# Technical White Paper

# Securing the AI Action Plane

***An Architectural Framework for Identity, Governance, and Trust in Enterprise AI***

Prepared by

**Donnie Wendt, DSc, CISSP, AIGP**

July 2026

---

**Author's Note**

This paper explores architectural approaches for governing enterprise AI systems that can perform actions on behalf of users. While the Model Context Protocol (MCP) serves as a motivating example throughout the paper, the proposed framework is intended to apply broadly to enterprise AI systems that interact with organizational resources through delegated authority. The concepts presented reflect the author's perspective on emerging AI governance challenges and are intended to stimulate discussion regarding practical approaches to securing increasingly autonomous AI environments.

---

## Executive Summary

Artificial intelligence is rapidly evolving from a technology that generates information to one that performs actions on behalf of users. Modern AI systems can retrieve enterprise data, invoke APIs, execute workflows, update business applications, and coordinate activities across multiple systems. Protocols such as the Model Context Protocol (MCP) accelerate this evolution by providing a standardized mechanism for AI models to discover and interact with enterprise tools and services. As AI systems become active participants in business operations, they introduce new security challenges that extend beyond those addressed by traditional application security.

Much of the current discussion surrounding AI security focuses on individual implementation controls such as authentication, encrypted communications, prompt injection defenses, and secure API design. While these controls remain essential, they represent only individual components of a broader governance architecture. Organizations are no longer securing applications that simply process information; they must govern AI systems capable of making decisions and performing actions that directly affect enterprise operations.

This paper introduces the concept of the **AI Action Plane**, the architectural layer where AI systems move beyond generating information and begin performing actions on behalf of users. Securing this environment requires more than traditional security controls. It requires a governance architecture that establishes trust, defines operational boundaries, and continuously validates that trust as AI systems, enterprise environments, and business requirements evolve.

The remainder of this paper develops a governance framework organized around three fundamental questions: Who can act? What actions can they perform? How is trust continuously verified?

Although the discussion is framed around the Model Context Protocol as a representative example, these principles extend beyond any single protocol or implementation. As AI agents become increasingly autonomous and interact with more enterprise systems, organizations will need governance architectures that remain effective regardless of the underlying models, vendors, or orchestration technologies.

Rather than treating AI security as another extension of traditional application security, this paper argues that organizations should recognize the AI Action Plane as a distinct architectural domain that requires its own governance model.

---

## The Three Questions of the AI Action Plane

As AI systems shift from generating information to performing actions, organizations must rethink their approach to enterprise security. Traditional security architectures have long focused on protecting applications, data, and infrastructure. While these responsibilities remain essential, AI introduces a new challenge. Organizations must now govern how intelligent systems make decisions, invoke tools, and interact with enterprise resources on behalf of users.

Many organizations approach AI security by implementing individual technical controls such as authentication, authorization, encryption, logging, or prompt injection defenses. These controls remain necessary, but they are not sufficient on their own. Security leaders also need an architectural framework that explains how these controls work together to establish, govern, and preserve trust throughout the AI lifecycle.

This paper proposes that every AI interaction within the AI Action Plane can be evaluated by answering three fundamental questions:

| Fundamental Question | Architectural Pillar | Primary Objective |
|---|---|---|
| Who can act? | Identity-Centric Governance | Establish Trust |
| What actions can they perform? | Constrained Execution | Govern Behavior |
| How can organizations verify trust? | Continuous Assurance | Preserve Trust |

These questions intentionally build upon one another. Identity-Centric Governance establishes confidence that every AI action originates from an authenticated identity operating under explicitly delegated authority. Constrained Execution defines the operational boundaries within which AI systems may act, ensuring autonomy remains governed by organizational policy and acceptable risk levels. Continuous Assurance then validates that these assumptions remain true throughout the operational lifecycle as users, models, enterprise systems, and business requirements evolve.

Taken together, these three questions transform AI security from a collection of individual technical controls into a cohesive governance architecture. Rather than asking whether a particular protocol or implementation is secure, organizations begin asking whether they can confidently establish trust, govern AI behavior, and continuously verify that trust as AI systems become increasingly autonomous.

The remainder of this paper explores each of these questions in detail and demonstrates how they collectively provide an architectural framework for securing the AI Action Plane.

---

## The Emergence of the AI Action Plane

The first generation of enterprise generative AI largely functioned as an information platform. Organizations deployed chatbots to answer questions, summarize documents, generate reports, and assist employees with research and content creation. While these applications introduced important security considerations, particularly around data protection and prompt injection, their ability to directly affect enterprise systems remained relatively limited. In most cases, AI produced information while humans remained responsible for deciding whether and how to act upon it.

Enterprise AI is now entering a fundamentally different phase. Rather than simply generating information, AI agents are increasingly expected to retrieve data, execute workflows, invoke APIs, update enterprise applications, and coordinate activities across

multiple systems. Protocols such as the Model Context Protocol (MCP) accelerate this transition by providing a standardized mechanism for AI models to discover and interact with external tools, enterprise services, and organizational data sources. As these capabilities mature, AI systems become active participants in business operations rather than passive information assistants.

This transition fundamentally changes the organization's attack surface. Traditional enterprise applications execute workflows explicitly designed and tested by software developers. Business logic is predefined, authorization paths are well understood, and interactions between systems are generally predictable. AI agents operate differently. They evaluate context, select tools dynamically, determine sequences of actions, and generate requests that may vary significantly even when responding to similar user prompts. While this flexibility creates tremendous opportunities for productivity and innovation, it also introduces new trust relationships that traditional application security architectures were never designed to govern.

The result is the emergence of what can be viewed as the **AI Action Plane**. The AI Action Plane represents the layer within an enterprise architecture where AI systems move beyond generating information and begin performing actions on behalf of users. It encompasses every interaction in which an AI agent retrieves enterprise data, invokes a business function, executes a workflow, or communicates with another system through standardized interfaces such as MCP. Every tool exposed through an MCP server effectively becomes part of this action plane, extending the AI's ability to influence enterprise operations.

The AI Action Plane is not another network segment or technology stack. It is a conceptual layer representing every enterprise interaction in which AI systems move beyond generating information to performing actions. Thinking in terms of an action plane helps security leaders shift their focus from protecting AI models to governing AI behavior. This perspective provides a common architectural lens regardless of the specific models, frameworks, or protocols an organization adopts.

Unlike traditional integration platforms, the AI Action Plane introduces an additional layer of decision-making. Rather than following static business rules, AI systems determine which tools to invoke, when to invoke them, and how to sequence multiple actions to accomplish a user's objective. This dynamic behavior significantly complicates traditional security models. Simply authenticating the user or encrypting network communications no longer provides sufficient assurance that the resulting actions remain appropriate, authorized, or aligned with organizational policy.

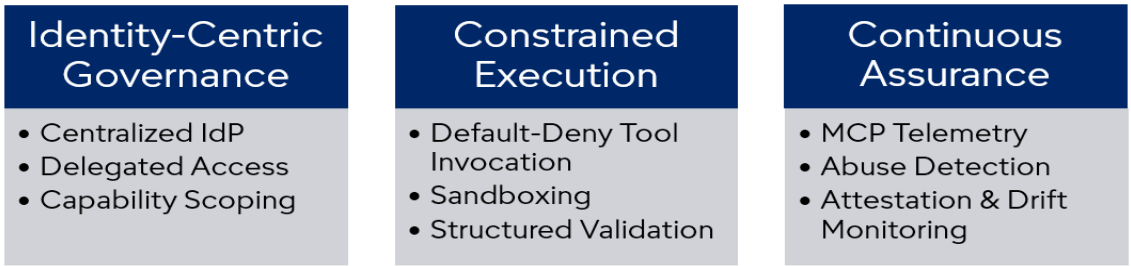
Many discussions surrounding MCP security focus on individual implementation controls such as OAuth, mutual TLS, encrypted communications, or prompt injection defenses. These controls are important and should be foundational to any secure implementation. However, focusing exclusively on individual controls risks overlooking the broader architectural challenge introduced by AI-driven action. Organizations are no longer securing a traditional application programming interface; they are **governing an autonomous decision-making system** that can interact with enterprise resources in ways previously reserved for human users or explicitly programmed software.

This distinction requires a shift in perspective. Security teams should move beyond asking whether an AI system can securely connect to an enterprise resource and begin asking whether the AI should be permitted to perform a given action in the first place. That question extends beyond authentication and authorization into governance, accountability, operational oversight, and continuous verification of trust.

For this reason, securing the AI Action Plane begins with three fundamental questions:

- **Who can act?**
- **What actions can they perform?**
- **How is trust continuously verified?**

These questions provide a simple framework for evaluating every AI interaction, regardless of the underlying model, vendor, or implementation technology. They also form the architectural foundation for the three security pillars presented throughout the remainder of this paper: **Identity-Centric Governance**, **Constrained Execution**, and **Continuous Assurance**.




**Key Insight**

Traditional application security focuses on protecting software from unauthorized access and misuse. Securing the AI Action Plane expands that responsibility to governing how AI systems are authorized to make decisions, invoke tools, and perform actions on behalf of users. Authentication, authorization, encryption, and monitoring remain essential, but they become supporting controls within a broader governance architecture rather than the architecture itself.

## Identity-Centric Governance

### *Establishing Trust Through Delegated Authority*

Every secure AI interaction begins with a single question: Who is authorized to act? Organizations authenticate users before granting access to applications, authorize those users to perform specific functions, and maintain audit records of their activities. The introduction of AI agents does not eliminate these principles. Instead, it expands them. As AI systems begin acting on behalf of users rather than simply responding to requests, identity evolves from an authentication problem into a governance challenge.

Identity-Centric Governance establishes the trust relationships that determine who, or what, is authorized to perform actions within the AI Action Plane. Rather than treating authentication as an isolated technical control, Identity-Centric Governance integrates authentication, delegated authority, authorization, and accountability into a unified governance model that spans every AI interaction. The objective is to maintain confidence that every action an AI system performs remains attributable, authorized, and consistent with organizational policy.

One of the most important principles of Identity-Centric Governance is that AI agents should never become independent identities within the enterprise. An AI assistant may recommend actions, retrieve information, or execute workflows, but it should do so on behalf of an authenticated user, service, or business process. The AI should inherit authority through explicit delegation rather than possess broad standing privileges of its own. Maintaining this distinction preserves accountability by ensuring every action can ultimately be traced back to a responsible human or organizational process.

This delegated model also aligns naturally with established enterprise identity architectures. Rather than introducing proprietary authentication mechanisms, organizations should leverage existing identity providers supporting standards such as OAuth 2.0, OpenID Connect (OIDC), SAML, and mutual TLS. These technologies already provide the mechanisms needed to authenticate users, issue short-lived access tokens, and establish trusted communications between AI clients, MCP servers, and enterprise applications. Reusing existing identity infrastructure reduces operational complexity while enabling organizations to extend established governance policies into AI-enabled environments.

Authentication alone, however, does not establish trust. Organizations must also determine the scope of authority being delegated to an AI system. This introduces an important distinction between **identity** and **capability**. Two users may possess identical identities yet authorize very different actions depending on the task they intend the AI to

perform. An AI assistant helping an employee summarize meeting notes should not automatically receive the same access as an AI agent authorized to update customer records or initiate financial transactions. Identity answers *who* is requesting an action; capability defines *what* authority accompanies that request.

For this reason, organizations should narrowly scope and time-limit delegated authority. Organizations should issue tokens that authorize only the specific tools, methods, resources, and operations required to complete the requested task. Once the task is complete, those permissions should expire automatically rather than remain available for future requests. Short-lived credentials, combined with explicit capability scoping, significantly reduce the opportunity for privilege escalation and unauthorized reuse of delegated authority.

Identity-Centric Governance should also extend beyond users to encompass every component participating in an MCP interaction. AI clients, MCP servers, enterprise applications, and supporting services should authenticate one another before performing sensitive operations. Mutual authentication helps prevent unauthorized systems from impersonating trusted components and ensures AI agents communicate only with approved enterprise resources. As organizations deploy more MCP servers, establishing trust between systems becomes as important as authenticating individual users.

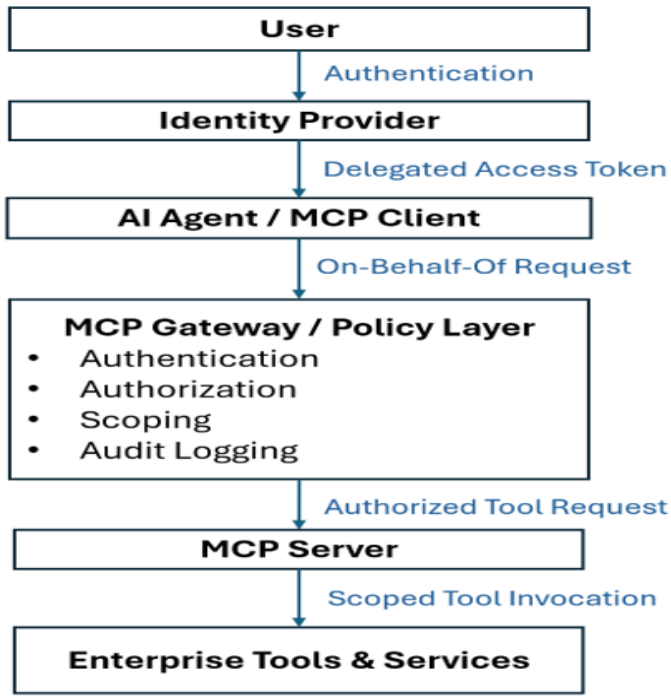
Another important consideration is the lifecycle management of delegated authority. Identity should not be viewed as a static configuration established during deployment. User roles evolve, projects conclude, applications are retired, and organizational responsibilities change. Governance processes should therefore include mechanisms to approve new AI capabilities, periodically review delegated permissions, revoke obsolete access, and document changes through centralized audit records. These governance activities help ensure that AI capabilities continue to reflect current business needs rather than historical configurations that gradually accumulate unnecessary privileges.

Perhaps the greatest benefit of Identity-Centric Governance is accountability. Every action performed through the AI Action Plane should be attributable to an authenticated identity, an approved delegation, and a specific business purpose. When organizations can answer *who requested an action, what authority was delegated, which resources were accessed, and why the action occurred*, they establish transparency that supports regulatory compliance, incident investigation, and executive confidence in enterprise AI systems.

Identity-Centric Governance therefore provides the trust foundation upon which the remaining architectural pillars are built. Once organizations establish confidence in **who** is

authorized to act, they can focus on **what** those AI systems should be permitted to do. That responsibility belongs to the second pillar: **Constrained Execution**.

***Figure 2: The MCP Trust Chain***




Identity follows the user. Authority is delegated. Every action remains attributable. architecture rather than the architecture itself.

---

## Constrained Execution

***Governing AI Behavior Within Explicit Operational Boundaries***

Trust alone is insufficient. Organizations must also define the precise boundaries within which AI systems may operate. This is where traditional authorization models begin to show their limitations. Historically, software applications execute business logic explicitly written by developers. If a user clicks a button, the application performs a predefined sequence of operations that has been designed, tested, and validated. AI agents operate differently. They dynamically select tools, determine the sequence of operations, and adapt their behavior based on context, retrieved information, and model reasoning. While this flexibility enables powerful new capabilities, it also introduces uncertainty into how enterprise resources are accessed and used.

Constrained Execution provides the architectural framework for governing AI behavior after identity has been established. Rather than assuming that an authenticated AI system should have unrestricted access to available tools, Constrained Execution limits every action to explicitly authorized capabilities that align with business purpose, organizational

policy, and acceptable risk. The objective is to ensure that every action remains predictable, explainable, and appropriate for the task being performed.

A fundamental principle of Constrained Execution is **default deny**. AI systems should never assume that every available tool is accessible simply because it exists within an MCP environment. Instead, each tool, API, workflow, or enterprise service should remain unavailable until it has been explicitly approved for a specific identity, role, or business function. This approach mirrors established security practices such as firewall rule management and zero trust architecture, where access is granted intentionally rather than inherited by default. As organizations deploy hundreds of MCP servers and thousands of available tools, default deny becomes essential for preventing unnecessary expansion of the AI Action Plane.

Capability scoping is equally important. Authorization should extend beyond determining whether an AI agent can access a system. It should define precisely **which operations** the AI may perform once connected. Consider an AI assistant interacting with a customer relationship management system. The assistant may be authorized to retrieve customer information or create a draft support case, while actions such as deleting records, modifying financial information, or changing account ownership remain prohibited. Authorization therefore occurs at the level of business capabilities rather than simply granting broad application access.

This principle becomes even more important when AI systems interact with enterprise APIs. Rather than exposing an entire service, organizations should expose only the specific methods required to accomplish approved business tasks. An MCP server should present carefully curated capabilities instead of unrestricted interfaces. For example, an AI assistant responsible for scheduling meetings may need access to calendar availability and event creation but no ability to delete existing appointments or modify organizational settings. Restricting access at the method level significantly reduces the impact of prompt injection, model error, or compromised AI behavior.

Constrained Execution should also include rigorous validation of every request before an action is performed. Traditional applications rely on predefined user interfaces that naturally limit input. AI systems generate requests dynamically, making structured validation even more important. Validate parameters against expected schemas, allowable values, business rules, and contextual constraints before requests reach enterprise applications. Organizations should treat AI-generated output with the same level of scrutiny traditionally applied to external user input, recognizing that AI systems may unintentionally generate malformed, unexpected, or unsafe requests.

Because AI systems rely heavily on natural language, organizations should also recognize that **prompts are inputs** and therefore should not be implicitly trusted. Prompt injection attacks exploit the model's ability to interpret instructions rather than vulnerabilities in traditional software. Malicious instructions embedded within retrieved documents, emails, websites, or external data sources may influence how an AI agent selects tools or sequences actions. Constrained Execution reduces the impact of these attacks by ensuring that tool invocation remains governed by explicit policy rather than solely by model reasoning. Even if an attacker successfully manipulates a prompt, the AI should remain unable to perform actions beyond its authorized capabilities.

Not every AI action carries the same level of risk. Organizations should therefore implement **risk-based execution policies** that align oversight with an action's potential business impact. Low-risk activities, such as retrieving public information or summarizing internal documents, may execute automatically with minimal oversight. Moderate-risk operations involving sensitive business data may require additional validation or contextual checks. High-risk actions, including modifying regulated data, executing financial transactions, changing security configurations, or approving privileged access, should require explicit human approval before execution. View human oversight as an adaptive governance mechanism rather than an obstacle to automation.

The runtime environment itself also plays an important role in Constrained Execution. AI agents should operate in isolated execution environments that restrict access to local file systems, operating system resources, network connectivity, and sensitive credentials unless those resources are explicitly authorized. Sandboxing, network segmentation, containerization, and resource quotas limit the impact of compromised AI behavior while reducing opportunities for lateral movement within enterprise environments. These controls become increasingly important as AI agents begin orchestrating workflows across multiple business systems.

Finally, organizations should recognize that Constrained Execution is not intended to eliminate AI autonomy. Rather, it establishes the boundaries within which autonomy can safely exist. Just as guardrails along a highway enable vehicles to travel safely at higher speeds, well-defined execution constraints allow organizations to increase AI capability without proportionally increasing operational risk. By clearly defining what actions AI systems may perform and continuously enforcing those boundaries, organizations create an environment where innovation and security reinforce one another rather than compete.

Constrained Execution therefore represents the operational control layer of the AI Action Plane. Identity-Centric Governance establishes **who** may act. Constrained Execution determines **what** those actors may do. Together they provide the trust necessary for

organizations to embrace increasingly capable AI systems while maintaining confidence that enterprise resources remain protected. The final architectural pillar, **Continuous Assurance**, extends this model by ensuring that trust is maintained long after AI systems have been deployed.

---

## Continuous Assurance

***Preserving Trust Throughout the AI Lifecycle***

Trust established today cannot be assumed tomorrow. Enterprise environments are constantly changing. New tools are deployed, user responsibilities evolve, AI models are updated, vulnerabilities are discovered, and business requirements shift. A secure AI implementation at deployment may gradually become less secure as these changes accumulate. For this reason, organizations should view trust as something that must be continuously earned, not permanently granted.

Continuous Assurance provides the architectural framework for validating that AI systems continue operating within established governance boundaries throughout their operational lifecycle. Unlike traditional monitoring, which often focuses on detecting technical failures or security events, Continuous Assurance evaluates whether AI systems continue to behave as intended, operate within approved constraints, and maintain the level of trust originally established through Identity-Centric Governance and Constrained Execution. The objective is not simply detecting attacks, but preserving confidence that AI systems remain aligned with organizational policy, business intent, and acceptable levels of risk.

Continuous Assurance is built on **comprehensive observability**. Organizations cannot govern what they cannot see. Every interaction within the AI Action Plane should generate enough telemetry to reconstruct the full sequence of events tied to an AI-driven action. This includes user identity, delegated authority, prompts, tool discovery, tool invocation, authorization decisions, returned results, policy evaluations, and execution outcomes. Capturing these events enables organizations to understand not only what actions occurred, but why those actions were performed and whether they aligned with approved business objectives.

Observability alone, however, does not establish assurance. Organizations should analyze AI behavior over time to identify deviations from expected operating patterns. Traditional security analytics focus on indicators such as failed logins, malware, or suspicious network traffic. AI systems add another behavioral dimension. Security teams should monitor which tools are invoked, how often they are used, the sequence of actions, the

types of data accessed, and whether AI agents begin demonstrating behaviors inconsistent with their intended purpose. Unexpected increases in privileged operations, unusual combinations of tool usage, or repeated attempts to access unauthorized resources may indicate prompt injection, model manipulation, compromised credentials, or configuration drift.

Configuration integrity represents another essential component of Continuous Assurance. MCP servers, AI agents, identity policies, authorization rules, and execution constraints should not be viewed as static configurations deployed once and forgotten. Organizations should continuously verify that these components remain consistent with approved baselines. Organizations should review changes to available tools, authorization policies, system prompts, retrieval sources, or supporting infrastructure through established change management processes and validate them before entering production. Configuration drift has long been recognized as a security concern within traditional IT environments; the same principle applies to AI systems whose behavior depends heavily on configuration and surrounding context.

Continuous Assurance should also encompass the operational health of the AI ecosystem itself. Identity providers, MCP servers, enterprise APIs, retrieval systems, and supporting infrastructure all contribute to the trustworthiness of AI interactions. Organizations should regularly evaluate software versions, apply security patches, rotate credentials, validate certificates, and confirm that trusted integrations continue meeting organizational security requirements. As AI ecosystems become increasingly interconnected, maintaining trust requires continuous attention to every component participating in the AI Action Plane.

Another important aspect of Continuous Assurance is validating that AI behavior remains aligned with business expectations as models evolve. Foundation models are updated, prompts are refined, retrieval sources expand, and business processes change over time. Even small modifications may alter how an AI agent selects tools or responds to user requests. Organizations should periodically reevaluate AI workflows through structured testing, red team exercises, adversarial simulations, and business validation to confirm that the AI continues producing safe, reliable, and policy-compliant outcomes. Continuous Assurance therefore extends beyond infrastructure monitoring to include ongoing validation of AI decision-making itself.

When assurance mechanisms identify abnormal behavior, organizations should have predefined response procedures that can restore trust quickly. Incident response for AI systems differs from traditional cybersecurity because the objective may not be to isolate compromised software but to suspend delegated authority, disable specific MCP servers, revoke authorization tokens, restrict available tools, or temporarily remove an AI agent from

production while investigating the underlying issue. Organizations should rehearse these actions in advance and integrate them into existing incident response plans so they can respond quickly without unnecessarily disrupting business operations.

Continuous Assurance also supports governance by providing measurable evidence that AI systems operate responsibly over time. Audit records, behavioral metrics, policy compliance reports, and periodic risk assessments provide executives, regulators, and business stakeholders with confidence that AI systems continue operating within approved governance boundaries. Rather than treating governance as a one-time approval process, Continuous Assurance transforms governance into an ongoing operational discipline supported by objective evidence and measurable outcomes.

Perhaps the most important principle of Continuous Assurance is recognizing that trust is never permanent. Every AI interaction either reinforces or weakens organizational confidence in the system. Security architectures should therefore assume that trust must be continuously evaluated as AI capabilities evolve, enterprise environments change, and new threats emerge. Organizations that embrace this mindset move beyond reactive monitoring toward proactive governance that enables innovation while preserving confidence in increasingly autonomous AI systems.

Continuous Assurance completes the architectural framework for securing the AI Action Plane. Identity-Centric Governance establishes who is trusted to act. Constrained Execution defines the boundaries within which those actions may occur. Continuous Assurance continuously validates that those assumptions remain true throughout the operational lifecycle. Together, these three pillars provide a governance architecture capable of supporting enterprise AI systems as they become increasingly autonomous, interconnected, and business critical.

---

## Putting It All Together

Security architectures are most effective when their individual components reinforce one another rather than operate independently. Identity management without authorization can establish who is requesting access but cannot determine whether the requested action is appropriate. Authorization without monitoring can restrict actions initially but cannot detect changes in behavior over time. Likewise, monitoring alone cannot compensate for weak identity controls or excessive privileges. The three pillars presented throughout this

paper are therefore intended to function as an integrated governance architecture rather than a collection of independent security controls.

Identity-Centric Governance establishes the trust foundation of the AI Action Plane. It answers the question, **who is authorized to act?** By leveraging enterprise identity providers, delegated authority, and policy-driven governance, organizations ensure that every AI action remains attributable to an authenticated identity and an approved business purpose. Identity becomes the anchor that establishes accountability throughout the lifecycle of every AI interaction.

Constrained Execution builds upon that trust by answering the second question: **what actions are authorized?** Once identity has been established, AI systems should operate only within explicitly defined operational boundaries. Rather than granting broad access to enterprise applications or APIs, organizations expose carefully scoped capabilities that align with business objectives and acceptable risk. Least privilege, structured validation, runtime isolation, and human oversight collectively ensure AI autonomy stays within well-defined limits rather than granting unrestricted access.

Continuous Assurance completes the governance architecture by answering the final question: **can we continue trusting the system to operate within those boundaries?** AI systems, enterprise environments, and business processes evolve continuously. Identity relationships change, tools are added, models are updated, and new threats emerge. Continuous Assurance provides the visibility, behavioral analysis, operational governance, and incident response capabilities necessary to validate that AI systems continue operating safely long after deployment. Trust therefore becomes a continuously evaluated property rather than a one-time approval decision.

When combined, these three pillars create a governance model that extends well beyond the Model Context Protocol itself. The framework applies equally to AI assistants, autonomous agents, retrieval-augmented generation (RAG) systems, orchestration platforms, and future AI technologies that have yet to emerge. While implementation details may evolve, every enterprise AI system capable of performing actions will ultimately require mechanisms to establish trusted identities, constrain authorized behavior, and continuously validate operational trust.

Perhaps the framework's greatest strength is its ability to support innovation rather than restrict it. Organizations often struggle to balance security with the desire to adopt emerging AI capabilities quickly. Security teams naturally focus on reducing risk, while business leaders seek to maximize the value AI can provide. By separating governance into three complementary architectural pillars, organizations gain a practical way to enable AI

adoption while maintaining confidence that appropriate controls remain in place. Security becomes an enabler of responsible innovation rather than a barrier to progress.

This perspective also shifts the conversation from technology to governance. Rather than asking whether an organization has implemented OAuth, encrypted communications, prompt injection defenses, or centralized logging, leaders can instead ask three far more meaningful questions:

- **Can we confidently identify who is authorized to perform AI-driven actions?**
- **Have we explicitly defined and enforced the actions those AI systems are permitted to perform?**
- **Do we continuously verify that AI systems remain trustworthy as they evolve?**

These questions provide a common language for security architects, developers, governance teams, executive leadership, and auditors. They encourage organizations to evaluate AI systems from a governance perspective rather than viewing security as a checklist of disconnected technical controls.

As AI systems become increasingly autonomous, this architectural approach becomes even more important. Future AI agents will interact with dozens or even hundreds of enterprise systems simultaneously, coordinate activities across business processes, and make increasingly complex operational decisions. Governing these environments through isolated security controls will become progressively more difficult. Organizations will instead require governance architectures that establish trust, define operational boundaries, and continuously validate AI behavior regardless of the underlying implementation technology.

The AI Action Plane offers one way to view this emerging landscape. Identity-Centric Governance, Constrained Execution, and Continuous Assurance provide the architectural principles for securing it. Together they form a practical framework that organizations can apply today while remaining flexible enough to support the increasingly autonomous AI systems of tomorrow.

---

## Governance Considerations

### *Operationalizing the AI Action Plane*

Designing a secure architecture is only one component of responsible AI adoption. Organizations must also establish governance processes that ensure security controls

remain effective as AI capabilities evolve. Identity-Centric Governance, Constrained Execution, and Continuous Assurance provide the architectural foundation for securing the AI Action Plane, but governance provides the organizational discipline that transforms those architectural principles into repeatable operational practices.

One of the first governance responsibilities is establishing clear ownership for AI systems. Enterprise AI deployments often span multiple organizational boundaries, including business units, software engineering teams, infrastructure groups, data scientists, security organizations, legal counsel, privacy offices, and compliance functions. Without clear ownership, important decisions about AI capabilities, acceptable risk, and operational accountability can become fragmented. Every AI system should therefore have an identified business owner responsible for its intended use, supported by technical owners responsible for implementation, operational security, and lifecycle management.

Organizations should also integrate AI governance into existing architecture review processes rather than treating AI as a separate technology initiative. AI systems should undergo the same level of architectural scrutiny as other enterprise platforms, with additional consideration given to model selection, external integrations, delegated authority, data access, tool exposure, and operational autonomy. Security architecture review boards provide an appropriate forum for evaluating whether proposed AI capabilities align with organizational risk tolerance and established governance policies before deployment.

As organizations adopt MCP, AI agents, and other orchestration technologies, **tool governance** becomes equally important. Every tool exposed through an MCP server effectively expands the AI Action Plane by introducing additional capabilities the AI may invoke. Governance should therefore include formal processes to approve new tools, document their intended business purpose, identify required permissions, classify operational risk, and periodically review whether those capabilities remain appropriate. Treating AI tools as governed enterprise assets rather than simple software integrations helps prevent uncontrolled expansion of AI authority over time.

Vendor governance also becomes increasingly important as organizations rely on third-party AI services. Many AI solutions now combine foundation models, orchestration frameworks, retrieval systems, MCP servers, and specialized plugins developed by multiple vendors. Each component introduces its own security assumptions, operational dependencies, and supply chain risks. Organizations should evaluate AI vendors with the same rigor they apply to other critical technology providers, assessing security architecture, identity integration, data handling practices, logging capabilities, incident response processes, software update mechanisms, and compliance with organizational

security requirements. Vendor assessments should extend beyond the AI model itself to encompass the broader ecosystem supporting enterprise AI operations.

Governance should also incorporate structured AI risk assessments throughout the system lifecycle. Traditional application risk assessments often focus on software vulnerabilities, infrastructure exposure, and compliance requirements. AI systems introduce additional considerations, including prompt injection, indirect prompt manipulation, excessive tool authority, retrieval quality, model behavior, hallucinations affecting business decisions, and unintended automation of sensitive activities. Organizations should evaluate these risks before deployment and revisit them periodically as models, business processes, and threat landscapes evolve. Frameworks such as the NIST AI Risk Management Framework (AI RMF) provide valuable guidance for integrating these considerations into broader enterprise risk management programs.

Policy development represents another critical governance responsibility. Organizations should establish policies defining acceptable AI use, delegated authority, data classification requirements, approved AI platforms, model selection criteria, logging expectations, human oversight requirements, and incident response procedures. Rather than creating entirely new governance structures, many organizations can extend existing cybersecurity, data governance, and technology risk management policies to incorporate AI-specific considerations. This approach reduces operational complexity while ensuring AI governance remains aligned with broader organizational objectives.

Perhaps one of the most overlooked aspects of AI governance is education. Security architects, software developers, business leaders, and end users all interact with AI systems differently, yet each group influences the organization's overall security posture. Developers need guidance on secure AI implementation patterns and prompt handling. Security teams need to understand AI-specific attack techniques such as prompt injection, model manipulation, and excessive agency. Business leaders must understand both the opportunities and governance responsibilities associated with increasingly autonomous AI systems. End users should recognize that AI recommendations, while often valuable, should not replace professional judgment or organizational policy. Building AI literacy across the organization strengthens every layer of the governance architecture.

Governance should also establish meaningful performance metrics. Organizations often measure AI success using productivity improvements, response quality, or operational efficiency. Security governance requires additional metrics that evaluate trustworthiness and operational maturity. Examples include the percentage of AI systems reviewed through architecture governance, the number of approved versus unauthorized MCP tools, prompt injection detection rates, privileged tool usage, policy compliance, AI vendor assessment

completion, and incident response readiness. These measures provide objective evidence that governance processes are functioning as intended while identifying opportunities for continuous improvement.

Finally, organizations should recognize that AI governance is not a one-time implementation project. Governance should evolve alongside the organization's AI capabilities. As new foundation models emerge, additional tools become available, and AI agents assume greater operational responsibility, governance processes should be reassessed to ensure they continue supporting innovation while maintaining acceptable levels of organizational risk. Periodic reviews of governance structures, architectural standards, delegated authority, and operational metrics help ensure that the AI Action Plane remains aligned with changing business objectives and emerging security challenges.

Effective governance ultimately determines whether enterprise AI becomes a trusted business capability or an unmanaged source of operational risk. Organizations that combine sound architectural principles with disciplined governance processes can adopt increasingly autonomous AI systems while maintaining executive confidence, regulatory compliance, and organizational trust.

---

## Conclusion

Artificial intelligence is rapidly evolving from a technology that informs human decision-making to one that increasingly participates in business operations. Through standards such as the Model Context Protocol, AI systems can retrieve enterprise data, invoke APIs, execute workflows, and coordinate activities across multiple applications. This evolution promises significant improvements in productivity, efficiency, and innovation, but it also introduces a new category of security challenges. Organizations are no longer securing systems that merely process information. They are securing systems that can perform actions that directly affect business operations.

Traditional cybersecurity principles remain essential, but they are no longer sufficient on their own. Authentication, authorization, encryption, logging, and least privilege remain foundational controls, but they must now operate within a broader governance architecture designed specifically for autonomous and semi-autonomous AI systems. The AI Action Plane represents this new operational reality, requiring organizations to move beyond protecting applications toward governing how AI systems interact with enterprise resources on behalf of users.

This paper has presented a practical architectural framework for securing the AI Action Plane through three complementary pillars. **Identity-Centric Governance** establishes trust by ensuring every AI action is performed on behalf of an authenticated identity with explicitly delegated authority. **Constrained Execution** governs AI behavior by limiting every action to approved capabilities that align with organizational policy and acceptable risk. **Continuous Assurance** preserves trust over time through ongoing validation, operational oversight, behavioral monitoring, and governance processes that adapt as AI systems and enterprise environments evolve.

While this paper has focused on the Model Context Protocol as an example of the emerging AI ecosystem, the framework extends well beyond any single protocol or technology. New orchestration frameworks, AI agent platforms, and integration standards will undoubtedly emerge over the coming years. Regardless of the underlying implementation, organizations will continue facing the same fundamental governance questions: **Who is authorized to act? What actions are permitted? Can the organization continue trusting the AI system as it evolves?** By grounding security decisions in these questions rather than in specific technologies, organizations can develop governance architectures that remain effective even as the AI landscape changes.

Perhaps the most significant shift enterprise AI introduces is that trust becomes an operational capability rather than a static security control. In traditional applications, trust is often established during authentication and maintained through predefined software logic. AI systems require a more dynamic approach. Trust must be established through identity, constrained through explicit authorization, and continuously validated throughout the operational lifecycle. Organizations that recognize trust as an ongoing process rather than a one-time event will be better positioned to safely adopt increasingly autonomous AI capabilities.

The emergence of the AI Action Plane represents one of the most significant architectural shifts since organizations first connected enterprise applications to the Internet. Just as the growth of web applications required new approaches to identity management, application security, and network architecture, the rise of AI agents demands new approaches to governance, authorization, and operational assurance. Organizations that begin building these capabilities today will be better prepared to embrace the next generation of enterprise AI while maintaining the security, accountability, and trust that modern business operations require.

Ultimately, enterprise AI success will not be determined solely by the sophistication of its models or the number of tasks it can automate. It will depend on whether organizations

can confidently answer three deceptively simple questions every time an AI system performs an action:

- **Who is authorized to act?**
- **What actions are they permitted to perform?**
- **How do we know that trust remains justified over time?**

The future of enterprise AI will not be secured by stronger models alone. Organizations that establish trusted identities, constrain AI actions, and continuously validate trust will be best positioned to realize the promise of increasingly autonomous AI while protecting the enterprise resources those systems are entrusted to use.

## Five Principles for Securing the AI Action Plane

1. **Treat AI as an actor, not just an application.**
   Govern AI systems that can invoke tools and execute workflows as participants in business processes rather than traditional software components.
2. **Delegate authority rather than granting standing privilege.**
   AI should always act on behalf of authenticated users or approved business processes using narrowly scoped, time-limited permissions.
3. **Constrain AI actions through explicit policy.**
   Intentionally expose and govern tools, methods, and enterprise resources based on business purpose and acceptable risk.
4. **Continuously validate trust.**
   Continuously evaluate trust through monitoring, behavioral analysis, governance, and operational oversight.
5. **Build governance that outlasts today's technology.**
   Secure AI architectures should be based on enduring governance principles rather than any single protocol, model, or vendor implementation.