FIOR

Authentication for the Intelligent Age

Securing MCP-Based Agentic AI Systems: A Runtime Trust and Governance Architecture for Enterprise AI

Response to NSA "Model Context Protocol (MCP): Security Design Considerations for AI-Driven Automation"

Author: FIOR Group
Date published: May 2026
Version: 1.4

Executive summary

The emergence of Model Context Protocol (MCP)-enabled AI systems represents a major architectural transition in enterprise computing. AI agents are increasingly able to autonomously invoke tools, access enterprise systems, chain workflows, interact with other agents, and execute long-running delegated actions with minimal human oversight.

Recent guidance[1] from the U.S. National Security Agency (NSA) highlights that these systems introduce fundamentally new cybersecurity risks that are not adequately addressed by traditional identity, API security, IAM, or network-centric controls.

The core challenge is that autonomous AI agents behave differently from conventional software workloads:

- they self-initiate actions,
- dynamically alter execution paths,
- invoke external tools,
- operate asynchronously,
- and make independent runtime decisions.

This creates a new requirement for continuous runtime trust, cryptographic identity, orchestration governance, and persistent verification across the entire agent lifecycle.

The FIOR AI Gateway architecture addresses these requirements by establishing a cryptographically anchored trust layer for agentic systems, enabling enterprises and governments to deploy MCP-based AI safely at scale.

[1] https://www.nsa.gov/Portals/75/documents/Cybersecurity/CSI_MCP_SECURITY.pdf?ver=bmgiSbNQLP6Z_GiWtRt6bg%3d%3d

The Security Challenge of MCP-Based Systems

MCP architectures accelerate interoperability between AI models, tools, enterprise applications and autonomous agents. However, they also create a substantially expanded attack surface.

Unlike conventional APIs or applications, agentic systems can:

- autonomously invoke privileged functions,
- chain actions across multiple systems,
- consume untrusted context,
- delegate tasks to secondary agents,
- and execute persistent workflows over extended periods.

This creates several critical security concerns:

Emerging Risk Area	Security Impact
Tool poisoning	Compromised tools manipulate agent decisions
Prompt/context injection	Agents execute malicious instructions
Unauthorized tool invocation	Agents exceed intended authority
Multi-agent chaining	Trust breaks across orchestration paths
Excessive delegated privilege	Autonomous privilege escalation
Supply-chain compromise	Malicious agents or connectors introduced
Lack of runtime identity	Inability to continuously verify agents
Session persistence abuse	Long-running agent compromise
Insufficient provenance	No trusted record of actions or origins
Configuration drift	Runtime behaviour diverges from policy

Traditional IAM and API gateways were designed primarily for:

- human authentication,
- static applications,
- and deterministic service interactions.

They were not designed to continuously evaluate autonomous machine behaviour in real time.

A Runtime Trust Architecture for Autonomous AI

The FIOR AI Gateway introduces a dedicated runtime trust and governance layer for agentic systems.

This architecture establishes:

- persistent cryptographic identity,
- continuous trust verification,
- policy-driven orchestration governance,
- and authenticated agent-to-agent communication.

The objective is not simply to authenticate access at session initiation, but to maintain trust continuously throughout autonomous execution.

Core architectural capabilities include:

1. Persistent Agent Identity

Every AI agent, tool, orchestration component and service is assigned a unique cryptographically verifiable identity.

This enables:

- continuous authentication,
- secure delegation,

- trusted orchestration,
- and verifiable provenance.

Unlike conventional workload identity, the identity persists across:

- asynchronous workflows,
- delegated execution,
- multi-agent interactions,
- and long-running autonomous tasks.

2. Runtime Attestation

All participating entities can be continuously validated during execution.

The architecture verifies:

- software integrity,
- orchestration state,
- policy alignment,
- execution provenance,
- and trust posture.

This enables rapid identification of:

- compromised agents,
- modified orchestration chains,
- unauthorized tools,
- or anomalous behaviour.

3. Trusted Tool Invocation

MCP ecosystems rely heavily on dynamic tool usage.

The gateway architecture establishes:

- authenticated tool registries,
- cryptographically signed tool identities,
- policy-controlled invocation rights,
- and runtime verification before execution.

This reduces exposure to:

- tool poisoning,
- unauthorized tool substitution,
- and malicious runtime redirection.

4. Delegated Authority Governance

Autonomous systems increasingly perform actions on behalf of users, organizations and other agents.

The architecture enforces:

- least-privilege delegation,
- scope-limited authority,
- runtime revocation,
- and policy-based behavioural constraints.

This prevents:

- uncontrolled privilege propagation,
- unrestricted orchestration chaining,
- and unauthorized lateral movement.

5. Continuous Trust Evaluation

Trust is evaluated dynamically throughout execution rather than assumed after initial authentication.

The system continuously analyses:

- behaviour,
- orchestration relationships,
- policy compliance,
- trust scoring,
- and execution context.

This supports adaptive governance for:

- autonomous agents,
- dynamic workflows,
- and evolving runtime conditions.

Alignment with Emerging Security Guidance

The architecture aligns closely with emerging guidance from:

- NSA cybersecurity advisories,
- NIST AI RMF principles,
- OWASP guidance for agentic AI,
- Zero Trust architecture models,
- and evolving AI governance frameworks.

Key alignment areas include:

Security Objective	Architectural Response
Continuous verification	Runtime identity validation
Least privilege	Delegated authority control
Supply-chain security	Signed provenance and attestation

Auditability	Immutable telemetry and action traceability
Zero Trust enforcement	Persistent runtime trust evaluation
Autonomous governance	Policy-driven orchestration control
Multi-agent security	Authenticated agent-to-agent trust
Tool security	Trusted invocation and validation

Enterprise and Sovereign Deployment Benefits
The architecture is particularly relevant for:

- government AI systems,
- critical infrastructure,
- telecoms,
- defence,
- financial services,
- healthcare,
- and sovereign cloud environments.

Key benefits include:

Reduced AI Operational Risk

Continuous runtime governance reduces the likelihood of autonomous compromise, policy drift and unauthorized behaviour.

Stronger Regulatory Alignment

FIOR AI Gateway Supports evolving requirements for:

- AI accountability,
- operational resilience,
- software provenance,
- and secure autonomous systems.

Secure AI Adoption at Scale

FIOR AI Gateway enables organizations to safely operationalize:

- autonomous agents,
- AI orchestration platforms,
- and MCP-enabled ecosystems.

Improved Audit and Governance

FIOR provides traceability across:

- agent decisions,
- tool invocation,
- orchestration paths,
- and delegated authority chains.

Conclusion

MCP-based systems represent a foundational shift in enterprise computing architecture. As AI agents become increasingly autonomous, identity and trust can no longer be treated as static authentication events.

Autonomous systems require:

- persistent identity,
- continuous verification,
- runtime governance,
- cryptographic provenance,
- and policy-driven orchestration control.

The AI Gateway architecture establishes these capabilities as a dedicated trust layer for agentic AI systems, enabling enterprises and governments to deploy autonomous AI securely, transparently and at scale.

A Secure MCP Reference Architecture

We outline a reference security architecture for enterprises deploying MCP-enabled autonomous AI systems. The architecture is designed to address emerging security requirements associated with AI agents, orchestration frameworks, delegated autonomy, runtime trust validation and secure tool invocation. *cont..*

Architecture Overview

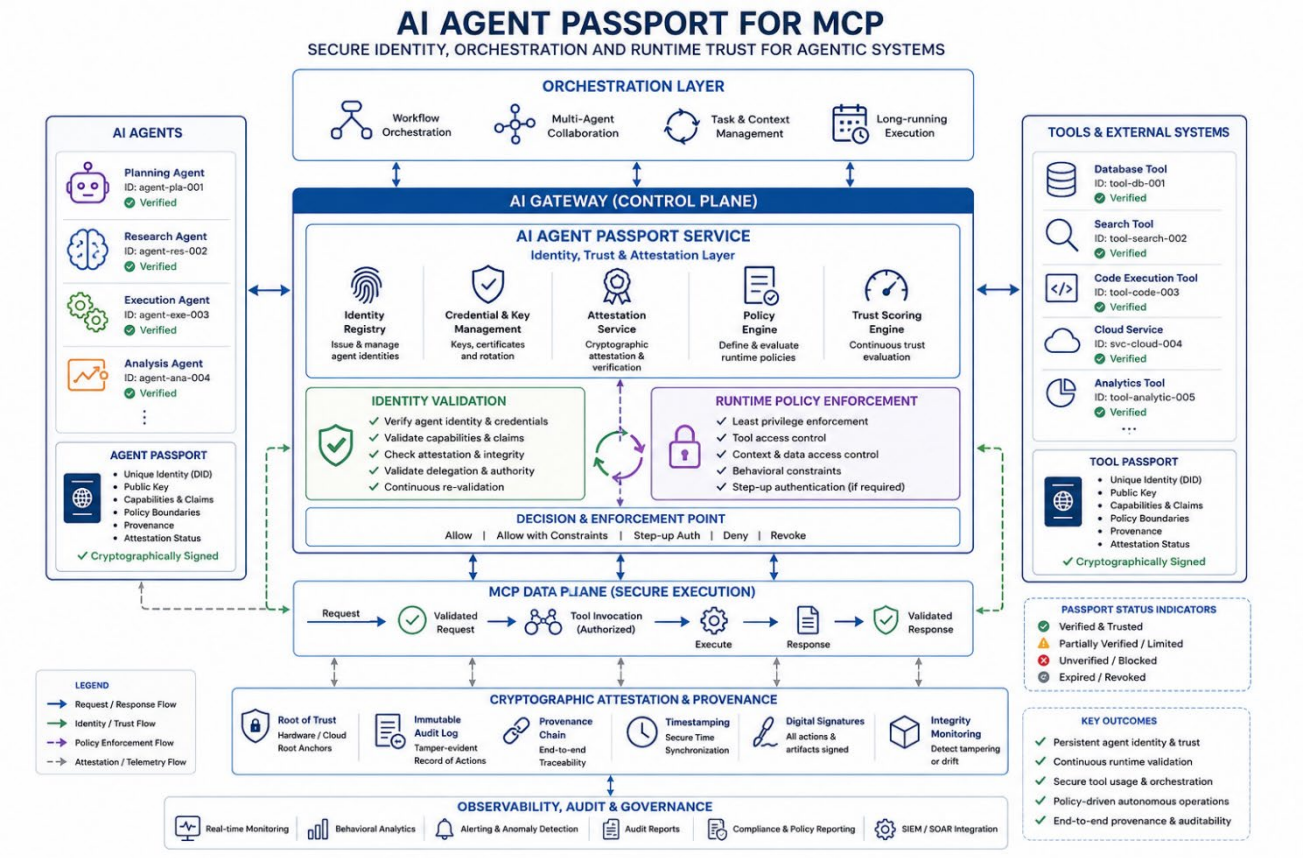

Figure 1: Secure MCP Reference Architecture

Core Architectural Principles

Persistent Runtime Identity: Every AI agent and orchestration component maintains a cryptographically verifiable identity throughout runtime execution.

Continuous Trust Evaluation: Trust posture is continuously reassessed based on behavior, delegation patterns, policy compliance and operational context.

Secure Tool Invocation: All tools, APIs and services are validated through trusted provenance and attestation controls before execution.

Least Privilege Governance: Autonomous systems operate within dynamically constrained permissions aligned to task-specific requirements.

Delegated Authority Control: Multi-agent orchestration chains are governed to prevent uncontrolled privilege propagation or unauthorized actions.

Immutable Telemetry: All orchestration activity, policy decisions and runtime events are logged through tamper-resistant audit systems.

Functional Architecture Layers

Enterprise Interaction Layer: Interfaces with enterprise users, applications, APIs and operational environments.

AI Gateway Runtime Trust Layer: Provides runtime identity, policy enforcement, orchestration governance and continuous trust validation.

MCP Orchestration Layer: Coordinates autonomous AI agents, workflow execution and delegated tool usage.

Tool and Service Integration Layer: Connects trusted enterprise systems, cloud services, SaaS platforms and operational technologies.

Security Services Layer: Delivers cryptographic attestation, threat analytics, policy enforcement and trusted tool management.

Governance and Telemetry Layer: Provides auditability, compliance monitoring, forensic telemetry and operational governance visibility.

Security Objectives

- Prevent unauthorized AI agent behavior.
- Control autonomous orchestration and delegation chains.
- Secure tool invocation and external service access.
- Provide continuous runtime governance for AI systems.
- Support sovereign cloud and critical infrastructure environments.
- Enable regulatory auditability and operational assurance.
- Support future post-quantum cryptographic migration requirements.

Strategic Relevance

The emergence of MCP-enabled AI systems fundamentally changes enterprise security requirements by introducing autonomous software entities capable of self-directed operations. Traditional IAM and API-centric security models are insufficient to govern these environments independently. The FIOR AI Gateway architecture therefore provides a foundational runtime trust layer supporting persistent identity, orchestration governance, cryptographic provenance and continuous behavioral validation.