MCP Overview

Overview of an MCP

A reusable layer that sit's on top of a systepms API tooling that allows multiple agents to interact the system

- Gives non-deterministic agents a way to make secure deterministic API callout.

- Enforces system security by limiting API exposure through a whitelisted subset of tools.

- Re-usable across agents and can even be built to connect to multiple system instances i.e Salesforce MCP

- Provides scalability and re-usability of agent to external system interactions outside of individual agent builds

Agents vs. MCPs

Agent

Interprets the user prompt and intent

Chooses which MCP server and tool to call

Determines what information or action it needs

Supplies structured parameters (within its best understanding)

Reasons over the MCP response to decide next steps

MCP

Does not reason about user intent

Does not choose tools on the agent's behalf

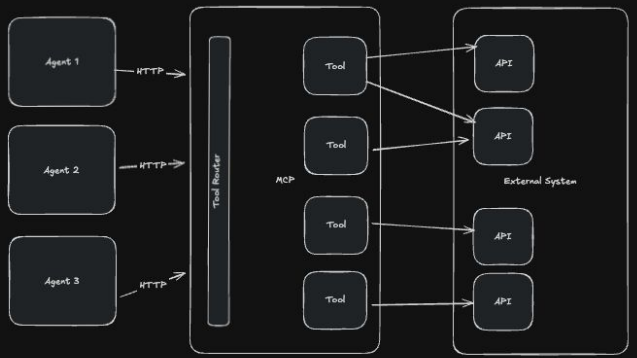
Translates the agent's request into:

System-specific APIs

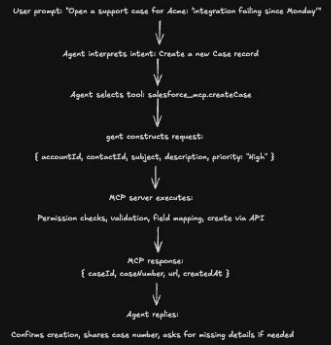
Authenticated, permission-checked calls

Enforces guardrails, policies, and rate limits

Normalizes responses for safe agent consumption



Example



Overview of an MCP

A reusable layer that sits on top of a systems API tooling that allows multiple agents to interact the system

- Gives non-deterministic agents a way to make secure deterministic API callout.

- Enforces system security by limiting API exposure through a whitelisted subset of tools.

- Re-usable across agents and can even be built to connect to multiple system instances i.e Salesforce MCP

- Provides scalability and re-usability of agent to external system interactions outside of individual agent builds

Agents vs. MCPs

Agent

Interprets the user prompt and intent

Chooses which MCP server and tool to call

Determines what information or action it needs

Supplies structured parameters (within its best understanding)

Reasons over the MCP response to decide next steps

MCP

Does not reason about user intent

Does not choose tools on the agent's behalf

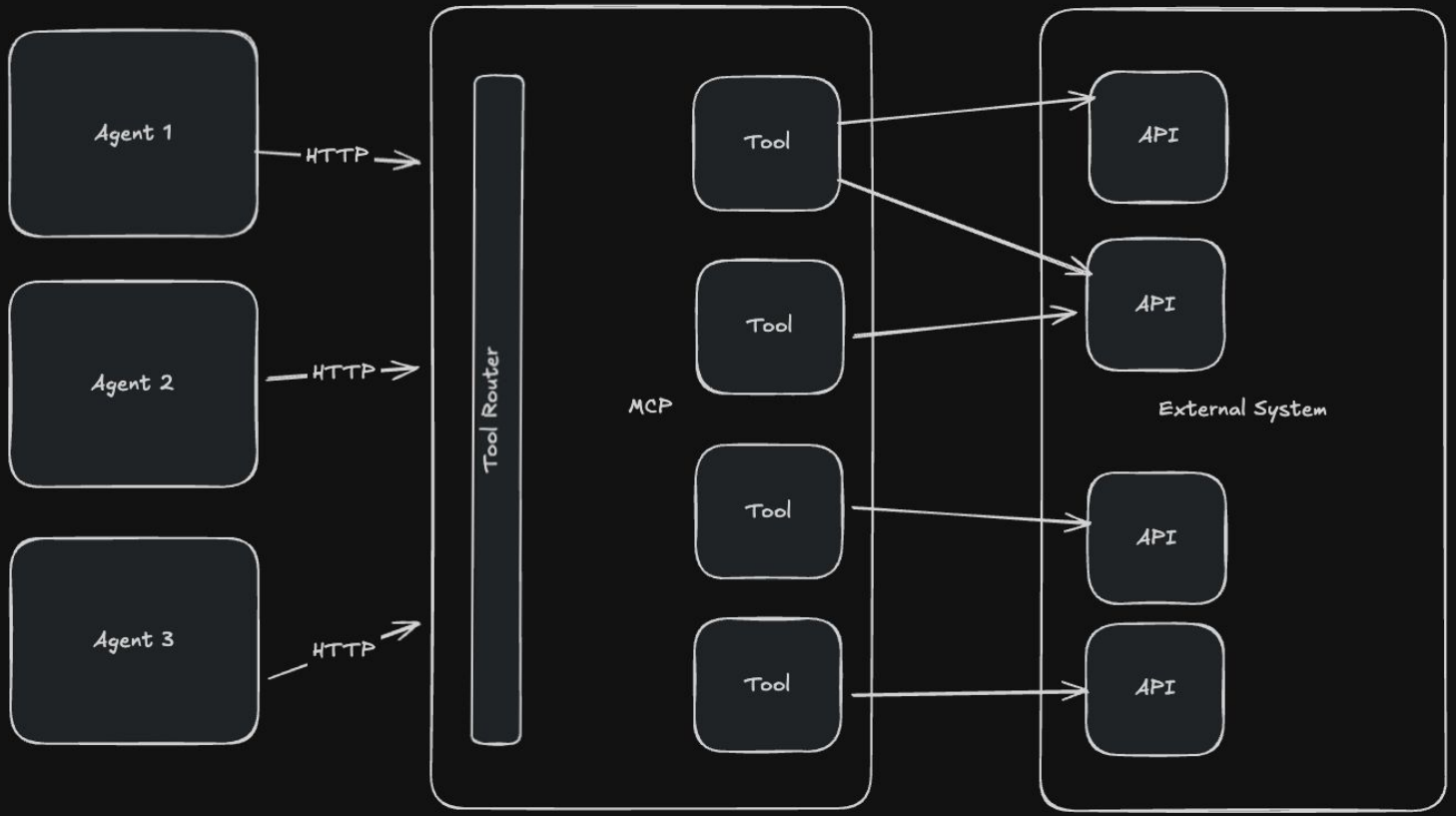
Translates the agent's request into:

System-specific APIs

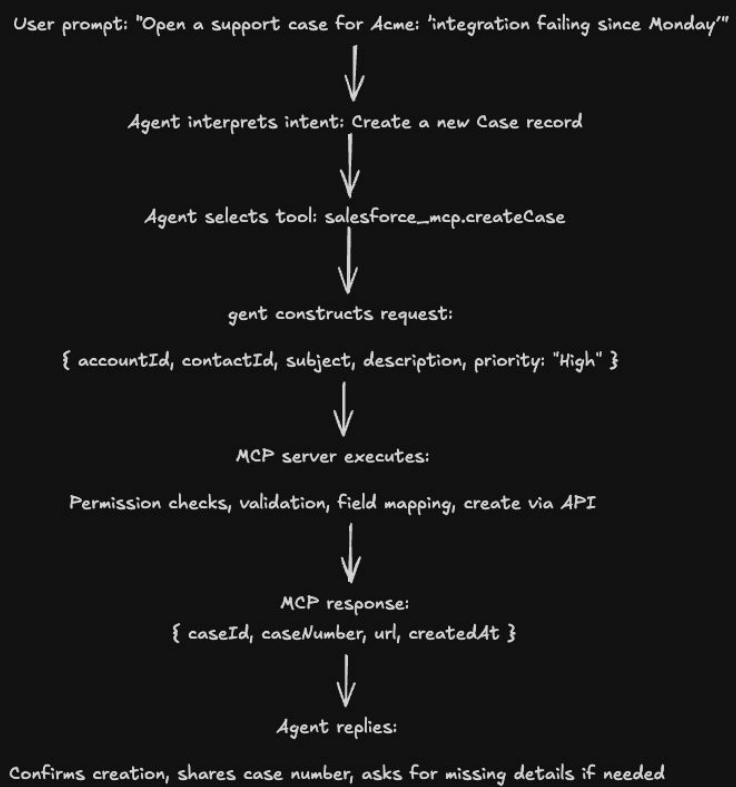
Authenticated, permission-checked calls

Enforces guardrails, policies, and rate limits

Normalizes responses for safe agent consumption


Agent 1
Agent 2
Agent 3
HTTP
HTTP
HTTP
Tool Router
MCP
Tool
Tool
Tool
Tool
API
API
API
API
External System

Example


User prompt: "Open a support case for Acme: 'integration failing since Monday'"
Agent interprets intent: Create a new Case record
Agent selects tool: salesforce_mcp.createCase
gent constructs request:
{ accountId, contactId, subject, description, priority: "High" }
MCP server executes:
Permission checks, validation, field mapping, create via API
MCP response:
{ caseId, caseNumber, url, createdAt }
Agent replies:
Confirms creation, shares case number, asks for missing details if needed