# Autonomy with Guardrails: Operating Agentic Automation in High-Risk Production Systems

Conf42 SRE 2026

**Ajay Athitya Ramanathan**

Data & AI Engineer

FourthSquare

# Presentation Overview

- **1. What is Happening ?** - The Rise of Agentic Automation
- **2. Why it's Risky ?** - Autonomy in High-Risk Systems
- **3. Why Old Approaches Break ?** - The Limits of Traditional Automation
- **4. How We Think About Solving It** - Principles of Guardrails Autonomy
- **5. What We Actually Build** - Core Guardrails for Safe Agentic Systems

# 1. What is Happening ? - The Rise of Agentic Automation

Artificial intelligence is moving from **assistive systems to autonomous systems**.

**Early AI systems:**

- Answered questions
- Summarized data
- Generated text

**Modern agentic systems can:**

- Plan multi-step tasks
- Call APIs
- Operate Workflows
- Coordinate across systems

Platforms LangGraph/LangChain, Microsoft Agent Framework, CrewAI, allow agents to Execute **Plan → Act → Observe →Iterate Loops**, enabling autonomous workflow execution.

**Key shift:**

AI Assistants → AI Operators

# 2. Why it's Risky ? - Autonomy in High-Risk Systems

When AI systems gain the ability to act, risk increases dramatically.

**Examples of high-risk environments:**

- financial operations
- healthcare systems
- infrastructure management
- enterprise automation
- customer data platforms

**Major risks include:**

- hallucinated actions
- prompt injection attacks
- unauthorized data access
- cascading automation failures
- compliance violations

**The engineering dilemma:**

AI = probabilistic reasoning
Production systems = deterministic reliability

Operating autonomous AI safely requires **control systems around non-deterministic models**.

# 3. Why Old Approaches Break ? - The Limits of Traditional Automation

Traditional automation systems assume:

- deterministic logic
- predictable workflows
- static decision trees

Examples include ETL pipelines, batch jobs, and rule engines. Agentic systems differ fundamentally:

| Traditional Automation | Agentic Automation |
|---|---|
| Fixed workflow | Dynamic planning |
| Deterministic logic | Probabilistic reasoning |
| Predefined actions | Tool discovery |
| Static scripts | Adaptive agents |

Existing automation infrastructure was not designed for **software that reasons about what to do next**.

# 4. How We Think About Solving It - Principles of Guardrails Autonomy

The solution is not removing autonomy but **bounding it with safety mechanisms**.

The design philosophy:

Maximum reasoning freedom - Strict execution boundaries

Agents can think freely, but **every action must pass through safety controls** before execution.

Guardrails transform free-form model outputs into **controlled and verifiable operations**. A typical enterprise agent architecture looks like this:

# 5. What We Actually Build - Core Guardrails for Safe Agentic Systems

Autonomy in high-risk production systems isn't just about letting agents act—it's about **building strong guardrails that enforce safety, compliance, and reliability**. In this section, we focus on the **practical building blocks** of a safe agentic system.

### Programmable Safety

- Prompt Injection
- Structured Output
- Programmable Validations
- LLM-as-a-Judge

### Identity & RBAC

- Least Privilege
- Managed Identities
- Session-based Tokens
- Time-bound Access

### Privacy & Safety

- Handling PII
- Redaction
- Masking
- Local LLM
- Synthetic Data

### Human-in-the-loop

- Require Approvals
- Allow Lists
- Irreversible Actions
- Action-level Logs

### Observability & Evals

- Log Calls
- Active Safety Scoring
- Passive Evals
- Unified Audit Trails

# Thank you