TrustAI Case Study

TRUSTLOGIX

Securing Autonomous AI Agents in a Major Financial Services Firm with TrustAI

Executive Summary

A leading financial services institution set out to scale agentic AI across its internal operations—without compromising regulatory compliance, data security, or identity governance. By partnering with **TrustLogix**, the firm operationalized a secure, governed AI agent ecosystem that enabled its AI engineers to rapidly deploy dozens of **purpose-driven micro-agents** for automation, analytics, and decision support.

With **TrustAI**, the organization empowered teams to innovate safely—ensuring every AI agent operated with **least-privilege access**, enforced **purpose-based data controls**, and maintained **end-to-end identity propagation** across complex, multi-cloud data environments. The result: accelerated AI adoption with provable compliance, real-time visibility, and centralized control.

However, prior to TrustLogix, the firm faced significant barriers to scaling agentic AI—rooted in a fundamental lack of visibility, governance, and control.

The Challenges:

1. The "Agent Blind Spot"

The firm lacked visibility and control over autonomous AI agents, creating critical barriers to scaling agentic AI initiatives safely.

- **Unbounded agent permissions:** AI agents and service accounts inherited overly broad privileges, increasing the risk of unauthorized data exposure and creating new insider-threat vectors invisible to traditional monitoring tools.
- **Regulatory ambiguity:** The firm could not demonstrate how sensitive financial and personal data was accessed or processed by autonomous agents—creating audit and compliance risk (SOC2, SOX, data residency).
- **Non-human identity sprawl:** Rapid growth of bots, agents, and automation workflows introduced unmanaged non-human identities outside traditional IAM governance models.

- **Non-human identity sprawl:** Rapid growth of bots, agents, and automation workflows introduced unmanaged non-human identities outside traditional IAM governance models.

Without scalable governance, agentic AI projects risked being paused or blocked entirely by security and legal teams. Existing tools could not transition from broad database-level filtering to precise table and row-level access, nor could they map agent behavior to business context or compliance obligations.

2. A Fragmented and Opaque Data Landscape

The firm's data is siloed across a highly varied landscape, creating a massive governance challenge where the security team does not know how effectively permissions are being enforced.

- **On-Premises Structured Sources:** Legacy databases including Oracle, SQL Server, and MySQL.
- **Cloud Data Platforms:** Modern high-scale environments like Snowflake, and data lakes like AWS S3
- **Unstructured Data sources:** Unstructured data residing in file systems such as OneDrive.

3. Balancing Innovation with Zero-Trust Security

Opening up these varied systems is vital for the firm's success in implementing AI Agents, yet it creates new threat vectors. The security team required a way to ensure that agents—which operate at machine speed—only interact with sensitive data under proper permissions.

TrustAI addresses this by enforcing Zero Trust principles, ensuring that AI agents inherit only the specific, scoped entitlements of the human requester, preventing them from accessing "Business sensitive" objects set up for different purposes.

The Solution: TrustAI Agent-Aware Governance

TrustAI serves as the centralized authorization conduit within the firm's existing AI stack, integrating with **LiteLLM** and **LLama Index**.

- **Native Policy Enforcement:** TrustAI establishes accountability across the varied data landscape by correlating the chain: **User** → **Agent** → **Query** → **Data**. It applies granular row-level filtering and masking natively at the source using the **TrustLogix MCP Server**.
- **TrustAI Copilot (AI-Powered Security Assistant):** Moving beyond static dashboards, the Copilot provides:
  - **Interactive Investigations:** Analysts use natural language prompts to identify specific data an agent accessed on behalf of a user.
  - **Root-Cause Analysis:** Complex telemetry is transformed into human-understandable explanations for security violations.
  - **Automated Policy Recommendations:** Copilot suggests remediation steps, such as creating new masking policies or revoking over-privileged roles based on observed risks

Technical Integration Architecture

The integration follows a structured request-response flow to maintain the **User** → **Agent** → **Query** → **Data** accountability chain:

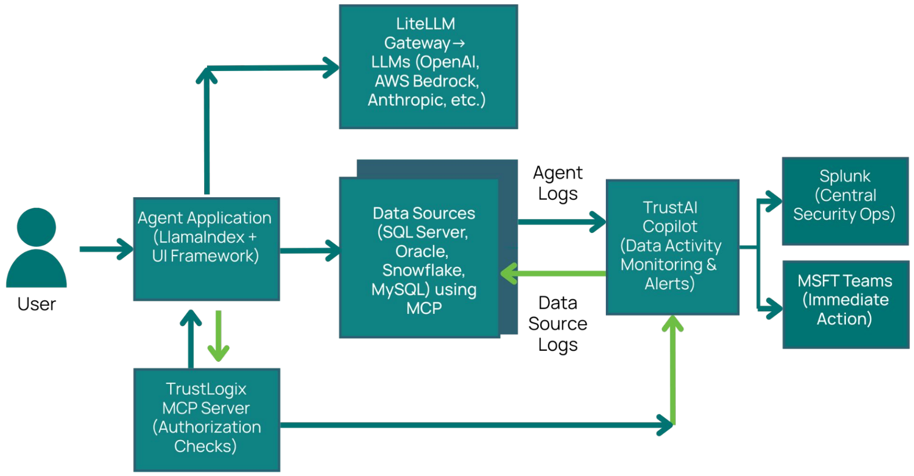
- **LiteLLM Gateway :** All AI interactions are routed through a centralized LiteLLM gateway. This gateway handles user identity headers, ensuring that every request is tied to a specific human initiator rather than a broad service account.
- **Llama Index Orchestration:** For agents built with the Llama Index framework, TrustAI provides scoped, just-in-time entitlements. This orchestration ensures that when an agent performs Retrieval-Augmented Generation (RAG), it only accesses data authorized for that specific requester.
- **Model Context Protocol (MCP) Server:** The TrustAI MCP Server acts as the core intelligence layer. Agents request authorization decisions from TrustAI MCP server to confirm authorization before an LLM can execute a query against data sources like Snowflake, SQL Server or file systems like Onedrive.
- **Policy Decision Engine:** The MCP server evaluates user identity and intent against defined Attribute-Based Access Control (ABAC) policies. It returns an Allow/Deny decision to the agent in real-time.

Monitoring and Data Protection Flow

TrustAI overlays these integrations with continuous behavioral monitoring to prevent "Agent Blind Spots":

- **Behavioral Baselining:** The system learns "usual" patterns for each agent, such as SQL command types and typical query volumes.
- **Native Policy Enforcement:** Access rules utilize a hybrid metadata framework combining native database definitions with **TrustLogix Data Classification**. This allows for granular row-level filtering and masking to be applied natively at the source.
- **Telemetery and Alerting:** TrustAI ingests agent activity logs and correlates them with data source logs. Deviations—such as an agent attempting unauthorized joins—stream alerts to **Microsoft Teams** or **Central SIEM (Splunk)** for investigation.

Secure AI Agent with TrustAI Monitoring


LiteLLM Gateway→ LLMs (OpenAI, AWS Bedrock, Anthropic, etc.)
User
Agent Application (LlamaIndex + UI Framework)
Data Sources (SQL Server, Oracle, Snowflake, MySQL) using MCP
Agent Logs
Data Source Logs
TrustAI Copilot (Data Activity Monitoring & Alerts)
Splunk (Central Security Ops)
MSFT Teams (Immediate Action)
TrustLogix MCP Server (Authorization Checks)

Business Outcomes: Trusted AI at Scale

By leveraging TrustAI for policy enforcement and data activity monitoring, the firm achieved significant measurable results:

- **Accelerated Development:** The use of the **TrustAI Copilot** and no-code policy frameworks allowed AI engineers to build and deploy dozens of automated mini-agents with immediate security approval.
- **Operational Intelligence:** The system established a behavioral "brain" that flags deviations from learned patterns, allowing the firm to fix misconfigured agents before data exposure occurred.
- **Compliance at Speed:** Analysts can now investigate risks via natural language (e.g., *"Which agent queries violated firms data confidentiality policies?"*), streaming alerts directly to **Microsoft Teams** and **Splunk** for rapid response.
- **Zero Performance Friction:** The **proxyless, agentless architecture** ensures no impact on inference speed for time-sensitive financial workflows.

Visit trustlogix.ai/accelerate-ai to learn more and request a demo