

MCP SERVER

NO CODE

CLOUD HOSTED

Helpshift MCP

Manage support issues and user data instantly.

Helpshift MCP connects your AI agent directly into your customer support platform. Manage issues, run user updates, and check documentation status—all from natural conversation. Instead of manually logging into a dashboard to track tickets or update profiles, you just talk to your agent. It lets you get full visibility on ticket history, manage user accounts in bulk, and keep up with FAQs without leaving your development environment.

A+ Quality Score 100/100

in-app-support

ticket-management

user-profiles

automation

customer-experience

audit-logs



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.

Helpshift MCP

11 tools available

Cloud-hosted on Vinkius

Your AI client handles complex support operations using Helpshift data. You no longer need to jump between your CRM, your knowledge base, and a dedicated ticketing system just to answer one question. Your agent acts like a highly skilled Support Ops team member, reading the ticket history, checking if the user's profile is correct, or even finding the right FAQ article for a customer. If you use Vinkius, this MCP connects all of those moving parts into one place. You can tell your agent to find every open issue related to payments and then update their status to 'Investigating.' It handles it. The system doesn't just give you data; it lets you perform actions—like adding a reply or running bulk updates—all through simple chat commands. This is about turning complex, multi-step administrative tasks into one conversation.

Core Capabilities

01 — Manage Support Tickets

Create new issues and retrieve detailed information, including the full action history for any support case.

03 — Maintain Knowledge Base Content

Browse all published FAQ articles and list the main categories of documentation available.

02 — Update User Data in Bulk

Apply changes to many user accounts at once or perform large-scale profile creations using a dedicated API.

04 — Monitor Operational Status

Check the status of large batch tasks or monitor overall support volume across the platform in real time.

One Click on Vinkius — From Prompt to Execution

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

- 01 Subscribe to this MCP and enter your Helpshift API key and domain credentials.
- 02 Connect your agent; it authenticates with your support platform's data streams.
- 03 You simply talk to your agent, telling it what needs doing (e.g., 'Check the audit log for issue 5501'). The agent performs the action and gives you a direct status update.

The bottom line is that you get to manage complex support operations by speaking naturally to your AI client, rather than clicking through multiple dashboards.

Built For

This MCP is built for technical teams and operational managers who spend too much time juggling data across different internal tools. It targets the pain of manual context switching.

Customer Support Manager

Needs to instantly pull up full issue transcripts and audit logs when a client calls with an escalated problem.

Product Operations Specialist

Must run bulk updates on user accounts or sync identity data for large batches of new users.

Technical Support Agent

Needs to add a follow-up message and change an issue's status (like 'Resolved') without leaving the chat interface.

What Changes When You Connect

- 01 Instead of exporting issue logs to Excel, you can ask your agent to pull the full action history for any ticket. This means you get immediate context on who touched a case and why, using `get_issue_audit_logs`.

-
- 02** Managing user profiles used to be a manual slog. Now, you use `bulk_user_action` to update dozens of records at once, handling identity syncs or profile fixes without leaving your chat window.
-
- 03** You keep documentation accurate by having the agent list all published FAQs using `list_faqs`. This ensures that when you talk to a client, you are referencing the most current knowledge base content.
-
- 04** When something pops up, you don't need to manually create tickets. Use `create_issue` directly through your conversation with your agent; it gets logged and assigned immediately.
-
- 05** You can change the state of a ticket—move it from 'Open' to 'Resolved' or 'Rejected'—using `update_issue_status`. It's an action command, not just a data read.
-

Real-World Applications

Need to understand why a major bug report was delayed?

An agent asks their MCP client: 'List issues and then get the audit logs for issue 5501.' The agent immediately tells them the ticket details and shows that the status was manually changed multiple times, pinpointing exactly when it stalled.

A customer asks about payment errors, but you aren't sure which knowledge article applies.

The support rep tells their agent: 'Search for FAQs related to payments.' The agent uses `'list_faqs'` and reads the two most relevant articles aloud, allowing the rep to quote accurate documentation immediately.

A marketing campaign requires updating user data across thousands of accounts.

The Product Ops specialist uses their agent to run a bulk action: 'Update all users in Segment X with the new subscription tier.' The agent confirms the process started and provides `'get_bulk_task_status'` until everything is done.

You need to start tracking a new bug report that came in via Slack.

Instead of opening a separate ticket form, the agent executes `'create_issue'`, populating all the necessary fields (app ID, title, body) directly from your prompt. The issue is live and ready for triage.

Patterns to Avoid

Treating it like a simple database query.

X AVOID

Asking the agent, 'Give me all open issues.' This only gives you raw data; it doesn't give context or allow for follow-up actions.

✓ INSTEAD

First, use `list_issues` to get an overview. Then, select a specific ID and ask the agent to run `get_issue_audit_logs`. This sequence of tools provides the full story: who did what and when.

Manually updating user records in batches.

X AVOID

Having to download a CSV, open your CRM, update 50 rows, and re-upload it. This is slow, error-prone work.

✓ INSTEAD

Use the agent to run `bulk_user_action`. You pass the list of changes as data, and the MCP handles the connection and execution in the background.

Assuming a ticket's status is always correct.

X AVOID

Thinking an issue that was marked 'Open' is still open, when it might have been rejected or needs specific internal follow-up.

✓ INSTEAD

Always confirm the current state by using `get_issue_details` and then use `update_issue_status` only after verifying the correct transition.

The Right Fit

Use this MCP if your core job involves reacting to, modifying, or tracking support tickets and user identities. Specifically, if you find yourself doing three things in one day: 1) Reading a long thread of issue activity; 2) Correcting or adding data to a user's profile; and 3) Checking documentation accuracy—this is your tool. Don't use it if your primary workflow is solely about building reports on historical volume (other specialized BI tools are better for that). If you only need to read static data, like listing all apps (`list_registered_apps`), a simple API connector might suffice. But since this MCP lets you perform actions—like adding a reply via `add_issue_message` or changing the status with `update_issue_status`—it's built for active operations.

The Context Switching Tax

Right now, when an issue comes in, you get a notification. You open your ticketing system to read the details. Then, you switch over to your user management dashboard to check if that customer's account is active. After that, you might have to go back to your knowledge base portal just to find the correct policy article to quote. This process requires three or four full tabs, five clicks minimum, and a lot of mental context switching.

With this MCP, you tell your agent everything in one chat. You prompt it: 'Check issue 5501's details, see if user X is valid, and find the FAQ about payment errors.' The agent handles the sequence, pulls all three data points together, and gives you a single, actionable answer without you ever leaving your main workflow.

Helpshift MCP: Actionable Insights

The manual steps that vanish are the constant transitions between platforms. You stop copy-pasting IDs from one dashboard into another, and you don't have to manually run separate reports for user data versus ticket history.

You get a single source of truth in your chat interface. It's not just reading; it's performing. This capability fundamentally changes how quickly you can respond to complex operational problems.

Helpshift MCP with 11 Tools

These tools allow you to perform every major operational task in Helpshift—from creating new issues to updating user data—all via your AI client.

#	TOOL	DESCRIPTION
01	<code>add_issue_message</code>	Adds a direct reply or internal note to an existing support issue.
02	<code>bulk_user_action</code>	Performs bulk operations like creating or updating multiple user profiles at once.
03	<code>create_issue</code>	Starts a brand new support ticket using required details like the app ID and subject line.
04	<code>get_issue_audit_logs</code>	Retrieves the full history of actions taken on a specific issue, showing who did what and when.
05	<code>get_bulk_task_status</code>	Checks if a large user profile update job completed successfully or is still running.
06	<code>get_issue_details</code>	Pulls up all the main information about one specific support ticket, like who reported it and when.
07	<code>list_registered_apps</code>	Shows a list of every application that is connected to or registered within your Helpshift account.
08	<code>list_faqs</code>	Retrieves the titles and summaries of all published Frequently Asked Questions articles.
09	<code>list_issues</code>	Lists a summary of current support issues, helping you track overall ticket volume.
10	<code>list_faq_sections</code>	Shows the main categories or sections used to organize your documentation library.
11	<code>update_issue_status</code>	Changes the overall status of a ticket, marking it as 'Resolved' or 'Rejected'.

See It in Action

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

U List all active issues and show their audit logs.



I've retrieved your active issues. You have 3 open cases, including 'App crash on iOS' (ID: 5501) and 'Payment failure' (ID: 5505). I've also fetched the audit log for 5501, which shows it was reassigned to the Engineering team 2 hours ago. Would you like the full transcript for either case?

U Add a reply to issue ID 5501: 'We are investigating the logs now'.



Message added! I've successfully sent your reply to issue 5501. The customer will receive this update in their app. Would you like to set the status to 'Resolved' or add any internal tags?

U Search for FAQ articles related to 'subscription renewal'.



Searching FAQs... I found 2 relevant articles: 'How to renew your plan' and 'Troubleshooting renewal errors'. Would you like the full content of 'How to renew your plan' to share with a customer?

Frequently Asked Questions

01 How do I use the Helpshift MCP to list all active issues?

You run `list_issues` through your agent. It returns a summary of current tickets, which is useful for quickly monitoring overall support volume and spotting urgent cases.

02 Can I update user profiles using the Helpshift MCP?

Yes, you can use `bulk_user_action` to perform profile operations on multiple users at once. This is great for large identity syncs or rolling out new features to a segment.

03 What tool do I use with Helpshift MCP to see ticket history?

To check the full action trail, you must run ``get_issue_audit_logs``. This shows who changed the status or added messages, giving you a complete record of the case.

04 Does Helpshift MCP help with documentation?

You can list available content using ``list_faqs`` to see all published articles. You can also use ``list_faq_sections`` to browse the major categories of your knowledge base.

05 How do I mark a ticket as resolved with Helpshift MCP?







Use the ``update_issue_status`` tool in your prompt. You tell the agent the issue ID and the new status (like 'Resolved'), and it updates the record for you.

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>"helpshift": { "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

Helpshift 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 Helpshift. 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	Helpshift MCP
Server ID	019d75af-b7cc-7087-aff5-f5db44b4486c
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/helpshift.