plainID
THE IDENTITY SECURITY COMPANY

Secure by Design
Build Agent AI Responsibility with A Policy-Based Access Control

Download

TABLE OF CONTENTS

Secure by Design: Build Agent AI Responsibility with A Policy-Based Access Control

1. The New Threat Landscape of Agentic AI

1.1. Introduction: A Paradigm Shift in Enterprise Operations and Risk

Agentic AI systems, a new class of autonomous software entities capable of planning, reasoning, and executing complex, multi-step tasks with minimal human supervision, are no longer conceptual. Their adoption is an inevitable paradigm shift in enterprise operations. This evolution can be understood across three key stages: from single-task **Interactive Agents** that execute well-defined commands; to goal-oriented **Autonomous Agents** capable of multi-step problem solving; and finally, to sophisticated **Digital Employees** that engage in dynamic, learning-driven collaboration and are deeply integrated into organizational structures. This progression from assistive tool to autonomous collaborator fundamentally redefines the enterprise attack surface. As these "Digital Employees" become integral to business workflows, they gain access to an organization's most valuable data and systems, creating an unprecedented intersection of operational power and security risk. This new reality demands a complete re-evaluation of our approach to identity and access management.

1.2. The Failure of Traditional IAM in the Agentic Era

Traditional Identity and Access Management (IAM) paradigms, built on protocols like OAuth 2.1 and SAML, were designed for predictable human users and static applications. They are fundamentally inadequate for the dynamic, autonomous, and often ephemeral nature of Agentic AI. Merely adapting these old protocols is insufficient; their core assumptions break down when faced with the complexities of autonomous agents.

Key shortcomings of traditional IAM include:

- **Coarse-Grained and Static Permissions:** Traditional protocols rely on predefined, static scopes or roles (e.g., read_all_data) that are far too broad for the fluid operational needs of AI agents. Agents require granular, resource-specific, and context-aware permissions that can change moment-by-moment, such as accessing "all files in the /taxinfo directory." Static scopes lead to massive over-privileging. Furthermore, heavy reliance on assertions creates performance bottlenecks for machine-speed authentication requirements, where AI agents may need to authenticate 148 times more frequently than human users.
- **Lack of Agent Identity Recognition:** Conventional protocols do not distinguish between human users and AI agents, making it *impossible* to apply agent-specific policies, track agent actions separately, or understand the full context of an automated workflow. Every AI agent, as a Non-Human Identity (NHI), requires its own distinct identity. Agent IDs

represent a subset of NHIs that are uniquely autonomous and goal-driven, a distinction critical for establishing a foundation of Zero Trust.

- **The Confused Deputy Problem:** This critical vulnerability arises when an agent inherits broad system permissions instead of being constrained by the specific entitlements of the user it acts for. An agent running on a system with privileged access could be manipulated into performing unauthorized actions that the user themselves could not, creating a significant security gap that traditional IAM fails to address.
- **Inadequate Delegation Support:** AI agents often act on behalf of users, spawn sub-agents, or collaborate in complex chains. Traditional IAM protocols lack the mechanisms to securely model and trace these complex delegation chains, making it nearly impossible to maintain clear accountability and enforce the principle of least privilege throughout a multi-agent workflow.

1.3. Quantifying the Risks: Privilege Escalation, Data Exposure, and Auditability Gaps

Uncontrolled agentic systems introduce a new class of enterprise risks that can lead to catastrophic security breaches, costly business disruptions, and lasting reputational damage. Proactively addressing these threats is not just a best practice but a business necessity.

- **Privilege Escalation** Agentic AI systems frequently operate with permissions that exceed the requesting user's authorized access levels. Because traditional security models often fail to enforce the user's specific entitlements within the agent's operational context, agents can become over-privileged, creating direct pathways to unauthorized data, restricted actions, and system disruption.
- **Data Exposure** In architectures like Retrieval-Augmented Generation (RAG), AI agents query internal knowledge bases, documents, and databases to enrich their responses. This creates a high-risk intersection where an agent can access, summarize, and inadvertently expose sensitive or regulated data that the end-user is not authorized to see. For many organizations, oversharing due to inappropriate access controls is emerging as the major obstacle to wider GenAI rollouts, leading to compliance violations, substantial financial penalties, and irreversible harm to brand trust.
- **Lack of Auditability** The complexity of AI actions, including dynamic, multi-step reasoning and calls to multiple external tools, makes it exceedingly difficult to trace and audit what services and data were accessed. This lack of a clear, verifiable audit trail complicates forensic investigations, makes proving compliance with regulations impossible, and undermines the ability to ensure that AI systems are operating as intended.

To counter these risks, organizations must adopt a "**Security by Design**" philosophy, embedding proactive, identity-aware security controls into the core of their AI architecture from the very inception of any project.

2. A Modern Framework for AI Governance: Policy-Based Access Control (PBAC)

2.1. Introduction: Establishing a Foundation of Zero Trust for AI

For too long, security has been treated as a patch, something added after systems are built, tested, and shipped. But in today's landscape of accelerating digital risks and autonomous AI agents, that model no longer works. Security must be, by Design meaning **embedding security into every decision, architecture, and process,** not bolting it on later.

To address the security gaps inherent in the agentic era, a new architectural foundation is required. Policy-Based Access Control (PBAC) provides the strategic solution, enabling organizations to move beyond static, perimeter-based security and implement a Zero Trust model tailored for AI. In an agentic environment, security cannot be an afterthought; it must be dynamic, context-aware, and identity-centric. Where traditional IAM's static scopes lead to over-privileging, PBAC delivers granular, identity-aware authorization; where traditional protocols are blind to agent identity, PBAC makes it a cornerstone of every decision. This approach provides the framework to ensure every action taken by an AI agent, whether accessing data, using a tool, or generating a response, is explicitly verified against centrally managed policies in real time, establishing the necessary guardrails for responsible AI adoption.

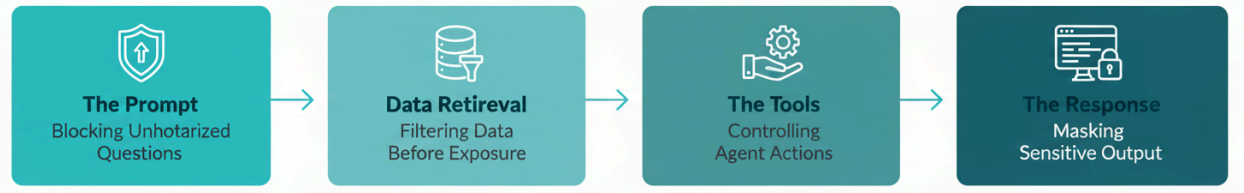
2.2. The Core Tenets of Policy-Based Access Control

PBAC is uniquely suited for governing Agentic AI because its core principles are designed to handle the complexity and dynamism that traditional models cannot. These tenets empower organizations to build AI responsibly, with auditable, consistent, and fine-grained access decisions.

- ***Identity-Aware Authorization:*** PBAC leverages rich identity attributes of all actors, including human users and non-human AI agents, to make highly specific and context-driven access decisions. By understanding the roles, departments, and other contextual factors of both the user and the agent, it ensures that only the right actors can access the right data at the right time.
- ***Granular Access Control:*** Policies are designed to enable precise, fine-grained control over data, tools, and actions. This moves security beyond overly broad roles and static scopes, allowing permissions to be defined at the level of individual data records or API functions, thereby enforcing the principle of least privilege.
- ***Enhanced Security Through Context:*** PBAC excels at incorporating real-time contextual factors into authorization decisions. Policies can dynamically adapt based on variables such as user location, device security, data sensitivity, or even anomalous agent behavior. This contextual awareness is critical for reducing the risk of unauthorized access in a constantly changing operational environment.

- ***Centralized Management and Scalability:*** PBAC automates and centralizes the management of access policies. This allows security and governance teams to define rules that are enforced consistently across a distributed and complex technology ecosystem. As the number of AI agents and applications explodes, this centralized approach ensures that security can scale without becoming unmanageable.

3. Implementing Identity-Aware Controls Across the Full AI Lifecycle



3.1. Introduction: The Imperative of End-to-End Enforcement

Securing Agentic AI is not about placing a single gateway in front of a model. It requires embedding fine-grained, identity-aware controls throughout the entire operational flow, from the initial prompt to the final generated response. Each stage of an AI workflow represents a potential point of data exposure or unauthorized action, making a layered, defense-in-depth strategy essential. Adopting **one platform to control access across the full AI flow** is the only way to ensure that security and compliance are maintained from end to end. This proactive approach enforces controls at the earliest possible point and at every subsequent control point in the workflow.

3.2. Stage 1: The Prompt - Blocking Unauthorized Questions

Security must begin the moment a prompt is submitted. Following the security principle of implementing controls as early as possible in the process, a PBAC framework can evaluate the user's question against their identity and entitlements before any data is retrieved or any model is invoked. This first line of defense determines whether the user is authorized to ask the question in the first place. For example, a prompt from a junior employee asking about their manager's salary should be blocked at this initial stage, preventing the system from wasting resources or triggering a downstream flow of events that could lead to data leakage.

3.3. Stage 2: The Data Retrieval - Filtering Data Before Exposure

In Retrieval-Augmented Generation (RAG) architectures, the system searches internal knowledge bases and databases to enrich the AI's response. This is a primary vector for data leakage. A PBAC approach secures this process by applying identity-aware policy checks directly to the data sources. Crucially, this enables the proactive filtering of sensitive structured

and unstructured data *before* it is retrieved and sent to the Large Language Model (LLM). This method contrasts sharply with weaker, emerging techniques that perform filtering *after* retrieval. By fetching only authorized data based on the user's identity and context, this approach ensures the AI agent never gains access to unauthorized information, eliminating a critical security vulnerability.

3.4. Stage 3: The Tools - Controlling Agent Actions and Agency

A significant risk with autonomous agents is "Excessive Agency," where an agent uses authorized tools to perform unauthorized actions. For instance, an agent that can use a ServiceNow MCP server might decide to close long-pending tickets with no relevant control. A policy management platform can mitigate this risk by controlling access to external tools and services, such as those using the Model Context Protocol (MCP). Policies enforce access to authorized tools based on the verified identity and context of both the human user and the AI agent, ensuring that agency remains within predefined, acceptable bounds.

3.5. Stage 4: The Response - Masking Sensitive Output in Real Time

The final layer of defense is applied to the AI-generated output. Even with controls at the prompt, data, and tool stages, a generative model can inadvertently blend restricted insights into an otherwise benign reply. Dynamic response masking automatically redacts or masks sensitive information, such as Personally Identifiable Information (PII), in the AI-generated response based on the user's entitlements. For instance, if an analysis of employee performance exposes a home address, that specific data point is masked before the response is delivered. This ensures the final output is secure and compliant, preventing data leakage through the LLM's synthesized answers.

4. Architecting an Enterprise-Grade Authorization Platform for AI

4.1. Introduction: From Policy to Practice

Moving from a conceptual framework to a functional security architecture requires an enterprise-grade platform capable of implementing policy-based controls at scale. Such a platform must combine centralized management with distributed enforcement to deliver consistent, real-time authorization across a complex and often hybrid AI ecosystem. It serves as the single source of truth for access decisions, ensuring that policies are defined in one place and enforced everywhere. The following components and capabilities are essential for building a robust, enterprise-ready authorization service for Agentic AI.

4.2. Key Architectural Components

A dynamic authorization service is composed of several key components that work in concert to manage and enforce access policies for Agentic AI workflows.

- **Policy Administration Point (PAP):** This is the centralized interface for creating, managing, and governing access policies. The PAP provides a user-friendly environment where both technical and non-technical stakeholders can define access rules that govern AI interactions.
- **Policy Information Point (PIP):** The PIP is responsible for retrieving real-time contextual attributes about users, agents, data, and the environment. It integrates with various sources like data catalogs and identity providers to supply the Policy Decision Point with the rich information needed to make informed authorization decisions.
- **Policy Decision Point (PDP):** As the "brain" of the architecture, the PDP is the engine that evaluates access requests from agents against the defined policies in real time. It takes the request and the contextual data from the PIP to render a "grant" or "deny" decision for every action at every stage of the AI pipeline.
- **Policy Enforcement Point (PEP):** The Policy Enforcement Points are the distributed components that enforce the PDP's decisions. These authorizers intercept access requests at each stage of the AI pipeline, prompt, data retrieval, tool usage, and response generation, and ensure that the decision from the PDP is carried out.

4.3. Essential Business and Technical Capabilities

An enterprise-grade platform for Agentic AI security must deliver a comprehensive set of capabilities that address the needs of both governance leaders and technical architects.

Business Capabilities (For Governance, Risk, and Compliance)	Technical Capabilities (For Security Architecture and Development)
Enterprise-Grade Scalability: Handles the performance and complexity demands of large-scale, hybrid enterprise AI deployments.	**Identity-Aware Enforcement for All Actors:** Ties every action within the AI workflow to a verified identity and its entitlements, with full context for both human users and non-human AI agents (NHIs) to build a Zero Trust foundation.
Full Visibility & Auditability: Provides a complete, traceable audit trail of the entire agentic AI flow and all interactions to satisfy compliance and governance needs.	**Dynamic Response Masking:** Automatically redacts or masks sensitive information in AI-generated responses in real time based on user entitlements, preventing data leakage through model outputs.

Unified Policy Management: Enables non-technical stakeholders to define and approve policies using a business-readable language, while providing developers with Policy-as-Code and low-code framework support.	**Proactive Data Filtering:** Prevents data exposure by filtering sensitive structured and unstructured data *before* it is retrieved by the AI agent.
Policy Simulation & Investigation: Allows teams to simulate "what-if" scenarios to analyze how policy changes will affect an AI agent's access rights and investigate the full reasoning behind any authorization decision.	**Low-Code Support for AI Development Frameworks:** Provides ready-to-use libraries and connectors that simplify the integration of authorization controls directly into AI development frameworks.

5. Conclusion: Enabling Responsible and Secure AI Adoption

5.1. Summary of the Path to Responsible Agentic AI

As enterprises accelerate their adoption of Agentic AI, it is clear that traditional security models designed for human-centric interactions are no longer sufficient. The autonomous, dynamic, and complex nature of these systems requires a fundamental shift toward a security framework that is identity-aware, context-driven, and enforced end-to-end. A comprehensive Policy-Based Access Control (PBAC) framework is the only viable foundation to mitigate critical risks like privilege escalation, data exposure, and auditability gaps. By embedding granular, real-time controls at every stage of the AI workflow, from prompt to response, organizations can harness the transformative power of AI while ensuring its usage is secure, compliant, and fully explainable. In the autonomous enterprise, identity-aware authorization is no longer just a security feature, it is a core business enabler and a competitive necessity.

5.2. Key Takeaways for Security and Compliance Leaders

To navigate the new landscape of Agentic AI, security and compliance leaders should prioritize the following strategic actions:

1. **Build AI with Identity-First Security:** Mandate that every AI action is bound to a verified user’s identity and entitlements. This enforces a Zero Trust foundation where permissions are explicit and tied to a known actor, not inherited from the underlying system.
2. **Centralize Control Across the Full AI Flow:** Implement a unified policy engine to manage access decisions at every stage: prompt, data retrieval, tool usage, and response

generation. This ensures consistent, auditable, and scalable governance across all AI-driven processes.

3. **Minimize Risk with Dynamic, Context-Aware Enforcement:** Leverage real-time, context-aware controls to reduce data exposure, misuse, and unauthorized actions by AI agents. Policies that adapt to runtime context are essential for mitigating the dynamic threats posed by autonomous systems.
4. **Enable Auditing and Observability:** Ensure your AI security architecture provides full traceability of access decisions and agent behavior. Comprehensive logging and visibility are non-negotiable for meeting regulatory requirements and supporting forensic analysis.