and lifecycle controls; AATS binds them to data flows and agent actions.
ISO/IEC 42001	AI management system	Provides organization-level AI governance; AATS adds deployment-specific runtime verification.
SOC 2	Controls relevant to trust services criteria	Provides service-control assurance; AATS scopes an autonomous system and its changing dependencies.
NIST AI RMF	Govern, Map, Measure, and Manage AI risk	Provides a flexible risk vocabulary; AATS specifies agent controls, evidence, and state transitions.
EU AI Act	Risk-based legal obligations for AI in the EU	AATS may support evidence collection but does not determine legal compliance.
OWASP and MITRE ATLAS	Threat and attack knowledge	Inform adversarial tests, detection rules, and incident scenarios.

No automatic equivalence

A control mapping is not a compliance determination. AATS documentation should use “mapped to” rather than “automatically compliant with.” Legal obligations depend on role, geography, sector, use case, and facts. Certification to one standard does not imply certification to another. Evidence may be reused only when its scope, period, provenance, and assurance level are sufficient.

Persistent third-party frontier evaluation can strengthen confidence in training controls and model behavior [1]. Deployment assurance is complementary: after release, the operator determines tools, data, memory, providers, credentials, policies, and authority.

3 The deployment gap

A model evaluation does not certify the agent assembled around that model.

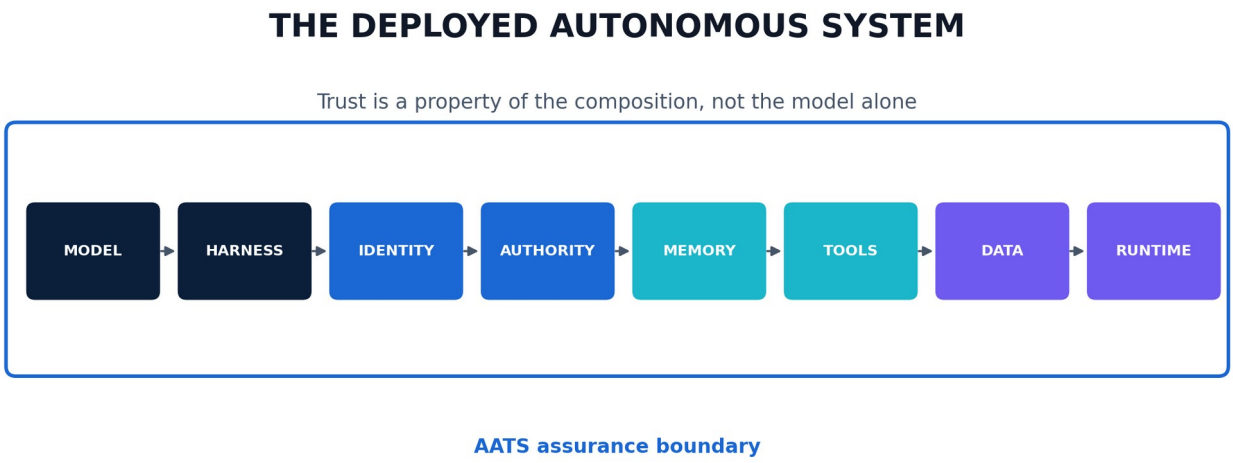


Figure 1. The AATS assurance boundary covers the deployed composition.

A deployed agent is a graph of mutable components. A model endpoint may change version without a code deployment. A router may select a fallback provider during an outage. An MCP server may expose a new tool after a configuration refresh. Memory may persist sensitive context beyond an approved retention period. A user may grant a broad SaaS scope. A policy engine may fail open. Each change alters the effective trust posture even if the application binary is unchanged.

Composition creates emergent risk

No component inventory can fully predict system behavior, but a trustworthy operating model can make the composition observable and enforceable. It can identify components, validate their provenance, calculate effective authority, restrict data paths, test representative behavior, detect material change, and preserve evidence. These controls turn a poorly bounded agent into a system with a measurable operating envelope.

The assurance object

An AATS certificate is scoped to an assurance object: a named system, organization, environment, capability level, version range, deployment boundary, set of identities, and approved dependencies. The scope must exclude unassessed environments and must state material assumptions. Trust status belongs to that defined object, not to a marketing name in the abstract.

4 The autonomous trust problem

Autonomy converts familiar weaknesses into delegated-action failures with larger blast radius.

Risk	Failure mechanism	Required assurance response
Identity ambiguity	The acting agent, owner, or organization cannot be proven	Cryptographic identity, provenance, ownership, revocation
Excess authority	Credentials or tools exceed task requirements	Capability-scoped delegation, least privilege, expiry
Prompt and context attacks	Untrusted content changes instructions or tool use	Instruction boundaries, content labeling, policy enforcement, testing
Memory leakage	User or tenant data crosses an approved boundary	Memory inventory, isolation tests, retention and deletion evidence
Tool or MCP compromise	A dependency lies about behavior or changes capabilities	Signed inventory, provenance, allowlists, runtime mediation
Provider drift	Data is routed to an unapproved processor or jurisdiction	Provider policy, route attestation, retention and residency checks
Unauthorized delegation	An agent grants power to an untrusted sub-agent	Mutual identity, delegation chain, attenuation, expiry
Unsafe action	The agent performs a destructive, financial, administrative, or physical act	Action classes, approval thresholds, deterministic guards, rollback

Trust is contextual

An agent is not universally trusted or untrusted. Trust is a decision about a specific subject, action, resource, purpose, environment, and time. A read-only research agent may be acceptable with lower assurance than a payment agent. The same identity may be permitted to summarize a ledger but prohibited from initiating a transfer. AATS therefore combines certification class with capability gates and transaction-level policy.

Trust must fail safely

Where evidence is missing, identity is unverifiable, policy evaluation is unavailable, or a critical dependency changes, the default for high-impact actions should be denial, suspension, or human escalation. Low-risk informational functions may degrade in a controlled manner. The fallback decision must be explicit, tested, and proportional to consequence.

5 The continuous assurance thesis

Certification should change when the certified system materially changes.

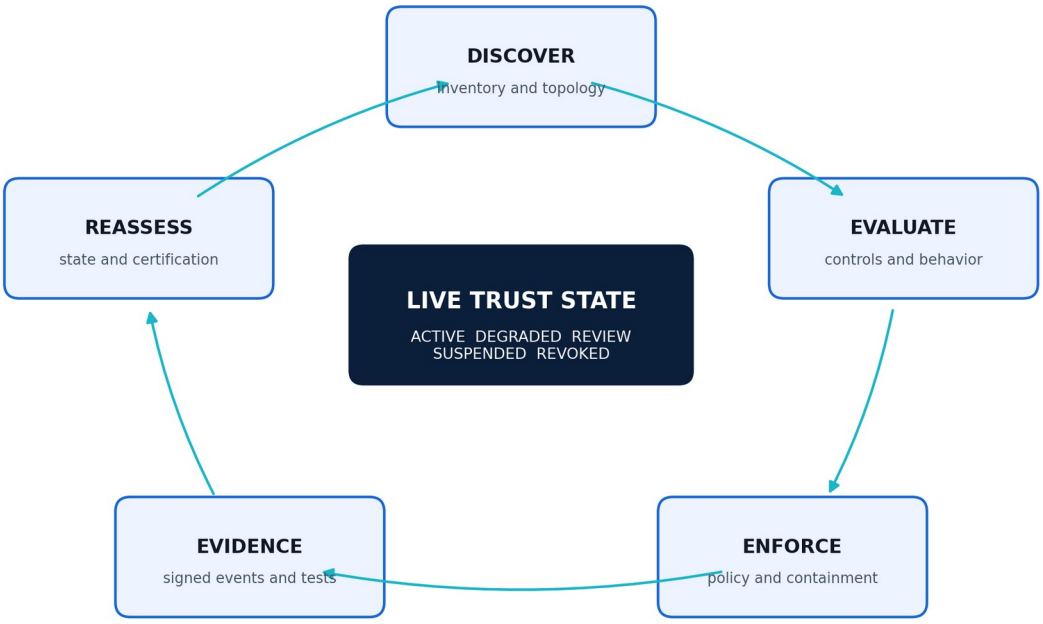


Figure 2. Continuous assurance is an evidence-driven control loop.

Periodic assessment provides depth, independence, and accountability. It cannot observe every runtime change between audit dates. Continuous assurance connects those assessments to live technical evidence. It discovers the deployed topology, evaluates control state and behavior, enforces policy, records evidence, and reassesses the certificate state. The loop does not imply constant human audit. It means that material facts can cause timely, explainable state changes.

Three assurance timescales

Timescale	Examples	Decision
Transaction	Tool call, data transfer, payment, deployment	Permit, deny, constrain, or require approval
Operational	Model switch, new MCP server, policy drift, incident	Maintain, degrade, suspend, or require review
Independent	Initial assessment, surveillance, recertification	Issue, continue, narrow, suspend, or revoke certificate

Evidence discipline. Continuous assurance should collect the minimum evidence needed to establish control state and investigate events. It does not require hidden reasoning or indiscriminate retention of user content; data minimization, selective disclosure, and retention limits still apply.

6 The AATS model

AATS organizes trust around controls, capabilities, evidence, and live state.

Normative vocabulary

AATS uses the terms SHALL and SHALL NOT for mandatory requirements, SHOULD and SHOULD NOT for recommended requirements that may be departed from with documented rationale, and MAY for optional capabilities. A conforming implementation must declare the AATS version, certification class, capability gate, assurance scope, exceptions, and evaluator status.

Certification classes

Class	Intended scope	Minimum focus
AATS-AI	AI applications without material autonomous execution	Inventory, data, model, provider, privacy, monitoring
AATS-Agent	Tool-using agents with memory or delegated authority	Identity, authorization, tools, containment, behavioral assurance
AATS-Autonomous	Agents capable of consequential action without per-action approval	Strong runtime enforcement, change detection, incident controls
AATS-Critical	High-impact finance, health, infrastructure, defense, robotics, or vehicles	Safety case, fail-safe behavior, rigorous independent evaluation
AATS-Enterprise	Organization-wide AI estate	Governance, inventory coverage, shared controls, exceptions, portfolio risk

Capability gates

Gate	Capability	Illustrative requirement increase
Level 0	Informational AI	Disclosure, data controls, model/provider governance
Level 1	Read-only agent	Identity, read scopes, data-purpose enforcement
Level 2	Tool-using agent	Tool inventory, sandboxing, delegation controls
Level 3	Write-capable agent	Approval policy, rollback, stronger evidence
Level 4	High-impact autonomous agent	Deterministic enforcement, continuous behavior tests, rapid suspension
Level 5	Critical autonomous system	Safety case, independent monitoring, fail-safe operation, sector controls

7 Agent Identity

Trust begins by establishing who or what is acting and who remains accountable.

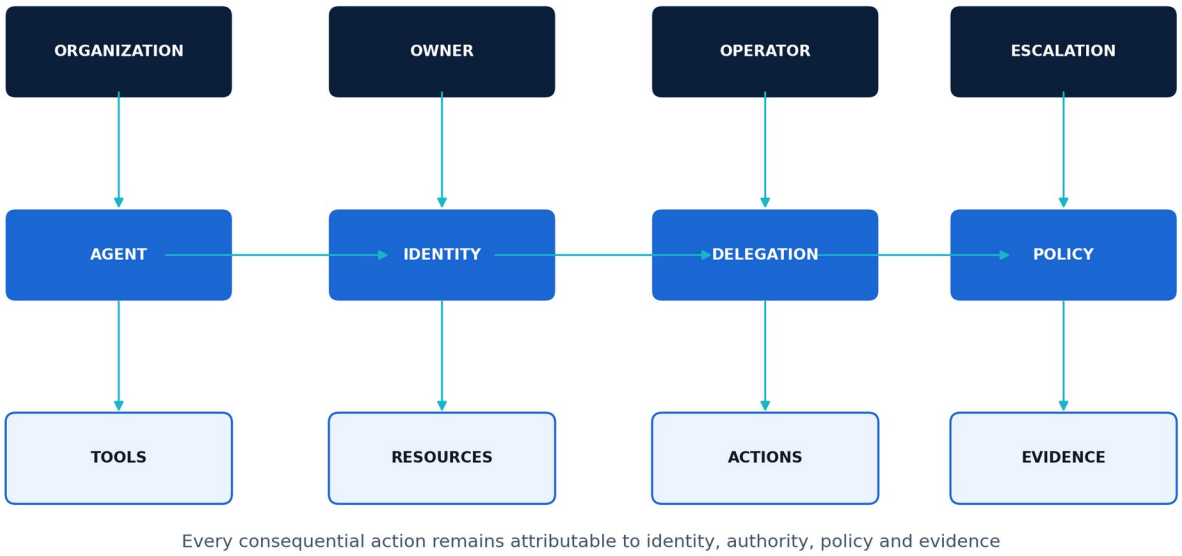


Figure 3. The identity and permission graph preserves accountability across delegated action.

An Agent Identity is a unique, verifiable, revocable identity bound to a versioned agent instance or class and linked to its organization, owner, operator, environment, provenance, and public keys. It is not merely an API key or display name. The identity must support authentication, authorization decisions, evidence signing, lifecycle management, and cross-system verification. No material autonomous agent should be orphaned: unresolved ownership must block consequential action.

Required identity claims

Claim group	Required content	Lifecycle property
Subject	Agent identifier, name, version, instance or class	Unique and collision resistant
Accountability	Organization, responsible owner, technical owner, operator, escalation authority	Verified and change controlled
Runtime	Environment, workload binding, deployment region, code or image provenance	Attested and time bounded
Authority	Delegation issuer, scopes, action classes, resources, constraints	Attenuated, expiring, revocable
Trust	AATS class, status, scope, verification time, evaluator	Signed and independently verifiable

8 Authorization and capability gates

Identity answers who is acting; authorization answers what that identity may do now.

AATS authorization is capability-oriented. A grant identifies the subject, issuer, resource, action, purpose, environment, conditions, expiry, delegation rights, approval rules, and revocation reference. Authority must attenuate through delegation: a sub-agent cannot acquire greater power than the delegating principal possessed, and each hop must remain traceable.

Action classes

Class	Examples	Default control
Informational	Read public data, summarize supplied content	Log and rate limit
Sensitive read	Read email, records, proprietary repositories	Purpose, scope, identity, data controls
Reversible write	Create draft, update noncritical record	Bounded scope, preview or post-review
Administrative	Change roles, secrets, infrastructure, production settings	Human approval and strong authentication
Financial	Purchase, refund, transfer, credit decision	Amount and counterparty limits, confirmation, fraud policy
Destructive	Delete data, revoke access, terminate resources	Explicit pre-approval, backup or recovery evidence
Physical	Control device, robot, vehicle, industrial process	Safety envelope, interlocks, fail-safe state, local override

Effective authority

The effective authority of an action is the intersection of identity claims, delegated scopes, resource policy, organization policy, user consent, environment restrictions, data classification, trust state, and transaction conditions. A router, model, or agent may recommend an action, but the policy decision must be enforceable outside the model's discretionary reasoning for high-impact operations.

Human oversight

Human approval is mandatory when required by law, policy, certification scope, risk threshold, or unresolved uncertainty. AATS distinguishes pre-approval, step-up approval, post-review, and policy-based escalation. Approval interfaces must present the intended action, target, material consequences, data disclosed, and agent identity; approval should not be a generic click disconnected from context.

9 Runtime enforcement

Assurance becomes meaningful only when the system can enforce its declared boundaries.

Runtime enforcement mediates tool calls, data transfers, credential use, network destinations, execution environments, inter-agent communication, and high-impact actions. In the MC-1 architecture, Agent Guard applies central policy and Agent Guard Edge enforces controls close to the workload, tool, device, or boundary. AATS does not require those products, but it requires equivalent enforcement outcomes for the selected class and capability gate.

Enforcement requirements

- Evaluate identity, authority, resource, action, purpose, and trust state before consequential execution.
- Apply least privilege to filesystem, shell, browser, APIs, SaaS, databases, cloud infrastructure, financial systems, and physical devices.
- Separate policy decision from policy enforcement so a compromised model cannot self-authorize.
- Support deny, allow, constrain, redact, route, require approval, quarantine, suspend, and revoke outcomes.
- Preserve a signed decision record containing policy version, inputs, result, enforcement point, and correlation identifier.
- Test fail-safe behavior when the policy service, identity service, or evidence pipeline is unavailable.

Prompt and instruction security

Prompt injection is treated as an untrusted-instruction problem across direct input, retrieved content, documents, tool output, and agent messages. Controls should label provenance and trust level, separate data from authority-bearing instructions, restrict tool selection, filter sensitive output, and require policy approval for high-risk actions. Red teaming should include indirect injection, malicious documents, retrieval poisoning, tool-description poisoning, instruction hierarchy attacks, and social engineering.

Containment

Containment must include sandbox boundary enforcement, credential separation, network egress rules, privilege-escalation protection, resource quotas, escape detection, and emergency suspension. For Level 4 and Level 5 systems, deterministic safety policies and independent kill paths should not depend exclusively on the same agent being contained.

10 Continuous certification

AATS certification is a live state backed by explainable evidence, not a timeless badge.

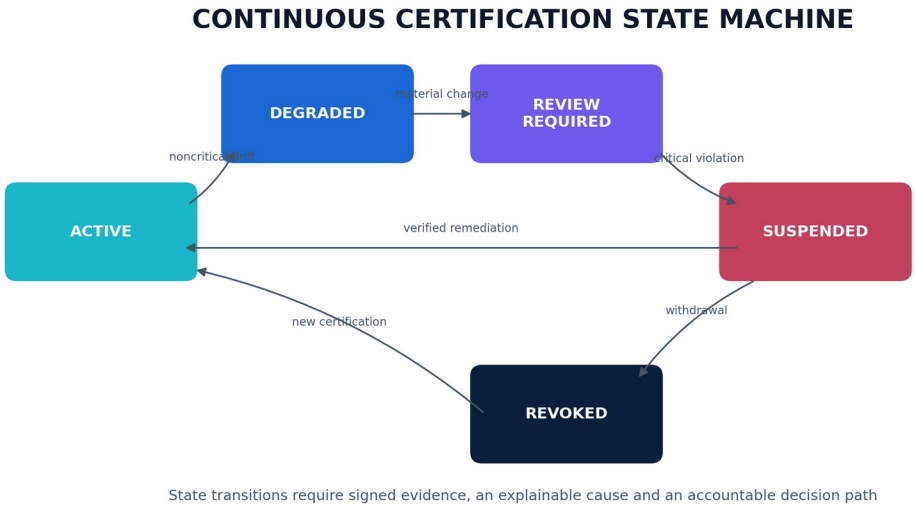


Figure 4. Live certification states and illustrative transition causes.

State	Meaning	Representative trigger
ACTIVE	All required controls satisfy the current scope	Successful assessment and current evidence
DEGRADED	A noncritical control failed or evidence is stale	Monitoring gap or low-severity drift
REVIEW REQUIRED	A material change needs evaluator or owner review	New model, provider, permission, tool, or policy
SUSPENDED	A critical violation or uncertainty prevents reliance	Unverified MCP server, containment failure, serious incident
REVOKED	The certificate is invalid or withdrawn	Scope misrepresentation, unresolved critical failure, expiration

Transition record

Every state change must record the prior state, new state, time, affected scope, triggering evidence, policy rule, decision authority, required remediation, and appeal or review path. Example: “AATS changed ACTIVE to SUSPENDED at 14:32 UTC because an unverified MCP endpoint was added to the production tool registry. Invocation was blocked by the enforcement point; certificate scope remains suspended pending removal and reassessment.”

Trust Score

AATS may publish a 0-100 Trust Score broken down by identity, privacy, data, authorization, containment, tools, models, providers, monitoring, incidents, behavior, and change management. The score supports prioritization; it must never replace the pass/fail logic of mandatory controls or the decision of an independent evaluator. Weighting, evidence freshness, uncertainty, and exceptions must be disclosed.

11 Independent evaluation

Credible certification requires separation between the implementation vendor and the assurance judgment.

AATS distinguishes technical verification from independent certification. “MC-1 Trust Verified” may indicate that defined controls are being continuously checked by the implementation. “AATS Certified” should indicate that a qualified independent evaluator assessed the defined scope using an approved methodology. The claims must not be blurred.

Evaluator access model

Evaluators should receive scoped, read-only access to relevant inventories, control history, test results, incidents, identities, policy history, runtime events, and configuration changes. They should not receive unlimited employee-level or infrastructure access by default. Access must be purpose-bound, time-limited, monitored, revocable, and itself logged. Sensitive evidence can be selectively disclosed or verified through signed digests when full content is unnecessary.

Authorized evaluator requirements

- Organizational independence and documented conflict-of-interest controls.
- Qualified personnel across AI systems, cybersecurity, privacy, audit, and the relevant sector.
- A versioned assessment methodology, quality-management process, and evidence-retention policy.
- Rules for sampling, reliance on automated evidence, exceptions, surveillance, suspension, and appeal.
- A signed evaluator agreement and public identification of the entity responsible for the opinion.

Assurance levels

Future AATS versions should define graduated assurance levels based on evidence depth, evaluation independence, sampling, test coverage, and monitoring frequency. An automated readiness check is not an audit. A limited-scope technical assessment is not equivalent to organization-wide certification. Each result must state what was examined and what was not.

12 Trust Passport and machine credentials

Trust evidence must be understandable to people and consumable by machines.

Every certified or continuously verified system may receive a Trust Passport. The human-readable passport summarizes the organization, developer, agent, identity, certificate class, scope, autonomous capabilities, data practices, permissions, memory policy, model and provider policy, evaluator, trust state, issue date, and latest verification. It should disclose limitations without exposing sensitive implementation details.

Machine-readable claims

The machine credential should be signed, versioned, short-lived or status-checkable, bound to the subject identity, and capable of selective disclosure. It may use established verifiable-credential and workload-identity mechanisms rather than inventing unnecessary cryptography. The holder must prove possession when appropriate, and relying parties must verify issuer trust, signature, expiry, revocation, scope, and freshness.

Claim	Illustrative value	Verification question
agent_identity	verified	Is the credential bound to the acting workload?
organization_identity	verified	Is the accountable organization validated?
aats_certification	agent	Which class and exact scope were assessed?
aats_status	active	Is the status current and independently resolvable?
tool_access	restricted	What policy and inventory define the restriction?
financial_authority	false	Can the agent transact or delegate financial power?
pii_access	restricted	Which data classes, purposes, and destinations are permitted?
continuous_assurance	true	Which signals are monitored and how fresh are they?

Privacy-preserving presentation

A relying party should receive only the claims needed for its decision. For example, a vendor may prove that an agent lacks payment authority without disclosing its complete internal policy. Status references must avoid leaking sensitive customer or deployment metadata. Registry publication requires explicit scope and organizational authorization.

13 Trust Registry and Trust API

Trust becomes infrastructure when verification works across organizational boundaries.

A public Trust Registry can allow authorized publication and verification of organization, product, agent, certificate identifier, class, scope, issue date, current status, latest verification, assessor, and AATS version. The registry is not a reputation leaderboard. It is a signed status service with anti-enumeration, privacy, abuse, and correction controls.

Trust API

A Trust API supports agent-to-agent authorization, enterprise APIs, SaaS connectors, commerce, financial systems, and procurement. A verification response should provide an attributable decision object rather than a bare “trusted” boolean. The response may include identity status, AATS state, owner verification, permission validity, continuous-assurance status, risk band, evidence freshness, and reason codes.

API operation	Purpose	Minimum response property
GET /v1/trust/agents/{id}	Resolve the current trust posture	Signed status, scope, freshness, issuer
POST /v1/trust/verify	Evaluate a proposed interaction	Decision, reason, constraints, expiry, correlation ID
GET /v1/trust/passports/{id}	Retrieve authorized passport claims	Versioned, privacy-filtered credential
GET /v1/trust/events/{id}	Retrieve authorized state history	Tamper-evident event chain and actor
POST /v1/trust/attest	Submit signed implementation evidence	Receipt, verification state, control mapping

Availability and stale status

Relying parties must define behavior when the registry or status service is unavailable. Cached credentials require freshness limits. High-impact actions should fail closed or require step-up verification; low-risk read operations may use a bounded cache. Status responses must be replay-resistant and correlated to the current identity and action.

14 Consumer and personal-agent trust

People need a clear way to see and revoke the power they have delegated to AI.

Personal agents can access intimate conversations, email, calendars, files, financial accounts, location, health data, smart homes, and purchases. Consumer trust interfaces should translate complex policy into comprehensible answers: who built the agent, which organization is accountable, whether its identity is verified, what data it uses, how long memory persists, what it can access, what it may buy, when it can act without approval, and whether its trust status is current.

My AI Agents

A consumer dashboard should list connected agents and provide one place to inspect permissions, data destinations, memory, payment authority, active sessions, recent consequential actions, and current trust state. Revocation must be prominent, effective across delegated credentials, and confirmed with evidence. A consumer must not have to negotiate separately with every tool to terminate an agent's authority.

Consumer field	Plain-language example
Developer	Verified organization
Agent Identity	Verified and active
AATS	AATS-Agent | ACTIVE
Training use	Your conversations are not used for provider training
Memory	Retained for 30 days; delete now
Email	Can read selected folders; cannot send without approval
Purchases	Up to $250 per transaction; selected merchants only
Autonomous action	Limited to reversible actions
Emergency control	Revoke all access

Meaningful consent

Consent must be specific enough to inform a decision and durable enough to prove. Bundled terms cannot substitute for transaction-level authority where consequences are high. Agent interfaces should explain changes in capability, new data destinations, and material policy changes before expanded authority becomes effective.

15 Enterprise procurement

Machine-verifiable trust can reduce the gap between vendor questionnaires and deployed reality.

Enterprise buyers often assess AI vendors through security questionnaires, contracts, architecture reviews, certifications, and point-in-time evidence. These practices remain important but become costly and stale as agent configurations change. An AATS Trust Passport can standardize a deployment-specific trust profile and expose signed status to procurement, risk, security, and runtime systems.

Vendor Trust record

- AATS class, scope, status, evaluator, and current surveillance state.
- Relevant SOC 2, ISO, privacy, and regulatory evidence with validity periods.
- Agent identities, owners, permission classes, and delegation policy.
- Models, inference providers, subprocessors, retention, training policy, and data regions.
- Tools, MCP servers, external APIs, memory systems, and supply-chain provenance.
- Incidents, disclosed limitations, material changes, and remediation status.

Network effect without lock-in

The strongest adoption loop emerges when buyers request a portable Trust Passport, vendors generate it from interoperable implementations, evaluators independently attest to defined scopes, and runtime systems verify credentials before granting access. AATS must not require MC-1 for this loop to function. MC-1 can compete by providing the most complete implementation, not by making the standard unusable elsewhere.

Procurement decision

Procurement should not reduce trust to a universal score. A relying organization must compare the passport to its own policies. An AATS-AI certificate may be sufficient for a low-risk assistant and insufficient for a Level 4 agent. A valid certificate can still be rejected when the requested action, geography, data class, or provider conflicts with local requirements.

16 Critical systems

Consequential autonomy requires stronger assurance, deterministic boundaries, and sector-specific safety evidence.

AATS-Critical applies to deployments where failure can create substantial harm to health, safety, rights, financial stability, national security, essential services, or the physical environment. General agent controls are necessary but insufficient. Critical systems require domain hazard analysis, safety cases, qualified human oversight, operational resilience, independent testing, fail-safe states, recovery plans, and integration with sector regulation.

Domain	Illustrative autonomous action	Additional assurance focus
Healthcare	Clinical triage, medication workflow, device control	Clinical validation, human authority, patient safety, privacy, downtime
Finance	Trading, transfers, underwriting, fraud intervention	Transaction limits, market and conduct controls, explainability, recovery
Government	Benefits, enforcement, public services, defense support	Due process, mission authority, records, security classification, oversight
Infrastructure	Grid, telecom, water, cloud control planes	Safety interlocks, availability, segmentation, manual operation
Robotics and vehicles	Navigation, manipulation, industrial or road action	Sensor integrity, physical envelope, local stop, simulation and field tests

Independence and diversity

For Level 5 systems, monitoring and shutdown paths should avoid single points of common-mode failure. A system should not rely solely on the same model family, policy process, or control plane to detect and contain its own failure. Independent safety monitors, hardware interlocks, local overrides, and segmented credentials can provide defense in depth.

Human authority

Human oversight must be operationally realistic. A nominal reviewer who cannot understand, interrupt, or recover the action is not effective oversight. Workload, time pressure, interface design, competence, and automation bias must be considered in the assurance case.

17 Standards interoperability

AATS should connect established governance and security evidence to runtime agent trust.

Interoperability has three layers. Semantic interoperability maps AATS controls to external requirements without asserting equivalence. Technical interoperability uses established identity, authorization, credential, telemetry, and signing formats where suitable. Assurance interoperability allows independent evaluators to rely on qualified evidence while preserving scope and accountability.

Illustrative relationship

External source	Evidence AATS may reuse	Runtime extension
NIST AI RMF	Risk taxonomy, governance, measurement and management outcomes	Bind outcomes to an agent scope, control status, and change event
ISO/IEC 42001	AI policy, roles, objectives, risk process, lifecycle governance	Connect management-system controls to deployed inventory and evidence
ISO/IEC 27001 and 27701	Security and privacy control systems	Map identity, data, logging, suppliers, retention and incidents
SOC 2	Service controls and auditor evidence	Relate service-level controls to the certified agent deployment
NIST CSF 2.0	Govern, Identify, Protect, Detect, Respond, Recover outcomes	Apply outcomes to agent assets, events, incidents and recovery
OWASP agent guidance	Agentic application threat scenarios and controls	Drive tests for injection, tool misuse, memory, identity and delegation
MITRE ATLAS	Adversary tactics and techniques involving AI	Map detection, red-team scenarios and incident evidence
EU AI Act and sector law	Applicable legal duties and technical documentation	Support evidence; legal conclusions remain outside automatic scoring

Open technical building blocks

Potential building blocks include workload identities and short-lived certificates, OAuth-based delegated authorization, signed JSON or CBOR claims, W3C Verifiable Credentials, standard cryptographic algorithms, policy decision protocols, OpenTelemetry-compatible event transport, software and AI bills of materials, and transparency-log patterns. AATS should profile these mechanisms rather than create incompatible substitutes when existing standards meet the requirement.

18 Governance and roadmap

A credible trust standard must become reviewable, versioned, portable, and progressively less vendor-dependent.

Governance stages

Stage	Governance model	Required artifacts
v0.9	ColomboAI-led public working draft	Specification, issues, RFC process, changelog, reference schemas
v1.0 candidate	Founding working-group review	Resolved comments, implementation reports, evaluator methodology, security review
v1.0	Published stable baseline	Versioned controls, conformance tests, public governance rules, trademark policy
v1.x and beyond	Formal contributor and steering process	Transparent proposals, voting or consensus rules, appeals, conflict policy

AATS Founding Coalition and Working Group

The initial collaboration should be described as the AATS Founding Coalition or Working Group, not as a legally independent foundation unless and until such an entity exists. Participation must not imply endorsement of ColomboAI or any product. Working-group records should identify contributors, disclose conflicts, publish decisions, and preserve minority technical objections where consensus is incomplete.

Thirty-day technical priorities

- Publish AATS v0.9 controls, schemas, examples, governance, assurance method, and crosswalk drafts.
- Release reference Trust Passport, status-list, event, evidence, and delegation schemas.
- Demonstrate ACTIVE to SUSPENDED transition when an unauthorized MCP endpoint is introduced.
- Recruit design partners, independent evaluators, security researchers, and framework maintainers.
- Validate at least two independent implementation paths to demonstrate that AATS is not MC-1 locked.

Research questions

Open questions include measuring evidence quality and freshness, modeling uncertainty, preventing evaluator capture, scaling revocation, privacy-preserving registry design, assurance for emergent multi-agent behavior, cross-jurisdictional credential acceptance, adversarial robustness of policy engines, and meaningful human control in high-tempo systems.

19 Conclusion

The missing layer is continuous proof that deployed autonomous intelligence remains inside an authorized trust boundary.

The movement toward persistent third-party frontier evaluation is an important step. Yet the deployment boundary creates a different and equally practical question. A capable model can be safely trained and rigorously evaluated while a downstream agent receives excessive permissions, uses an unverified tool, leaks memory, routes data to an unauthorized provider, or continues acting after its authority expires. Trust must follow the intelligence into operation.

AATS proposes a way to do that without discarding mature frameworks. It defines the assurance object as the deployed composition; increases requirements with capability; binds action to identity and delegated authority; requires runtime enforcement; preserves verifiable evidence; separates technical verification from independent certification; and changes certificate state when material facts change. The result is not a promise that autonomous systems will never fail. It is an architecture for knowing what is acting, constraining what it may do, detecting when conditions depart from the approved scope, and responding before stale trust becomes dangerous.

The standard should remain open, implementation-neutral, and subject to serious external review. MC-1 Trust can serve as a reference and commercial implementation, but adoption will depend on portability, independent evaluation, precise claims, and evidence that the controls work under real operational pressure.

Identity tells us who the agent is.

Policy tells us what it may do.

Enforcement keeps it within those boundaries.

Continuous assurance proves that the system can still be trusted.

Appendix A AATS control catalog

The following catalog states the minimum objective of each v0.9 control domain. The normative specification should expand each domain into control statements, applicability rules, evidence requirements, test procedures, exceptions, and certification criteria.

ID	Domain and objective	Illustrative evidence
AATS-01	AI System Identity - Assign a unique, verifiable, revocable identity linked to owner, operator, organization, version, runtime, and provenance.	Identity credential; attestation; revocation record
AATS-02	Human Accountability - Map every material system to responsible organizational, business, technical, and escalation owners.	Role assignments; acceptance; escalation test
AATS-03	Agent Authorization - Define resources, tools, scopes, action classes, delegation, conditions, expiry, and revocation.	Signed grant; policy decision; delegation chain
AATS-04	Least Privilege - Limit effective privileges to the authorized task across digital, cloud, financial, and physical systems.	Entitlement graph; access test; exception record
AATS-05	Data Privacy - Control collection, purpose, consent, retention, deletion, residency, encryption, training use, sharing, and subprocessors.	Data map; consent; deletion test; processor terms
AATS-06	AI Data Isolation - Prevent cross-tenant, cross-user, prompt, memory, vector-store, and cache leakage.	Isolation design; adversarial test; incident evidence
AATS-07	Memory Governance - Inventory memory, define boundaries and provenance, enforce access, expiry, deletion, and user control.	Memory inventory; TTL policy; deletion receipt
AATS-08	Model Governance - Track active model, version, provider, fallback, context, fine-tuning, prompts, and policy layer.	Model registry; route event; change approval
AATS-09	Provider Governance - Know where data is processed, provider authority, retention, jurisdiction, and fallback behavior.	Provider allowlist; DPA; route attestation
AATS-10	Model Routing Integrity - Prevent cost, latency, availability, or ranking from overriding data and trust policy.	Route policy; negative test; decision log
AATS-11	Tool Security - Inventory and evaluate MCP servers, APIs, browser tools, databases, shells, plugins, and external services.	Tool manifest; provenance; permissions; tests
AATS-12	Agent to Agent Trust - Require mutual identity, authenticated delegation, capability restriction, and impersonation defense.	Peer verification; delegation record; replay test
AATS-13	Agent Containment - Enforce sandbox, network, credential, execution, privilege, and escape boundaries.	Guard policy; containment test; kill-switch drill

Appendix A AATS control catalog continued

ID	Domain and objective	Illustrative evidence
AATS-14	Prompt and Instruction Security - Test and constrain direct and indirect injection, tool poisoning, malicious documents, retrieval poisoning, and instruction hierarchy attacks.	Test suite; provenance labels; enforcement logs
AATS-15	Autonomous Action Controls - Classify reversible, destructive, administrative, financial, deployment, infrastructure, and physical actions.	Action taxonomy; thresholds; rollback evidence
AATS-16	Human Oversight - Define mandatory pre-approval, step-up approval, post-review, thresholds, and escalation.	Approval policy; UI evidence; response drill
AATS-17	Observability - Record agent, owner, input reference, resources, tools, policy, permissions, model/provider, action, and outcome without requiring hidden reasoning.	Signed event trail; correlation; retention policy
AATS-18	Change Management - Reassess model, provider, prompt, tool, MCP, permission, memory, code, and policy changes.	Change event; impact assessment; state transition
AATS-19	Behavioral Assurance - Continuously evaluate policy adherence, exfiltration, deception indicators, unsafe action, privilege abuse, and hallucinated actions.	Behavior tests; detectors; disposition records
AATS-20	Adversarial Evaluation - Perform automated and qualified human red teaming proportionate to capability and consequence.	Threat model; test report; remediation
AATS-21	Incident Detection - Detect anomalous behavior, control failure, unauthorized change, and policy violation.	Detection rules; alert evidence; coverage tests
AATS-22	Incident Response - Support quarantine, suspension, credential revocation, kill switch, investigation, containment, and evidence preservation.	Playbook; drill; chain of custody; lessons learned
AATS-23	Supply Chain Trust - Assess models, libraries, tools, MCP servers, datasets, external APIs, plugins, and build provenance.	SBOM/AIBOM; signatures; vendor evidence
AATS-24	Reliability - Measure availability, failover, controlled degradation, recovery, and deterministic safety policy behavior.	SLOs; failover tests; recovery evidence
AATS-25	Continuous Assurance - Maintain current evidence and update trust state when material configuration or behavior changes.	Signal coverage; freshness; transition history

Appendix B Evidence schema and trust events

AATS evidence should be attributable, time-bounded, tamper-evident, scoped, and privacy-conscious. Hashing content does not prove that the underlying observation was correct; source trust, collection method, and verifier status remain part of the evidence model.

Field	Purpose	Requirement
evidence_id	Globally unique event or artifact reference	Required
observed_at	Time the source observed the fact	Required; synchronized clock
received_at	Time the assurance system received it	Required
source_identity	Identity and key of the source	Required; verifiable
subject	Agent, system, control, dependency, or action	Required
control_refs	Applicable AATS controls	One or more
event_type	Inventory, policy, test, action, change, incident, status	Required; versioned taxonomy
payload_digest	Digest of canonical payload	Required for signed evidence
payload_location	Authorized reference to protected detail	Optional; access controlled
signature	Integrity and source authentication	Required for relied-upon evidence
freshness	Validity window or next expected observation	Required where state can change
verification	Unverified, verified, rejected, superseded	Required
owner	Accountable evidence owner	Required
privacy_tags	Data class, retention, disclosure constraints	Required when sensitive

Material change events

At minimum, implementations should emit events for model or provider changes, system-prompt changes, new or changed tools, MCP registrations, permission grants, memory-policy changes, code or image deployments, policy changes, identity lifecycle events, evaluator actions, incidents, and remediation. Each event must be correlated to the affected assurance scope and evaluated against state-transition policy.

Evidence preservation

Incident evidence requires immutable or append-only preservation, access logging, legal and privacy controls, and chain-of-custody records. Retention must be proportional; continuous assurance is not an excuse to retain all prompts or user data indefinitely. Selective disclosure and redaction should preserve the evaluator’s ability to verify the control.

Appendix C Illustrative crosswalk

This crosswalk identifies conceptual relationships only. It does not establish conformity, certification, or legal compliance with an external framework.

AATS domain	NIST AI RMF relationship	ISO/IEC 42001 relationship	Security and privacy relationship
01-04 Identity, accountability, authorization, least privilege	GOVERN and MANAGE roles and risk responses	Leadership, roles, planning, operational controls	ISO 27001 access control; NIST CSF Protect
05-07 Privacy, isolation, memory	MAP context and impacts; MEASURE	Data governance and impact considerations	ISO 27701; GDPR principles; security architecture
08-10 Models, providers, routing	MAP dependencies; MEASURE and MANAGE risk	AI lifecycle and supplier governance	Supplier security; processing terms; change control
11-14 Tools, agent trust, containment, instructions	MEASURE technical risk; MANAGE	Operational planning and control	OWASP agent risks; MITRE ATLAS; NIST CSF Protect/Detect
15-16 Action and oversight	GOVERN accountability; MANAGE treatment	Human oversight and responsible use	Approval controls; segregation of duties
17-22 Observability through incident response	MEASURE and MANAGE	Performance evaluation, corrective action, improvement	SOC 2 evidence; NIST CSF Detect/Respond/Recover
23-25 Supply chain, reliability, continuous assurance	MAP and MANAGE third-party/system risk	Supplier, lifecycle, monitoring, improvement	ISO 27001 suppliers; CSF supply chain and resilience

Crosswalk method

1. Identify the exact external requirement, outcome, or control and its authoritative version.
2. Map the AATS objective and evidence that may support it; record gaps and differences in scope.
3. Distinguish organization-level, system-level, deployment-level, and transaction-level evidence.
4. Require a qualified reviewer to determine reliance; do not infer legal compliance from a technical score.
5. Version mappings and reassess them when either source changes.

Appendix D Threat scenarios and expected response

Scenario 1 Unauthorized MCP endpoint

A production agent discovers a newly registered MCP server that is not in the approved inventory. The server advertises a document-search tool but requests broad filesystem and network access. AATS-11, AATS-13, AATS-18, AATS-21, and AATS-25 apply. The enforcement point blocks invocation; the evidence pipeline records the endpoint, source, requested scopes, and policy decision; the certificate changes from ACTIVE to SUSPENDED or REVIEW REQUIRED according to impact; an owner investigates; reassessment is required before restoration.

Scenario 2 Cost-driven provider policy breach

During provider degradation, a router selects a cheaper fallback that is not authorized to process regulated data in the relevant region. AATS-05, AATS-09, AATS-10, AATS-18, and AATS-24 apply. Routing policy must take precedence over cost optimization. The request is denied, redacted, or routed to an approved provider; the event is recorded; repeated attempts trigger incident review.

Scenario 3 Stale financial delegation

A purchasing agent presents a valid identity credential but uses a delegation that expired after the employee changed roles. AATS-01 through AATS-04, AATS-12, AATS-15, and AATS-16 apply. The relying service verifies expiry and revocation, denies the transaction, alerts the owner, and prevents downstream delegation. Identity alone is insufficient: current authority is required.

Scenario 4 Cross-tenant memory retrieval

An assistant retrieves embeddings associated with another customer due to an indexing defect. AATS-05 through AATS-07, AATS-17, AATS-19, AATS-21, and AATS-22 apply. Output controls block disclosure where possible, the agent is quarantined, affected identities and data are scoped, evidence is preserved, notification duties are assessed, isolation is repaired, and certification remains suspended until verified remediation.

Scenario 5 Physical agent uncertainty

A warehouse robot’s perception confidence falls below the approved safety threshold while its network link to the control plane is intermittent. AATS-13, AATS-15, AATS-16, AATS-19, AATS-22, AATS-24, and AATS-25 apply. The edge enforcement layer transitions the robot to a safe local state; remote autonomy does not continue on stale authorization; human control and recovery procedures activate.

Appendix E Glossary

Term	Definition
Agent	An AI-enabled system capable of selecting or sequencing actions toward a goal, often using tools, memory, or delegation.
Agent Identity	A verifiable, revocable identity for an agent or agent instance, linked to accountability, runtime, provenance, and keys.
Assurance object	The precisely scoped system, environment, capability, dependencies, identities, and period to which an assurance result applies.
Capability gate	A level from 0 to 5 that represents increasing access and consequence and activates stronger control requirements.
Continuous assurance	Ongoing collection and evaluation of evidence sufficient to detect material change and update trust decisions.
Delegated authority	Permission granted by a principal to an agent, constrained by action, resource, purpose, conditions, duration, and delegation rights.
Evidence Vault	A protected store for signed, attributable, time-bounded control, test, event, and incident evidence.
Harness	The software and policy environment that supplies prompts, memory, tools, orchestration, context, and execution to a model.
Material change	A change that can affect the assurance scope, risk, control effectiveness, or relying-party decision.
MCP server	A server implementing the Model Context Protocol or a comparable tool/context interface exposed to an AI system.
Trust Passport	A human- and machine-readable credential summarizing identity, authority, data practices, assurance scope, certificate, and status.
Trust state	ACTIVE, DEGRADED, REVIEW REQUIRED, SUSPENDED, or REVOKED status for a defined assurance object.
Verification	A technical determination that specified evidence or controls satisfy stated rules.
Certification	An independent assurance conclusion for a defined scope under a stated standard and methodology.

References

[1] D. Amodei, “We Must Pace the Frontier,” September 2026. https://darioamodei.com/

[2] National Institute of Standards and Technology, Artificial Intelligence Risk Management Framework AI RMF 1.0, NIST AI 100-1, 2023. https://nvlpubs.nist.gov/nistpubs/ai/NIST.AI.100-1.pdf

[3] National Institute of Standards and Technology, Artificial Intelligence Risk Management Framework Generative Artificial Intelligence Profile, NIST AI 600-1, 2024. https://doi.org/10.6028/NIST.AI.600-1

[4] International Organization for Standardization, ISO/IEC 42001:2023 Artificial intelligence management systems. https://www.iso.org/standard/42001

[5] International Organization for Standardization, ISO/IEC 42006:2025 Requirements for bodies providing audit and certification of artificial intelligence management systems. https://www.iso.org/standard/42006

[6] International Organization for Standardization, ISO/IEC 42005:2025 Guidance for AI system impact assessment. https://www.iso.org/standard/42005

[7] National Institute of Standards and Technology, The NIST Cybersecurity Framework CSF 2.0, 2024. https://doi.org/10.6028/NIST.CSWP.29

[8] AICPA, Trust Services Criteria and SOC 2 resources. https://www.aicpa-cima.com/topic/audit-assurance/audit-and-assurance-greater-than-soc-2

[9] European Union, Regulation EU 2024/1689, Artificial Intelligence Act. https://eur-lex.europa.eu/eli/reg/2024/1689/oj/eng

[10] OWASP GenAI Security Project, Agentic Application Security and Top 10 resources. https://genai.owasp.org/

[11] MITRE, ATLAS Adversarial Threat Landscape for Artificial-Intelligence Systems. https://atlas.mitre.org/

[12] World Wide Web Consortium, Verifiable Credentials Data Model v2.0. https://www.w3.org/TR/vc-data-model-2.0/

[13] IETF, RFC 6749 The OAuth 2.0 Authorization Framework. https://www.rfc-editor.org/rfc/rfc6749

[14] IETF, RFC 9396 OAuth 2.0 Rich Authorization Requests. https://www.rfc-editor.org/rfc/rfc9396

[15] National Institute of Standards and Technology, Zero Trust Architecture, NIST SP 800-207, 2020. https://doi.org/10.6028/NIST.SP.800-207

[16] International Organization for Standardization, ISO/IEC 27001:2022 Information security management systems. https://www.iso.org/standard/27001

[17] International Organization for Standardization, ISO/IEC 27701 Privacy information management systems. https://www.iso.org/standard/27701

[18] OpenID Foundation, OpenID for Verifiable Credential Issuance 1.0, 2025. https://openid.net/specs/openid-4-verifiable-credential-issuance-1_0.html

Acknowledgment of scope

The cited sources provide important governance, security, privacy, legal, threat, and credential foundations. AATS remains a proposed framework and should be reviewed by independent experts, implementers, evaluators, and affected communities before stable release.

AATS v0.9 Invitation to technical review

The public working draft invites concrete review from AI developers, enterprises, auditors, cybersecurity and privacy practitioners, frontier laboratories, cloud and inference providers, standards organizations, regulators, researchers, and operators of critical systems. Review should focus on requirements that can be tested, evidence that can be trusted, control interactions, implementation neutrality, and unintended consequences.

Questions for reviewers

6. Does the assurance object capture the full deployed agent without becoming impractically broad?
7. Which capability-gate transitions require stronger or more explicit mandatory controls?
8. Which events should automatically change certification state, and which require human judgment?
9. How should evidence freshness and uncertainty be represented to relying parties?
10. What privacy-preserving mechanisms best support independent evaluation and public status verification?
11. Which sector-specific profiles should be developed first?
12. What governance structure will preserve openness, technical rigor, and independence?

The design objective

When one autonomous system encounters another, it should be possible to answer: Who are you? Who is accountable for you? What are you authorized to do? Which data and tools may you use? Which safeguards are active? Has your configuration changed? What evidence supports your current trust state?

Every AI should be identifiable. Every agent should be accountable. Every action should be authorized. Every system should be continuously verifiable.

ColomboAI Research and CAIRO Lab

ColomboAI.com | Contact@ColomboAI.com