The MCP tab

Model Context Protocol servers are the tools your AI assistant can call — file access, issue trackers, databases, browsers. They are also the most direct route from a prompt to something happening in the real world, which is why they get their own governance.

Local MCP governance

The card at the top shows how many local servers are governed. **Govern local MCP servers** puts them behind the Sentilai shim.

Before it edits anything it **backs up the configuration file** and it reverts exactly that file if you stop governing. Claude Code must be restarted afterwards.

Stop governing restores the backup — unless your organization has locked this down.

The server list

Every MCP server found on this machine, with its name, its project path, and how it is started. Servers configured in Cursor are listed separately, because Cursor keeps its own configuration.

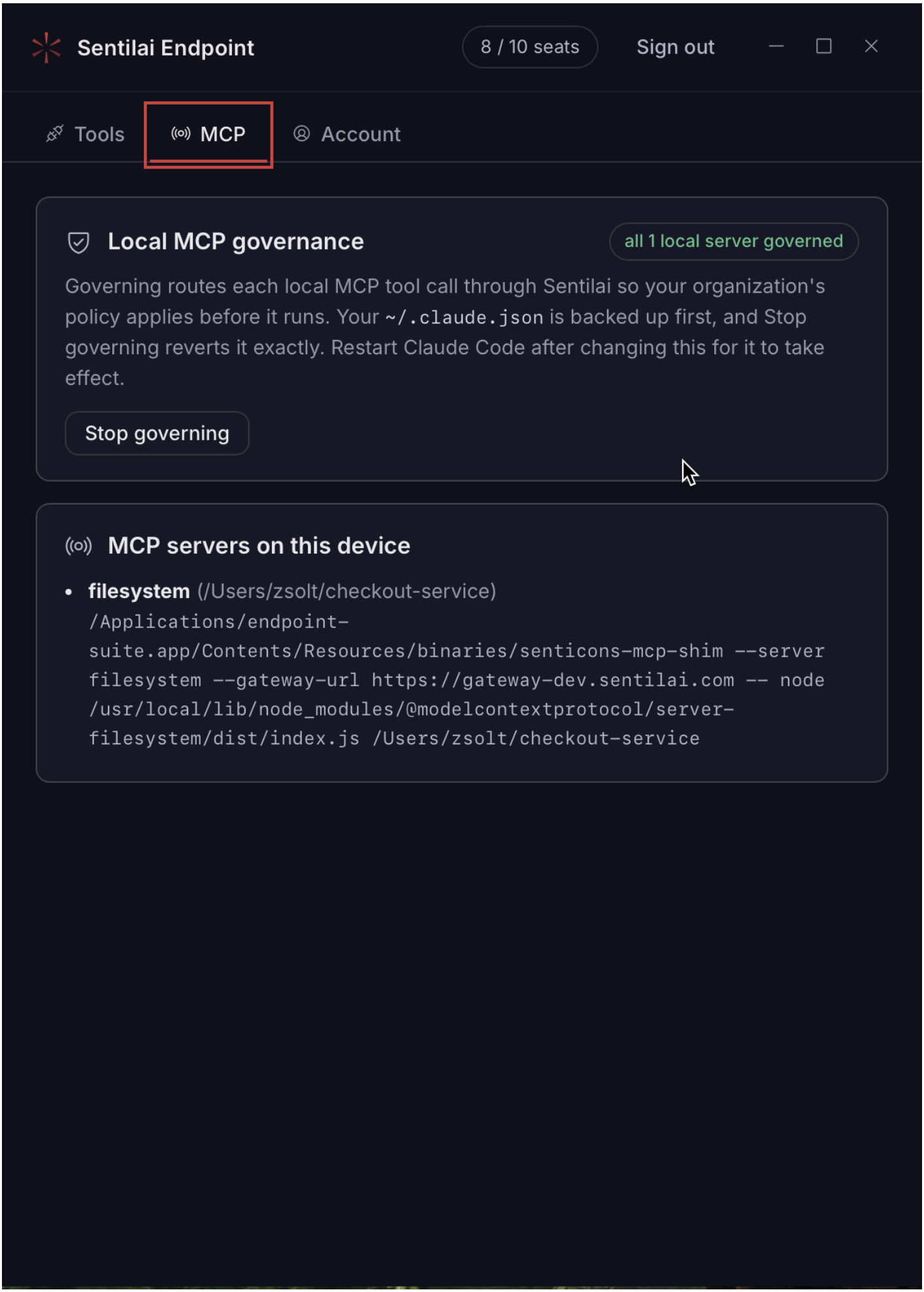
"Scanning…" means the sweep is still running; "None found" means this machine has no local MCP servers configured.

What the shim actually does

Once governed, an AI tool calling an MCP server goes tool ? Sentilai shim ? real server. The shim only inspects **tool calls**; everything else passes through untouched.

For each call it asks the Gateway for a decision and caches the answer for as long as your admin's **Local MCP decision cache** setting allows. When a call is blocked, the agent gets a clean error naming the server and tool, so the assistant can tell the developer what happened instead of failing mysteriously.

The shim also watches the server's advertised tool list, which is what makes it possible to notice when a tool's description or schema changes underneath you.



The MCP tab — local MCP governance, and every MCP server this device has been seen using.

Revision #8
Created 2026-08-02 10:19:45 UTC by Sentilai Docs
Updated 2026-08-06 05:49:00 UTC by Sentilai Docs