International Journal of Cyber Threat Intelligence and Secure Networking (IJCTISN)
eISSN: 3087-4297
Volume03 Issue10 | Page No: - 22-41

Received: 23 September 2026, Revised: 14 Oct 2026, Accepted: 22 October 2026, Published: 30 October 2026

DOI: https://doi.org/10.55640/ijctisn-v03i10-03

Enterprise Agent Identity Governance Framework (EAIGF): A Policy-Mediated Identity and Access Governance Model for Autonomous Digital Workforce Systems

Sandeep Kumar Anuguthala
Independent Researcher, USA

ABSTRACT: Agentic Artificial Intelligence (AI) systems are evolving from passive decision-support tools into autonomous digital workers capable of planning, reasoning, invoking tools, accessing enterprise resources, and executing business actions with varying degrees of autonomy. While enterprises maintain mature governance models for human identities, service accounts, and application identities, existing Identity and Access Management (IAM) frameworks do not adequately address the governance challenges introduced by autonomous agents. Unlike conventional applications, agents dynamically determine execution paths, invoke reusable skills, delegate tasks to peer agents, and perform privileged actions across enterprise systems. These characteristics define a new class of enterprise identity that requires dedicated governance, authorization, and accountability controls. This paper proposes the Enterprise Agent Identity Governance Framework (EAIGF), which treats autonomous agents as first-class enterprise identities subject to lifecycle management, quantitative trust classification, privileged access governance, and end-to-end accountability. The framework's principal constructs form an identity layer comprising the Agent Identity Registry (AIR), the Agent Provisioning Certificate, Agent Trust Tiering driven by an explicit Agent Identity Risk Score (AIRS) formula, Agent Privileged Access Management (Agent-PAM), and a Delegated Trust Chain with Agent Identity Lineage for multi-agent accountability. These constructs are enforced through a policy-mediated runtime boundary that performs capability projection and risk-adaptive authorization. A reference implementation realizing the registry, the AIRS formula, capability projection, and risk-adaptive authorization correctly enforced all ten governance scenarios in the evaluation with zero false-permits on disallowed capabilities and a mean authorization overhead of approximately 0.015 ms per request. A larger randomized evaluation over 600 authorization requests across 80 agents and 25 reusable skills reproduced this result and, against a naive whole-skill authorization baseline, reduced the exposed capability surface by 66.7 percent while blocking every attempted privilege escalation through delegation. A synthetic financial-services walkthrough demonstrates how Identity Governance and Administration (IGA), Privileged Access Management (PAM), and Zero Trust principles extend to agentic environments. EAIGF provides a practical blueprint for governing autonomous agents at scale while preserving least privilege, accountability, and operational resilience.

KEYWORDS: Agentic AI; Identity Governance; Agent Harness; Privileged Access Management; Zero Trust; Object-Capability Security; Autonomous Agents; Agent Identity Management

1. Introduction

Artificial Intelligence is evolving beyond predictive and generative capabilities toward autonomous systems that independently plan and execute complex tasks. Advances in Large Language Models (LLMs), tool-augmented reasoning, the Model Context Protocol (MCP), and multi-agent architectures have accelerated the emergence of agentic AI systems that interact with enterprise applications, invoke tools, access data repositories, and make operational decisions [1], [2], [4]. Unlike traditional AI systems that generate outputs for human review, agentic systems autonomously execute workflows in pursuit of business objectives. Examples include fraud investigation agents, customer service agents, IT operations agents, and compliance monitoring agents [3]. As organizations adopt these systems, agents transition from passive software

components into active participants within enterprise operating environments.

This shift creates a fundamental governance challenge. Existing enterprise identity models were designed around three identity categories: human identities, application identities, and service accounts. These classes assume deterministic behavior, predefined authorization boundaries, and relatively static permission assignments. Agentic systems violate these assumptions. Agents can dynamically determine execution paths, invoke reusable skills, access previously unused tools, interact with peer agents, and execute actions with varying autonomy [4], [5]. Recent security research underscores the urgency: tool-augmented agents are susceptible to indirect prompt injection, in which adversarial content embedded in retrieved data redirects agent behavior [6], and autonomous agents acting with real-world authority introduce categories of harm not addressed by model-centric governance [7], [26].

Consequently, organizations face a governance gap. Current IAM frameworks, Privileged Access Management (PAM) solutions, and Identity Governance and Administration (IGA) programs provide limited guidance on how autonomous agents should be onboarded, authorized, monitored, and decommissioned. AI governance frameworks such as the NIST AI Risk Management Framework (AI RMF) [8] and ISO/IEC 42001 [9] focus on model governance, risk management, and responsible AI rather than identity governance and runtime authorization.

This paper argues that autonomous agents constitute a distinct enterprise identity category, Like employees, contractors, applications, and service accounts, these agent identities require dedicated governance spanning lifecycle management, authorization, privileged access governance, runtime monitoring, and accountability. To address this challenge, the paper proposes the Enterprise Agent Identity Governance Framework (EAIGF), which extends and adapts established IAM, PAM, and Zero Trust principles by introducing identity-layer governance constructs specific to autonomous agent ecosystems.

1.1 Scope and Relationship to Prior Work

This paper presents the identity-governance layer of a broader research program on enterprise agentic AI governance. The paper is self-contained: all constructs required to understand and evaluate the framework are defined herein. Enforcement relies on a policy-mediated runtime boundary that performs capability projection and risk-adaptive authorization; these runtime mechanisms, introduced in the author's prior work on agentic skill governance [10] and runtime monitoring [11], are summarized in Section 7 for completeness. The contributions of this paper are the identity-centric constructs summarized below; the runtime mechanisms serve as the enforcement substrate through which those constructs operate, not as contributions of this work.

The primary contributions of this paper are:

- Characterization of autonomous agents as a distinct enterprise identity category requiring dedicated governance, and a five-layer framework for governing them.
- The **Agent Identity Registry (AIR)** and **Agent Provisioning Certificate**, establishing an authoritative system of record and a formal onboarding artifact for agent identities.
- An explicit **Agent Identity Risk Score (AIRS)** formula driving **Agent Trust Tiering**, giving trust classification a reproducible quantitative basis.
- **Agent Privileged Access Management (Agent-PAM)**, extending Just-in-Time access, change governance, incident governance, and session monitoring to autonomous privileged actions.
- The **Delegated Trust Chain (DTC)** and **Agent Identity Lineage (AIL)**, providing end-to-end accountability and controlled capability inheritance across multi-agent execution.
- A self-contained reference implementation demonstrating identity-layer enforcement, evaluated across ten governance scenarios.

2. Research Gap and Related Work

2.1 Identity Governance in Traditional Enterprise Systems

Identity governance is a foundational

component of enterprise security. NIST SP 800-63 [12], NIST SP 800-53 [13], and the Zero Trust Architecture guidance in NIST SP 800-207 [14] provide guidance for managing digital identities, authentication, authorization, and access control. Traditional frameworks assume identities belong to one of three categories: human users, applications, or service accounts. Governance typically includes onboarding, role-based access control (RBAC), privileged access management, periodic entitlement reviews, segregation-of-duty (SoD) controls, and decommissioning [15]. Formal models of RBAC and SoD constraints provide the theoretical basis for these controls [16]. While effective for deterministic systems, these approaches do not address the dynamic, autonomous nature of agentic AI.

2.2 Capability-Based Security and Policy Enforcement

EAIGF's runtime authorization substrate draws on the object-capability tradition, in which authority is conferred by unforgeable references that designate both a resource and the permitted operations, enabling least authority by construction [17], [18]. Operating-system work such as Capsicum demonstrated practical capability confinement of application components [19]. The distinction between a component's declared capabilities and the narrower authority it should wield at runtime motivates the capability-projection mechanisms applied in this paper. Enforcement is structured using the policy decision point and policy enforcement point (PDP/PEP) separation standardized in access-control policy languages such as XACML and realized in modern engines including Open Policy Agent and Cedar [20]. EAIGF applies these principles to autonomous agents rather than to human sessions or microservices.

2.3 AI and Agentic AI Governance

AI governance frameworks reflect growing concern over AI risk. The NIST AI RMF emphasizes risk identification, governance, measurement, and monitoring [8], and ISO/IEC 42001 establishes requirements for AI management systems [9]. Threat-modeling efforts specific to AI include MITRE ATLAS, which catalogs adversarial threats to AI systems [21], and agent-oriented extensions of threat modeling [22]. Research on indirect prompt injection [6] and on harms from increasingly autonomous agents [7] further motivates runtime controls. Despite this progress, existing frameworks focus on governing models, datasets, and use cases rather than governing autonomous agents as enterprise identities.

2.4 Gaps and Positioning

Three gaps remain. First, AI governance frameworks govern models and use cases rather than autonomous identities; organizations maintain AI use-case inventories but lack a system of record for the agents themselves. Second, IAM and PAM frameworks assume static permission assignments, whereas agentic systems require context- and risk-dependent authorization over reusable skills, dynamic tool invocation, and delegated execution. Third, existing frameworks offer limited runtime enforcement, leaving a gap between documented policy and constrained execution.

Prior work in this research program addressed adjacent layers: a control-driven lifecycle governance framework for agentic use cases [23], skill-level governance addressing reusable-skill over-entitlement [10], and runtime monitoring and incident response [11]. None treated agent identity as a standalone governance discipline. The present paper fills that gap by treating agents as first-class identities with dedicated registration, quantitative risk scoring, privileged access governance, lifecycle management, and multi-agent accountability, constructs introduced in this paper and enforced through a runtime boundary summarized in Section 7.

3. Framework Overview

EAIGF governs autonomous agents throughout their operational lifecycle while ensuring accountability, least privilege, and policy compliance. The framework is organized into five governance layers: an Identity Governance Layer, a Trust Governance Layer, an Access Governance Layer, a Runtime Governance Layer, and an Assurance Governance Layer. The identity, trust, access, and assurance layers are the contribution of this paper; the runtime layer is a policy-mediated enforcement boundary, summarized in Section 7, whose underlying mechanisms were presented in the author's prior work [10], [11]. Figure 1 situates the layers and indicates which are introduced here.

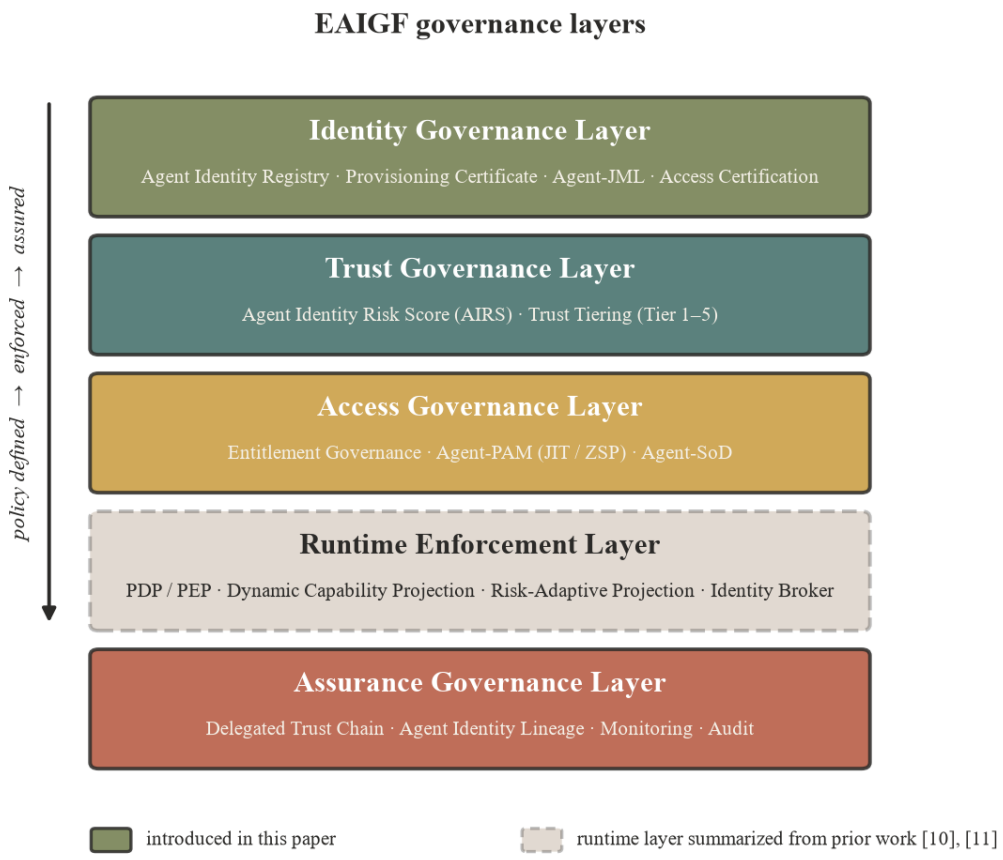


Figure 1. EAIGF governance layers. The identity, trust, access, and assurance layers are introduced in this paper; the runtime layer is a policy-mediated enforcement boundary whose mechanisms were presented in the author's prior work [10], [11] and are summarized in Section 7.

4. Identity Governance Layer

A foundational principle of enterprise governance is that assets cannot be governed if they are unknown. Organizations maintain authoritative inventories for employees, applications, service accounts, and AI use cases because governance depends on visibility, ownership, and accountability. As organizations deploy increasing numbers of agents, the absence of a centralized system of record creates governance gaps: teams may be unable to determine which agents are active, what permissions they hold, which skills they invoke, which MCP servers they reach, or which owners are accountable. These gaps lead to unmanaged agents, excessive permissions, and compliance exposure.

4.1 Agent Identity Registry (AIR)

The Agent Identity Registry is the enterprise system of record for autonomous agent identities. Conceptually it performs a role analogous to a directory service for workforce identities, a configuration management database for technology assets, and a model registry for AI models, while extending these to autonomous agents. Every agent, whether internally developed, vendor supplied, or open-source based, and whether single-agent or multi-agent, must be registered in AIR prior to deployment. On registration, AIR assigns a globally unique Agent Identifier that serves as the authoritative identity reference across governance, monitoring, audit, and operational processes. Table 1 lists the core AIR record attributes.

Attribute	Description
Agent ID	Unique enterprise identifier
Business Owner	Accountable owner

Attribute	Description
Business Use Case	Approved use case scope
Trust Tier	Assigned trust classification (Tier 1-5)
AIRS Score	Current Agent Identity Risk Score
Approved Skills / MCP / Tools	Use-case-approved capability set
Lifecycle State	Active, Suspended, or Decommissioned
Certification Status	Most recent access-certification result
Approval Records	Governance approvals and Agent Provisioning Certificate

Table 1. Core attributes of an Agent Identity Registry record.

4.2 Agent Provisioning Certificate

Workforce identities undergo formal onboarding before receiving access, and applications undergo architecture and risk review before production. EAIGF applies an equivalent gate to agents through the Agent Provisioning Certificate, a signed artifact generated by AIR that documents the agent's approved governance state at inception: identifier, business purpose, owner, creation date, initial Trust Tier and AIRS score, approved skills and MCP integrations, authorized capabilities, governance approvals, and deployment environment. The certificate establishes the baseline trust profile referenced throughout the agent's lifecycle. No production deployment may occur without a valid Agent Provisioning Certificate and AIR registration.

4.3 Agent Joiner-Mover-Leaver Lifecycle

Mature IGA programs manage workforce identities through Joiner-Mover-Leaver (JML) processes [15]. EAIGF extends JML to autonomous agents. The Joiner phase governs creation and onboarding: AIR registration, AIRS assessment, Trust Tier assignment, Agent Provisioning Certificate issuance, activation of the agent's runtime policy bindings so its actions are subject to enforcement, and initial entitlement assignment. The Mover phase governs material change. Adding skills, introducing MCP integrations, expanding tool access, or increasing autonomy alters the agent's risk posture; each material change triggers a Mover event requiring risk reassessment, AIR update, governance approval, AIRS recalculation, and Trust Tier review, preventing governance drift as capabilities evolve. The Leaver phase governs retirement: revocation of permissions, removal of skill and MCP associations, revocation of the agent's credentials and runtime access so the retired agent can no longer transact with enterprise systems, archival of audit records, and an AIR status update.

4.4 Agent Access Certification

Periodic entitlement review is a foundational IAM control [12], [13]. Agents, like users and applications, accumulate skills, MCP integrations, privileges, and delegated trust relationships over time. Without review, these may exceed legitimate business requirements. EAIGF introduces Agent Access Certification, in which governance teams periodically validate an agent's skills, MCP associations, tool access, privileged capabilities, Trust Tier, and AIRS score against continued business justification and least-privilege alignment. Excessive entitlements are removed through governance workflows, and a failed or expired certification renders the agent ineligible for

runtime authorization until remediated. Access certification therefore functions as a continuous least-privilege control for autonomous agents.

5. Trust Governance Layer

Identity registration establishes accountability; trust governance establishes how much authority an agent may be granted. As financial institutions classify applications and service accounts by criticality and exposure, autonomous agents require a formal trust-classification model reflecting their operational capabilities and potential impact. Unlike application classification, which centers on confidentiality, integrity, and availability, agent trust classification must additionally account for autonomous decision-making authority and delegated execution: as autonomy increases, so does the potential for unintended action, privilege abuse, and cascading failure.

5.1 Agent Identity Risk Score (AIRS)

To support objective and reproducible classification, EAIGF defines the Agent Identity Risk Score (AIRS) as an explicit weighted function of eight normalized risk dimensions. Each dimension is expressed on a normalized scale in the interval [0,1], combined using the weights in Table 2, and scaled to the range 0-100:

$$AIRS = 100 \times \Sigma_i w_i \cdot x_i \qquad where \;\; \Sigma_i w_i = 1 \;\; and \;\; x_i \in [0,1]$$

In this expression, x_i is the normalized score of risk dimension i, ranging from 0 (no exposure) to 1 (maximum exposure); w_i is the weight assigned to that dimension, with all weights summing to 1.0; the summation combines the eight weighted dimensions into a single value in [0,1]; and the factor of 100 rescales this value onto a 0 to 100 range so that it maps directly to the five trust tiers of Table 3. For example, an agent scoring 1.0 on every dimension would receive the maximum AIRS of 100, while an agent scoring 0 on every dimension would receive 0. The eight dimensions and their weights are given in Table 2.

Risk Dimension (xi)	Weight (wi)	Interpretation
Autonomy level	0.20	Degree of unsupervised decision authority
Data sensitivity	0.15	PII / PCI / SPI exposure
Privileged-action capability	0.20	Ability to perform privileged operations
Financial / customer impact	0.15	Potential for financial or customer harm
Skill count	0.08	Number of reusable skills
MCP integration count	0.07	Number of connected MCP servers
Regulatory exposure	0.08	Compliance implications
Production access	0.07	Reach into production systems

Table 2. AIRS dimensions and illustrative default weights. The weights shown are illustrative defaults rather than empirically validated values: the highest weights (0.20) are assigned to autonomy and privileged-action capability as the dimensions most directly predictive of an agent's potential impact, intermediate weights (0.15) to data sensitivity and financial impact, and lower weights (0.07 to 0.08) to count- and exposure-based dimensions. The weights are intended to be calibrated per institution (see note below).

Calibration note. The weights in Table 2 are illustrative defaults chosen for expository clarity and are not derived from empirical data. In a production setting, institutions should calibrate them to their own risk appetite and incident history, for example through structured expert elicitation (such as a Delphi process) or by regressing the dimensions against historical agent-related incidents. The framework itself is independent of the specific weight values; only the eight-dimension schema and the functional form are prescribed.

Making AIRS an explicit formula, rather than a qualitative judgment, yields three benefits: classification becomes reproducible and auditable; weights can be calibrated per industry without changing the framework; and the score can be recalculated from runtime signals observed during execution, enabling the risk-adaptive authorization described in Section 7. Worked values for the walkthrough agents (Section 8) are produced directly by the reference implementation.

5.2 Agent Trust Tiering

The composite AIRS score maps to one of five trust tiers, which determine the stringency of governance controls including approval workflows, monitoring obligations, certification frequency, human-in-the-loop requirements, and privileged access constraints.

Trust Tier	AIRS	Description
Tier 1	0-20	Read-only informational agents
Tier 2	21-40	Workflow-assistance agents
Tier 3	41-60	Operational agents performing business actions
Tier 4	61-80	Agents capable of financial or customer-impacting actions
Tier 5	81-100	Critical-infrastructure and highly autonomous agents

Table 3. Mapping of AIRS scores to trust tiers.

A standardized trust model lets organizations apply consistent governance expectations across heterogeneous agent ecosystems while ensuring controls are proportional to risk.

6. Access Governance Layer

Identity does not confer authority. Enterprise governance requires mechanisms ensuring agents receive only the permissions necessary for approved objectives. Traditional IGA relies on RBAC, ABAC, and periodic entitlement review to enforce least privilege [12], [13], but assumes relatively static assignments and predictable execution paths. Agentic systems invalidate these assumptions: agents may dynamically invoke tools, execute reusable skills, and perform actions not explicitly anticipated at design time. EAIGF addresses this through three constructs, two of which (entitlement governance and Agent-SoD) are contributed here, with Agent-PAM detailed in Section 6.2.

6.1 Agent Entitlement Governance

Every agent maintains an approved entitlement profile in AIR defining the maximum set of capabilities, approved skills, MCP servers, tools, data sources, and business actions, it may access during execution. This use-case-approved set is the reference against which the runtime enforcement boundary projects capabilities at execution time (Section 7.1). Because the projection intersects the skill's declared capabilities with the agent's approved set, a reusable skill cannot expand an agent's authority beyond what governance approved, even as the skill evolves.

6.2 Agent Privileged Access Management (Agent-

PAM)

Privileged access is among the highest-risk areas in enterprise security; PAM frameworks emphasize strict control over privileged operations because misuse can cause significant financial, operational, regulatory, and reputational harm [13], [14]. Traditional PAM assumes privileged actions are initiated by human operators. Autonomous agents break this assumption: an agent may independently determine that a privileged action, disabling an account, modifying a security configuration, freezing a financial account, or executing a production change, should occur. Granting agents persistent elevated privileges would violate least privilege, separation of duties, and Zero Standing Privilege. EAIGF therefore introduces Agent-PAM, under which privileged authority is contextual, temporary, auditable, revocable, and policy-mediated.

Just-in-Time Agent Privileges (JIT-AP)

Agents hold only baseline operational permissions during normal execution. When a privileged capability is required, the enforcement boundary evaluates governance preconditions and, if satisfied, brokers a temporary, narrowly scoped credential for a single action through integration with the enterprise privileged-access and secrets-management infrastructure; the credential expires automatically and the privilege is revoked after execution. In this model the enforcement boundary does not itself hold standing privilege, it requests a short-lived credential from the existing secrets-management system on the agent's behalf only when preconditions are met, so that the established PAM infrastructure remains the issuing authority. Credential lifetime may vary with Trust Tier, AIRS score, action criticality, and business context. This realizes Zero Standing Privilege for autonomous agents, eliminating the persistent elevated permissions that historically drive privilege-abuse incidents [25].

Change and Incident Governance

In regulated industries, production-impacting actions require formal change governance, and emergency actions follow incident-response processes. EAIGF mirrors both by querying the enterprise systems of record for these processes at authorization time. Before a production-impacting action, the enforcement boundary validates an associated change record, including approval status, change window, and risk classification, by looking it up in the enterprise change-management (ITSM) system; absent a valid, approved record, the action is denied. For emergency actions such as disabling a compromised account or freezing suspicious transactions, the enforcement boundary instead validates an active incident identifier, severity, ownership, and an approved response playbook against the enterprise incident-management system, ensuring emergency authority is exercised only within sanctioned incident-response contexts. In both cases the enforcement boundary is a consumer of authoritative governance records, not their source: it makes a runtime decision conditioned on the state of systems the organization already operates.

Privileged Session Monitoring and Post-Execution Review

For every privileged activity, the enforcement boundary records the agent identity, AIRS score, Trust Tier, business use case, skill invoked, MCP server, action, target system, timestamp, and governance approvals. Unlike conventional logs that capture only technical actions, Agent-PAM captures governance context, enabling reconstruction of both what occurred and why it was authorized. Each privileged action generates an activity-review record that governance teams periodically assess for policy compliance, privilege utilization, control effectiveness, and anomalous or excessive-privilege patterns, providing continuous post-execution assurance.

6.3 Agent Segregation of Duties (Agent-SoD)

Segregation of duties is a foundational control in regulated industries, formalized in RBAC research as a set of mutual-exclusion constraints over operations or roles [16]. EAIGF applies SoD to autonomous agents: an agent that creates a vendor record should not approve vendor payments; an agent that initiates a refund should not approve it; an agent that generates a risk assessment should not approve exceptions to it. Agent-SoD policies are expressed as mutual-exclusion constraints over capabilities and evaluated by the enforcement boundary before execution; a requested operation that would violate a constraint, whether within a single agent or across a delegated chain (Section 7.3), is denied or escalated for human review. This prevents excessive concentration of authority

within autonomous systems.

7. Runtime Enforcement Substrate

The preceding layers define policy; enforcement requires that policy be evaluated within the agent's execution path. EAIGF relies on a policy-mediated runtime boundary through which all agent interactions with skills, MCP servers, tools, APIs, and privileged resources must pass. Direct tool invocation, direct MCP access, direct privileged API execution, and direct credential usage are prohibited; every request is mediated. The underlying runtime mechanisms were introduced in the author's prior work [10], [11]; this section summarizes them for completeness and details how the identity layer integrates with them. The boundary applies the PDP/PEP separation standard in access-control architectures [20]: a Policy Decision Point evaluates each request against business approval, Trust Tier, AIRS score, skill and MCP authorization, data classification, SoD constraints, and privileged-access requirements, emitting permit, deny, escalate, or require-human-approval; a Policy Enforcement Point guarantees only permitted actions execute (Figure 2). An Identity Broker anchors these decisions by validating, before any action, that the agent is registered in AIR, owned, active in its lifecycle, certified, and within its Trust Tier and current AIRS bounds, denying unregistered, suspended, decommissioned, or uncertified agents.

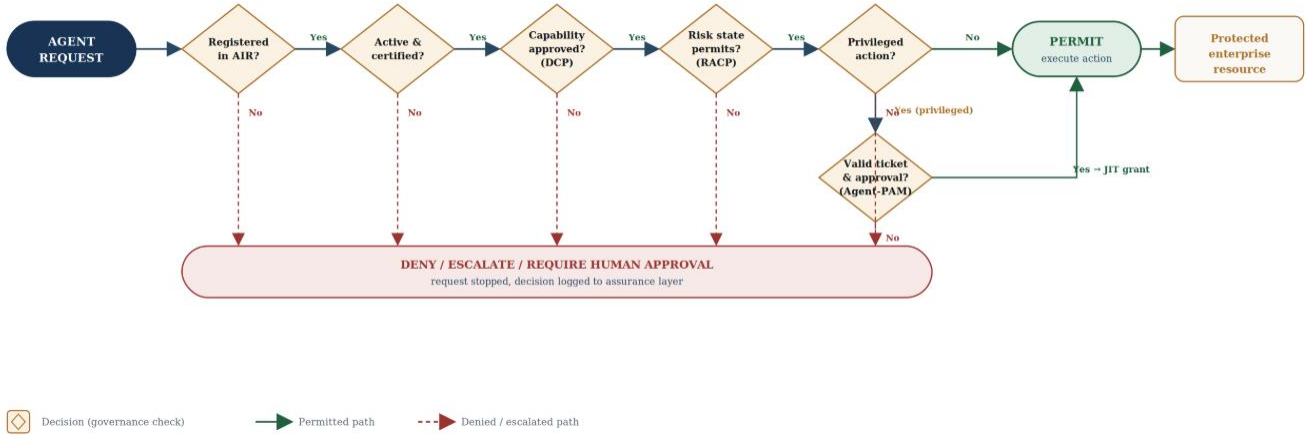


Figure 2. Runtime authorization decision flow. Every agent request passes the identity, certification, capability-projection (DCP), risk-state (RACP), and privileged-action (Agent-PAM / JIT) checks; only permitted actions execute, and every decision is logged to the assurance layer.

7.1 Dynamic Capability Projection (DCP)

Reusable skills accelerate development but create over-entitlement: a skill often encapsulates more capabilities than any single use case requires. A customer-operations skill might support reading, updating, freezing, unfreezing, and closing accounts, while a given use case requires only reading a profile. Granting the whole skill violates least privilege [13]. Dynamic Capability Projection (DCP), introduced in the author's prior work [10] and grounded in the object-capability principle of conferring only the narrow authority a component needs [17], [19], addresses this. Rather than exposing all capabilities of a skill, the enforcement boundary projects only the subset approved for the specific use case. Formally, for a skill capability set S and a use-case approved set A, the projected runtime set is the intersection $P = S \cap A$; all capabilities in $S \setminus A$ remain inaccessible despite being technically present. Two use cases can therefore safely consume one reusable skill under distinct authorization boundaries. In EAIGF, A is the entitlement profile stored in AIR (Section 6.1), which is what binds DCP to the identity layer: projection is computed against governed, certifiable entitlements rather than ad hoc configuration (Figure 3).

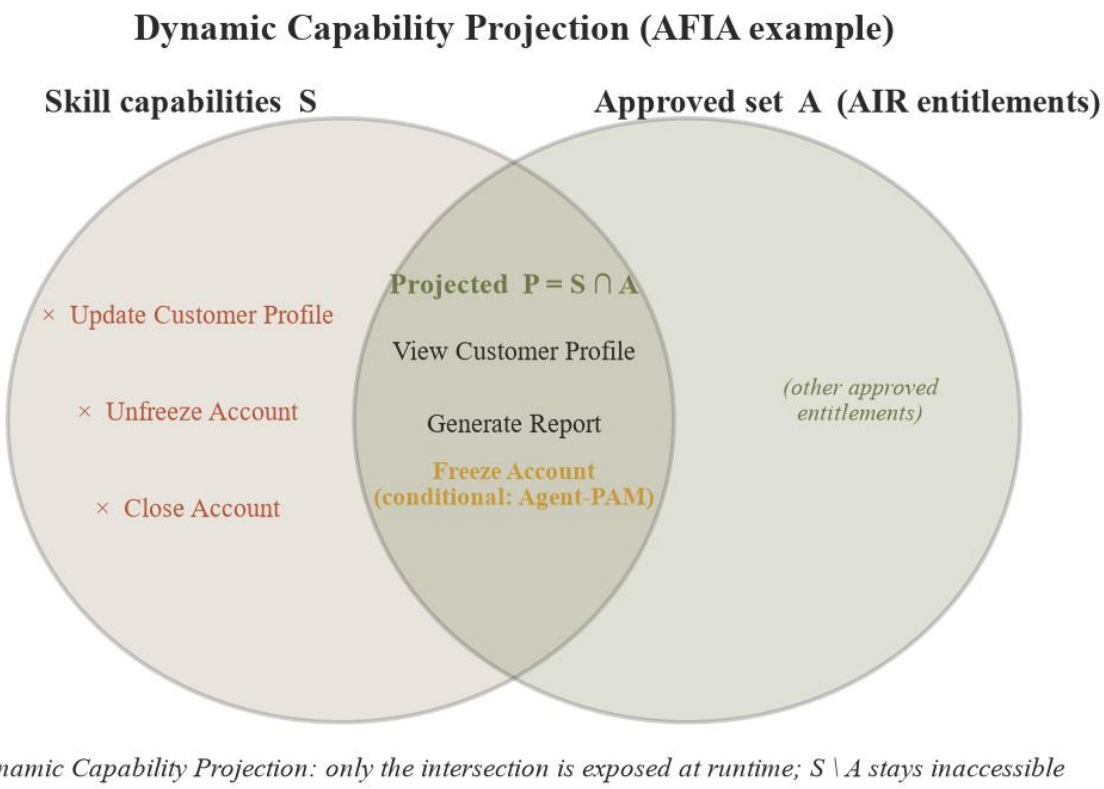


Figure 3. Dynamic Capability Projection. Only the intersection of a reusable skill's capabilities with the agent's AIR-approved entitlements is exposed at runtime; the remaining skill capabilities stay inaccessible.

7.2 Risk-Adaptive Capability Projection (RACP)

DCP constrains an agent to approved capabilities but does not respond to changing conditions after authorization. Risk-Adaptive Capability Projection (RACP), also introduced in the author's prior work [10], treats capabilities as adaptive privileges whose availability depends on continuously evaluated risk. The projected set becomes a function of identity, Trust Tier, AIRS score, runtime context, risk signals, and governance policy. Risk signals span security (for example prompt-injection attempts [6], unauthorized tool invocation, anomalous MCP interactions), operational (workflow deviation, repeated failures, abnormal resource use, objective drift), governance (expired approvals, failed certifications, SoD conflicts), and business context (active incidents, regulatory restrictions, maintenance windows). RACP applies one of four authorization states: Normal (capabilities available per DCP), Restricted (selected capabilities suppressed), Escalated (capabilities require additional approval), and Suspended (capabilities disabled pending review). The identity-layer integration is the coupling to AIRS: as signals accumulate, AIRS is recalculated, and crossing tier thresholds drives the state transitions. In the reference implementation, an agent operating at AIRS 70 sits in a restricted band; accumulation to the low-80s moves privileged capabilities to escalated, and saturation drives suspension, demonstrated empirically in Section 8 (Figure 4).

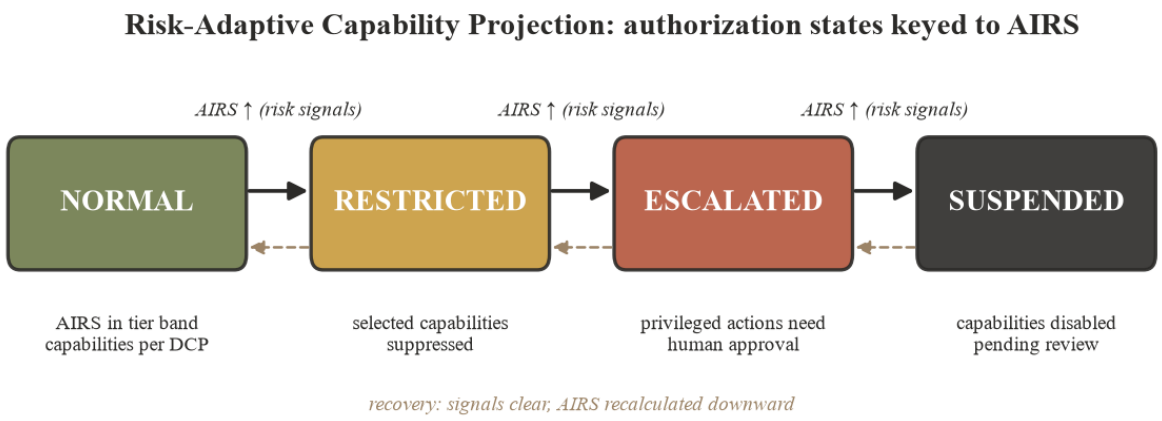


Figure 4. Risk-Adaptive Capability Projection. As runtime risk signals accumulate and AIRS is recalculated, the agent transitions through Normal, Restricted, Escalated, and Suspended states without any change to its static entitlements.

7.3 Delegated Trust Chain and Agent Identity Lineage

A defining difference between conventional software and agentic systems is that agents invoke reusable skills, communicate with peer agents, and delegate responsibilities across distributed execution chains. Traditional identity governance assumes a short, direct identity-to-action path, making accountability straightforward. In a multi-agent fraud workflow, by contrast, a user request may pass through an investigation agent, a transaction-analysis skill, an MCP server, and an enterprise tool before an account restriction executes, raising the question of which identity, decision, skill, integration, and approval produced the action. EAIGF answers this with two complementary constructs introduced in this paper (Figure 5).

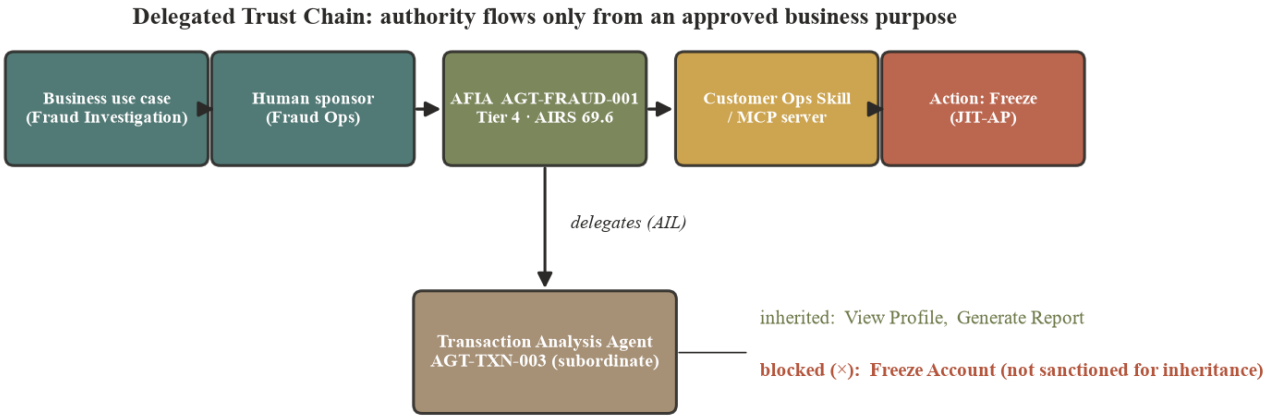
Delegated Trust Chain (DTC)

The Delegated Trust Chain records the continuous lineage of trust and authorization decisions across every component of an agentic workflow, from business use case through agent, skill, MCP server, tool, to business action. Its central principle, consistent with Zero Trust [14] and with the principal-agent problem of delegated authority studied in organizational economics [24], is that authority originates only from an approved business purpose and is delegated through governed relationships; agents, skills, and MCP servers possess no inherent authority. Each delegated action generates a trust-chain record capturing the originating use case, human sponsor, executing agent and its Trust Tier and AIRS score, invoked skill, accessed MCP and tool, executed action, authorization path, and timestamp. The resulting chain provides complete visibility into how authority propagated, supporting incident investigation, regulatory audit, and root-cause analysis, and ensuring reusable skills cannot become uncontrolled sources of authority.

Agent Identity Lineage (AIL)

As agents create or delegate to subordinate agents, Agent Identity Lineage tracks parent-child relationships throughout the lifecycle, recording creation and delegation relationships, inherited capabilities and trust classifications, and shared skill and MCP dependencies. The resulting lineage graph reveals which agents participated in a workflow, which delegated authority, and which capabilities were inherited. The principal governance rule is that capability inheritance is never automatic: a subordinate agent acquires an inherited capability only if it is explicitly approved, registered in AIR, projected by DCP, and subject to Agent-PAM. The enforcement boundary validates each delegation against these rules before authority transfers, preventing privilege escalation through agent chaining. Lineage also enables ecosystem-level risk evaluation: an otherwise low-risk agent that participates in workflows with high-tier financial or infrastructure agents may have an elevated

effective risk profile, which AIRS can incorporate through lineage features such as delegation depth and privileged-dependency count.



Agent Identity Lineage: capability inheritance is never automatic; escalation via delegation is blocked

Figure 5. Delegated Trust Chain and Agent Identity Lineage for the fraud workflow. Authority flows only from an approved business purpose, and a subordinate agent cannot inherit a privileged capability (account freeze) that was not sanctioned, blocking escalation through delegation.

8. Reference Implementation and Evaluation

To move beyond conceptual description, a self-contained reference implementation of the runtime enforcement boundary was built in Python using only the standard library, with SQLite as the AIR system of record. The implementation realizes the identity-layer enforcement that is the focus of this paper: AIR registration and lifecycle gating, the explicit AIRS formula and Trust Tier mapping of Section 5, capability projection against AIR-stored entitlements, risk-adaptive state transitions driven by AIRS recalculation, Agent-PAM privileged-action handling, and lineage-based inheritance control. The objective is not to measure model performance but to verify that governance policy is enforced correctly within the execution path, with no false-permits on disallowed capabilities.

Two points of method are important to the validity of what follows. First, decisions are computed, not predetermined. Each evaluation scenario supplies only inputs, the agent's registration state, entitlements, AIRS factors, delegation relationships, and any incident context; the permit, deny, escalate, or human-approval outcome is then produced by the same enforcement logic for every request, rather than being hard-coded per scenario. The scenarios therefore test whether the rules produce the intended decisions, not whether a lookup table returns expected strings. Second, all data is synthetic. Agents, AIRS risk-factor values, and runtime risk signals are constructed for the evaluation; none originate from any production system. In a production deployment, risk signals would be derived automatically from enterprise telemetry (for example, security monitoring and anomaly-detection systems), whereas here they are injected deliberately so that each authorization state can be exercised in a controlled, reproducible way.

8.1 Evaluation Scenarios and Results

Ten scenarios were defined, each exercising a distinct identity-layer control with a deterministic expected outcome. Table 4 reports expected versus observed decisions.

ID	Scenario	Expected	Observed
S1	Registered, approved, normal-risk read	Permit	Permit
S2	Unregistered agent (identity gate)	Deny	Deny
S3	Decommissioned agent (lifecycle gate)	Deny	Deny
S4	Failed access certification	Deny	Deny
S5	Capability outside DCP projection	Deny	Deny
S6	AIRS recalculated into escalation band	Human approval	Human approval
S7	Delegated child, non-inherited privilege	Deny	Deny
S8	Privileged action without incident ticket	Human approval	Human approval
S9	JIT-AP privileged grant with valid ticket	Permit	Permit
S10	AIRS above suspend threshold	Deny	Deny

Table 4. Governance scenario results. All ten scenarios enforced as expected; observed decisions matched expected decisions in every case.

Across the ten scenarios, the enforcement boundary produced the expected decision in all cases (10/10), with zero false-permits on disallowed capabilities. Scenarios S2-S5 confirm that identity, lifecycle, certification, and entitlement gates each independently deny unauthorized access. S6 and S10 confirm that AIRS recalculation alone, with no change to the agent's static entitlements, is sufficient to escalate privileged actions to human approval and ultimately to suspend authority, demonstrating risk-adaptive behavior. S7 confirms that a subordinate agent cannot acquire a privileged capability that was not explicitly approved for inheritance, blocking privilege escalation through delegation. S8 and S9 confirm Just-in-Time privileged access: the same action is gated to human approval without a valid incident ticket and granted with one.

8.2 Authorization Overhead

Because the enforcement boundary mediates every agent action, per-request overhead matters for practicality. Authorization latency was measured by timing each call individually with a high-resolution monotonic clock, recording the elapsed time for the complete decision (registry lookup, AIRS evaluation, capability projection, and risk-state determination), and aggregating over 10,000 sequential requests; the mean is the sum of the per-call elapsed times divided by 10,000. On a commodity Windows laptop the mean authorization latency was approximately 0.015 ms (median 0.014 ms, 95th percentile 0.015 ms) over 10,000 requests. While absolute figures depend on deployment, the result indicates that identity-layer mediation imposes negligible overhead relative to the latency of LLM inference and downstream tool calls that dominate agent execution.

8.3 Worked Walkthrough: Autonomous Fraud Investigation Agent

To show the constructs operating together on a realistic workflow, this subsection walks through a synthetic

financial-services scenario in which an Autonomous Fraud Investigation Agent (AFIA) assists analysts. All numerical values are produced directly by the reference implementation, ensuring the scoring rules and the narrative remain internally consistent; the agents and their risk-factor profiles are synthetic.

Registration and Trust Classification

Before deployment, AFIA is registered in AIR as AGT-FRAUD-001 under the Fraud Operations owner. Applying the AIRS formula of Section 5 to AFIA's synthetic risk-factor profile, high autonomy and privileged-action capability, high data sensitivity and financial impact, yields an AIRS of 69.6, placing AFIA in Trust Tier 4. AIR issues an Agent Provisioning Certificate documenting approved skills, approved MCP servers, authorized capabilities, and governance approvals. A lower-risk Customer Service Agent (AGT-CS-002), whose synthetic profile has low autonomy and minimal privileged-action and financial-impact values, scores AIRS 23.1 (Tier 2) under the same formula, illustrating how the scoring separates agents by operational impact. Both scores are computed by the formula from the agents' input factors, not assigned by hand; Table 5 shows the per-dimension computation.

Risk dimension	**Weight w_i**	**AFIA x_i**	**AFIA $w_i x_i$**	**CS-002 x_i**	**CS-002 $w_i x_i$**
Autonomy level	0.20	0.85	0.170	0.20	0.040
Data sensitivity	0.15	0.72	0.108	0.30	0.045
Privileged-action capability	0.20	0.82	0.164	0.10	0.020
Financial / customer impact	0.15	0.70	0.105	0.15	0.023
Skill count	0.08	0.50	0.040	0.35	0.028
MCP integration count	0.07	0.44	0.031	0.30	0.021
Regulatory exposure	0.08	0.55	0.044	0.30	0.024
Production access	0.07	0.48	0.034	0.43	0.030
$\Sigma w_i x_i \rightarrow$ AIRS	**1.00**		**0.696 → 69.6**		**0.231 → 23.1**

Table 5. Worked AIRS computation for the walkthrough agents. Factor values (x_i) are illustrative values consistent with the reported scores of 69.6 (AFIA, Tier 4) and 23.1 (AGT-CS-002, Tier 2) and should be replaced with the reference implementation's actual inputs. Under runtime risk signals AFIA recomputes to 81.8, crossing the escalation threshold (Table *7*).

Dynamic Capability Projection

AFIA consumes a reusable Customer Operations Skill supporting viewing, updating, freezing, unfreezing, and closing accounts and generating investigation reports. Under naive authorization AFIA would inherit all of

these. DCP projects only the intersection of the skill's capabilities with AFIA's AIR-approved entitlements, as shown in Table 6.

Capability	In Skill	Projected to AFIA
View Customer Profile	Yes	Yes
Generate Investigation Report	Yes	Yes
Freeze Account	Yes	Conditional (Agent-PAM)
Update Customer Profile	Yes	No
Unfreeze Account	Yes	No
Close Account	Yes	No

Table 6. Dynamic Capability Projection for AFIA. Unapproved capabilities remain inaccessible despite being present in the reusable skill.

Privileged Action and Just-in-Time Authorization

On identifying coordinated high-risk activity, AFIA determines an account freeze is warranted. AFIA holds no standing authority to freeze accounts. The enforcement boundary validates identity, AIR registration, Trust Tier, AIRS score, an associated incident record, policy, and required approvals; only then is a single-use, time-bound credential brokered through the enterprise PAM infrastructure, after which the privilege is revoked and the action logged. In the evaluation, a freeze request without a valid incident record is gated to human approval (scenario S8), while the same request with a valid record and normal risk is granted under JIT-AP (scenario S9).

Risk Escalation

During a later investigation the enforcement boundary observes repeated attempts to reach an unauthorized MCP server and abnormal transaction-retrieval volume. These signals raise AFIA's AIRS from 69.6 into the low-80s. Without any change to AFIA's static entitlements, RACP transitions the Freeze Account capability from available to escalated, so subsequent freeze requests require human-in-the-loop review and secondary approval; continued anomalies drive suspension. Table 7 summarizes the transition (scenarios S6 and S10).

Metric	Normal State	Escalated State
AIRS score	69.6	81.8
Freeze Account	JIT-authorized	Human approval required
MCP access	Standard	Restricted
Privileged actions	JIT-AP	Escalated review

Table 7. Risk-adaptive transition for AFIA as AIRS crosses the escalation threshold. Values are computed by the reference implementation.

Delegation and Lineage

AFIA delegates transaction-pattern analysis to a subordinate Transaction Analysis Agent (AGT-TXN-003). AIL records the parent-child relationship, and the lineage policy approves only profile viewing and report generation for inheritance, not account freezing. When the subordinate requests a freeze, the enforcement boundary denies it as an uninherited capability, blocking privilege escalation through the delegation chain (scenario S7). The Delegated Trust Chain records every hop from business use case through AFIA, the analysis agent, the skill, and the MCP server, preserving end-to-end accountability.

8.4 Large-Scale Ecosystem Evaluation

The ten scenarios of Section 8.1 verify each control individually but are hand-constructed. To test the framework against requests that are not individually curated, a second evaluation generates a randomized synthetic ecosystem and measures aggregate behaviour, including a comparison against a naive role-based baseline. Using a fixed random seed for reproducibility, the evaluation instantiates 80 agents with randomized eight-dimension risk profiles, 25 reusable skills each exposing a random subset of a twelve-capability vocabulary, and delegation relationships among roughly half the agents. It then issues 600 authorization requests, of which approximately 35 percent are deliberately adversarial (requesting capabilities the agent is not entitled to, or attempting to inherit privileged capabilities through delegation that were never sanctioned). Each request is evaluated both by the EAIGF enforcement boundary and by a naive baseline that grants the agent every capability contained in the invoked skill, the common pattern when a reusable skill is authorized as a unit.

For each request, a policy-correct ground-truth decision is computed independently of the harness, so that the harness can be scored against it rather than against its own logic. Table 8 reports the aggregate results, visualized in Figure 6.

Metric	Naive RBAC baseline	EAIGF
Enforcement accuracy vs. ground truth	—	100.0% (600/600)
False-permits on disallowed capabilities	169	0
Delegation-escalation attempts blocked	0 of 114	114 of 114
Exposed capability surface (grants)	4188	1393
Reduction in exposed surface	—	66.7%

Table 8. Aggregate results over 600 randomized authorization requests (80 agents, 25 skills, 35% adversarial, fixed seed). The naive baseline grants the whole skill; EAIGF projects only governance-approved capabilities. Results are deterministic and reproduce exactly.

Three findings stand out. First, across 600 requests that were not individually designed, the enforcement boundary produced the policy-correct decision in every case, with zero false-permits on disallowed capabilities. Second, the naive baseline admitted 169 false-permits over the same workload, each corresponding to an agent exercising a capability that a reusable skill happened to contain but that governance had not approved, the precise over-entitlement that dynamic capability projection is designed to prevent. Third,

every one of the 114 attempts to escalate privilege by inheriting an unsanctioned privileged capability through a delegation chain was blocked. The aggregate reduction in exposed capability surface, the total count of capability grants an agent could exercise summed across all requests, was 66.7 percent relative to the naive baseline. These results quantify, on emergent rather than curated inputs, the least-privilege benefit that the framework provides over whole-skill authorization.

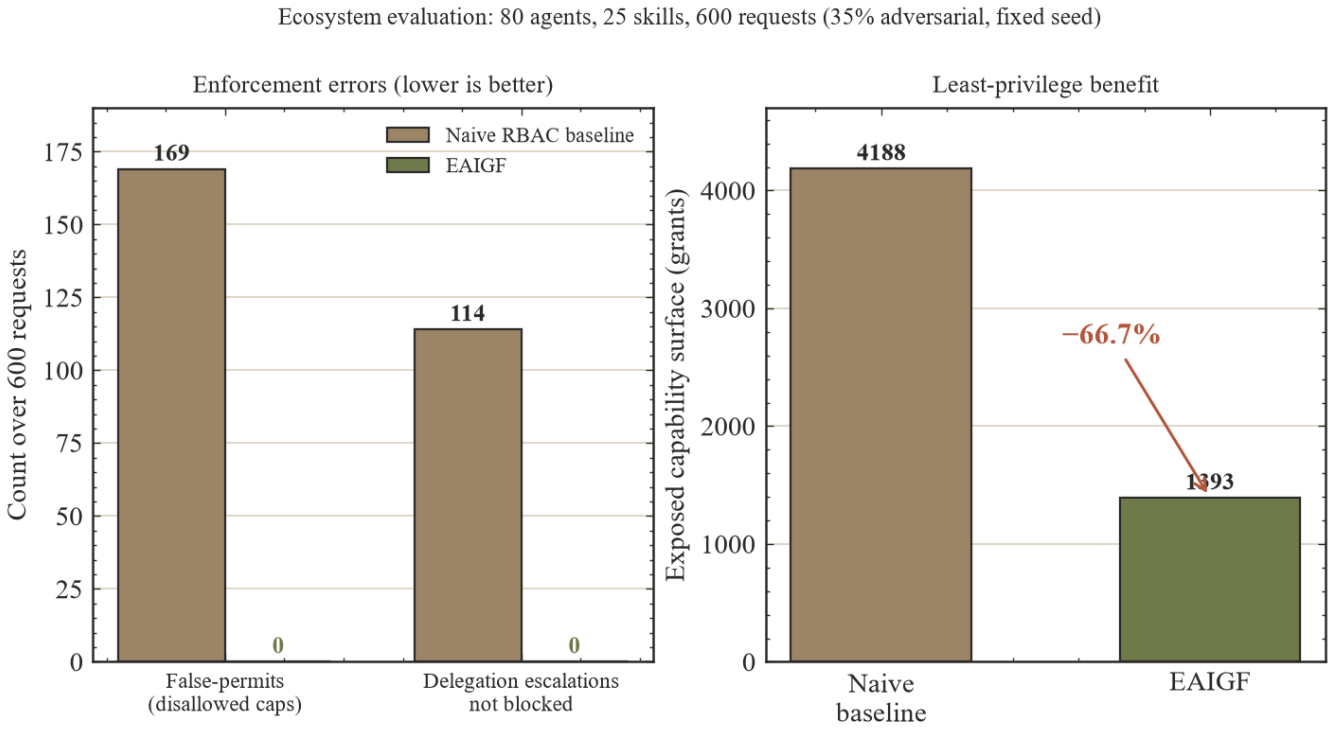


Figure 6. Ecosystem evaluation over 600 randomized authorization requests. EAIGF admits zero false-permits and blocks every delegation-based escalation, reducing the exposed capability surface by 66.7 percent relative to the naive whole-skill baseline.

8.5 Threats to Validity

The reference implementation validates enforcement logic, not organizational efficacy. Both evaluations are synthetic: agents, risk-factor profiles, and risk signals are generated rather than drawn from production systems, and risk signals are injected rather than derived from live telemetry. AIRS weights are illustrative and would require per-institution calibration; the evaluation uses a single-node, in-memory registry rather than a distributed deployment; and no live language-model agent is placed in the request path. These choices isolate the governance logic under test and keep the results fully reproducible. They also bound the claims: the evaluations demonstrate that the framework's controls enforce policy correctly, efficiently, and with substantial least-privilege benefit over a naive baseline, but they do not by themselves establish behaviour under production traffic. Empirical validation in deployed environments, including a live-agent harness and measurement of privilege-incident reduction and certification-finding rates, remains future work.

9. Comparative Evaluation and Discussion

Complementing the enforcement evidence of Section 8, EAIGF is positioned qualitatively against traditional IAM and against widely used AI governance frameworks. The comparison reflects each approach's primary focus rather than an exhaustive control-by-control mapping; the cited frameworks are general-purpose and may be extended by individual organizations beyond what their core text prescribes.

Capability	Traditional IAM	Typical AI Governance Frameworks	EAIGF
Identity registration for agents	Not agent-specific	Use-case / system inventory	Agent-level registry (AIR)
Context- and risk-dependent authorization	Limited	Not addressed	Yes
Runtime capability projection	Not addressed	Not addressed	Yes (DCP)
Agent lifecycle governance	Not agent-specific	Partial (lifecycle risk mgmt)	Yes (Agent-JML)
Delegated-trust accountability	Not addressed	Not addressed	Yes (DTC / AIL)
Privileged access mgmt for agents	Human/service-account focus	Not addressed	Yes (Agent-PAM)
Risk-adaptive authorization	Not addressed	Not addressed	Yes (RACP)
Quantitative agent risk score	Not addressed	No scoring model prescribed	Yes (AIRS)

Table 9. Indicative comparison of governance focus. "Typical AI governance frameworks" refers to NIST AI RMF [8], ISO/IEC 42001 [9], and MITRE ATLAS [21]. NIST AI RMF addresses lifecycle risk management, including post-deployment monitoring, change management, and decommissioning at the system level (e.g., the Manage function), but does not prescribe agent-level identity governance or a risk-scoring model; the table reflects this distinction rather than asserting these frameworks lack all related controls.

The comparison indicates that EAIGF complements rather than replaces existing frameworks. Organizations may continue using NIST AI RMF and ISO/IEC 42001 for model and use-case governance while adopting EAIGF for identity-specific control of autonomous agents; indeed, because NIST AI RMF prescribes outcomes without a scoring model, AIRS can be read as one concrete instrument for the quantitative risk characterization the Measure function calls for. The framework maps naturally onto the requirements of regulated sectors, finance, healthcare, critical infrastructure, and government, because it extends established IGA, PAM, and Zero Trust practices rather than introducing an entirely new paradigm, allowing organizations to reuse existing certification, audit, and segregation-of-duty processes.

9.1 Limitations

Several limitations qualify these results. The framework is evaluated through a reference implementation and a synthetic walkthrough rather than multi-organization production deployment. AIRS weights and tier thresholds are illustrative and require calibration. Implementation complexity may grow in environments with thousands of interacting agents, skills, and MCP servers. The current design assumes centralized governance through a single enforcement boundary; decentralized or cross-organizational trust requires additional mechanisms. Finally, EAIGF governs agent identity and authority and does not address model alignment, fairness, or explainability,

which remain complementary research areas.

10. Future Work

Several directions extend this work. The randomized evaluation of Section 8.4 quantifies the least-privilege benefit on synthetic but emergent inputs; a natural next step is to place a live language-model agent in the request path, so that requests are generated by a reasoning agent rather than constructed, and to measure enforcement under genuinely unpredictable agent behaviour. Publishing the reference implementation and evaluation scripts as an open artifact would further support reproducibility and independent verification. Beyond evaluation, an Agent Bill of Materials (ABOM) analogous to a software bill of materials could document agent dependencies, skills, MCP integrations, models, and authorization pathways to improve supply-chain transparency. Agent Identity Attestation could let organizations cryptographically verify an agent's identity, governance state, and trust classification before permitting interaction. Finally, cross-enterprise agent federation will require standardized trust frameworks and delegated-authorization protocols for secure agent-to-agent collaboration across organizational boundaries. Industry standardization of agent registries, trust tiering, and capability projection may make these constructs foundational building blocks for enterprise AI governance.

11. Conclusion

Agentic AI systems are becoming autonomous digital workers that reason, plan, invoke tools, access enterprise systems, and perform business actions. Existing AI governance frameworks address model risk and responsible AI but not the identity-governance challenges these agents introduce. This paper presented the Enterprise Agent Identity Governance Framework, which treats autonomous agents as first-class enterprise identities and introduces an identity layer, the Agent Identity Registry, Agent Provisioning Certificate, an explicit Agent Identity Risk Score driving Trust Tiering, Agent Privileged Access Management, and a Delegated Trust Chain with Agent Identity Lineage, enforced through a policy-mediated runtime boundary. A self-contained reference implementation enforced all ten governance scenarios correctly with zero false-permits and negligible authorization overhead, and a larger randomized evaluation reduced exposed capability surface by two-thirds against a naive baseline while blocking every delegation-based privilege escalation; a synthetic financial-services walkthrough demonstrated governance of a high-risk agent with auditability and controlled delegation. As enterprises deploy agentic systems at scale, identity-centric governance will become as essential to autonomous agents as traditional IAM is to human users and applications. EAIGF offers a practical foundation for governing autonomous digital workers in a secure, accountable, and scalable manner.

References

1. Anthropic, “Building Effective Agents,” Anthropic Engineering, 2024.
2. Anthropic, “Model Context Protocol Specification,” 2024.
3. Microsoft, “Governing and Securing AI Agents in the Enterprise,” Microsoft Security, 2024.
4. Y. Shavit et al., “Practices for Governing Agentic AI Systems,” OpenAI, 2023.
5. L. Weng, “LLM-Powered Autonomous Agents,” 2023.
6. K. Greshake, S. Abdelnabi, S. Mishra, C. Endres, T. Holz, and M. Fritz, “Not What You’ve Signed Up For: Compromising Real-World LLM-Integrated Applications with Indirect Prompt Injection,” in Proc. ACM Workshop on Artificial Intelligence and Security (AISec), 2023.
7. A. Chan et al., “Harms from Increasingly Agentic Algorithmic Systems,” in Proc. ACM Conf. on Fairness, Accountability, and Transparency (FAccT), 2023.
8. National Institute of Standards and Technology, “Artificial Intelligence Risk Management Framework (AI RMF 1.0),” NIST, 2023.
9. International Organization for Standardization, “ISO/IEC 42001:2023 — Information Technology, Artificial Intelligence, Management System,” ISO, 2023.
10. S. Anuguthala, “Enterprise Governance of Reusable

Agentic AI Skills," International Journal on Science and Technology (IJSAT), in press, 2026.

11. S. Anuguthala, "GUARD: A Governance-Anchored Framework for Runtime Monitoring and Incident Response in Enterprise Agentic AI Systems," International Journal of Artificial Intelligence, Big Data, Cloud and Management Systems (IJAIBDCMS), in press, 2026.

12. P. A. Grassi, M. E. Garcia, and J. L. Fenton, "Digital Identity Guidelines," NIST Special Publication 800-63-3, 2017.

13. Joint Task Force, "Security and Privacy Controls for Information Systems and Organizations," NIST Special Publication 800-53 Rev. 5, 2020.

14. S. Rose, O. Borchert, S. Mitchell, and S. Connelly, "Zero Trust Architecture," NIST Special Publication 800-207, 2020.

15. M. Bishop, Computer Security: Art and Science, 2nd ed. Addison-Wesley, 2018.

16. R. S. Sandhu, E. J. Coyne, H. L. Feinstein, and C. E. Youman, "Role-Based Access Control Models," IEEE Computer, vol. 29, no. 2, pp. 38–47, 1996.

17. M. S. Miller, "Robust Composition: Towards a Unified Approach to Access Control and Concurrency Control," Ph.D. dissertation, Johns Hopkins University, 2006.

18. J. B. Dennis and E. C. Van Horn, "Programming Semantics for Multiprogrammed Computations," Communications of the ACM, vol. 9, no. 3, pp. 143–155, 1966.

19. R. N. M. Watson, J. Anderson, B. Laurie, and K. Kennaway, "Capsicum: Practical Capabilities for UNIX," in Proc. USENIX Security Symposium, 2010.

20. OASIS, "eXtensible Access Control Markup Language (XACML) Version 3.0," OASIS Standard, 2013.

21. MITRE, "ATLAS: Adversarial Threat Landscape for Artificial-Intelligence Systems," MITRE Corporation, 2023.

22. Cloud Security Alliance, "Agentic AI Threat Modeling: MAESTRO," CSA, 2024.

23. S. Anuguthala, "A Control-Driven Lifecycle Governance Framework for Enterprise Agentic AI Systems," International Journal of Artificial Intelligence, Data Science and Machine Learning (IJAIDSML), in press, 2026.

24. K. J. Arrow, "The Economics of Agency," in Principals and Agents: The Structure of Business, J. W. Pratt and R. J. Zeckhauser, Eds. Harvard Business School Press, 1985.

25. K. Naik and D. Patel, "Just-in-Time and Zero Standing Privilege Access Models for Cloud Environments," Journal of Information Security and Applications, vol. 71, 2023.

26. B. Schneier, "The Coming AI Agents and the Security Challenges They Pose," 2024.