RUNTIME POLICY DECISION ENGINE

Control thousands of AI agents with one set of policies

Build AI agent policies in Obsidian. The Policy Decision Engine applies them consistently at runtime across coding agents, agent platforms, MCP connections, and the gateways you already run.

01

One set of policies

Each enforcement point gets its decision from the same policies.

02

Decisions with context

Policies know who the agent acts for and how you label your data.

03

A record of each decision

Show the board which agent actions your policies stopped.

THE PROBLEM

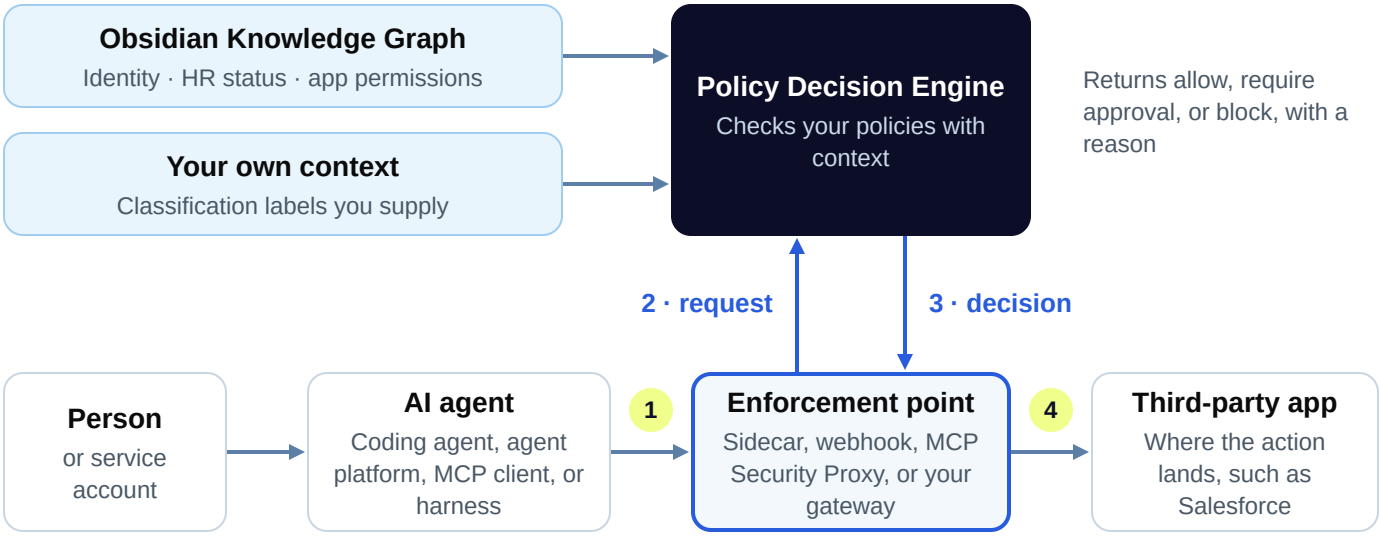
AI agent policies are missing or disjointed, and the gaps grow with every new agent

Each rule is rebuilt in each coding agent, agent platform, MCP connection, and gateway, and each copy behaves differently.

- **No runtime check.** Some agent tools have nowhere to apply the rule.
- **Another route open.** Blocked in one tool, the data is reachable through another.
- **Identity unmapped.** The tool can't see who the agent acts for.
- **All-or-nothing.** Gateways allow or deny the agent, not the action.

HOW IT WORKS

The decision happens at the enforcement point, before the action reaches the app



1. The agent calls a tool with its ID, invoker, and arguments.

2. The enforcement point pauses the action and asks.

3. The engine checks your policies with context and returns a reason.

4. The action proceeds, waits for approval, or stops, and is logged.

ENFORCEMENT POINTS

Apply your policies where each team's agents run

Endpoint sidecar

Checks each Claude Code, Cowork, or Codex tool call before it runs.

Platform webhook

Copilot Studio asks the engine before an agent acts.

MCP Security Proxy

Checks each MCP tool call before it reaches the app.

Policy Decision API

Your gateway asks for a decision and enforces it, even for agents without an Obsidian connector.

CONTEXT IN ACTION

A contractor’s agent tries to drop a critical table. The engine blocks it on the spot.

THE POLICY **Contractors’ agents can’t drop critical tables**

1	Nancy, a contractor, asks Claude Code to tidy up Snowflake.	Claude Code
2	The agent tries to drop a table your team labeled critical.	DROP TABLE
3	Obsidian checks the call and sees what Snowflake can’t.	Contractor, from HR · Critical, your label
4	The table stays, and Nancy sees the policy author’s message.	Blocked

IN THE PRODUCT

Write policies as precise as your business rules

Each policy sets which agents it covers, what triggers it, the context it checks, and what happens on a match.

Policy	App	Outcome
Block access to unapproved MCP servers Prevents agents from accessing MCP servers that have not been r...	Microsoft Copilot Stu...	Block
HR Terminated Users Block terminated employees from accessing downstream MCP ser...	MCP Security	Block
Block drop table requests for contractors	Claude code	Block
Block local sensitive file access	Claude code	Block
Email Block Outbound emails to non-corporate accounts is disabled for AI age...	Claude code	Require approval
Sensitive Data in Payload (ARR) Blocked by company policy: outbound payload contains customer/...	Claude code	Block

Talk to us about taking control of your agents

Request a Demo

Scan to book a demo