MCP SERVER | NO CODE | CLOUD HOSTED

GraphHopper MCP

Calculate complex paths and solve logistics problems.

GraphHopper MCP gives your AI client full control over complex geospatial data. Calculate precise driving routes, figure out the service area reachable in specific times (isochrones), and solve massive fleet routing problems—all through natural conversation. It handles everything from translating addresses to coordinates to auditing GPS tracks against actual streets.

A+ Quality Score 100/100

routing | pathfinding | openstreetmap | fleet-optimization | geospatial-data | navigation

MAINTAINED BY VINKIUS CLOUD FOR AI AGENTS · JUNE 2026
vinkius.com/mcp/graphhopper

SCAN TO VIEW

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.

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.

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.

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	Ed25519	24h TTL	V8 Isolate
Encryption at rest	PKI vault signatures	Ephemeral session keys	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.

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.

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.

06 — Honeytoken Trap System

Phantom credentials are injected into isolated environments. If a honeytoken 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.

Agent Activity

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

FinOps

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

Tool Health

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

Firewall & DLP

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

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.

GraphHopper MCP

10 tools available
Cloud-hosted on Vinkius

Stop wrestling with spreadsheets or juggling multiple API calls just to plan a delivery route. This MCP connects your AI client directly to advanced geospatial engine capabilities, letting you treat location data like any other piece of information. You can ask it to calculate the best multi-stop path for a vehicle, generate polygons showing all addresses reachable within 10 minutes, or even validate if a GPS track was slightly off by snapping it perfectly onto known road vectors. Need to know what coordinates mean? Use your agent to perform reverse geocoding and get the exact street name attached. It's about taking control of complex logistics and mapping without touching code. When you connect this MCP via Vinkius, you gain access to a powerful set of tools that lets you talk through difficult routing problems—whether it’s solving for vehicle capacity or calculating massive travel matrices between dozens of points.

Core Capabilities

01 — Calculate complex routes and paths

Determines the optimal driving directions, distances, and times between any set of GPS coordinates.

02 — Define service areas (Isochrones)

Maps out a precise boundary or polygon showing all points you can reach from one spot within a specified time limit.

03 — Solve fleet logistics problems

Optimizes routes for multiple vehicles, checking constraints like total travel time and vehicle capacity to find the most efficient plan.

04 — Translate addresses to coordinates (Geocoding)

Converts human-readable street names or addresses into precise latitude and longitude pairs needed for mapping software.

05 — Verify location boundaries

Takes raw GPS pin locations and matches them back to specific named streets, ensuring accurate place identification.

One Click on Vinkius — From Prompt to Execution

Available at **vinkius.com/mcp/graphhopper** — connect your AI agent in three steps.

01 Subscribe to this MCP on Vinkius and enter your GraphHopper API Key into your AI client.

02 Ask your agent a question about location or routing in plain English (e.g., 'What's the best route for three stops?').

03 The MCP processes the request, uses its underlying tools to calculate the solution, and delivers the structured map data or optimized path back to your chat interface.

The bottom line is you tell your AI agent what you need, and it figures out the complicated math of geography for you.

Built For

Anyone whose job involves moving things or people across a physical space. If you spend time looking at maps, calculating distances, or planning logistics in spreadsheets, this is for you. Stop manually copying coordinates and starting to talk directly to the data.

Logistics Manager

Uses the MCP to solve complex delivery problems, generating multi-stop routes and verifying if a fleet can hit all time windows without manual calculations.

GIS Analyst

Performs spatial analysis by calculating isochrone reachability boundaries for site selection or identifying the best service zone using natural language prompts.

Supply Chain Planner

Calculates routing matrices and verifies distance tables across entire distribution networks, letting them compare multiple complex pathways in real time.

What Changes When You Connect

01 Stop manual calculation. Instead of building massive spreadsheets to compare distances, simply ask the agent for a 'routing matrix' using `calculate_routing_matrix` and get instant comparisons across all your points.

02 Pinpoint service areas instantly. You don’t have time to draw polygons by hand; use `calculate_distance_isochrone` to map exactly what customers can reach in 15 minutes from a new depot location.

03 Handle multi-stop routes easily. Instead of calling separate APIs for every leg, the agent uses tools like `calculate_heavy_route` to calculate optimal paths for dozens of stops in one go.

04 Clean up dirty data. If your field team sends back messy GPS tracks, use `snap_gpx_to_road` to snap those raw points perfectly onto actual street vectors—no manual cleaning required.

05 Verify addresses with confidence. When you get coordinates, don't guess what they mean; run `reverse_geocode` and get the precise street name attached, every time.

Real-World Applications

Determining a new depot location

A city planner needs to find the best spot for a new warehouse. They ask the agent to generate a 'reachability polygon' around three potential sites, using `calculate_reachability_polygon`. This immediately shows them which site covers the highest density of target customers within a 20-minute drive.

Optimizing daily deliveries

A logistics manager has 15 packages and 3 trucks with different capacities. They tell their agent to 'solve the vehicle routing problem.' The MCP uses `submit_vrp_optimizer` and returns a perfect, constrained schedule that minimizes total mileage.

Verifying field data

A developer receives a large GPX file from a drone flight path. They ask the agent to 'audit the GPS track.' The MCP runs `snap_gpx_to_road`, which instantly corrects any minor positional jumps, making the raw data reliable for billing.

Comparing distribution networks

A supply chain planner needs to compare the travel time between 5 warehouses and 10 client sites. They ask the agent to calculate a 'routing matrix,' receiving a synchronous table of all distances and times immediately.

Patterns to Avoid

Using simple map widgets

✗ AVOID

Trying to find the optimal route for 10 stops by drawing lines on Google Maps or using basic online trip planners.

✓ INSTEAD

Don't waste time with visual estimates. Use `calculate_heavy_route` or `submit_vrp_optimizer` via your agent. These tools handle complex, multi-stop geometry and constraints that simple map widgets ignore.

Manual coordinate lookup

✗ AVOID

Getting a list of coordinates from a client and manually cross-referencing them on a physical street map to find the official address.

✓ INSTEAD

Just hand those coordinates to your agent. It runs `reverse_geocode` and spits out the exact, verifiable street name in seconds.

Treating data as simple points

✗ AVOID

Thinking that a single GPS pin is enough for logistics planning.

✓ INSTEAD

Real-world planning requires understanding boundaries. Use `calculate_distance_isochrone` to understand the entire service area, not just one point.

The Right Fit

Use this MCP if your job involves physical space and moving things: calculating optimal paths, defining service zones, or solving complex scheduling problems based on location data. You need it when you have multiple stops, limited vehicle capacity, or must analyze coverage over a large area. Don't use it just because you need to look up the quickest route between two points; for simple point-to-point travel, basic mapping tools are enough. If your

primary goal is merely reading data from a database, don't use this MCP; instead, use a dedicated record management or data retrieval type tool.

The headache of location planning today

Right now, figuring out logistics means jumping between five different interfaces. You pull coordinates from one sheet, input them into another mapping service to check a route, and then manually export the resulting data into a third program to solve for vehicle capacity. It’s copy-pasting hell.

With this MCP, you stop clicking through dashboards. You just talk to your agent: 'Find the best routes for these 12 stops using three trucks with limited space.' The system does all that heavy lifting and returns a perfect plan instantly.

GraphHopper Gives You Control Over Every Map Detail

You no longer have to guess if the raw GPS data is accurate. If your field team sends back messy coordinates, you simply ask the agent to 'audit the track.' The MCP runs `snap_gpx_to_road`, which cleans up every minor jump and makes the path perfect.

This means you spend zero time cleaning data and 100% of your time making decisions. You get reliable, actionable geographic truth without ever opening a spreadsheet or debugging an API call.

GraphHopper: 10 Geospatial Tools

These tools give you the power to analyze location data—calculating routes, defining boundaries, and solving complex logistics problems through natural language commands.

#	TOOL	DESCRIPTION
01	`search_geocode`	Pinpoints the correct geographic area for specific routes within the GraphHopper engine.
02	`calculate_distance_isochrone`	Creates a boundary map (polygon) showing all points accessible from one location in a given time frame.
03	`calculate_reachability_polygon`	Generates structured rules that define the active reachability area for site planning or delivery zone mapping.
04	`snap_gpx_to_road`	Corrects imprecise GPS data by snapping raw tracking files perfectly onto known street paths.
05	`calculate_routing_matrix`	Generates deep arrays of travel times and distances to analyze complex logistics tables between multiple points.
06	`reverse_geocode`	Matches GPS coordinates back to specific, named street properties for accurate location verification.
07	`calculate_url_route`	Retrieves lightweight and explicit step-by-step directions between two points.
08	`calculate_heavy_route`	Identifies detailed arrays that span native paths for complex, multi-stop geometry calculations.
09	`submit_vrp_optimizer`	Runs an automated check to solve complex vehicle routing problems using predefined constraints like time limits.
10	`poll_vrp_solution`	Retrieves the final structured results confirming the best alternative routes for delivery planning.

See It in Action

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

U Calculate a car route between '40.71, -74.00' and '40.75, -73.98'

Route calculated! I've found the optimal driving path. Total distance: 3.5 miles. Estimated time: 12 minutes. I can provide the step-by-step turn directions for you now.

U Show me the 10-minute reachability zone from central Berlin

Isochrone generated! I've mapped the area reachable within 10 minutes from your location in Berlin. I can provide the polygon coordinates or verify if specific addresses fall within this boundary.

U Reverse geocode these coordinates: '48.85, 2.35'

Coordinates matched! The location for 48.85, 2.35 corresponds to 'Rue de Rivoli, Paris, France'. It is located in the 4th arrondissement. Would you like more venue details for this spot?

Frequently Asked Questions

01 **How do I use GraphHopper MCP for single-stop routing?**

You can calculate simple routes using `calculate_url_route`. Just ask your agent to find the best path between two coordinates, and it will handle the basic distance and time calculations.

02 **What is the main difference between GraphHopper MCP and a standard mapping API?**

A standard API gives you a route; this MCP lets you solve complex problems. It can calculate reachability polygons or optimize for vehicle capacity, which simple APIs can't do.

03 **Can I use GraphHopper MCP to check if an area is reachable?**

Yes. Use `calculate_distance_isochrone` to map out the exact boundary (polygon) of all points you can reach from a starting location within your specified time limit.

04 **Do I need multiple tools in GraphHopper MCP for fleet optimization?**

No, the agent handles it. You just tell it to solve the vehicle routing problem; it automatically uses tools like `submit_vrp_optimizer` and then reads the final answer back using `poll_vrp_solution`.

05 **Does GraphHopper MCP work with my custom coordinates?**

Absolutely. The agent accepts raw GPS data and can process it, whether you need to find a street name (`reverse_geocode`) or clean up the path itself (`snap_gpx_to_road`).

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 `"graphhopper": { "url": "…" }`
Windsurf	MCP Settings → `mcp_settings.json` → 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

GraphHopper 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 →

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 GraphHopper. 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	GraphHopper MCP
Server ID	`019d75a4-31ad-704b-830f-0811622d0b9f`
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/graphhopper.