TrustAI Solution Brief

TRUSTLOGIX

Securing Agentic AI in the MCP Era

Problem Statement

Enterprises face a conflict between **AI-driven innovation and data security.**

Use Case: In one case, a large global manufacturing enterprise building critical components for high-growth industries rapidly adopted AI agents across engineering, operations, and development workflows.

Business teams deployed agents to automate tasks such as issue management, CI/CD pipelines, and data access across systems like GitHub, Jira, and internal platforms. While this accelerated productivity, it introduced new challenges for security teams.

Agents operated across multiple systems using existing credentials, often without clear ownership, fine-grained access controls, or real-time governance. This created gaps in visibility, increased risk of over-permissioned access, and exposed sensitive data across both cloud and on-prem environments.

Business leaders are accelerating adoption of agentic AI to improve productivity and maintain competitive advantage. **Security teams** are responsible for protecting sensitive data, enforcing compliance, and preventing uncontrolled access. Meanwhile, data teams are evolving their established access and control frameworks to manage a landscape where agents provide a broader range of users with direct, real-time interaction with organizational information.

Thousands of AI agents are being developed, accessing data across multiple systems, including sensitive and regulated sources. This growth is outpacing visibility, policy enforcement, and control.

AI agents operating through Model Context Protocols (MCPs) interact with enterprise systems in ways traditional identity and security models were not designed to govern. Recent incidents, including Claude code leak and Vercel's AI-related data leaks and unauthorized system access events, highlight these risks.

Dilemma:
How to move fast with agentic AI without compromising control over enterprise data?

Gartner predicts that by 2028, more than 50% of enterprises will adopt AI security platforms to secure third-party AI services and protect custom-built AI applications.

Securing AI requires a new approach that operates at machine speed and is purpose-built for agentic, MCP-driven environments.

Major Pain Points

This challenge is driven by three gaps:

1. Agent Sprawl and BYOA (Bring Your Own Agents)

- Growth of internal and third-party agents across teams
- Independent adoption of Claude, GPT, and custom agents
- No centralized visibility, ownership, or governance
- Agents accessing enterprise data without consistent controls
- Hard-coded enforcement slows detection and remediation
- Limited auditability, monitoring, and real-time oversight causing large potential compliance and regulatory risks
- Agents automating CI/CD workflows (issue management, repo access, pipelines) without centralized governance

2. MCP Proliferation with Weak Access Governance

- Increase in MCP servers across disparate teams and third-party vendors.
- Ungoverned access paths creating shadow integrations into enterprise systems.
- Expanded attack surface across cloud and on-prem
- **No standardized, fine-grained authorization across MCPs; each MCP enforces different levels of access control**
- **Inconsistent, non-standardized authorization:** OAuth provides only coarse-grained access, leaving fine-grained control entirely dependent on how each vendor designs their MCP tools:
  - *Granular Tooling (e.g., GitHub, Atlassian):* Expose highly specific tools (e.g., read-only vs. write), allowing gateways to enforce strict, action-level policies.
  - *Coarse Tooling (e.g., Snowflake):* Expose broad actions (e.g., SYSTEM_EXECUTE_SQL). If a gateway allows this tool, the agent's blast radius relies entirely on the underlying database permissions being perfectly configured, magnifying the risk of over-privileged accounts.

3. Legacy Identity Models Are Insufficient

- **Static IAM vs. Dynamic Agents:** IAM and OAuth were built for deterministic "Point A to Point B" interactions. They cannot govern the non-linear, multi-step reasoning paths agents take.
- **Lack of Intent-Based Control:** Traditional systems verify *who* is asking, but not *why*. There is no mechanism to validate "inferred intent" or ensure the agent's actions align with the original user request.
- **No Real-Time Policy Enforcement:** Legacy identity cannot intercept a mid-thought "reasoning" step that violates security policy before the data is accessed.
- **Identity is a Perimeter, Not a Guardrail:** Knowing an agent is "authorized" is no longer enough; we must be able to authorize the *logic* of the request in real-time.

How Enterprises Address This Today?

1. Security and Identity Systems (IAM, OAuth, IGA, SIEM)

Usage:

- IAM / OAuth authenticate agents and grant access
- IGA manages identity lifecycle and access reviews
- SIEM aggregates logs and monitors activity

Limitations:

- Authentication and post-facto monitoring, **not inline enforcement**
- Static, role-based access without agent context
- No real-time, agent-level authorization
- Limited cross-system visibility

2. Data and Analytics Platforms (Snowflake, Databricks, Power BI, SQL Server, Oracle)

Usage:

- RBAC, masking, row/column-level security
- Audit logs track data access

Limitations:

- Controls are siloed per platform
- No consistent model for AI access to on-prem data
- No centralized policy layer
- Limited cross-system visibility
- No enforcement across multi-step workflows

3. Business and Developer Tools (Atlassian, GitHub, GitLab)

Usage:

- Access via user or service account permissions
- Exposed to agents via MCPs
- Agents used to automate CI/CD workflows

Limitations:

- Coarse access controls
- MCPs create ungoverned access paths
- No fine-grained control over agent actions
- Limited auditability of agent activity
- Access controls are not scoped to agent identity, leading to over-permissioned CI/CD automation

The Gap

Existing identity and data systems remain necessary for authentication, governance, and monitoring.

They lack:

- Real-time, inline enforcement
- Context-aware, purpose-driven authorization
- Unified governance across MCP interactions and multi-system workflows

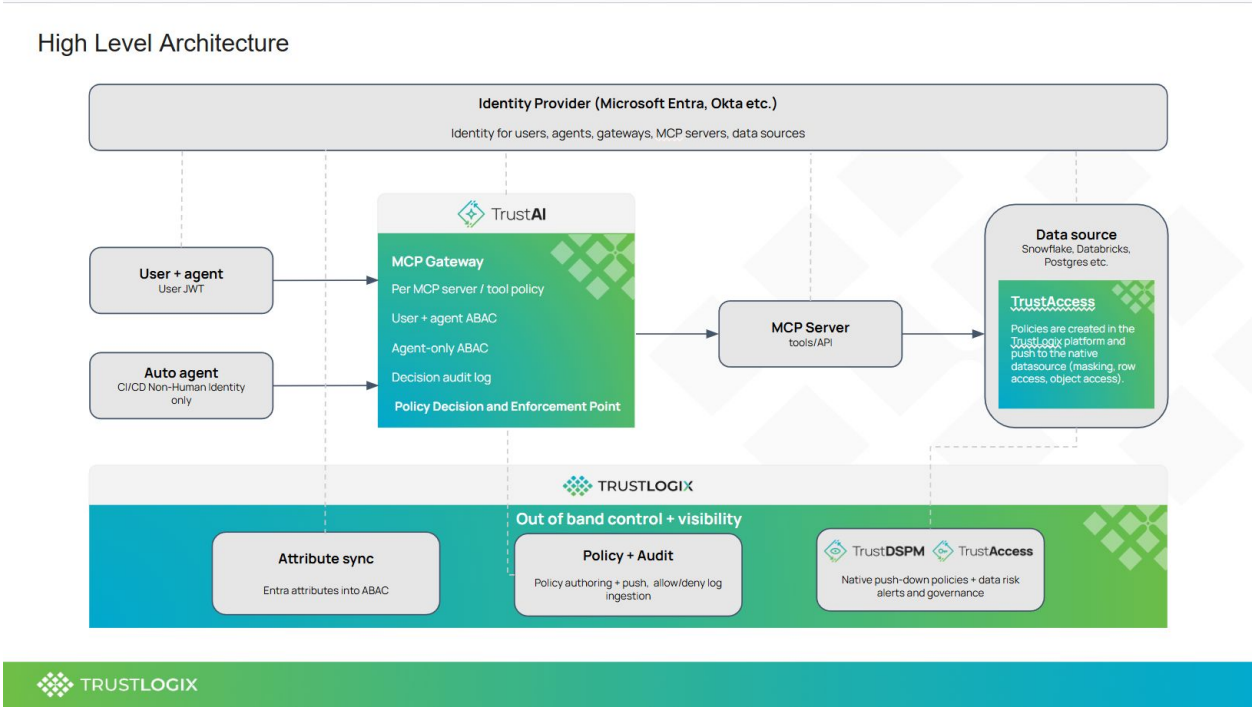
As a result, enterprises face visibility gaps, inconsistent policy enforcement, and increased risk as agent adoption scales.

Solution

TrustLogix TrustAI provides a centralized policy enforcement layer that governs how AI agents and MCPs access enterprise systems. It sits between agents, MCPs, and enterprise data, ensuring every request is controlled before execution.

TrustAI acts as a centralized policy enforcement plane that brings together the objectives of Business, Security, and Data teams. Built as a native extension of the TrustLogix platform, it provides a unified control point to govern AI agent interactions with sensitive data while directly honoring and extending existing enterprise entitlements.

By evaluating the intent and context of every MCP request in real-time, TrustAI enforces fine-grained, data-layer controls, including row-level filtering, dynamic masking, and column-level security across Snowflake, Databricks, and other core systems. This ensures that agents only "see" the data they are entitled to, while providing Data teams with the DSPM-driven visibility and risk governance needed to scale AI without compromising the underlying data architecture.



TrustAI Control Plane

TrustAI is a control plane for agents with two core components:

1. TrustAI MCP Gateway (Access Governance Layer for AI)

MCP Gateway

- Centralized PEP intercepts calls to data, APIs, RAG indexes, and MCP tools
- Enforces policy before data leaves the system

Policy Decision Engine

- Define and enforce access policies across systems

Real-Time Authorization

- Allow or deny every request before data access

Agent & MCP Governance

- Control all MCP interactions through a single layer

Unified Visibility

- Track user → agent → action → data

2. Guardian Agent (Natural Language Risk Detection and Investigation)

- Natural language risk detection and instant remediation
- Natural language introspection of agent activity
- Investigation of agent activity through a unified interface

Coverage Across Systems

TrustLogix secures and governs data across

- Data platforms (Snowflake, Databricks)
- Analytical tools (Power BI)
- Developer tools (Git, Atlassian)
- CI/CD workflows (GitHub Actions, pipelines, automated issue management)
- APIs and services

On-Prem Security

Direct MCP connections to on-prem systems introduce risk:

- Legacy systems lack modern controls
- Sensitive data exposure
- Access bypasses security layers

TrustLogix enforces control before access:

- No direct MCP → on-prem access
- Policy validation on every request
- Consistent enforcement across hybrid environments

Why TrustLogix

- The missing control layer for agentic AI access
- Fine-grained data-level access based on compliance
- Enforces policy before every MCP request
- Governs all agent and MCP interactions centrally
- Prevents direct agent-to-data access without policy enforcement
- Enforces access control before any MCP-driven data access
- Natural language risk detection and instant remediation
- Active data and agent monitoring
- Secure CI/CD automation by enforcing agent-level access in development workflows
- Authentication and post-facto monitoring, **not inline enforcement**

How TrustLogix Solves It

Three phases of AI governance and security maturity

Phase 01: Register — Agent Identity & Registration

- Agent discovery and registration
- Lifecycle governance (onboarding, ownership, changes, decommission)
- Authentication and credentials (OAuth, token management, service accounts)
- Identity posture management (detect orphaned credentials, over-privileged roles)
- ISO 42001: A.3 Internal Organization, A.9 Responsible Use

Phase 02: Authorize — Policy-Based Access Control

- TrustAI Gateway intercepts MCP, API, data, and tool calls
- Policy Decision Point for real-time evaluation
- Context-aware authorization (identity, data sensitivity, purpose, location, runtime signals)
- MCP and tool security (whitelisting, rate limiting, masking, prevent unauthorized connections)
- Data-layer controls (row, column, masking, purpose-based access)
- ISO 42001: A.6 AI Lifecycle, A.7 Data for AI Systems

Phase 03: Govern — AI Risk Management

- Monitor agent data access, tool calls, and MCP activity
- Right-size access based on behavior
- Kill-switch for real-time policy revocation (pause, quarantine, terminate access)
- Audit evidence (logs, impact assessments, control evidence)
- ISO 42001: A.5 Impact Assessment, A.8 Monitoring

The Outcome

- Eliminate uncontrolled MCP access
- Enforce least-privilege agent access
- Apply consistent policies across systems
- Secure data access across cloud and on-prem