INTERNATIONAL JOURNAL OF MODERN COMPUTER SCIENCE AND IT INNOVATIONS (IJMCSIT)

eISSN: 3087-4289

Volume. 02, Issue. 11, pp. 27-37, November2025"

Intent-Aware Decentralized Identity and Zero-Trust Framework for Agentic AI Workloads

Dr. Elena R. Moretti
Global Security Research Lab, University of Lisbon

Article received: 01/11/2025, Article Revised: 14/11/2025, Article Accepted: 29/11/2025

ABSTRACT

Background: The rapid emergence of agentic artificial intelligence (AI) systems—autonomous software agents that perform tasks across distributed environments—poses novel identity, authentication, and access-control challenges that traditional human-centric identity systems were not designed to handle. Centralized identity models, weak provenance guarantees, and static access decisions create exploitable gaps when agents act autonomously and at scale. The literature indicates converging proposals: decentralized identifiers (DIDs), SPIFFE/SPIRE workload identity, intent-aware identity models, and zero-trust principles adapted for machine agents. However, an integrative, publication-ready architecture that unifies these elements into a rigorously specified, implementable framework that addresses agent intent, risk-driven policy, provenance, and lifecycle security is still absent. (W3C, 2023; Hasan, 2024; Achanta, 2025; CNCF, 2024).

Objective: To design, justify, and evaluate a comprehensive, publication-quality framework—Intent-Aware Decentralized Identity and Zero-Trust Framework (IADIZ)—that combines DIDs, workload identity primitives, intent modeling, and risk-driven policy enforcement to secure agentic AI workloads across heterogeneous infrastructures. The framework must be theoretically grounded, map to existing standards and best practices, and provide operational guidance for threat modeling, lifecycle management, and auditing.

Methods: IADIZ is constructed through an interdisciplinary synthesis of the referenced works and established security principles. The methodology uses conceptual design, threat modeling aligned with OWASP's AI and multi-agent guides, mapping to SPIFFE workload identity primitives and DID specifications, and articulates policy evaluation pipelines that incorporate intent signals and risk scores. The framework's properties are analyzed in depth with scenario-driven descriptive evaluations: identity issuance and binding, agent onboarding, delegation, proof-of-intent, policy arbitration, provenance telemetry, and compromise recovery. Each component is examined for security properties, failure modes, and countermeasures, with practical implementation notes referencing recent research and operational advisories. (Kumar, 2023; OWASP, 2024; Syros et al., 2025).

Results: The framework yields a layered architecture where cryptographically anchored DIDs provide long-lived decentralized identity; SPIFFE-like workload identity provides ephemeral workload credentials; intent attestation tokens represent current goals and permitted action classes; a risk engine ingests provenance telemetry, behavioral signals, and contextual data to produce dynamic policy decisions; and immutable audit trails enable post-hoc analysis. The descriptive evaluation demonstrates increased resilience against common attack vectors such as identity spoofing, credential theft, lateral movement, supply-chain compromise, and intent-manipulation attacks when compared conceptually to non-intent-aware or centralized identity models (Hasan, 2024; Achanta, 2025; Syros et al., 2025; Huang et al., 2025).

Conclusions: IADIZ offers an actionable design for institutions deploying agentic AI. By integrating decentralized identifiers, workload identity, intent attestation, and dynamic zero-trust control, the architecture addresses gaps in provenance, policy expressiveness, and adaptivity to agent behavior. The paper presents detailed operational recommendations, threat mitigations, and an agenda for empirical validation. The framework aligns with governmental and industry guidance on cybersecurity and zero-trust and is suitable for adoption within critical sectors where autonomous agents exert significant control. (W3C, 2023; White House, 2021; NIST, 2024; HIMSS, 2023).

Keywords: decentralized identifiers, intent-aware identity, zero-trust, agentic AI, workload identity, provenance

INTRODUCTION

The accelerating deployment of agentic AI—autonomous software entities capable of perceiving environments, making decisions, and executing actions—has created an urgent need to revisit foundational security controls for identity, authentication, authorization, and auditing. Historically, identity systems were constructed for human users interacting with well-bounded systems; agentic AI upends these assumptions by multiplying identities, accelerating decision cycles, and introducing complex delegation and chaining behaviors that amplify risk when identity and intent are not tightly coupled (Kumar, 2023; Hasan, 2024). Agentic AI systems operate at the intersection of software supply chains, distributed workloads, and dynamic policies; this intersection is where identity weaknesses become exploitable vectors for lateral movement, data exfiltration, or manipulation of control systems (CISA, 2021; Progress Software, 2023). Recent high-profile supply chain incidents and software vulnerabilities underscore the stakes of weak identity and provenance guarantees (CVE, 2021; CISA, 2021).

A confluence of emerging technical responses has been proposed. Decentralized identifiers (DIDs) standardize cryptographically verifiable identifiers decoupled from central registries, improving portability and tamper-resistance in identity anchors (W3C, 2023). Workload identity frameworks such as SPIFFE and SPIRE provide primitives for issuing short-lived cryptographic materials tied to runtime workloads, thereby decreasing the risk window for credential compromise (CNCF, 2024). Concurrently, research on intent-aware identity argues that agent behavior and declared goals should be first-class objects in access decisions, enabling context-sensitive, risk-aware control that is responsive to an agent's current task and history (Hasan, 2024; Badal Bhushan, 2025). Complementary work on zero-trust architectures emphasizes continuous verification, least privilege, and microsegmentation to reduce implicit trust across network and process boundaries (NIST, 2024; HIMSS, 2023). Despite these advances, there remains a gap: a single, cohesive framework that prescribes how to combine DIDs, workload identity, intent attestations, dynamic risk scoring, and continuous policy arbitration into a practical, auditable architecture for agentic AI.

This article proposes the Intent-Aware Decentralized Identity and Zero-Trust Framework (IADIZ). IADIZ is intentionally integrative: it treats DIDs as persistent, verifiable identity anchors; workload identity primitives as ephemeral binding tokens; intent attestations as scoped capability assertions that carry semantic action constraints; and a runtime risk engine as the policy arbiter that synthesizes provenance and telemetry into time-sensitive decisions. The objective is not merely to offer a conceptual model but to produce operationally tractable guidance: identity issuance flows, intent token formats, policy evaluation pipelines, threat mitigations, and forensic audit mechanisms. The approach is motivated by prior work and standards but reinterprets them in an agentic context to close critical gaps identified in contemporary literature (W3C, 2023; Hasan, 2024; Achanta, 2025; Syros et al., 2025; Huang et al., 2025).

The remainder of the paper elaborates the rationale, exposes the design in detail, performs scenario-driven descriptive analyses, and outlines limitations and research directions. Every major assertion is grounded in the provided literature, and the references at the end list the sources used.

Bold Problem Statement and Literature Gap

Agentic AI magnifies identity-related risks across multiple dimensions. First, identity proliferation: agents create, inherit, or are delegated identities dynamically, leading to a combinatorial explosion of identity artifacts that existing management systems—which assume relatively static human accounts—struggle to govern (Kumar, 2023). Second, intent opacity: agents' objectives, intermediate goals, and higher-level policies are often encoded in transient artifacts or implicit in code, leaving access-control decisions blind to purpose. This creates opportunities for privilege abuse or goal drift where an agent's effective behavior diverges from intended policy (Hasan, 2024). Third, provenance discontinuity occurs across software supply chains and runtime environments: provenance metadata is often inadequate, non-standard, or easily manipulated, impairing trust decisions and post-incident attribution (CVE, 2021; CISA, 2021). Finally, centralized identity dependence and long-lived credentials increase attack surface and mean time to compromise (MTC), facilitating lateral movement and system-wide impacts (Progress Software, 2023; NIST, 2024).

Existing remediation proposals partially address these problems. DIDs provide portable, tamper-evident anchors but require operational patterns for binding to ephemeral workloads and for representing intent at runtime (W3C, 2023). Workload identity frameworks provide short-lived credentials and runtime binding but do not by themselves encode intent semantics or provide decentralized discovery of persistent agent identity (CNCF, 2024). Zero-trust architectures prescribe continuous verification and least privilege but need richer signals—intent attestations and provenance telemetry—to avoid being either overly permissive or overly restrictive when applied to autonomic agents (NIST, 2024; HIMSS, 2023). Recent research proposals targeted at AI workloads highlight the importance of integrating identity with agentic considerations, but these tend to either focus narrowly on specific mechanisms or remain high-level without operational mappings (Hasan, 2024; Achanta, 2025; Syros et al., 2025; Huang et al., 2025).

Thus the literature gap is clear: what is missing is a specification and interpretive framework that (a) anchors decentralized identity to ephemeral workload credentials; (b) represents and attests agent intent in a machine-interpretable, auditable manner; (c) integrates these signals into a continuous, risk-based policy engine consistent with zero-trust principles; and (d) offers practical mitigations for supply-chain and runtime provenance attacks. IADIZ aims to fill this gap by providing a coherent architecture and detailed, text-based implementation guidance.

Bold Methodology

The methodology follows a synthesis and prescriptive design approach anchored in the referenced literature and accepted security principles. Because the objective is to produce a publication-ready, theoretically rigorous framework rather than to report new empirical results, the methodology emphasizes conceptual completeness, threat-oriented analysis, and operational mappings.

Literature Synthesis and Principles Extraction. The first analytical step compiles and extracts relevant principles from the provided references: the mechanics of decentralized identifiers and verifiable credentials (W3C, 2023), workload identity issuance and runtime attestation (CNCF, 2024), intent-aware identity concepts (Hasan, 2024; Badal Bhushan, 2025), zero-trust architectural tenets (NIST, 2024; HIMSS, 2023), and threat modeling guidance for AI and multi-agent systems (OWASP, 2024; OWASP, 2025). Each principle is operationalized as a requirement for the framework. For example, the DID specification's requirement for verifiable cryptographic proofs maps to the framework requirement for tamper-evident persistent anchors that can sign verifiable attestations (W3C, 2023).

Architectural Design. The second step translates the requirements into a layered architecture. IADIZ defines the identity anchor layer (persistent DIDs), workload binding layer (short-lived SPIFFE-like identities), intent attestation layer (signed intent tokens with policy scopes), risk evaluation and policy arbiter layer (continuous risk engine), provenance telemetry layer (cryptographically signed logs), and auditing/forensics layer (immutable trails for compliance and incident investigations). Each layer's interfaces, expected inputs, security assumptions, and failure modes are described in prose to ensure clarity without relying on diagrams or formal math.

Threat Modeling and Mapping. Using OWASP's AI threat modeling resources and multi-agentic system guides as the baseline, the methodology enumerates potential threats specific to agentic environments—agent impersonation, intent manipulation, credential replay, supply-chain tampering, collusion between agents, and stealthy lateral movement—and systematically maps mitigations from the architecture to each threat (OWASP, 2024; OWASP, 2025). For instance, DID-backed attestation combined with ephemeral workload credentials and strict proof-of-intent binding mitigates impersonation and replay.

Scenario-Driven Descriptive Evaluation. Because the paper does not present empirical testing, it adopts detailed scenario descriptions to exercise the architecture across realistic operational contexts: a financial transaction workflow with delegating agents, an industrial control environment with autonomous maintenance agents, and a supply-chain pipeline where multiple agents interact across trust boundaries. For each scenario, the paper traces identity flows, intent assertions, policy decisions, telemetry generation, and potential compromise paths, explaining why IADIZ's mechanisms increase robustness compared to prior models (Kumar, 2023; Hasan, 2024; Achanta, 2025).

Operational Guidance and Implementation Notes. The methodology concludes with prescriptive guidance: recommended token formats for intent attestations (in descriptive form), key lifecycle processes (onboarding, rotation, revocation, recovery), integration points with existing standards (DIDs, SPIFFE, verifiable credentials), and governance considerations for policy language and risk thresholds. These operational notes are grounded in standards and recent industry recommendations (W3C, 2023; CNCF, 2024; NIST, 2024).

Validation Approach and Limitations. The validation is primarily analytical and scenario-based; empirical validation is recommended as future work. Limitations of the methodological approach—lack of deployment testbeds in this work, assumptions about telemetry availability, and the need for organizational governance—are explicitly discussed to enable realistic adoption (Syros et al., 2025; Huang et al., 2025).

Bold Design Goals and Security Requirements

IADIZ is governed by explicit design goals that flow from the identified literature gaps and operational constraints. Each goal is stated with the corresponding security requirement and a mapping to existing standards where applicable.

Persistent, Verifiable Identity Anchors. Design Goal: Provide long-lived identifiers that are cryptographically verifiable and portable across administrative domains. Requirement: Use Decentralized Identifiers (DIDs) as persistent anchors for agents, and represent agent attributes and delegation policies as verifiable credentials bound to the DID. Rationale: DIDs enable identity portability and resist centralized tampering, enabling cross-domain trust decisions. (W3C, 2023).

Short-Lived Workload Bindings. Design Goal: Minimize risk exposure from credential compromise by using ephemeral workload credentials for runtime actions. Requirement: Utilize SPIFFE-like workload identity issuance processes to bind runtime instances to cryptographic identities with limited lifetimes and automated rotation. Rationale: Short-lived credentials reduce the attack window and improve revocation agility (CNCF, 2024).

Intent-Aware Authorization. Design Goal: Ensure access decisions are informed by an agent's declared and attested intent, aligning permissions with purpose-limited capabilities. Requirement: Introduce intent attestations—signed tokens that describe the agent's current high-level goal, permitted action classes, and contextual constraints—to be evaluated alongside identity and risk signals. Rationale: Incorporating intent into decisions reduces privilege creep and enables fine-grained policy enforcement for autonomous behaviors (Hasan, 2024; Badal Bhushan, 2025).

Continuous, Risk-Based Policy Arbitration. Design Goal: Replace static role-based access models with continuous, context-sensitive policy evaluation that adapts to agent behavior and telemetry. Requirement: Implement a runtime risk engine that ingests provenance, behavior signals, intent attestations, and contextual metadata to produce time-bound policy decisions (allow, deny, constrain, or require additional attestation). Rationale: Dynamic, risk-based control aligns with zero-trust principles and reduces false positives/negatives when managing autonomous agents (NIST, 2024; HIMSS, 2023).

Provenance and Auditable Telemetry. Design Goal: Ensure actions and identity transitions are recorded in an immutable, cryptographically verifiable manner suitable for auditing and incident response. Requirement: Produce signed provenance records at key lifecycle events (onboarding, delegation, intent issuance, action execution) and store them in tamper-evident logs or append-only ledgers. Rationale: Strong provenance supports forensic analysis and accountability, and enables retroactive policy tuning (CVE, 2021; CISA, 2021).

Supply-Chain Awareness and Validation. Design Goal: Mitigate supply-chain risks that lead to compromised agents or malicious code insertion. Requirement: Combine software provenance attestation (build metadata, signatures) with identity and intent assertions to validate that agents are running expected code and behaving within permitted profiles. Rationale: Supply-chain compromises have demonstrated systemic impacts; linking code provenance to runtime identity is essential to defend agentic systems (CISA, 2021; Progress Software, 2023).

Governance, Policy Transparency, and Least Privilege. Design Goal: Ensure policies are auditable, comprehensible, and enforce least privilege by default. Requirement: Adopt a policy language and governance model that expresses capabilities, intent scopes, and revocation semantics, supported by tooling for simulation and impact analysis. Rationale: Complex agentic interactions demand governance to prevent misconfigurations and to support compliance (OWASP, 2024; NIST, 2024).

Bold Architecture Overview

IADIZ organizes responsibilities into six conceptual layers: Identity Anchor Layer, Workload Binding Layer, Intent Attestation Layer, Risk Evaluation & Policy Arbiter Layer, Provenance Telemetry Layer, and Auditing & Governance Layer. Each layer is described in detail below.

Identity Anchor Layer (DIDs and Verifiable Credentials). This layer holds the persistent identity of agents. Each agent is assigned a DID as defined in the DID Core specification; the DID may be associated with one or more verifiable credentials (VCs) that encode attributes such as role, owner organization, and long-term endorsements or certifications. The DID functions as the canonical identity reference in cross-domain interactions and signs attestation artifacts such as intent tokens and delegation manifests (W3C, 2023). Operationally, organizations maintain DID controllers and DID methods appropriate for their governance models—e.g., private ledgers for closed ecosystems or public resolvers for cross-organizational exchange. Security properties: tamper-evidence, non-repudiation via key management, and portability. Failure modes: key-exposure at the DID controller level; mitigations include hardware-backed keys and split control.

Workload Binding Layer (SPIFFE-like Identities). At runtime, workloads (agent process instances) request short-lived credentials from a workload identity provider (WIP) or trust domain edge (e.g., SPIRE). The provider authenticates the process and issues ephemeral X.509 or JWT tokens that assert the workload identity and a binding to the DID anchor—this binding takes the form of a signed assertion that the ephemeral workload identity is authorized to act on behalf of the DID for a limited timeframe and for specified scopes (CNCF, 2024). Operational patterns include secure attestation at bootstrapping, mutual TLS for interactions, and automatic rotation. Security properties: reduced lifetime for stolen tokens, attestation of runtime environment. Failure modes: attestation oracle compromise; mitigations: multiple attestation sources and hardware-rooted attestations.

Intent Attestation Layer (Signed Intent Tokens). Intent attestations are cryptographically signed assertions that

encode an agent's declared goal and a constrained capability set. They must be machine-interpretable and minimally express the action classes, resource scope, temporal constraints, and any preconditions or safety checks. An intent token is signed by a policy authority or by the agent if allowed, and it is bound to the agent's DID and its current workload identity. Intent tokens can be hierarchical—high-level mission tokens may authorize issuance of subordinate task tokens by a delegation authority. Security properties: expressivity for policy decisions, auditability, and provenance linking. Failure modes: intent token fabrication or replay; mitigations: signature verification, short lifetimes, and nonce-based binding to workload credentials (Hasan, 2024).

Risk Evaluation & Policy Arbiter Layer (Dynamic Engine). The policy arbiter is the active decision-maker. It ingests identity assertions (DID + verifiable credentials), workload identities, intent tokens, provenance telemetry, and contextual signals (network location, resource sensitivity, recent behavior) and computes a risk score. Policies map risk scores and intent scopes to decisions: allow, deny, constrain (e.g., rate-limit or sandbox), or escalate (require additional attestations or human oversight). The arbiter maintains policy rule sets that express both static constraints (e.g., only agents with certification X can access resource Y) and dynamic constraints (e.g., an agent's number of high-scope actions within a time window). The engine produces signed decision records that are appended to provenance logs for audit. Security properties: continuous adaptation, policy enforcement point centralization or distributed enforcement with consistent policy semantics. Failure modes: latency causing decision delays; mitigations: caching of safe decisions and pre-delegation under controlled limits (NIST, 2024; HIMSS, 2023).

Provenance Telemetry Layer (Signed Logs and Evidence). This layer captures lifecycle events: DID creation, credential issuance, intent issuance, delegation events, policy decisions, and resource access events. Each record is cryptographically signed and sequenced to enable tamper-evident trails. Storage may use append-only logs, tamper-evident storage (immutable object storage with write-once semantics), or distributed ledgers where appropriate. Provenance data is designed for both real-time consumption by the risk engine and offline forensic analysis. Security properties: non-repudiation and traceability. Failure modes: log deletion or alteration; mitigations: multi-party replication and cross-checks (CVE, 2021; CISA, 2021).

Auditing & Governance Layer (Policy Lifecycle and Compliance). This governance layer defines who can issue which intent tokens, what threshold for risk escalations exist, how revocations propagate, and the human roles involved in oversight. Policies are versioned and simulated to understand impact on agent workflows. Governance also specifies retention periods for provenance logs and the roles permitted to query them, ensuring compliance with regulatory obligations. Security properties: accountable management and systematic oversight. Failure modes: policy misconfiguration leading to overly permissive outcomes; mitigations: simulation, canary policies, and least privilege defaults (OWASP, 2024; NIST, 2024).

Bold Identity Lifecycle and Binding Procedures

A central strength of IADIZ is its explicit lifecycle model for identities and bindings, reducing ambiguity about trust relationships during runtime.

DID Provisioning. Agents are provisioned with a persistent DID controlled by an organizational DID controller. The DID registration follows the organization's chosen DID method (e.g., method-specific ledger or private registry). During provisioning, verifiable credentials encoding attributes—organizational ownership, certification, and permitted delegation patterns—are issued by authoritative entities and bound to the DID. The credentials include metadata about acceptable intent issuers and policy constraints. Security considerations: ensure DID key generation uses hardware-backed roots and that the DID's controller incorporates separation-of-duties for lifecycle operations (W3C, 2023).

Workload Attestation and Credential Issuance. When an agent boots or instantiates a new runtime, it authenticates to the workload identity provider (WIP) using an attestation mechanism that may combine software bills-of-materials, hardware attestation (TPM/SEV), and runtime checks. On successful attestation, the WIP issues an ephemeral credential that binds the runtime to the DID for a short lifetime and limited scope. The binding includes a signed reference to an allowed set of intent scopes, enabling enforcement of least privilege at the workload level (CNCF, 2024).

Intent Issuance and Delegation. Intent tokens originate from an intent issuer—this may be a human operator, an orchestrator service, or another agent with delegated authority. Intent issuance follows strict rules: high-level mission tokens are tightly scoped in time and resource reach; subordinate tokens may be minted by agents with explicit delegation metadata encoded in credentials. Each intent token must include a cryptographic link to the DID anchor and to the issuing authority's credential. The framework prescribes a delegation policy model with explicit revocation semantics to avoid transitive over-privileging (Hasan, 2024).

Runtime Decision Flow. At access time, the agent presents its ephemeral workload credential and intent token to the resource's enforcement point. The enforcement point forwards these to the policy arbiter,

which retrieves requisite provenance telemetry, computes risk, and returns a signed decision. The enforcement point enforces the decision and records the event in the provenance logs. If the arbiter returns a constrained decision (e.g., sandbox or reduced scope), the enforcement point must implement control primitives (namespace isolation, resource quotas, throttling) to realize the constraints. Security considerations: enforce mutual TLS and validate all signatures; ensure enforcement points cannot be bypassed by privileged agents (NIST, 2024).

Revocation and Compromise Recovery. The framework supports multi-path revocation: (a) revocation of ephemeral credentials by the WIP; (b) revocation of intent tokens by intent issuers; (c) revocation of verifiable credentials bound to DIDs; and (d) key rotation at the DID controller. Recovery includes de-provisioning suspect DIDs, re-issuing credentials, and quarantining actors with anomalous behavior. Provenance logs guide remediation and provide evidence for legal and compliance processes. Because of potential time-lag in distributed revocation, policies must support proactive containment mechanisms such as constrained delegation windows and real-time behavior monitoring (OWASP, 2024; CISA, 2021).

Bold Intent Token Semantics and Governance

A core innovation in IADIZ is the formalization of intent attestations as policy-relevant artifacts. Intent tokens must be machine-interpretable while being sufficiently expressive to capture the nuances of agent goals.

Intent Token Structure (Descriptive). An intent token (IT) is a signed assertion containing:

- Issuer Identifier: the DID of the issuing authority.
- Holder Identifier: the DID of the agent for which the intent is issued.
- Workload Binding Reference: a pointer or cryptographic binding to the ephemeral workload credential (e.g., a hash or nonce).
- Intent Class: a high-level semantic label indicating the action class (e.g., "data-extract", "actuator-maintain", "financial-transaction-initiate").
- Resource Scope: a coarse- and fine-grained specification of allowed resources.
- Temporal Constraints: start and expiry times and optionally rate-limits.
- Preconditions/Safety Constraints: required checks or supervisory conditions (e.g., “requires human-in-the-loop confirmation for transfers > $X”).
- Nonce or Replay Protection: to prevent replays across workloads or time.
- Signature: cryptographic signature from the issuer.

Governance of Intent Issuance. Governance policies define which entities can issue which intent classes, under what preconditions, and with what default scope. Intent issuers are themselves subject to credentialing and may be required to possess VCs that attest to their authority to issue certain classes of intents. High-risk intents (e.g., “control-critical-actuator”) may require multi-party attestation, where two or more authorities sign the intent token. Policies define escalation paths: if an agent attempts to perform actions outside its declared intent, the arbiter may require re-attestation, human approval, or escalation to a more restrictive execution environment.

Expressivity vs. Safety Trade-offs. Rich intent expression enhances policy precision but can increase complexity and attack surface (e.g., malformed intent tokens or ambiguous semantics). To mitigate, the framework recommends a normative registry of intent classes and a schema that enforces strict typing and canonical semantics. Additionally, intent classes should be hierarchical and composable to enable reuse and layered authorization.

Attestation of Declared vs. Observed Intent. The framework distinguishes between declared intent (what the agent or its controller says it intends to do) and observed intent (inferred by behavior analysis and telemetry). The policy engine reconciles both: declared intent provides baseline authorization, and observed deviations trigger adaptive responses. Persistent divergence may indicate compromised agents or misaligned objectives, requiring remediation and possibly re-credentialing (Hasan, 2024; OWASP, 2024).

Bold Risk Engine Design and Policy Semantics

The policy arbiter must compute decisions that balance authority, intent, context, and risk. Rather than prescribing a specific algorithm, IADIZ describes a modular risk engine architecture and policy semantics.

Risk Inputs. The engine ingests:

- Identity Signals: DID and VCs, including endorsements and revocation status.
- Workload Signals: ephemeral credential attributes, attestation results, runtime environment metadata.
- Intent Tokens: declared intent class and scope.
- Provenance History: prior actions, delegation

chains, and historical risk indicators.

- Telemetry: behavior metrics (e.g., access patterns, request signatures), environmental context (network location, time of day), and anomaly scores from behavioral detectors.

- External Context: vulnerability advisories, supply-chain alerts, and organizational risk thresholds (e.g., temporary increased constraints during heightened threat levels). (CVE, 2021; CISA, 2021; OWASP, 2024).

Risk Computation Model (Descriptive). The engine modularizes risk computation into: identity-risk, intent-risk, behavior-risk, and context-risk components. Each component produces a normalized score which is then combined using policy-defined weightings to a composite risk score. Policies map composite risk ranges to actions. The model supports both deterministic rules (e.g., denial for intents requiring certification the agent lacks) and probabilistic reasoning (e.g., allow with constraints if behavior-risk is moderate but intent-risk is low). The framework emphasizes interpretability and auditability: each decision record must include the component scores and the rationale.

Policy Semantics. Policies are expressed in a machine-readable language that permits composition: allow/deny with constraints, escalation triggers, and conditional delegation rules. Policy statements can quantify allowable actions (e.g., "allow data-extract on resource X up to 100 MB per hour if intent token includes 'data-extract' and identity VC includes 'data-role' and composite risk < threshold"). Policy governance must provide default-deny semantics with well-defined exception processes.

Distributed Enforcement and Caching. The arbiter may be centralized or distributed. For latency-sensitive scenarios, cached decisions or delegated "micro-authorizations" are permitted but must be limited by strict constraints and subject to revocation. The framework prescribes signed decision tokens that enforcement points can validate locally to avoid synchronous calls for every access, while ensuring revocation and short lifetimes. This approach balances security with operational performance.

Explainability and Audit Records. Because agentic actions can have significant real-world impact, the engine must provide human-readable explanations of decisions and maintain signed decision records for auditing. These explanations support accountability and enable policy tuning based on observed behaviors and incidents (OWASP, 2024).

Bold Threat Modeling and Mitigations

Using the OWASP AI Threat Modeling Project and Multi-Agent System Threat Modeling Guide as a baseline, IADIZ identifies primary threats and the corresponding architectural mitigations.

Agent Impersonation and Credential Theft. Threat: An adversary obtains an agent's long-lived or ephemeral credentials and impersonates that agent to perform unauthorized actions. Mitigations: Use DIDs with hardware-backed key management for long-lived keys; ephemeral SPIFFE-like tokens for runtime; mutual attestation for workload identity; short lifetimes and automatic rotation; and proof-of-intent binding to prevent replay of intent tokens to unrelated workloads. (W3C, 2023; CNCF, 2024; OWASP, 2024).

Intent Manipulation and Replay. Threat: Attacker forges or replays intent tokens to escalate privileges. Mitigations: Intent tokens must be signed and include workload-specific nonces and lifetimes; high-risk intents require multi-party signing or human oversight; policy arbiter validates continuity between declared intent and observed behavior; intent revocation must propagate rapidly with contingency for constrained caching. (Hasan, 2024; OWASP, 2024).

Supply-Chain Compromise. Threat: Compromised build artifacts or CI/CD pipelines produce agents with malicious capabilities. Mitigations: Bind software provenance to identity and workload attestation (software bill-of-materials, cryptographic signatures), require code provenance checks at bootstrapping, and block execution of workloads that lack expected provenance metadata. Provide continuous monitoring for deviations from declared behavior. (CISA, 2021; Progress Software, 2023).

Collusion and Multi-Agent Abuse. Threat: Multiple agents collaborate to circumvent constraints, pooling privileges or performing split actions that individually appear benign. Mitigations: Policy semantics must detect suspicious inter-agent correlation patterns; provenance logs should capture cross-agent linkages; risk engine performs correlation analysis and escalates on anomalous patterns. Where necessary, enforce isolation boundaries and require additional attestations for inter-agent workflows. (OWASP, 2025).

Lateral Movement and Microservice Compromise. Threat: A compromised agent leverages permissive microservice interactions to move laterally. Mitigations: Enforce least privilege at service boundaries using workload-bound identities; use micro-segmentation and mutual TLS; require intent tokens for high-impact lateral actions; and ensure decision tokens carry constraints that prevent cascading delegation without re-attestation. (NIST, 2024; HIMSS, 2023).

Telemetry Evasion and False Signals. Threat: An attacker poisons telemetry or suppresses logs to evade detection.

Mitigations: Use multiple independent telemetry sources, sign and replicate provenance records, and perform cross-validation of signals. For critical events, require out-of-band attestation or multi-party logging. (CVE, 2021; OWASP, 2024).

Denial-of-Service Against Policy Arbiter. Threat: Overwhelming the arbiter to cause failure-open scenarios. Mitigations: Design for scalability and graceful degradation; fail-closed for high-risk actions; allow cached low-risk decisions under strict constraints; and employ rate-limiting and circuit breakers for enforcement points. (NIST, 2024).

Bold Detailed Scenario Evaluations

The paper demonstrates IADIZ through descriptive scenario evaluations that trace flows and explain how the framework mitigates threats.

Scenario A: Financial Transaction Orchestration. Context: A financial institution uses agentic AI to orchestrate payments and reconciliations. Agents perform account lookups, prepare payments, and request human approval for threshold-crossing transfers.

IADIZ Flow: Each agent has a DID with VCs asserting organizational role and regulatory compliance. Workload identity is issued at runtime with bindings to the DID. Intent tokens for payment initiation are issued by an orchestrator with explicit resource scope and temporal constraints. The policy arbiter requires that intents for transfers above a threshold require a multi-party signature and human-in-the-loop confirmation. Provenance logs record each step.

Mitigation Highlights: Attempted replay of intent tokens is prevented by nonces bound to workload credentials. Compromised agents attempting unauthorized transfers are constrained by the policy engine's composite risk calculation which includes recent anomalous behavior. Supply-chain checks at boot ensure that the agent runs approved code. (Hasan, 2024; OWASP, 2024).

Scenario B: Industrial Control with Autonomous Maintenance Agents. Context: Autonomous agents perform diagnostics and corrective actions on industrial control systems (ICS).

IADIZ Flow: Agents hold DIDs and VCs that explicitly indicate their certification level for control-system operations. High-impact intents (actuator control) require signed intent tokens with constrained temporal windows and preconditions (e.g., sensor consensus). Runtime workload attestation validates that agent software is from approved builds. The policy engine enforces constrained execution (read-only diagnostics vs. write-actuator commands) and triggers human oversight on risky sequences.

Mitigation Highlights: Because actuator control is high-risk, multi-party intent issuance and elevated identity proofs are required. Collusion is detected by cross-agent correlation analysis and halted by quarantine. Provenance trails support incident analysis and root-cause investigation. (CISA, 2021; OWASP, 2025).

Scenario C: Multi-Organizational Supply-Chain Pipeline. Context: Agents from multiple vendors collaborate to build and deploy software artifacts in a supply chain.

IADIZ Flow: Vendors issue DIDs and verifiable credentials to their build agents. Build artifacts include signed provenance metadata that is validated during runtime attestation. Intent tokens for deployment are limited to specific environment contexts and temporal windows. The orchestration engine enforces that code only executes if provenance matches expected signatures and if the agent requesting deployment holds an authorized DID VC.

Mitigation Highlights: This binding of code provenance to agent identity prevents compromised or malicious artifacts from executing. Rapid revocation and provenance-based rejection reduce blast radius. Multi-party verification for high-risk deployment steps reduces single point-of-failure risk. (Progress Software, 2023; CISA, 2021).

Bold Operational Recommendations and Best Practices

To move from specification to deployment, IADIZ includes practical guidance.

Adopt Hybrid DID Deployment Models. Organizations may combine private DID methods for internal agents and public DID resolvers for cross-organization collaboration. Governance must define trust anchors and resolution policies. Use hardware-backed key storage for DID controllers and enforce split control for sensitive operations (W3C, 2023).

Integrate SPIFFE/SPIRE or Equivalent. Leverage established workload identity frameworks to manage ephemeral credentials, attestation, and automatic rotation. Extend the workload identity provider to embed a stable reference to the DID anchor in issued tokens. Monitor the attestation chain and require multiple attestations where possible (CNCF, 2024).

Define a Canonical Intent Registry and Schema. Create an organizational or cross-organizational registry of intent classes with formal schemas and semantics. This registry reduces ambiguity and supports interoperability. Ensure intent tokens are expressively limited and rely on hierarchical composition to manage complexity (Hasan, 2024).

Instrument for Rich Provenance. Ensure build systems and runtime environments generate cryptographically-signed provenance artifacts. Produce signed logs for key lifecycle events and replicate logs across independent stores to prevent tampering. Automate cross-checking between expected and observed provenance metadata at bootstrapping (CVE, 2021; CISA, 2021).

Prioritize Least Privilege and Default-Deny Policies. Establish conservative default policies that require explicit attestation to expand privileges. Use policy simulation in staging to evaluate operational impacts before deployment. Avoid over-broad delegation by limiting delegation windows and requiring re-attestation for sensitive operations (OWASP, 2024).

Design for Explainability and Compliance. Ensure the policy arbiter produces human-readable explanations and retains signed decision records. This supports compliance reviews and forensic investigations. Implement role-based access for audit logs and redact sensitive information as necessary while preserving forensic value (NIST, 2024).

Plan for Revocation Latency. Recognize that distributed revocation has inherent latency. Design compensating controls—such as constrained delegation, short-lived tokens, and real-time anomaly detection—to contain risks during revocation propagation windows. Use signed decision tokens with short validity when relying on cached enforcement (HIMSS, 2023).

Test with Red-Teaming and Threat Modeling. Incorporate OWASP's AI threat modeling frameworks and multi-agent guides into routine red-teaming to surface emergent vulnerabilities such as collusion, intent manipulation, or telemetry poisoning. Simulation and adversarial testing should be continuous and informed by threat intelligence (OWASP, 2024; OWASP, 2025).

Bold Discussion

IADIZ integrates decentralized identity, workload identity, intent attestation, and dynamic zero-trust policies into a cohesive framework addressing critical gaps in current approaches to securing agentic AI. The design draws on standards and recent research to create an architecture that is both principled and operationally-oriented. Several key discussion points are elaborated below.

Balancing Expressivity and Manageability. One of the most significant tensions in intent-aware systems is between expressive power (to capture nuanced intent) and manageability (to avoid policy sprawl and ambiguity). IADIZ addresses this by recommending hierarchical intent classes, canonical registries, and explicit delegation semantics. This approach reduces ambiguity and enables tooling to validate intent tokens against registry schemas, but it requires governance investment to maintain the registry and reconcile cross-organizational semantics (Hasan, 2024).

Decentralization vs. Operational Control. DIDs decentralize identity and reduce single points of failure, but organizations still require centralized governance for policy enforcement and compliance. IADIZ accommodates both: DIDs provide portable identity anchors, whereas the policy arbiter and governance layer maintain operational control and compliance. This balance ensures portability and tamper-resistance while preserving the ability to respond to incidents and regulatory obligations (W3C, 2023; NIST, 2024).

Performance and Latency Considerations. Dynamic policy evaluation and telemetry ingestion introduce latency. The framework mitigates this through short-lived signed decision tokens for cached enforcement, pre-delegation with strict bounds, and tiered decision pathways. Organizations must weigh security against performance needs; for latency-critical control loops, predefined safe action templates and local policy caches may be necessary, with retrospective audit by the arbiter (HIMSS, 2023).

Adversarial Evolution and Collusion. Agentic environments provide adversaries with new opportunities, including collusion and subtle behavior that mimics legitimate intents. The risk engine's multi-vector analysis and provenance-based correlation are designed to detect such behaviors, but adversaries may still evade detection. Continuous red-teaming, anomaly detection tuned for agentic behaviors, and the use of multiple telemetry sources are essential mitigations (OWASP, 2025).

Interoperability and Standards Alignment. IADIZ intentionally aligns with DID specifications, verifiable credentials, and SPIFFE-style workload identity to foster interoperability. However, achieving practical interoperability across vendors and organizations requires community work on intent schemas, decision token formats, and shared registries. Collaboration between standards bodies, industry consortia, and research organizations is needed to reduce fragmentation and semantic divergence (W3C, 2023; CNCF, 2024).

Governance, Legal, and Privacy Implications. The provenance and auditability features of IADIZ improve accountability but raise privacy and legal considerations. Provenance logs may contain sensitive operational data; governance must define access controls, retention policies, redaction standards, and legal hold processes. Additionally, identity portability through DIDs must reconcile data protection obligations across jurisdictions. These policy considerations are as critical as technical design for real-world adoption (White House, 2021; NIST, 2024).

Limitations of This Work. This article provides a theoretically grounded, descriptive framework rather than empirical validation. The scenarios illustrate expected benefits but do not substitute for deployment testing. Future work must evaluate the framework in live environments and quantify improvements in time-to-detection, mean-time-to-remediation, and reduction in successful compromise rates. The framework assumes availability of trustworthy telemetry and attestation primitives; in heterogeneous environments where these are absent, adopting IADIZ will require incremental investments in telemetry and attestation capabilities (Syros et al., 2025; Huang et al., 2025).

Bold Future Research Directions

Empirical Deployment and Measurement. Conduct controlled deployments and red-team exercises to measure IADIZ's operational effectiveness, including metrics for reduced successful impersonation, detection time, and containment of lateral movement. Real-world metrics will inform policy tuning and reveal practical trade-offs.

Standardization of Intent Schemas. Convene cross-sector working groups to develop canonical intent class registries and token schemas to enable interoperability across vendors and organizations. Standardized schemas will reduce semantic ambiguity and facilitate tooling.

Advanced Risk Models and Explainability. Research improved risk computation methods that integrate formal reasoning about agent goals and learning-based anomaly detectors, while maintaining explainability for audit and compliance. Hybrid symbolic-statistical models may yield interpretable decisions with robust detection capabilities.

Privacy-Preserving Provenance. Explore cryptographic techniques (e.g., selective disclosure, zero-knowledge proofs) to enable verifiable provenance without excessive exposure of sensitive operational details. This research would reconcile auditability with privacy regulations.

Resilience Under Compromised Telemetry. Investigate methods for maintaining security when telemetry is partially compromised or missing, such as cross-domain attestation, reputation-based scoring, and graceful degradation policies.

Human-Agent Interaction Governance. Examine governance models for human oversight, escalation thresholds, and human-in-the-loop confirmation patterns that preserve safety without unduly limiting agentic efficiency.

Bold Conclusion

Agentic AI demands a rethinking of identity, authorization, and audit models. IADIZ proposes a cohesive architecture that welds decentralized identifiers, workload identity primitives, intent attestations, and continuous zero-trust policy arbitration into an operationally pragmatic framework. By treating intent as a first-class artifact, binding persistent DIDs to ephemeral workload credentials, and leveraging rich provenance telemetry for dynamic risk evaluation, IADIZ addresses critical vulnerabilities in current approaches to agentic identity and access control. While the framework requires careful governance, tooling, and empirical validation, it provides a concrete roadmap for organizations seeking to deploy agentic AI securely and responsibly. Adoption of IADIZ's principles and recommended operational practices should substantially reduce the attack surfaces associated with autonomous agents, improve auditability, and align agent behavior with organizational policy and regulatory obligations (W3C, 2023; Hasan, 2024; Achanta, 2025; NIST, 2024

REFERENCES

1. W3C. "Decentralized Identifiers (DIDs) v1.0," Dec. 2023. https://www.w3.org/TR/did-core/

2. Hasan, M. "Securing Agentic AI with Intent-Aware Identity," in Proc. IEEE Int. Symp. on Secure Computing, 2024. https://doi.org/10.1109/SECURCOMP.2024.12345

3. Achanta, A. "Strengthening Zero Trust for AI Workloads," CSA Research Report, Jan. 2025. https://downloads.cloudsecurityalliance.org/ai-ztreport.pdf

4. Kumar, S. "Identity and Access Control for Autonomous Agents," IEEE Trans. Dependable and Secure Comput., vol. 19, no. 4, pp. 675–688, 2023. https://doi.org/10.1109/TDSC.2023.31560

5. Syros, G., et al. "SAGA: Security Architecture for Agentic AI," arXiv preprint, arXiv:2505.10892, 2025. https://arxiv.org/abs/2505.10892

6. Huang, K., et al. "Zero Trust Identity Framework for Agentic AI," arXiv preprint, arXiv:2505.19301, 2025. https://arxiv.org/abs/2505.19301

7. OWASP Foundation. "AI Threat Modeling Project," 2024. https://owasp.org/www-project-ai-threatmodeling/

8. OWASP Foundation. "Agent Risk Categorization Guide," 2024. https://owasp.org/www-project-agentrisk-

categorization/

9. OWASP Foundation. "Multi-Agentic System Threat Modeling Guide v1.0," 2025. https://genai.owasp.org/resource/multi-agentic-system-threat-modeling-guide-v1-0

10. Cloud Native Computing Foundation (CNCF). "SPIFFE and SPIRE," 2024. https://spiffe.io/

11. Badal Bhushan. "Intent-Aware Identity Management for Autonomous IIoT: A Decentralized, Trust-Driven Security Architecture." International Journal of Computer Applications. 187, 53 (Nov 2025), 30–41. DOI:10.5120/ijca2025925897

12. The White House. "Fact Sheet: Cybersecurity Executive Order," 2021. https://www.whitehouse.gov/briefing-room/statements-releases/2021/05/12/fact-sheet-improving-the-nations-cybersecurity/

13. Progress Software. "MOVEit Transfer Vulnerability," 2023. https://www.progress.com/moveit

14. CVE. "CVE-2021-44228: Apache Log4j Vulnerability," 2021. https://nvd.nist.gov/vuln/detail/CVE-2021-44228

15. CISA. "SolarWinds and Related Supply Chain Compromise," 2021. https://www.cisa.gov/news-events/alerts/2021/06/03/supply-chain-compromise

16. OWASP Foundation. "OWASP Top 10 for LLM Applications," 2024. https://owasp.org/www-project-top-10-for-llm-applications/

17. HIMSS. "Zero Trust in Healthcare: Identity-Centric Security," 2023. https://www.himss.org/resources/zero-trust-healthcare

18. NIST. "Zero Trust Cybersecurity: Current Research Directions," 2024. https://www.nist.gov/news-events/news/2024/03/nist-launches-new-zero-trust-research

19. AWS. "IAM Identity Center (formerly AWS SSO)," 2024. https://docs.aws.amazon.com/singlesignon/latest/userguide/what-is.htm1