The Convergence Architecture – Part 2

From Six Building Blocks to One Integrated Architecture

Casey Plunkett, Co-Founder and CEO, Secure AI LLC · June 2026

Contact: casey.plunkett@trustwaire.ai | @CaseyPlunkettAI on X

"Once you see the strings on the marionettes you can never return to that moment in the performance when you did not see them..."

- Author Unknown

Part One of this series framed five choices available to enterprises deploying agentic AI. If you chose Option C, to evaluate the convergence architecture now, in controlled pilots, continue reading. What follows is the architecture that makes that choice operational.

Executive Summary — Part 2: The Convergence Architecture

The first paper argued that the prerequisite trust model for agentic AI rests on six interdependent building blocks, and that no collection of vendors delivers them as one integrated system. The Convergence Architecture fills this void. Legacy vendors are retrofitting middleware-era architecture to solve a fundamentally new problem, treating agentic AI governance as an identity problem and keeping the trust root external in a Certificate Authority, an identity provider, or a Zero Trust verification layer. The Convergence Architecture inverts that premise.

Convergence, in this sense, is the simultaneous alignment of multiple stakeholders on a single governance profile that drives every downstream enforcement and evidence layer. Not integration at the API level. Not workflow orchestration. Convergence at the trust root itself — the point where business intent, security policy, data classification, and compliance

mapping meet in one attributed, auditable profile before the first credential is ever issued. That is the capability the market currently lacks, and it is the foundation upon which every other capability in this paper is built.

The trust root is not a certificate authority or an identity provider. It is the business decision to deploy an agent, captured in a governance profile before the first credential is issued.

> *Insight Check:* Every other vendor treats agentic governance as an identity problem with an external trust root. It is not. It is an organizational problem with cryptographic enforcement, and the trust root is the business itself.

Six stakeholders converge on the profile: the business owner, the development team, the CISO, the data owner, the compliance officer, and the operations team. Together they define what the agent is approved to do. Two enforcement control points reference that profile — the Credential Router gates the credential at issuance, the AI Kill Switch terminates the agent at runtime if it deviates. Compliance evidence is generated as a byproduct of doing the governance, not collected after the fact.

TrustwAIre is the first instantiation of this architecture. The core: the Credential Router, the Multi-Stakeholder Governance Model, the Kill Switch, and the NIST 800-53 compliance mapping with forty active controls, is built and operational. Five additional compliance frameworks are targeted for design-partner co-development. The rest of this paper shows how it works and how to put it in front of your stakeholders.

Sections I and II present the architectural pattern. Sections III through V describe how TrustwAIre delivers it, how the six building blocks map to the implementation, and how to evaluate it.

I. The Architectural Inversion

In human-centric models, vendors have rarely entered the governance conversation at the modeling of entitlements and roles. As a result, the industry has been playing “whack a mole” for a quarter of a century. The challenge will become exponentially worse with Agentic AI. The first paper showed why the legacy approach won’t solve this expanding problem: every major entrant addresses a slice of the trust model, but none delivers the whole. IBM covers Control at issuance and partial Auditability; Cisco-Astrix covers Visibility; CrowdStrike-SGNL covers behavioral signal; Strata enforces policy at runtime. Each capability is real. None begins at the point where the business owner decides an agent

should exist. The question this paper answers is: what does the architecture look like that starts at business intent — before the agent's first credential is issued?

The first inversion is this: move the trust root from the identity layer, where every incumbent places it, to the point where business intent originates — and wire the governance profile to that origin.

Incumbent vendors will assert they already address this required inversion. Consider what they deliver in the following scenario. An autonomous lending agent at a regional bank makes five thousand credit decisions a day:

- HashiCorp Vault issues the agent a unique cryptographic identity, scopes each credential to a specific request, and expires it when the task completes. Real and necessary engineering. It does not answer who at the bank authorized this agent for production use, who reviewed the disparate-impact analysis, or where the documented business purpose lives.
- IBM Verify and Vault 2.0 unify the agent's identity with the human credit officer who delegated to it and log every transaction immutably. None of those capabilities depend on the business owner, the data owner, or the compliance officer having ever seen the agent's governance profile, let alone having converged on it before the first credential was issued.
- Cisco-Astrix discovers the agent and watches its behavior, which is excellent for finding agents that exist but silent on agents that should not have existed in the first place.
- CrowdStrike-SGNL detects drift in real time but does not know what the agent was supposed to do; without a governance baseline to compare against, observation produces noise.

Four real and useful capabilities, and all four enter after the agent already exists.

Vendors will assign a human owner after the agent is discovered and run certification campaigns after it is deployed. None requires the owner to have authored a governance profile before the agent's first credential was issued. The audit trail begins at registration, not at intent. And none of them answers the question the bank's board, regulator, and audit committee will all ask first: who at this organization approved this agent for production, on what authority, with what review, and where is the documented chain of approvals?

That question, the governance origination question, drives the Source of Reality in this paper: who can author policy, how credentials are issued, what runtime enforcement looks like, and what the compliance evidence actually proves.

> *Insight Check:* Identity tells you who an agent is. Governance tells you whether it should exist at all. The two are not the same architectural concern and conflating them is an architectural error many vendors are making.

II. The Multi-Stakeholder Governance Model

This model reinvents how enterprises have managed digital identity, access, and governance for the past several decades. This requires redefining roles, responsibilities, collaboration in the process, and separation of duties.

Trust Owner and Trust Custodian, Separated by Design

The **Trust Owner** is the individual or individuals accountable for the agent's existence and behavior. Typically the business owner who defined the use case, owns the business or technical outcome, and bears the consequences if the agent misbehaves. The Trust Owner authorizes what the agent is supposed to do, accepts the business risk, and answers to senior leadership, the board, the regulator, and the auditor when the agent makes a consequential decision.

The Trust Custodian isn't a person. It is the set of stakeholders who help the Trust Owner get agents into production faster, while establishing guardrails for those agents.

- Business analysts who model the process and entitlements.
- The development team that builds the agent or agent hierarchy and declares its technical profile.
- The data owner who classifies sensitivity.
- The compliance officer who maps the deployment to regulatory frameworks.
- The CISO, whose thresholds define when active review is required. This reduces friction by replacing mandatory per-agent review with threshold-based oversight: routine changes proceed within published bounds, while changes that cross CISO thresholds trigger active review automatically.

The Trust Custodian collaborates with the Trust Owner on the governance profile so the agent arrives in production without exposing the organization to undue risk. The Operations (spanning infrastructure and the SOC) function is downstream; they complete the feedback loop with the Trust Owner through the custodial team.

The Trust Owner answers audit's question of, "Who decided this agent should exist?" In collaboration with the Trust Owner, the Trust Custodian answers the question of, "Who helped ensure it was built safely?" This is the second inversion in the digital identity and

governance paradigm: Traditional governance places security and compliance downstream, reviewing what the business already built. The Convergence Architecture brings them upstream — the true *Shift Left* that DevSecOps has been pursuing for years — reviewing the governance profile while the agent is being designed for use in the process.

> *Insight Check:* The result is faster deployment with fewer surprises, not slower deployment with more gates. Conversely, if the heritage governance model is retrofitted for Agentic AI, accountability becomes ambiguous. The model collapses into either a business-led free-for-all or an IT-imposed governance bureaucracy that the business will route around, resulting in increased Shadow AI.

Living Governance: Business-Owned, Dynamically Managed and Automated with Division of Labor

This new architecture does not replace the current human-centric governance model. It is designed for Agentic AI and will coexist with that legacy model. What it replaces is the document-passing model: forms emailed to security, spreadsheets circulated for review, checklists attached in chains of approvals. It is not predicated on documents. It is not a record in a ticketing system. It is a real-time profile on which the Trust Owner and the Trust Custodian converge, simultaneously, with full attribution of who changed what, and when.

- The business owner specifies the agent's purpose, resources, authorization model, and risk classification — seeking counsel, as needed.
- The development team declares the agent's technical profile: LLM provider, framework, deployment environment, and agent capabilities.
- The data owner approves the data sensitivity classifications.
- The compliance officer maps the profile to regulatory controls.
- The CISO reviews thresholds, sets autonomy boundaries, and approves or holds activation.

Every stakeholder sees the same Source of Reality. Every change is timestamped and attributed. When the business owner makes a change, it is made on the same profile, with the same attribution, with the same downstream propagation. There is no second document, no second process, no second review cycle. The process is fully automated.

The profile includes independently configurable agent capability flags, such as shell access, self-modification, agent-to-agent delegation, credentials and secrets access, PHI data

access, and public cloud LLM deployment. These flags govern agents regardless of how they are built or deployed: standalone agents, orchestrator-to-sub-agent hierarchies, harness-managed workflows, or ephemeral agents that exist for a single task. Each flag can be gated by threshold tables that determine whether CISO review is required.

The profile also governs agent delegation and tool access. Sub-agents declared within an orchestrator profile inherit the parent's resource scope ceiling and cannot exceed it. Each sub-agent is independently configurable: its own LLM provider, TTL, authorization type, and resource permissions, all bounded by what the parent was approved to access. MCP-enabled tool access is governed at the profile level, with ungoverned tools detected and explicitly blocked at credential issuance.

> *Insight Check:* The CISO's governance authority scales with the granularity of the capability flags, not with the number of agents deployed. That distinction is what separates a governance model built for the deployment curve from one that collapses under it.

Bidirectional Integration with Business Process Tools

Business intent does not originate in the governance profile. It is typically codified in the business process design and mining tools where the Trust Owner and colleagues model the work the agent will perform. The Convergence Architecture is bidirectionally wired into several leading tools in the business process tier: e.g., Signavio, Camunda, Appian, Celonis, Pega, ServiceNow, UiPath, Power Automate, IBM watsonx Orchestrate, and the BPMN modelers.

There are two elements to this bidirectional integration. **First,** propagation from process design to profile: For example, the business analyst adds a new resource to the agent's scope. That change propagates to the governance profile through event-driven integration and triggers the appropriate threshold check. **Second,** propagation from the profile back to the process view. When a stakeholder modifies the profile, the change is reflected in the process view in real time, so the Trust Owner and the Custodian Team always see the current governance state. These are not batch synchronizations; they are schema-aware, event-driven connections that keep both views current as changes occur.

Without this bidirectional integration, the governance profile is disconnected from the business process that defines the agent's purpose, and governance becomes an after-the-fact IT exercise rather than a design-time business decision. With the integration, the

business process and the governance profile are the same Source of Reality, expressed in two views. The architecture does not permit them to drift apart.

> *Insight Check: As Part One argued, moving governance left requires the trust root to begin in the process design tool. The Convergence Architecture is the first to attempt that integration.*

Threshold-Delegated Authority

The CISO, CPO, Compliance Team, et al, should not review every agent profile. They should publish the rules that define when review is required and update them as business conditions change. The platform enforces those rules in real time. This process becomes essential as agents proliferate: multi-stakeholder collective reviews of every agent would be unsustainable at scale.

The Trust Custodian publishes threshold tables across four initial categories, with the expectation that categories will evolve as the agentic AI landscape matures:

- Core security thresholds (action types, risk levels, credential expiration windows)
- Agent business-level entitlement thresholds (per-resource and per-data-classification permissions)
- Development environment thresholds (whether technical review is required before activation)
- Agent capability thresholds (shell access, self-modification, agent-to-agent delegation, public cloud LLM deployment)

Each threshold is timestamped and attributed to the individual who last modified it. The tables are versioned and auditable, with optional hash values assigned.

Below the threshold, the Trust Owner acts autonomously. The agent's profile is created, modified, and activated within the delegation the CISO has explicitly published. Above the threshold, the platform automatically triggers Trust Custodian review, either actively (blocking activation until review completes) or passively (informing without blocking, for changes that reduce rather than expand risk). De-escalation never requires review.

The math forces this model. When agents outnumber employees — and the projections put that ratio well past five to one before the decade is out — any governance model in which a central security team authors policy for each agent will never be able to keep up with the

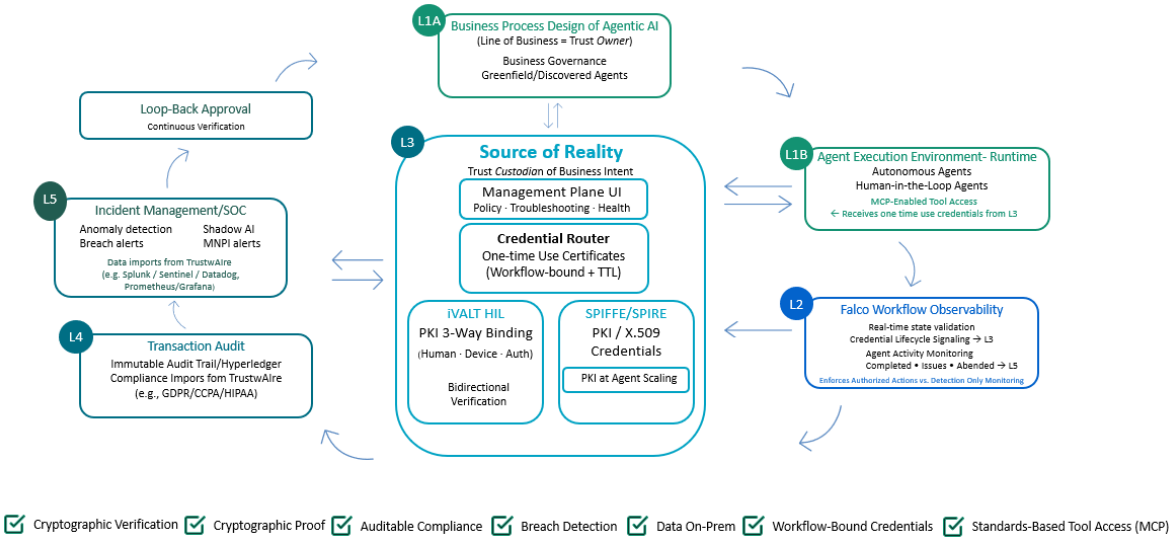
business. Threshold-delegated authority is the only model that scales because it decentralizes authorship to the business owners who define the work while keeping authority centralized in the thresholds the CISO publishes. The workload of the CISO and their peers scales with the number of rules, not the number of agents.

No Credential Without Active Governance

The Convergence Architecture requires that the governance profile lifecycle be a hard gate on credential issuance. No profile state other than Active permits a credential to be issued. There is no override flag, no emergency bypass, no "we will sort the paperwork out later." The deployment shortcut that has historically allowed Shadow AI to flourish is closed at the level of certificate issuance. Agents deployed outside this governed path will persist. The runtime enforcement layer described in Section III exists for that reason.

The architecture must also resolve the inevitable tension between Trust Owner and Trust Custodian. The business owner who needs the agent deployed and the CISO who must accept the risk will not always agree. The question is whether that disagreement is resolved through a shared, attributed, auditable process or through informal negotiation that leaves no record. Section III describes how TrustwAIre implements both the lifecycle gate and the conflict-resolution workflows.

The following diagram illustrates how Convergence Architecture delivers enforceable and provable trust. Business intent originates at L1A, converges through the multi-stakeholder governance model into the Source of Reality at L3, and flows outward as enforceable credentials to the agent execution environment at L1B. Falco observability at L2 and incident management at L5 close the enforcement loop. The transaction audit at L4 generates the tamper-evident compliance record. Every component references the same governance profile.



III. TrustwAIre: Delivering the Convergence Architecture

The preceding sections defined the pattern. This section describes how TrustwAIre, the instantiation of that architecture, delivers discovery, enforcement, compliance, and operational monitoring. Key questions to address: what does the implementation deliver, what trade-offs does it accept, and where are the boundaries of what it solves?

Enforcement begins with the governance profile. The profile defines what the agent is approved to do. Two control points reference it — one at credential issuance, one at runtime. Same profile, both gates. That is what makes them coherent.

Profile Lifecycle and Conflict Resolution

Every governance profile passes through a defined lifecycle: Draft, then Pending CISO Review, then Active, then optionally Suspended, Quarantined, or Superseded. Only one of those states permits credential issuance: Active. Active status requires joint confirmation from both the profile and the synchronization layer, which validates that the profile has been compiled into runtime policy and pushed to the enforcement layer.

This is not workflow logic. It is architectural enforcement. The Credential Router refuses to issue credentials for any profile not in Active status.

Four named workflows handle the tension between Trust Owner and Trust Custodian, each with explicit attribution and full audit trail:

- Profile review on demand for cases where the Trust Owner seeks CISO guidance before proceeding
- Override approval requiring joint business justification and CISO authorization for high-risk autonomous deployments
- Threshold-triggered automatic review when a profile change crosses a CISO-published boundary
- Executive break-glass for impasses where the business case requires escalation beyond the CISO. In this workflow, executive approval auto-issues the credential, so governance never becomes a bottleneck.

Every escalation is fully audited, so governance never becomes a rubber stamp. The architecture forces the conversation to happen on a shared record with full attribution; neither side can claim ignorance of the other's position.

The Credential Router: Enforcement at Issuance

The Credential Router orchestrates the entire issuance pipeline. When an agent requests a credential, the router evaluates the request against the governance profile through a

sequence of gates: profile is Active, threshold tables permit the request, the OPA policy engine evaluates the access decision, the SPIRE workload attestation verifies the agent's identity, and the iVALT human-in-the-loop attestation (where required by the profile) verifies the human approver.

The iVALT integration addresses the heritage-MFA failure mode identified in the first position paper. Heritage MFA verifies a human at session login. Every action after that runs on a stale attestation. The product binds five factors to the specific credential issuance event, in sub-second response time:

- A verified human
- A verified device
- A verified location
- A verified time window
- A verified request source to the specific credential issuance event

The five factors are cryptographically chained to the credential, so the resulting agent action is traceable to the human-of-record through the credential's entire lifecycle. TrustwAIre default configuration polls iVALT once per minute for up to five minutes. If the designated business owner does not respond, no PKI credential is issued. The agent is blocked. No credential can be issued through inaction or negligence; active attestation is required, every time.

If the iVALT service is temporarily unreachable, the credential request times out and is denied. For operational continuity, the CISO's office can approve an override through the governance workflow that bypasses the product and routes the credential directly to SPIRE. The override is logged, attributed, and visible in the audit trail. This is not multi-factor authentication; it is credential-bound human attestation, a distinct architectural primitive.

The credential that emerges is an X.509 certificate, not a token. PKI rather than persistent secrets, for the reasons Part One developed at length. The certificate is workflow-bound, single-use or TTL-expiring, mathematically self-verifying without callback to a central authority. It dies when the operation completes. No persistent access. No blast radius beyond the single operation. No token to steal. The trade-off the architecture accepts deliberately: a credential compromised mid-operation remains valid until its TTL expires. That lifetime can be measured in seconds, but it is not zero.

The three credential layers serve distinct architectural purposes. iVALT binds a verified human to the credential event. SPIFFE attests the workload at the kernel level and carries the governance proof. For resources within a service mesh, the SPIFFE certificate

authenticates directly through mTLS, with no secret on the wire. For legacy resources outside the mesh, Vault or equivalent delivers an ephemeral, scoped resource credential on TrustwAIre's instruction. These are separation of concerns, not redundancy. HashiCorp's own documentation recommends SPIFFE for workload attestation when authenticating to Vault, confirming these are complementary technologies.

Agent operations span a wide spectrum, from read-only queries to destructive actions like DELETE. The threshold tables let the enterprise define which operations require CISO authorization and which can be delegated. TrustwAIre's default configuration locks down destructive operations — the PocketOS scenario Part One described is closed at the credential gate.

The issuance pipeline also resolves a problem the recent wave of security innovation has not addressed: what happens when an agent spawns another agent. Deployment patterns increasingly delegate to sub-agents, and a governance model that stops at the first agent leaves the delegated work ungoverned. In the Convergence Architecture, a sub-agent requesting a credential inherits its scope from the parent and cannot exceed the parent's capability ceiling. The Credential Router enforces this at issuance: the sub-agent's request is evaluated not only against its own profile but against the bounds of the credential that authorized its parent. A parent restricted to read-only access cannot delegate write access it never held. The delegation chain is bounded by construction, and every link in it is attributed in the audit record.

An agent that requests a credential without a governance profile is denied at the gate and logged as an undeclared agent. The same applies to any agent without a parent orchestrator that has declared it as a sub-agent. There is no review queue. No profile means no credential. The denial is immediate.

Users and developers will find workarounds — deploying agents outside the governed path, or changing code on governed agents after approval. The Kill Switch addresses both: shadow-agent detection for the ungoverned, trust-but-verify runtime monitoring for the governed.

> *Insight Check:* The Credential Router governs agents that request credentials through the designed process. The Kill Switch governs agents that do not. Between the two, enforcement is continuous whether the agent cooperates or not.

The AI Kill Switch: Enforcement at Runtime

Issuance-time enforcement is the first control point. The second is runtime. The AI Kill Switch correlates the agent's live behavior, observed by Falco or the enterprise's existing AI monitoring tools, against the governance profile captured in the OPA policy. When behavior deviates from the profile, the Kill Switch acts. Three configurable enforcement modes apply: Kill plus Notify, Kill plus Quarantine with federal-grade evidence preservation, or Notify Only for grace periods during initial pilot deployment. The enforcement path is architected for sub-second latency. Cryptographic proof of every termination action.

A Kill Switch that fires on a false positive in production will kill adoption faster than the problem it solves. That is the central tension. The three modes exist to manage it. Notify Only lets the security team tune thresholds during initial deployment without disrupting operations. Kill plus Notify provides reversible enforcement. Kill plus Quarantine preserves federal-grade evidence for regulated industries where the enforcement action itself must be provable.

Enterprises deploying the Kill Switch into an environment with extensive Shadow AI face a real risk: legitimate, business-critical agents running without governance profiles. Summarily killing them is not an option. The 48-hour warning protocol gives organizations advance notice before destructive enforcement activates, so that legitimate agents can be registered before the Kill Switch treats them as unauthorized. The CISO selects the mode; the platform enforces it uniformly.

Two things distinguish this from currently available behavioral observability products. First, the Kill Switch terminates rather than alerts. Detection without enforcement is window dressing. Second, the Kill Switch references the same governance profile that issued the credential. Current products compare the agent's behavior to a generic threat model, because they have no access to the agent's business-approved baseline. The Kill Switch compares the agent's behavior to the profile its Trust Owner and Trust Custodian jointly approved. That comparison is the difference between a false-positive alert and a deterministic enforcement decision.

Effective Shadow AI detection requires multiple independent paths. No single detection method is sufficient: agents can be deployed through MCP tools, traditional CI/CD pipelines, direct infrastructure access, or ad hoc scripting. TrustwAIre supports detection across these vectors: for example, EDR-led detection through CrowdStrike Falcon, cloud-security-led detection through Wiz AI-SPM, CMDB/CI correlation through ServiceNow, and Falco kernel-level detection of LLM-pattern syscalls without a SPIFFE credential. An agent operating in production without a corresponding Active governance profile is, by architectural definition, unauthorized. The 48-hour warning protocol includes a downloadable signed memo for governance documentation.

The architecture is also monitor-agnostic for production observability. Detection finds unknown agents. Monitoring validates that governed agents continue to operate within their profiles. CrowdStrike, Splunk, Wiz, Cisco-Astrix, Datadog, Sentinel, Elastic, and Arctic Wolf are all supported for either function: any tool that produces a JSON event with agent name, action, and resource can integrate. The enforcement path is the same regardless of the signal source.

To make this concrete: a claims-triage agent at a healthcare payer holds a valid credential scoped to read-only access on patient demographics. At 2:47 AM, the agent attempts a write operation on the full patient record in the EHR. Falco detects the syscall, compares it against the OPA-compiled governance profile, and flags a scope deviation. The Kill Switch fires in under one second: credential revoked, agent quarantined, SOC webhook dispatched. The audit record, which captures the credential issued, the in-scope actions logged, the out-of-scope attempt detected, the Kill Switch activated, and the profile quarantined, is written as five SHA-256 hashed events, chained and anchored, in elapsed time of under two seconds.

When the CISO opens the compliance dashboard the next morning, the enforcement event is already classified against NIST 800-53 AC-3 and AC-6, with the full chain of custody from the business owner who authorized the agent's scope through the deviation that violated it. No log archaeology. No incident reconstruction. The proof was generated as a byproduct of the enforcement itself.

The Kill Switch enforces more than resource scope. It also enforces the thirteen (currently available) capability flags at runtime. If the CISO disallowed shell access on the governance profile and the agent spawns a shell, Falco detects the syscall and the Kill Switch fires. The capability declaration is a runtime enforcement boundary, not merely a governance checkbox.

The Kill Switch ships with full enforcement capability in evaluation mode. For production runtime monitoring, the client connects their existing monitoring infrastructure to a single API endpoint. Any tool that produces JSON events with agent name, action, and resource can integrate. No code changes to TrustwAIre are required. The integration is a configuration task, not a development project. This is the monitor-agnostic architecture in practice: the CISO chooses the signal source, TrustwAIre enforces the governance profile against it.

Enforcement without evidence is assertion. The next layer makes every enforcement action provable.

Tamper-Evident Compliance, Generated as a Byproduct

Compliance in TrustwAIre is not bolted on after the fact. It is built into the architecture. The evidence regulators, examiners, boards, and courts require is generated as a byproduct of doing the governance.

Every governance decision, credential issuance, runtime enforcement action, quarantine, and clearance is cryptographically hashed at the moment it occurs and chained to its predecessor in a tamper-evident hash chain (SHA-256 over canonical event data). For organizations running Hyperledger, the architecture is designed to anchor the hash chain to their existing Fabric deployment. The integration hooks are built; connecting to a client's Fabric network is a configuration task during pilot engagement. Over sixty event types span the full governance lifecycle. Tamper-evidence is not added later; it is the wire format. If any entry is altered, the chain breaks, the hash exposes the alteration, and the record cannot be quietly rewritten.

The compliance report is then a deterministic projection over an already-tamper-evident chain. When the CISO generates a report today and hands it to the auditor, and the auditor returns six months later asking how they know it has not been modified, the platform recomputes the hash from stored canonical data. If it matches, nothing changed. If it does not match, the system identifies exactly what is different. Deterministic, verifiable report integrity is a property evaluators can test in their first session. The auditor sees a per-agent chain of custody, filterable by event type, with one-click integrity verification. If the chain is intact, the platform confirms it. If it has been altered, the platform identifies what changed. The evidence is exportable in CSV, HTML, and Excel formats, and the sub-agent delegation chain is included.

I call this the **compliance-as-byproduct**. Every other compliance product on the market works backwards: collect evidence from external systems, assemble it manually or with workflow automation, attest that the assemblage represents the truth. OneTrust, ServiceNow GRC, Drata, Vanta, and the broader GRC tier are all evidence-collection platforms. In TrustwAIre, when the CISO opens the compliance view and sees "AC-3: Compliant, 847 enforcement events in the last 90 days," that is not a claim. It is a fact derived from the platform's own audit trail. And if the CISO disagrees, they override with a justification, and both the system's assessment and the CISO's interpretation are visible to the auditor.

> *Insight Check:* The Convergence Architecture does not collect evidence. It generates evidence as a byproduct of doing the governance.

The platform classifies events against NIST 800-53 Revision 5 today, with forty controls active. That mapping is built and available for evaluation. Five additional compliance frameworks (HIPAA, PCI DSS Version 4, EU AI Act Article 12, NIST AI RMF, and SOC 2 Type II) are on the roadmap, targeted for design partner co-development. The auto-classification engine is framework-agnostic; additional frameworks activate without architectural changes.

The Convergence Architecture in Practice

TrustwAIre is the first implementation of the Convergence Architecture. It is built and available for structured evaluation: fifteen integrated capability domains, delivered as ten Docker containers, running on-premises or in the customer's cloud environment. It is built on open components that customers can audit and replace: SPIRE for machine PKI, OPA for policy evaluation, SHA-256 tamper-evident hash chains with extensible Hyperledger anchoring, Falco for kernel-level runtime observability, with iVALT as the credential-bound human attestation primitive through strategic partnership. The same policy that gates credential issuance runs unchanged as an Envoy external authorizer and ports to an Istio service mesh without a rewrite, so enforcement lives inside the mesh rather than above it. For resources within the mesh, the SPIFFE credential authenticates directly. For legacy resources outside the mesh, TrustwAIre integrates with the enterprise's existing secrets infrastructure to deliver ephemeral, governed resource credentials. The Credential Router and Multi-Stakeholder Governance Convergence Model are TrustwAIre's proprietary core.

TrustwAIre does not require replacement of the identity, secrets, security, or compliance infrastructure the enterprise has already deployed. It is the connective tissue above them. As the enterprise deploys agents, those tools continue to serve their existing functions. Tools like HashiCorp Vault continue to manage secrets and credentials for human-centric and prompt-based generative AI use cases, and deliver ephemeral resource credentials to legacy systems in agentic AI deployments. CrowdStrike continues to monitor endpoint and cloud behavior. SailPoint and Okta continue to govern human identity and access — and can enforce role-based separation of duties on the TrustwAIre platform itself, controlling which stakeholders access which functions. What none of them delivers is agent governance at the architectural level: should this agent exist, who approved it, is it operating within its governance profile, and can you prove all of that to an auditor? That is what TrustwAIre adds to the stack.

The monitor-agnostic integration extends across the full enterprise stack: identity providers, SIEMs, behavioral sensors, and business process tools. This breadth is possible because TrustwAIre is anchored to none of them. A hyperscaler cannot replicate this, because a hyperscaler will not build first-class integrations with its competitors' identity stacks,

SIEMs, and monitoring tools. Microsoft will not build bidirectional Celonis integration in its governance product. IBM will not build native CrowdStrike Falcon enforcement in Verify. The governance abstraction layer must come from a vendor whose architectural incentive is neutrality across the enterprise stack, not preference for one provider's ecosystem.

> *Insight Check:* Vendor neutrality is not a feature; it is a structural requirement of the problem the Convergence Architecture solves.

Closing the Gaps Part One Identified

Part One assessed IBM's recent announcements and identified four gaps in its capabilities.

- None offer a single living governance profile in which all stakeholders converge in real time. The Convergence Model's living profile directly addresses this.
- None offer bidirectional integration with the business process design and mining tier. Nine bidirectional integrations are built and shipping.
- Existing vendors integrate at the workflow and API level, not at the trust root. Credential issuance, runtime enforcement, and audit carry the imprint of separate product architectures. In TrustwAIre, all three are driven from the same governance profile, the same data model, and the same policy language.
- The agent owner is not the inherent Trust Owner. The Convergence Architecture's Trust Owner / Trust Custodian separation makes business ownership a first-class architectural requirement, not an afterthought assigned during certification campaigns.

IV. How the Six Building Blocks Map to the Convergence Architecture

The progression is complete: from the six building blocks in the first paper, through the architectural pattern in Sections I and II, to the product that delivers it in Section III. What follows closes the loop.

- **Governance:** Multi-Stakeholder Governance Model — living profile, threshold-delegated authority, hard-gated lifecycle, bidirectional BPM integration.
- **Visibility:** Discovery Import — ingests agent inventory from CrowdStrike Falcon, Cisco-Astrix, ServiceNow CMDB, MCP discovery, and LangSmith.
- **Transparency:** Full-attribution governance — every action timestamped and attributed, from business intent through enforcement to audit.

- **Control:** Two enforcement control points — Credential Router at issuance, Kill Switch at runtime.
- **Automation:** Threshold-delegated authority — codifies human judgment into rules that execute at machine speed.
- **Auditability:** Compliance-as-byproduct — SHA-256 hashed events, deterministic report generation, hash-verified integrity.

Six building blocks. One architecture. One trust root.

V. What to Do Next

The pace of agentic AI adoption is outrunning the governance models designed to contain it. EU AI Act Article 12 record-keeping obligations take effect in 2026. DORA Article 21 is active. NIST is developing agent-focused security guidance. The Yale Chief Executive Leadership Institute published a governance framework in Fortune. Enterprises are deploying agents at scale, and regulators are responding in kind. TrustwAIre is purpose-built for this moment: an architecture that enables safe adoption of agentic AI while generating the compliance evidence those regulations will require.

For organizations ready to evaluate:

- **Review the collateral.** The TrustwAIre website includes detailed explanations of the architecture, functionality, process, and organizational impact, with product demos in video format.
- **Download an evaluation copy**. TrustwAIre ships as a Docker-based package that runs on-premises or in your cloud environment. No data leaves the environment. The evaluation is complimentary.
- **Acquiring TrustwAIre.** If you would like to acquire the product, contact us.
- **Customers across all segments.** SMB, enterprise, and public sector organizations are encouraged to download and evaluate. What we ask in return is structured feedback: what matters most, what is missing, and suggested enhancements for production deployment.
- **Consulting Assistance.** Available during pilot evaluation and ongoing assistance.
- **Consulting firms.** We are actively seeking integration partners for early adoption engagements. If your practice advises enterprises on agentic AI governance, security architecture, or regulatory compliance, we want to hear from you.

The organizations that recognize this pivot early will define how their generation of enterprise AI is governed. The organizations that wait will inherit a model designed for someone else's priorities.

Choose explicitly. Build deliberately. Govern at the speed of the agents — not at the speed of the quarterly review cycle.

About the Author

Casey Plunkett is Co-Founder and CEO of Secure AI LLC. At IBM, he served as Chief of Staff to the General Manager of Tivoli, then as Director of Global Sales for IBM Security, leading 1,300 specialists serving 15,000 customers across 160 countries. In that role, he integrated three segments into the company's first unified IAM suite and launched the Federated Identity Management product, growing it from zero to fifty million dollars in revenue in under a year. He also led due diligence and integration for six IBM acquisitions. At Oracle, as Senior Practice Director of North America Security Consulting, he created the IAM and Database Security Practice and led the Oracle Tech Surge that stabilized Healthcare.gov in 2013. He is the author of The Agentic AI Steamroller and has led more than six hundred global engagements in digital identity, privacy, and cybersecurity over two decades.

Casey Plunkett, Co-Founder and CEO, Secure AI LLC · June 2026

Part Two of a Two-Part Position Paper.

Part One: "The Six Building Blocks of Agentic AI Trust" available separately.