

MCP SERVER

NO CODE

CLOUD HOSTED

Google Home MCP

Control your entire house, from climate to security.

Google Home MCP connects your AI agent directly to all your Nest smart devices, giving you hands-free control over climate, security, and monitoring. Use it to set temperatures, view live camera feeds from doorbells and cameras, or check device status across multiple properties.

A+ Quality Score 100/100

smart-home

device-control

automation

thermostat-management

camera-monitoring



The infrastructure that powers AI agents in the real world.



Vinkius connects AI to the world's software through secure, enterprise-grade infrastructure — enabling real-world execution at scale, built on the Model Context Protocol (MCP).

Your AI Connections Run Through Vinkius Cloud

The world's largest
managed MCP catalog

Vinkius is the cloud infrastructure where AI agents connect to the software your business already runs. We handle the hosting, the security, the credentials, the uptime — you get agents that actually do things.

We operate the world's largest managed MCP catalog. Major SaaS platforms, CRMs, databases, and cloud providers — running, monitored, production-ready. This MCP server is hosted and maintained by the Vinkius Cloud for AI Agents.

The agent doesn't manage credentials, doesn't manage uptime, doesn't manage security. Vinkius does.

— Architecture principle

Four Pillars of the Vinkius Runtime

01 — Security by design

Credentials stay encrypted at rest via AES-256. The AI agent never touches raw keys — they're injected into a sandboxed V8 isolate at runtime. Actions are logged, and connections have an emergency kill switch.

03 — Deterministic observability

Eight immutable metrics per endpoint: request volume, p95 latency, error rate, active connections, cost attribution. A live payload feed logs every tool call with mutation detection.

02 — Built on MCP Fusion

This MCP server was built with **MCP Fusion**, the open-source framework (Apache 2.0) that powers the entire Vinkius catalog. Schema-as-firewall strips undeclared fields, compiled PII redaction runs at zero overhead, and cryptographic lockfiles produce git-diffable audit trails.

04 — Autonomous operations

Servers are deployed, monitored, and patched autonomously. New capabilities and security patches ship weekly. Zero-downtime deployments ensure continuous availability across all managed MCP servers.

AES-256

Encryption at rest

Ed25519

PKI vault signatures

24h TTL

Ephemeral session keys

V8 Isolate

Sandboxed execution

One Token. Instant Access.

Every MCP server on Vinkius is accessed through a **Connection Token**. Tokens are generated in the cloud dashboard and produce a unique MCP endpoint URL. Paste this URL into any MCP-compatible client — no SDK required.

A single token can serve **multiple AI clients simultaneously**, or you can issue separate tokens per client for granular access control. Each token tracks its own request count, last activity timestamp, and can be individually enabled or revoked.

MCP ENDPOINT

`https://edge.vinkius.com/{token}/mcp`

Claude



Cursor



VS Code



Windsurf



Grok



Gemini

Security Is the Architecture

Security in Vinkius is not a feature — it's the foundation of the runtime. The gateway enforces multiple independent protection layers between AI agents and third-party APIs.

01 — Ed25519 PKI Vault

Every workspace has an Ed25519 Master Key. Session keys are generated ephemerally (24h TTL) and signed by the Master Key. Credentials never leave the vault boundary.

02 — V8 Isolate Sandboxing

Tool code runs inside isolated-vm V8 isolates with 64 MB memory caps and per-request timeouts. No filesystem access, no network access except through the SSRF-guarded fetch bridge.

03 — SSRF Guard

All outbound HTTP requests are DNS-resolved and validated before execution. Private IP ranges (10.x, 172.16-31.x, 192.168.x, AWS metadata 169.254.x) are blocked at the network layer.

05 — Cryptographic Audit Trail

Every request is signed into a SHA-256 hash chain with Ed25519 signatures. Events form a tamper-proof, SIEM-exportable forensic record.

04 — DLP & PII Redaction

A ResponseGuard pipeline intercepts every tool response. Configurable redaction patterns strip sensitive fields (emails, SSNs, card numbers) before data reaches the AI agent.

06 — Honeypot Trap System

Phantom credentials are injected into isolated environments. If a honeypot is used outside Vinkius infrastructure, the server is quarantined instantly.

Emergency Kill Switch

EU AI Act Art. 14(1)
Compliant

The kill switch is an **emergency halt** mechanism — not a simple toggle. When triggered, it executes three actions atomically:

01 — Server deactivated

The MCP server is immediately taken offline across the entire cluster.

02 — All tokens revoked

Every connection token is invalidated. Total lockout — reconnection blocked until new tokens are issued.

03 — WebSocket connections killed

Active connections terminated via Redis pubsub broadcast. Propagates to every runtime node in the cluster.

Full Visibility. Zero Guesswork.

The Vinkius cloud dashboard includes a full MCP Governance suite — real-time analytics and security controls for production AI operations.

Control Plane

KPI dashboard with request volume, latency, success rate, token consumption, and AI-generated operational briefings.

FinOps

Cost tracking per tool, payload compression savings, budget optimization signals, and consumption trends.

Firewall & DLP

PII redaction activity, sensitive data protection counters, and security event timeline.

Agent Activity

Which AI clients are connecting, how often, and what they're doing — real-time session tracking.

Tool Health

Slowest and most error-prone tools, with actionable root-cause insights and performance baselines.

Incident Log

Error trends, failure rates, status-code breakdowns, and forensic audit trail access.

Get started at cloud.vinkius.com — connect your AI agent in under 60 seconds.

Google Home MCP

12 tools available

Cloud-hosted on Vinkius

This connector turns your smart home into a fully controllable system through natural conversation. Instead of opening separate apps for the thermostat, checking video feeds, and managing room data, your AI agent handles it all. You can ask it to adjust the heat in the living room or check if the front door camera sees movement. Need to know what devices you have? The MCP lists every Nest device, including cameras, thermostats, and doorbells across all structures. If you're building complex home automation routines, connecting through Vinkius gives your agent access to a centralized catalog of capabilities, making it easy for your AI client to manage everything from setting the optimal cooling temperature to streaming live video feeds on demand.

Core Capabilities

01 — Manage device locations

List and view all structures, rooms, and connected Nest devices in your home.

02 — Adjust climate settings

Change the operating mode (heat/cool/off) or set specific temperatures for any linked thermostat.

03 — Run live camera streams

Generate temporary streaming URLs so your agent can display real-time feeds from cameras and doorbells.

04 — Control auxiliary features

Set fan timers or place the thermostat in energy-saving eco mode manually.

One Click on Vinkius — From Prompt to Execution

Available at vinkius.com/mcp/google-home — connect your AI agent in three steps.

- 01** First, subscribe to this MCP and obtain your OAuth 2.0 token and Project ID from the Google Device Access console.
- 02** Next, point your AI client—like Cursor or Claude—to the Vinkius catalog using these credentials.
- 03** Your agent then uses the available tools to interact with your home's devices based on natural language commands.

The bottom line is that your AI client treats your entire smart home system like a single, controllable unit.

Built For

This connector is for anyone who manages physical properties or lives in an advanced smart home setup. If you're tired of juggling multiple manufacturer apps and manually checking feeds, this MCP gives your agent the central brain it needs.

Property Manager

Monitoring climate control and security across several buildings remotely, making sure every unit is set to the right temperature or that cameras are recording.

Smart Home Owner

Asking their agent simple questions like, 'It's getting cold in here, switch the living room thermostat to heat mode,' instead of physically adjusting a dial.

Automation Enthusiast

Building complex workflows that combine camera monitoring with automated climate adjustments based on detected activity or time of day.

What Changes When You Connect

- 01** Full Security Visibility: Use `generate_camera_stream` and `stop_camera_stream` to get live RTSP/WebRTC feeds instantly. You don't have to open a separate app just to check the front door camera feed.

-
- 02** Precise Climate Control: Need to raise or lower the temperature? The MCP lets you `set_thermostat_mode`, `set_thermostat_heat`, and `set_thermostat_cool` directly through your agent, regardless of which room you're in.
-
- 03** Energy Management: Don't waste power. You can use `set_thermostat_eco` to place the system in energy-saving mode, or use `set_fan_timer` if you just need fresh air for a while.
-
- 04** System Mapping: Before changing anything, `list_devices` lets your agent map out every single component—cameras, thermostats, and doorbells—so you know exactly what you're controlling.
-
- 05** Structured Monitoring: By using `list_structures` and `list_rooms`, your agent understands the physical layout of your home, ensuring commands like 'Set the living room thermostat...' hit the right target.
-

Real-World Applications

Checking on a vacant property after hours

A property manager asks their AI client to check if anything is unusual. The agent runs `list_devices` first, then uses `generate_camera_stream` for the main entrance feed and `set_thermostat_mode` to ensure the HVAC system doesn't waste energy by running at full blast.

Quickly adjusting climate when running late

You walk in the door and ask your agent to make it warmer. The system identifies the local thermostat using `list_devices` and executes `set_thermostat_heat`, reporting back on the current temperature status.

Preparing a home before guests arrive

A homeowner asks their agent to get ready. The MCP automatically runs `list_rooms`, checks the current settings using `get_device`, and then uses `set_thermostat_range` to dial in the perfect comfort zone for both heating and cooling.

Troubleshooting a camera feed issue

A security expert wants to check multiple feeds quickly. They ask the agent to `generate_camera_stream` for all available doorbells and then `stop_camera_stream` once they've finished reviewing the live data.

Patterns to Avoid

Assuming device location

✗ AVOID

Telling your agent, 'Turn off the thermostat,' when the system has multiple thermostats across different floors or wings of the house.

✓ INSTEAD

Always start by calling `list_devices` to see all available devices. Then, use `get_device` and specify the exact ID or room name to ensure you're targeting the correct unit before running `set_thermostat_mode`.

Forgetting to stop streams

✗ AVOID

The agent keeps generating live feeds for cameras indefinitely, causing resource overload and billing issues.

✓ INSTEAD

After `generate_camera_stream` provides the stream URL, remember to use the corresponding `stop_camera_stream` tool. This cleanly releases the stream token and frees up resources.

Mixing temperature commands

✗ AVOID

Trying to set both the heat and cool temperatures using separate tools without defining a clear comfort range.

✓ INSTEAD

If you're in HeatCool mode, use `set_thermostat_range`. This tool lets your agent define a single comfort envelope for both heating and cooling simultaneously.

The Right Fit

Use this MCP if your goal is to centrally manage multiple physical systems—specifically climate control (heat/cool) and security monitoring—from one conversational interface. You need the deep device knowledge provided by `list_devices`, `set_thermostat_mode`, and `generate_camera_stream`. Don't use it if you just need simple scheduling; for that, a dedicated calendar or routine automation tool is better. Also, don't use it to manage non-Google branded smart devices—it only works with Nest via the SDM API. If your task involves coordinating actions across completely separate systems (e.g., lights and sprinklers), you might need an integration layer that acts as a middleware connector, rather than just focusing on one brand.

Juggling multiple smart home apps is exhausting.

Today, managing your house means logging into the Nest app for cameras, the separate thermostat portal for climate control, and maybe a third dashboard just to check overall device status. Every time you want to know if the front door camera saw anything while also adjusting the temperature in the back bedroom, it's a sequence of clicks, tabs, and context switches.

With this MCP, those steps vanish. You tell your agent what you need—'Check the backyard and turn up the heat.' The system handles the entire flow: it pulls live video from the doorbell using `generate_camera_stream`, confirms the status with `get_device`, and adjusts the climate using `set_thermostat_mode`. It's all done conversationally.

Control your whole house with Google Home MCP

Manual tasks like listing every connected device or determining which structure needs attention are tedious and prone to error. You have to navigate the system just to get a complete inventory of what's working.

This MCP lets your agent run `list_devices` instantly, providing you with a definitive, structured map of every piece of hardware in use. It gives clarity where there used to be complexity.

Google Home MCP with 12 Tools

Use these tools to list devices, adjust thermostats, set fan timers, generate live camera feeds, and manage the physical structure of your smart home.

#	TOOL	DESCRIPTION
01	<code>generate_camera_stream</code>	This tool provides a temporary live stream URL so your agent can display video feeds from any Nest camera or doorbell.
02	<code>get_device</code>	Fetch detailed status and information for one specific Google Nest device using its ID or name.
03	<code>list_devices</code>	Retrieves a complete list of every Nest device, showing their type, room assignment, and current state.
04	<code>list_rooms</code>	Shows the physical organization of your home by listing all structured rooms within a location.
05	<code>list_structures</code>	Lists every building or structure connected to your Google Nest account.
06	<code>set_fan_timer</code>	Sets the fan timer on a thermostat, turning the fan on for a specified period of time.
07	<code>set_thermostat_range</code>	Defines both the heating and cooling comfort temperature range for a Nest thermostat in HeatCool mode.
08	<code>stop_camera_stream</code>	Properly terminates an active camera stream session using the token provided by <code>generate_camera_stream</code> .
09	<code>set_thermostat_cool</code>	Adjusts the cooling temperature setting on a Nest thermostat when it's in Cool or HeatCool mode.
10	<code>set_thermostat_eco</code>	Manually forces your Nest thermostat into energy-saving eco mode to reduce consumption.
11	<code>set_thermostat_heat</code>	Sets the desired heating temperature on a Nest thermostat when it's in Heat or HeatCool mode.
12	<code>set_thermostat_mode</code>	Changes the overall operating mode of the thermostat, setting it to HEAT, COOL, HEATCOOL, or OFF.

See It in Action

Real prompts you can use once this MCP is connected to your AI agent through Vinkius Cloud.

U List all my Nest devices and show me what thermostats I have.



I found 4 Nest devices in your home: 1) Nest Thermostat (Living Room) — currently in HEAT mode at 21°C, 2) Nest Camera (Front Door) — streaming ready, 3) Nest Doorbell (Backyard) — battery powered, 4) Nest Hub (Kitchen). Would you like to adjust any thermostat settings?

U Set my living room thermostat to 22°C heating mode.



Setting thermostat to HEAT mode at 22°C... Done! Your Living Room Nest Thermostat is now heating to 22°C. Current room temperature is 19.5°C. The HVAC system has activated and is heating. Estimated time to reach target: approximately 15 minutes.

U Show me the live feed from my front door camera.



Generating live stream URL for Front Door Nest Camera... Stream URL generated (RTSP). The stream is active for the next 5 minutes. Use any RTSP-compatible player to view the live feed. Remember to stop the stream when done to free up resources. Would you like me to stop the stream?

Frequently Asked Questions

01 How do I check what Nest devices are connected using the Google Home MCP?

You run `list_devices`. This tool retrieves a full inventory of every piece of hardware, including cameras, thermostats, and doorbells across your entire property.

02 Can the Google Home MCP adjust the temperature if I'm away?

Yes, you can use `set_thermostat_mode` or `set_thermostat_eco`. This lets your agent put the system into energy-saving mode when no one is home.

03 What do I need to view a live feed with `generate_camera_stream`?

The tool generates a temporary stream URL (RTSP). You must use this link immediately, and remember that you'll need to call `stop_camera_stream` when you are done viewing.

04 Does the Google Home MCP understand different rooms in my house?

Yes. By using `list_rooms`, your agent understands the physical layout of your home and can target specific devices within a structure.

05 What if I want to change both heat and cool settings at once? Do I use `set_thermostat_range`?

Yes, `set_thermostat_range` is the correct tool. It lets your agent define a single comfort envelope that covers both heating and cooling parameters simultaneously.

06 What Google Nest devices are supported by this API?

The SDM API supports: Nest Thermostats (Nest Learning Thermostat, Nest Thermostat E, Nest Thermostat 2020), Nest Cameras (legacy, battery, wired, with spotlight), Nest Doorbells (legacy battery, wired, battery), and Nest Hub displays. Each device exposes specific traits (e.g., `ThermostatMode`, `CameraLiveStream`) that can be queried and controlled.

07 How do I get OAuth 2.0 credentials for the Google SDM API?

Visit the [**Google Device Access console**](https://console.nest.google.com/device-access/) and create a project. You'll receive a Project ID. Then, set up OAuth 2.0 credentials in Google Cloud Console, link them to your Device Access project, and complete the OAuth flow to obtain an access token. The token expires after 1 hour and must be refreshed.

08 Can I control multiple thermostats in different locations?

Yes! The API returns all devices across all structures and rooms in your project. Use `list_devices` to see all thermostats with their room assignments, then use each device's unique ID to control them individually. You can set different modes and temperatures for each thermostat.

09 How long do camera stream URLs last?







Camera stream URLs (both RTSP and WebRTC) are temporary and typically expire after 5 minutes. The stream token returned in the response should be used immediately and the stream should be stopped with `stop_camera_stream` when no longer needed to free up device resources.

Go Live in 60 Seconds

Get your connection token from cloud.vinkius.com, then paste the endpoint URL into any MCP-compatible client.

YOUR MCP ENDPOINT

```
https://edge.vinkius.com/[TOKEN]/mcp
```

CLIENT	WHERE TO CONFIGURE
 Claude AI	Profile → Customize → Connectors → "+" → Add custom connector → Paste endpoint
 Cursor	Settings → Features → MCP Servers → "+ Add New MCP Server" → Type: SSE → Paste endpoint
 VS Code	Ctrl/Cmd+Shift+P → "MCP: Add Server" → add <code>"google-home": { "url": "..."</code>
 Windsurf	MCP Settings → <code>mcp_settings.json</code> → Add endpoint URL
 ChatGPT	Settings → Tools & plugins → Add MCP server → Paste endpoint
 Gemini	Extensions → Add MCP Server → Paste endpoint URL

ASK AN AI ABOUT THIS

Let your preferred AI explain this MCP server

-  **Ask ChatGPT** 
-  **Ask Claude** 
-  **Ask Perplexity** 
-  **Ask Gemini** 
-  **Ask Grok** 

READY TO CONNECT

Google Home is live on Vinkius Cloud.

Get your connection token, paste it into your AI agent, and
start building. No SDK. No deployment. Just results.

[Start at cloud.vinkius.com](https://cloud.vinkius.com) →

vinkius.com · support@vinkius.com

INDEPENDENT PLATFORM DISCLAIMER

Vinkius is an independent platform and is not affiliated with, endorsed by, sponsored by, verified by, or otherwise authorized by Google Home. All third-party trademarks, logos, and brand names are the property of their respective owners. Their use in this document is strictly for informational purposes to identify service compatibility and interoperability.

DOCUMENT INFORMATION

Generated	June 2026
MCP Server	Google Home MCP
Server ID	019d75a8-c0b5-70b0-af00-197965d676c9
Platform	Vinkius Cloud for AI Agents
Endpoint	https://edge.vinkius.com/{token}/mcp

LICENSE & USAGE

This document is generated automatically by the Vinkius PDF Engine. Content reflects the MCP server configuration at the time of generation and may change as updates are deployed. For the most current information, visit vinkius.com/mcp/google-home.